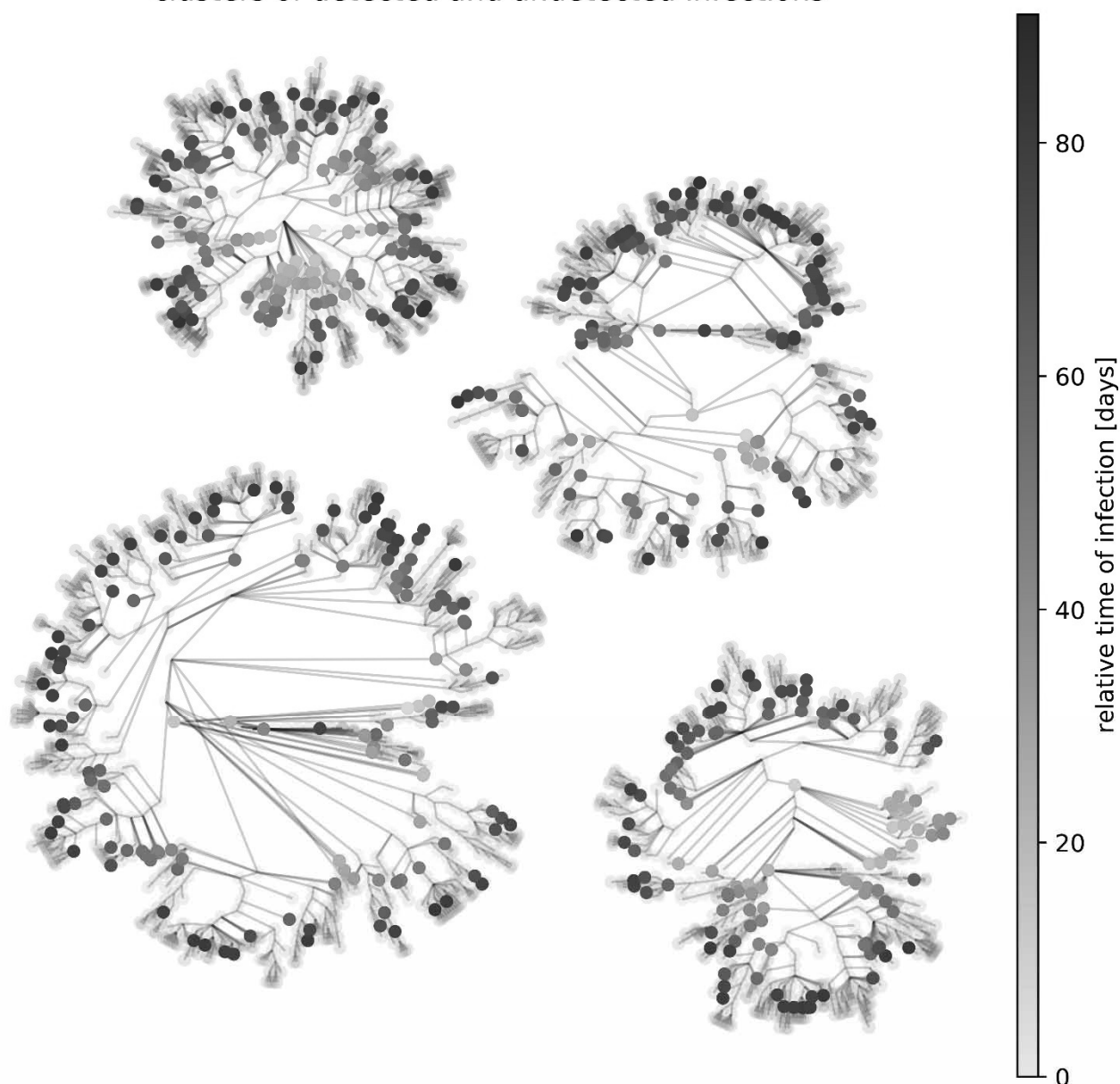


clusters of detected and undetected infections



Proceedings ASIM SST 2020

25. ASIM Symposium Simulationstechnik
14.10. - 15.10.2020, Online-Tagung

ASIM Mitteilung AM 174

ISBN eBook 978-3-901608-93-3

ARGESIM Report AR 59

DOI 10.11128/arep.59

Proceedings

ASIM SST 2020

25. ASIM Symposium Simulationstechnik

14.10. - 15.10.2020

Online-Tagung

Organisation: Fraunhofer IAIS, Sankt Augustin

Herausgeber: Christina Deatcu
Daniel Lückerrath
Oliver Ullrich
Umut Durak

ASIM Mitteilung AM 174
ISBN eBook 978-3-901608-93-3

ARGESIM Report AR 59
DOI 10.11128/arep.59

Cover: 'Visualisierung eines simulierten Transmissionsclusters der SARS-CoV-2 Pandemie'
aus SST Hauptvortrag Niki Popper & Martin Bicher, dwh GmbH Wien / TU Wien
'Simulation Based Decision Support - the COVID19 Crisis from a Modeller's Perspective'

Bibliographic Data:

ARGESIM Reports

Published by ARGESIM and ASIM, Arbeitsgemeinschaft Simulation,
Fachausschuss Simulation in der GI - Gesellschaft für Informatik

Series Editors:

Felix Breitenecker (ARGESIM / ASIM)
Math. Modelling and Simulation
Group, TU Wien
Wiedner Hauptstrasse 8-10
1040 Vienna, Austria

Thorsten Pawletta (ASIM)
CEA - Computational Engineering and
Automation, HS Wismar
PF 1210
23952 Wismar, Germany

Titel: Proceedings ASIM SST 2020

Untertitel: 25. ASIM Symposium Simulationstechnik,
14.10. - 15.10.2020, Online-Tagung,
Organisation: Fraunhofer IAIS, Sankt Augustin

Herausgeber: Christina Deatcu, Daniel Lückerrath,
Oliver Ullrich, Umut Durak

ARGESIM Report: AR 59

ASIM Mitteilung: AM 174

ISBN eBook: 978-3-901608-93-3

DOI: 10.11128/arep.59

individual DOIs for contributions: 10.11128/arep.59.a59nnn



© by ARGESIM / ASIM, Wien, 2020

Publisher ARGE Simulation News (ARGESIM)
c/o F. Breitenecker, Math. Modelling and Simulation Group, TU Wien
Wiedner Hauptstrasse 8 - 10, A - 1040 Vienna, Austria
Tel: +43-1-58801-10115, Fax: +43-1-58801-910115
Email: info@argesim.org; WWW: www.argesim.org

ASIM SST 2020 - Vorwort ASIM

Liebe Teilnehmerinnen und Teilnehmer, liebe ASIM-Mitglieder !

25. *ASIM Symposium Simulationstechnik* – normalerweise ein Anlass für zurückhaltendes oder weniger zurückhaltendes Feiern einer gelungenen Tagungsreihe.

Es begann 1982 – mit dem ASIM SST 1982, dem 1. Symposium Simulationstechnik in Erlangen, kurz nach der Gründung von ASIM. Das SST – das *Symposium Simulationstechnik* – präsentierte sich als ASIM-Jahrestagung, mit allen Richtungen und dem aktuellen State-of-the-Art der Simulationstechnik. Als ASIM Mitglied von EUROSIM, der Federation of European Simulation Societies wurde, wurde das SST jedes dritte Jahr ein Teil des *EUROSIM Congress*, der Tagung der europäischen Simulationsvereinigungen.

Das SST bereiste die Deutschen, Österreichischen und Schweizer Lande und wurde das Familientreffen der „Simulanten“ und die Anwendungsgebiete wurden immer breiter. Es entwickelte sich der Bedarf an eigenen Workshops und Tagungen der ASIM-Fachgruppen, sodass auch das SST sich wandelte – von der (jährlichen) Jahrestagung zu einer alle zwei Jahre stattfindenden Tagung zu Methodik und Anwendungen der Simulationstechnik in Abwechslung mit der ebenso alle zwei Jahre veranstalteten Tagung *Simulation in Produktion und Logistik* der namensgleichen ASIM Fachgruppe und parallel zu Workshops weiterer ASIM-Fachgruppen.

Das *Symposium Simulationstechnik* machte auch den Generationswechsel und die Konsolidierung von ASIM mit, versucht heute verstärkt die Unterstützung junger „Simulanten“ und geht in Richtung einer teilweise englischsprachigen Tagung, um dem Simulationsnachwuchs einen Einstieg in die europäische bzw. internationale Simulationswelt zu bieten.

Und heuer, 2020, das 25. *Symposium Simulationstechnik*, das nicht auf unsere Feiern angewiesen ist. Zunächst als klassische Tagung in Bayreuth konzipiert, feiert es sich selbst als erste virtuelle ASIM-Tagung vSST 2020, allen Corona-Irrungen und Wirrungen zum Trotz – wie es das folgende Vorwort der (nicht virtuellen) Organisatoren dokumentiert.

Es wird vermutlich auch ein 26. Symposium Simulationstechnik geben, aber vielleicht eine hybride Tagung – das *Symposium Simulationstechnik* kann aus der Corona-Krise lernen.

Dank allen Organisatoren der bisherigen 24 SST, und spezieller Dank den Organisatoren des 25. vSST 2020 – und ALLES GUTE, ASIM vSST 2020 zum 25. !

Prof. Dr. Felix Breiteneker, ASIM Sprecher

ASIM vSST2020 - Vorwort Organisatoren

„Dieses Corona-Virus verbreitet sich immer weiter. Die ASIM fällt aus.“ – „Es gibt mehrere ASIM-Konferenzen. Welche fällt aus?“ – „Alle.“

Ziemlich schnell war klar: Wir müssen etwas tun. Wir müssen die ausgefallenen Präsenzveranstaltungen – u.a. den Workshop zu Simulation in den Umwelt- und Geowissenschaften, die Fachgruppentagung zu Simulation Technischer Systeme und Grundlagen und Methoden in Modellbildung und Simulation sowie das 25. Symposium Simulationstechnik – zusammenführen und zumindest eine virtuelle Tagung, ein Virtuelles Symposium Simulationstechnik auf die Beine stellen.

Dann ging es los: Thorsten Pawletta von der HS Wismar und Walter Commerell von der HS Ulm gaben den Anstoß, Daniel Lückerrath und Oliver Ullrich vom Fraunhofer IAIS in Sankt Augustin trugen die Beiträge der Fachgruppentagung GMMS/STS bei und organisierten Anmeldungs- und Reviewprozess des Symposiums, Jochen Wittmann von der HTW Berlin brachte die Vorträge des Workshops zu Simulation in den Umwelt- und Geowissenschaften mit, Andreas Körner von der TU Wien die Paper der Fachgruppe Simulation und Edukation, Kurt Chudej von der Uni Bayreuth machte die Beiträge des ausgefallenen Symposium Simulationstechnik zugänglich, Oliver Rose von der Universität der Bundeswehr in München erklärte sich spontan bereit für die Konferenztechnik zu sorgen, Christina Deatcu von der HS Wismar und Umut Durak vom DLR Braunschweig bereiteten den Tagungsband vor. Sämtliche Mitglieder des Programmkomitees und des ASIM-Vorstands schlugen sich Wochenenden um die Ohren um die vielen Beiträge durchzuschauen und zu bewerten. Felix Breitenecker als ASIM-Sprecher koordinierte all diese Aktivitäten.

Und es hat sich gelohnt. Es gibt rekordverdächtige Teilnehmerzahlen, viele glänzende Beiträge; die Diskussionen in den einzelnen Sitzungen versprechen enorm interessant zu werden. Dazu freuen wir uns darauf viele altbekannte Kollegen und Freunde – wenn auch nur per Videokonferenz – im Rahmen des virtuellen Symposiums wiederzusehen.

Wir sind überzeugt: Irgendwann, wenn das alles vorbei ist, werden wir uns zu einem Wein im ASIM-Hauptquartier in Wien treffen und uns lachend Geschichten erzählen. „Weißt du noch, die Corona-Zeit? Das waren wirklich verrückte Jahre.“

Die Organisatoren vom Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme IAIS, Sankt Augustin

Oliver Ullrich
Daniel Lückerrath



Organisation:

Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme IAIS, Sankt Augustin

Mitorganisation:

ASIM / GI

Universität der Bundeswehr München

Technische Universität Wien

Technische Hochschule Ulm

Hochschule Wismar

Programmkomitee:

Felix Breiteneker, TU Wien

Kurt Chudej, Universität Bayreuth

Walter Commerell, Hochschule Ulm

Christina Deatcu, Hochschule Wismar

Umut Durak, DLR

Leo Gall, LTX Simulation GmbH, München

Lukas Hollenstein, Züricher Hochschule für Ang. Wissenschaften

Andreas Körner, TU Wien

Christoph Laroque, Hochschule Zwickau

Xiaobo Liu-Henke, Hochschule Ostfalia

Daniel Lückerrath, Fraunhofer IAIS, Sankt Augustin

Heinz-Theo Mammen, Hella KGaA Hueck & Co, Lippstadt

Thorsten Pawletta, Hochschule Wismar

Nicolas Popper, dwh GmbH, Wien

Oliver Rose, Universität der Bundeswehr München

Thomas Schramm, HafenCity Universität Hamburg

Michael Striebel, HTWG Konstanz

Oliver Ullrich, Fraunhofer IAIS, Sankt Augustin

Sigrid Wenzel, Universität Kassel

Jochen Wittmann, HTW Berlin

INHALT

Plenarvortrag

Zukunft modellieren: Science Fiction als soziale Simulation von Hyperobjekten <i>Lars Schmeink</i>	1
---	---

Grundlagen und Methoden in Modellbildung und Simulation I

Modelling of Non-standard Queuing Policies - An invitation to ARGESIM Benchmark C22 <i>Peter Junglas, Thorsten Pawletta</i>	9
Model Generation for Multiple Simulators Using SES/MB and FMI <i>Hendrik Folkerts, Thorsten Pawletta, Christina Deatcu</i>	13
Generative and Modular Simulation Models for Supply and Manufacturing Networks <i>Pavel Gocev, Tim Hellfeuer</i>	21
Entwicklung webbasierter graphischer Benutzeroberflächen für Simulationskerne <i>Svenja Hilbrich, Katharina Gerdes, Johannes Hinckeldeyn, Jochen Kreutzfeldt</i>	23

Grundlagen und Methoden in Modellbildung und Simulation II

NSA-DEVS: Combining Mealy behaviour and Causality <i>Peter Junglas</i>	33
Investigation on Stability Properties of Hierarchical Co-Simulation <i>Irene Hafner, Niki Popper</i>	41
A Stable but Explicit Cosimulation Coupling Method <i>Thilo Moshagen</i>	49
Model Order Reduction of Deterministic Microscopic Models - A Case Study <i>Matthias Rößler, Niki Popper</i>	57

Mathematische Verfahren in Modellbildung und Simulation I

Analyse, Simulation und optimale Steuerung eines Dengue-Fieber-Modells mit temporärer Kreuzimmunität <i>Mark Herath, Kurt Chudej</i>	63
Entwicklung und Analyse einer stufigen SVIR-Baummodellierung <i>Amelie Flothow, Kurt Chudej</i>	73
Simulation of a discharge electrode needle for particle charging in an electrostatic precipitator <i>Sebastian C.-Beckers, Julian Pawlik, Hikmet Eren, Adam Sanaf, Jürgen Kiel</i>	83

Mathematische Verfahren in Modellbildung und Simulation II

Analyse und Simulation des Kraftübertragungsverhaltens von Mecanum-Rädern <i>Marian Göllner, Xiaobo Liu-Henke, Ludger Frerichs</i>	89
Modellreduktion für hochviskose, nicht-isotherme Fluide mit freier Oberfläche <i>Edmon Skeli, Dirk Weidemann, Klaus Panreck</i>	99
Modelling and simulation of multi-physics applications in case of sudden transformations of material properties <i>Robert Courant, Jürgen Maas</i>	107

Maschinelles Lernen in der Simulation I

Reducing response time with data farming and machine learning <i>Falk Stefan Pappert, Oliver Rose</i>	113
Neural Network Application for Event Detection in Hybrid Dynamical Systems <i>Stefanie Winkler, Andreas Körner, Felix Breitenecker</i>	121
Reduction of Complexity in Q-Learning a Robot Control for an Assembly Cell by using Multiple Agents <i>Georg Kunert, Thorsten Pawletta, Sven Hartmann</i>	129
Visual Analytics for Data-Driven Analysis in Semiconductor Manufacturing <i>Patrick Boden, Sebastian Rank, Thorsten Schmidt</i>	137

Maschinelles Lernen in der Simulation II

Simulationsgestützte Auslegung von Reglern mithilfe von Machine Learning <i>Dominic Brown, Martin Strube</i>	141
Sequence to Sequence Modelle zur hochaufgelösten Prädiktion von Stromverbrauch <i>Benjamin Wörrlein, Steffen Straßburger</i>	149
Vorhersage und Regelung der Methanproduktion durch maschinelles Lernen David Wagner, Wolfgang Schlüter	159
Prediction of PM emissions during transient operation of marine diesel engines using artificial neural networks <i>Michèle Schaub, Michael Baldauf, Egon Hassel</i>	167

Echtzeitsimulation

Erweiterung der Entwicklungsplattform LoRra um eine Schnittstelle zum Internet der Dinge <i>Sven Jacobitz, Xiaobo Liu-Henke</i>	175
Echtzeitfähige Motorprozessmodelle für Schiffsmaschinen-Simulatoren <i>Georg Finger, Karsten Wehner, Egon Hassel, Steffen Loest, Michael Baldauf</i>	185
ROCS: A Realtime Optimization and Control Simulator <i>Andreas Britzelmeier, Matthias Gerds, Omid Moslehi Rad, Sonali Rani, Thomas Rottmann</i>	193
Additive process chain: From virtual design to real implementation <i>Martin Rambke, Sven Lippardt, Tobias Mussehl</i>	201

Simulation mechatronischer Systeme

Systemsimulation als Teil des Systems Engineering für Scheinwerfer- und Pedalsysteme auf Basis der Modellbeschreibungssprache Modelica <i>Heinz-Theo Mammen, Phillip Limbach, Thorsten Maschkio</i>	207
Anforderungsmanagement für die modellbasierte Entwicklung mechatronischer Systeme im digitalisierten und vernetzten Umfeld <i>Or Aviv Yarom, Jie Zhang, Christian Raulf, Xiaobo Liu-Henke, Thomas Vietor</i>	213
Method for the computer-aided design and simulation of hydrogel-based microfluidic chips <i>Andreas Voigt, Jörg Schreiter, Christian Mayr, Andreas Richter</i>	221
Simulation Study on Various Double Pendulum Configurations <i>Milena Sipovac, Stefanie Winkler, Andreas Körner</i>	227

Simulation in der Luft- und Raumfahrt

Modulare Entwicklungsplattform für elektrische Luftfahrtantriebe <i>Markus Henke, Jan Hoffmann, Lennard Waschke</i>	231
Verification of Drogue Detection during Autonomous Aerial Refueling in a Simulation Environment <i>Oliver Ellis, Umut Durak</i>	237
Simulation Based Execution of UAV Missions Sent Through Web Services <i>Siddhartha Gupta, Umut Durak</i>	239
Modeling and simulation of a multi-functional high-lift actuation system based on key performance data <i>Andreas Schäfer, René Hollmann, Oliver Bertram</i>	241

Simulation in der Fahrzeugtechnik und bei autonomen Systemen

Intelligente Zielführung elektrischer Fahrzeuge mit Brennstoffzelle als Range Extender
in vernetzten Verkehrssystemen 251
Sören Scherler, Xiaobo Liu-Henke

Konfliktfreie, selbstoptimierte Trajektorienplanung für ein fahrerloses Transportfahrzeug
zur Durchführung des autonomen Gütertransportes im Produktionsumfeld 259
Jie Zhang, Xiaobo Liu-Henke

Parameter-Optimierung eines Brake-by-Wire-Pedals 267
Jennifer Werner, Martin Düsing, Bernhard Bachmann, Ali Kemal Kücükayvuz

Simulation in der Energietechnik

Modelling and Simulation of All-Electric Machines and Renewable Electric
Power Systems in Agricultural Operations 273
Felix Klabunde, Christian Reinhold, Bernd Engel

Simulationsbasierte Dimensionierung von Regeneratoren für eine volatile
Hochtemperatur-Abwärmeverstromung 281
Wolfgang Schlüter, Jack Hanna, Konstantin Zacharias

Simulation eines MPR-basierten Energiemanagementsystems 289
Sebastian Schwarz, Andreas Rehkopf

Vorgehensmodell zur Simulation von gebündeltem Energiebedarf 295
Benjamin Jacobsen, Maximilian Stange

Simulation in der Elektronikentwicklung

Anwendung von Batterieanalysemethoden zur Validierung von Alterungsmodellen
in Lithium-Ionen-Zellen 303
Steffen Bazlen, Markus Blessing, Walter Commerell

Einsatz von Simulationen beim Entwurf leistungselektronischer Systeme 313
Robert Rohn, Thorben Schobre, Günter Tareilus, Regina Mallwitz

Lifetime modelling of electrical machines using the methodology of design of experiments 319
Lucas Vincent Hanisch, Markus Henke

Simulation in Produktion und Logistik I

Automatische Erstellung von digitalen Simulationszwillingen von Produktionssystemen <i>Walter Wincheringer, Tobias Sohny, Marec Kexel</i>	327
Ein simulationsbasiertes Optimierungssystem zur Priorisierung von Maschinenstillständen unter Einbeziehung eines Lookahead <i>Michael Hegemann, Stefan Nickel</i>	335
Simulationsgestützte Optimierung des Materialflusses in einem Aluminium-Gussbetrieb <i>Johannes Dettelbacher, Wolfgang Schlüter</i>	343
Potenziale der Ablaufsimulation für die Entwicklung von einer Pilot- zur Volumenfertigung am Beispiel der Heliateg GmbH <i>Felix Diener, Samuel Horler, Pierre Grzona, Philipp Wilsky</i>	349

Simulation in Produktion und Logistik II

Simulation-based Analysis of Dispatching Methods on Seaport Container Terminals <i>Anne Schwientek, Ann-Kathrin Lange, Carlos Jahn</i>	357
Simulation als Bestandteil eines BIM-basierten Vorgehens zur Planung der Baustellenlogistik im Großanlagenbau <i>Jana Stolipin, Ulrich Jessen, Jan Weber, Sigrid Wenzel, Markus König</i>	365
Referenzmodell basierend auf Wertstromsimulation zur Bewertung von Produktionssystemen in der Angebotsphase <i>Markus Rabe, Walter Wincheringer, Tobias Sohny</i>	373

Simulation in Produktion und Logistik III

A Simulation Study on the Performance of Wafer Fabs with Hot Lots Under WIP Balance and Due Date Control Policies <i>Zhugen Zhou, Oliver Rose</i>	381
Generisches Simulationsmodell für automatische Hochregallagersysteme <i>Walter Wincheringer, Marko Sekulic, Marec Kexel</i>	389
Einsatzmöglichkeiten der Rückwärtssimulation zur Produktionsplanung in der Halbleiterfertigung <i>Christoph Laroque, Christoph Löffler, Wolfgang Scholl, Germar Schneider</i>	397
Die Auswirkungen der Stornier- und Umbuchfunktion in Truck Appointment Systemen <i>Katharina Beck, Ann-Kathrin Lange, Carlos Jahn</i>	403

Urban Systems Simulation

- Modelling Urban Transportation Using Tree-Attribute-Matrix Models 411
Kilian Nickel, Daniel Lücknerath, Oliver Ullrich
- „Performance Evaluation of Timed Events in Railways“ in Österreich 421
Alexander Edthofer, Martin Bicher, Felix Breiteneker
- Creating Cloud Simulations for Urban Logistics 427
Richard Pump, Charline von Perbandt, Volker Ahlers, Arne Koschel
- Simulation und Bewertung unterschiedlicher Boarding-Strategien
am Beispiel des Airbus A320 433
Jürgen Wunderlich

Simulation in den Umwelt- und Geowissenschaften

- Modellieren mit Raumbezug: Spezifikation dynamischer Topologien mit den Mitteln
von Graphersetzungs-systemen 439
Jochen Wittmann
- Erreichbarkeitsgraphen als Werkzeug zur Visualisierung des Treibhausgasausstoßes
für die Verkehrsmittel Flugzeug, Auto, Bahn und Reisebus bei der Dienstreiseplanung 447
Malte Christiansen, Jochen Wittmann
- Applying Simulation to Advance Resilience of Historic Areas to Climate Change
and Natural Hazards 453
*Katharina Milde, Sonia Giovinnazzi, Daniel Lücknerath, Oliver Ullrich,
Maurizio Pollino, Erich Rome, Vittorio Rosato*
- Modellierung der Ausbreitung von Baumschädlingen nach aerochemischer
Insektizidanwendung mit den Mitteln eines Geoinformationssystems 461
Colja Krugmann, Majdi Abusaleh, René Krüger, Jochen Wittmann

Simulation und Modellbildung für die Ausbildung

- Modellbildung und Simulation als Grundlagenfach 467
Werner Maurer
- Teaching Application Area Oriented Mathematics in Engineering 473
Andreas Körner, Stefanie Winkler
- Data Science und Lineare Algebra – Didaktisch-Methodische Überlegungen 477
Thomas Schramm
- Heuristische Untersuchung der Abhängigkeit von Übungen und Vorlesungen
für den Prüfungserfolg 481
Corinna Modiz, Franziska Gorgas, Stefanie Winkler, Andreas Körner

Zukunft modellieren: Science Fiction als soziale Simulation von Hyperobjekten

Lars Schmeink¹

¹ HafenCity Universität, Überseeallee 16, 20457 Hamburg; lars.schmeink@hcu-hamburg.de

Abstract. Climate change, pandemics, digitalization – there are many hyperobjects, as Timothy Morton [1] describes them, that are too big and too complex for any one of us to fully grasp. On the one hand, these objects affect all of us, creating a need for “the immanence of thinking to the physical,” while at the same time being so large, complex, and beyond our metalanguage capabilities that they create an “absence of anything meaningfully like a ‘world.’” The language and data of science do not do justice to the personal appropriation of these issues – we are not experiencing r-factors and infection vectors; we are personally affected by living with social distancing measures and the breakdown on social contracts.

So, in order to grasp these hyperobjects and make sense of what the future will hold, the paper argues, we need to move beyond the modeling of pure scientific data and instead turn towards the social, political, and cultural aspects of each of these hyperobjects. We need to simulate the impact of scientific facts onto the fabric of our lives. And for this purpose, fictions are needed that leave behind the data driven empiricism and instead model a new social reality – and the paper argues, this is the purview of science fiction.

Einleitung

Die Zukunft zu modellieren ist nicht einfach. Für bestimmte Fragen scheint dies jedoch mit Hilfe von Simulationen und Modellen besser möglich als für andere. So haben Carl Benedikt Frey und Michael Osborne an der Universität Oxford die Entwicklung von Automatisierung und Digitalisierung im Bereich der Arbeitsprozesse untersucht und Berechnungen erstellt, welche Berufe auch in Zukunft mit hoher Wahrscheinlichkeit von Menschen ausgeübt werden. Aber kaum ein Modell beschäftigt sich damit, welche privaten und persönlichen Konsequenzen diese Entwicklung für die tausenden Betroffenen haben wird und was diese Veränderungen dann gesellschaftlich auslösen. Soziologische Studien können

hier vornehmlich stattfindende oder abgeschlossene Umbrüche dokumentieren, bieten aber kaum die Möglichkeit zukünftige Veränderungen zu antizipieren. An dieser Stelle ist es also hilfreich sich auf die Kultur als Gradmesser von Hoffnungen und Sorgen einer Gesellschaft zu berufen und mit Hilfe der Science Fiction (SF) – eines kulturellen Modus, der sich dezidiert mit technologischem Wandel und dessen sozialen Folgen beschäftigt – mögliche Szenarien der Zukunft zu entwickeln. Für die folgenden Überlegungen nutze ich daher eine für die Natur- und Ingenieurwissenschaften unorthodoxe Auslegung von Modellen, die uns aber dennoch dabei helfen kann, gesellschaftliche Zukunftsbilder begreifbar zu machen. Ich behaupte, die Science Fiction kann uns Modelle für das liefern, was anderweitig nur schwer greifbar ist.¹

Bevor ich aber zur SF und ihren Spezifika komme, möchte ich kurz erläutern, wo die Verbindung von Fiktionalität und Modellierung liegt. Zwei Punkte sind dabei wichtig. Zum einen, dass Kultur Realität abbildet. Zum anderen, dass Fiktionalität nicht gleichzusetzen ist mit Täuschung, Lüge oder Erfindung, sondern eine Beschreibung möglicher Handlungen, Zusammenhänge, oder Zustände darstellt. Zum ersten Punkt: Kultur ist ein Modell unserer Realität. Sie beschreibt, mal mehr, mal weniger detailliert, was potentiell passieren kann. Dabei nutzt sie mitunter auch Abstraktionen, Verkürzungen, oder Verfremdungen. Dies lässt sich beispielsweise daran festmachen, dass wir literarische Verkürzungen mit uns bekannten Mustern auffüllen. Wenn also in einem Roman der Satz steht „Wir gingen zum Abendessen in unser Lieblingsrestaurant“, dann können wir mit einiger Sicherheit trotz des niedrigen Detailgrades und der starken Verkürzung relativ genau sagen, wie der Abend abgelaufen ist – zumindest auf der rein technischen Seite der Handlungen. Im Großen und Ganzen dürfte etwa folgendes passiert sein: Ankunft im Restaurant, Platz nehmen am Tisch,

¹ Teile dieses Beitrags sind in leicht abgewandelter Form bereits erschienen. [2]

Auswahl, Anlieferung und Verzehr der Speisen, Bezahlung der Dienstleistung, Verlassen des Restaurants. Dazwischen gibt es viele Faktoren, die wir nicht genau bestimmen können. Und es gibt mögliche Abweichungen von den wahrscheinlichen Handlungen. Aber weil im Roman keine weiteren Details stehen, gehen wir beim Lesen davon aus, dass Abweichungen und Details nicht relevant für unsere weitere Auseinandersetzung mit dem Abend sind. Wir haben ein Modell des Restaurantbesuchs, abstrakt und detailarm, aber ein Modell.

Ähnlich funktioniert diese Realitätsmodellierung von Kultur in Filmen oder Serien. Wir sehen beispielsweise die Figuren in ein Auto steigen und die Straße entlang aus dem Bild fahren. In einer späteren Szene sehen wir dann dasselbe Auto anhalten und die Figuren steigen wieder aus. Auch hier ist die Abstraktion des Modells recht hoch, die Detaillichte ist gering. Aber doch gehen wir davon aus, dass zwischen beiden Szenen eine Fahrt liegen muss, dass das Auto an Ampeln gehalten hat, beschleunigt und gebremst hat, abgebogen ist, Spuren gewechselt hat – dass also all das passiert ist, was auf dem Weg von A nach B mit einem Auto passiert. Wiederum steht uns ein Modell abstraktes und detailarmes Modell einer Autofahrt zur Verfügung, was uns in wenigen Bildern vermittelt wurde.

Der zweite Punkt – Fiktionen sind nicht mit Täuschungen gleichzusetzen – ist hier auch schon angerissen. Die beschriebenen Figuren mögen nicht real sein, ihre Handlungen aber sind es: Restaurantbesuche und Autofahrten sind Teil unserer Realität, somit in ihrer Abstraktion keine Täuschungen. Weiter noch: Fiktionen bestimmen unser Miteinander. Wir sehen in Fiktionen Muster und Handlungen, die wir dann als Modell für die Realität nehmen und in dem Moment kopieren, da wir keine eigene Erfahrung mit der jeweiligen Handlung haben. Ein Beispiel für dieses Phänomen ist, dass viele Menschen vor deutschen Gerichten auf den „Einspruch“ warten, den es hierzulande in dieser Form nicht gibt, der aber auf US-amerikanischen Fiktionen in unserem Modell einer Gerichtsverhandlung vorkommt. Fiktion bestimmt also zu einem gewissen Grad unsere Erwartung an die Realität.

Die Funktion von Fiktionen unsere wahrgenommene Wirklichkeit möglichst nah abzubilden nennt man in der Literaturwissenschaft Mimesis. Mimetische Darstellungen in Literatur und Film sind das, was man im allgemeinen Sprachgebrauch auch als ‚realistisch‘ ansieht. Das ist oftmals der Fall, wenn etwa Zwischenmenschliches modelliert werden soll – also wie Menschen sich lieben, sich begegnen, aneinander verzweifeln. Oder bei historischen

Ereignissen, die im Nachgang möglichst getreu der Realität nacherzählt werden sollen. Aber was passiert, wenn wir komplexe und ‚ungreifbare‘ Dinge wie den Klimawandel, die Digitalisierung, oder die Globalisierung abzubilden versuchen? Mein Argument ist, dass hierzu die Science Fiction nötig ist. Kultur kann auch diese komplexen Objekte sinnvoll modellieren und so einem Massenpublikum erzählend näherbringen, wenn sie dafür auf die SF als Modus zurückgreift.

Die Literaturwissenschaftlerin Seo-Young Chu [3] argumentiert, dass die Science Fiction besonders leistungsstark darin sei, Welt zu modellieren. Aber SF sei nicht etwa der Gegenpol zur mimetischen Hochliteratur, sondern vielmehr müsse man SF als eine mimetische Form verstehen, die Abbilder bestimmter besonders schwer zu greifender Wirklichkeitsbezüge, sogenannter „cognitively estranging referents“ (5), generieren will. Also Bezüge zu Objekten, die für den Menschen nicht *vollständig kognitiv erfahrbare* sind und nur mit einem gewissen Maß an zusätzlicher kreativer Energie beschrieben werden können. Chu situiert diese im Mittel eines Spektrums zwischen Wissbarem und Unwissbarem (6). Auf der Seite des Wissbaren finden sich Alltagsgegenstände wie Werkzeuge, Essen, Kleidung usw., die eine realistische Literatur mit hoher Präzision mimetisch darzustellen vermag. Auf der Seite des Unwissbaren finden sich Konzepte, die außerhalb der menschlichen Kognition liegen und mimetisch überhaupt nicht darstellbar sind, wie etwa die Zeit des Urknalls oder Erfahrungen jenseits des Todes (7). In der Mitte dieses Spektrums aber finden sich Objekte, die „kognitiv verfremdend“ sind – erkennbar, allerdings kaum erfahrbare: „the sublime (e.g., outer space), virtual entities (cyberspace), realities imperceptible to the human brain (the fourth dimension), phenomena whose historical contexts have not yet been fully realized (robot rights), and events so overwhelming that they escape immediate experience (shell shock)“ (7). Die Bezugnahme auf diese Objekte in der Wirklichkeit ist nicht vollständig möglich und so bedürfen sie, wenn man so will, einer höheren literarischen Energie für eine mimetische Annäherung.

Die SF sorgt also dort für eine Annäherung an mimetische Repräsentation, wo die komplexen Wirklichkeitsbezüge für die Leser*in sonst nicht erfahrbare wären. Ich würde sogar so weit gehen, zu behaupten, dass viele Aspekte unserer aktuellen Lebensrealität solche Komplexität aufweisen und eben einer SF-mimetischen Darstellung bedürfen. Chu führt beispielsweise das Leben in ei-

ner stark medialisierten Welt an, in der die Grenzen zwischen virtueller und materieller Existenz dank allgegenwärtiger, digitaler Technologien verschwimmen (9). Und die schwer durchdringlichen ökonomisch-politischen Realitäten der spätkapitalistischen Weltwirtschaft mit ihren „too big to fail“-Banken, digitalen Monopolisten und globalisierten Märkten scheinen ebenfalls eine ideale Ausdrucksform in der SF zu finden. Im Folgenden möchte ich daher einige Beispiele solcher SF-nahen Objekte diskutieren. Im heutigen politischen Alltag lässt sich gut beobachten, wie die Komplexität einer globalisierten Welt selbst für Experten undurchschaubar geworden ist. Gerade in der aktuellen Corona-Situation zeigt das Fehlen von Auffangmechanismen und Krisenplänen, wie wenig tiefgehendes Verständnis von Marktzusammenhängen in Wirtschaft und Politik vorhanden ist. Die Komplexitäten des globalen Kapitalismus haben sich durch die Krisensituation als unwägbare erwiesen, besonders weil plötzliche Rückfälle in Zeiten strikter Grenzkontrollen und eingeschränkter Freizügigkeit in keinem Zukunftsszenario verlässlich berechnet worden sind. Und auch gesellschaftlich gesehen sind die Auswirkungen der Globalisierung komplex und überall, wenn auch diffus, spürbar – Automatisierung und Digitalisierung lassen auch in der bislang verschonten westlichen Welt die bestehenden Sicherheiten zusammenbrechen. Die Pandemie hat diesen Entwicklungen nur weiter Vorschub geleistet. Migrationsbewegungen und ökologische Veränderungen werden in Zukunft zu den wirtschaftlichen Prozessen hinzuaddiert werden müssen und lassen die Globalisierung so dem werden, was Timothy Morton [1] als Hyperobjekt bezeichnet.

Morton sieht Hyperobjekte als Objekte, die im Maßstab menschlicher Erfahrungen zu groß und zu weitreichend sind, um vollständig erfasst werden zu können (1). Er zählt beispielsweise das Sonnensystem, die Menge allen nuklearen Materials auf der Erde, die Lebensdauer von Styropor oder die globale Erderwärmung dazu. Ich möchte dazu auch die Globalisierung ergänzen, die Prozesse der Digitalisierung und deren Einfluss auf die Arbeitsmärkte, den Klimawandel, oder eben eine Pandemie wie Corona. Hyperobjekte sind, laut Morton, „viscous“ (also klebrig oder zähflüssig) und man kann ihnen nicht aus dem Weg gehen. Hyperobjekte lassen sich nicht ignorieren: „Every attempt to pull myself free by some act of cognition renders me more hopelessly stuck to hyperobjects“ (29). Wir müssen uns mit diesen komplexen und allgegenwärtigen Themen beschäftigen. Aber sie lassen

sich nicht gut beschreiben, nicht in klar greifbare Kategorien packen. Naturwissenschaftliche Modelle können einzelne Aspekte solcher Hyperobjekte visualisieren, aber man braucht spezielles Wissen, um die Fülle an Daten in Zusammenhang zu bringen und korrekt zu interpretieren. Die Wissenschaftskommunikation ist täglich damit beschäftigt, Narrative zu präsentieren, Beispiele zu liefern und für die Öffentlichkeit neue Ankerpunkte zu schaffen, um Befunde und Erkenntnisse über Hyperobjekte zu vermitteln. Doch wie sehr solche datenbasierte Vermittlung scheitern kann, lässt sich aktuell an der weltweit wachsenden Zahl der Corona-Leugner erkennen. Die Science Fiction macht jedoch mit Hilfe von Literatur, Film und Fernsehen Hyperobjekte für die Allgemeinheit erfahrbar, wie ich nun an einigen Beispielen aufzeigen möchte.

1 Globalisierung: William Gibsons *The Peripheral* und *Agency*

In seinen aktuellen Romanen *The Peripheral* [4] und *Agency* [5] beschreibt William Gibson das Hyperobjekt der Globalisierung, indem er dessen hyperbolisch erscheinende Komplexität literarisch vereinfacht und die Welt buchstäblich zu einem Spielball der Mächtigen macht. *The Peripheral* beschreibt zwei Zukunftsperspektiven: die eine in der nahen Zukunft, in einer ländlichen Gegend in den USA. Hier arbeitet die Protagonistin Flynn in einer von der Digitalisierung geprägten Welt für eine kolumbianische Firma als Sicherheitskraft. Doch statt in physischer Präsenz muss Flynn ihre Schutzobjekte per virtueller Realität überwachen. Ihre Arbeit gleicht einem Computerspiel, und als in dieser virtuellen Welt ein Mord geschieht, den Flynn beobachtet, gerät ihr Leben ganz real aus den Fugen.

In der anderen Zukunftsperspektive, mehr als 70 Jahre später, hat sich das weltweite System der Märkte komplett verändert. Ein globaler Zusammenbruch (sowohl ökologisch als auch ökonomisch) hat die Menschheit dezimiert; Fragmente der bestehenden Welt konnten mit Hilfe von Nanotechnologie neu aufgebaut werden, aber der Verlust der Erde als natürliche Ressource ist deutlich spürbar. In einem künstlich wiedererschaffenen London des 22. Jahrhunderts lebt eine reiche Elite, die vor allem mit Machtspielen und politischer Einflussnahme beschäftigt ist. Arbeitskraft, die für den Erhalt ihres Lebensstandards notwendig ist, ist entweder automatisiert oder wird aus der Vergangenheit eingekauft und

virtuell erbracht. Mittels Quantencomputern ist es technologisch möglich, sogenannte Kontinua zu erschaffen, Abzweigungen möglicher Welten, in denen die Zukunft die Vergangenheit beeinflusst und nach Belieben manipuliert. Flynn ist Teil einer solchen alternativen Zeitlinie, die von den reichen Eliten als Spielball für ihre Intrigen genutzt wird. Der Mord, den sie in der virtuellen Realität gesehen hat, ist im London des 22. Jahrhunderts real geschehen.

Auch in *Agency* geht es um diese alternativ-geschichtlichen Welten. Wiederum wird ein solcher Stub (also Stummel) – wie die Zukunft des 22. Jahrhunderts aus privilegierter Sicht die abweichenden Zeitlinien nennt – von zukünftigen Eliten als Experimentierfeld genutzt. Die Welt des Stubs steht kurz vor dem Atomkrieg, den Präsidentin Hillary Clinton mit allen Mitteln zu verhindern versucht. Der Stub gilt als wichtig, weil er die am weitesten zurückreichende Abweichung von der Zeitlinie darstellt. In ihm könnte sich zeigen, ob es möglich gewesen wäre, dass der ökonomische und ökologische Kollaps der Welt hätte verhindert werden können. Deswegen nutzen die Mächtigen der Zukunft eine militärisch entwickelte KI, um die Machtverhältnisse der Vergangenheit zu stabilisieren und den Krieg zu verhindern. Dabei wird aber vor allem deutlich, dass die Manipulation der Vergangenheit eine nostalgische Beschäftigung ist: also der Versuch, die in der eigenen Zeitlinie katastrophal zerstörten Systeme in anderen Zeitlinien zu erhalten und den Kollaps zu verhindern. Ein bestimmter Teil der Zukunft trauert dem heutigen Machtsystem, den freien Märkten und dem technologischen Erfindergeist hinterher – aus Sicht Gibsons haben wir heute das Potential, noch alles zum Besseren zu kehren.

Die Kontinua repräsentieren mimetisch das Hyperobjekt der Globalisierung in den Romanen. Die Mächtigen des 22. Jahrhunderts vermögen aufgrund ihres technologischen Fortschritts in der Zeit zurückzureichen und die Menschen dort zu instrumentalisieren. Dass in den Romanen die Nutzung von Kontinua als dekadente Subkultur von Oligarchen und Aristokraten aufgezeigt wird, die alternative Zeitlinien zur Unterhaltung oder Ausbeutung generieren, lässt tief blicken. Die Lebensbedingungen in den Stubs sind nicht von Bedeutung; die Menschen sind Figuren in einer virtuell erlebten ökonomischen Simulation. Das Leben in einem Stub wird von allwissenden und zu allem fähigen Mächten kontrolliert – ein Verständnis der eigenen Situation bleibt den Menschen verwehrt. Der Stub hat in etwa den Stellenwert eines komplexen wirtschaftlichen, gesellschaftlichen Computerspiels. Seine

Welt hat keinen Nutzen in direktem Bezug auf die ihn generierende Realität des 22. Jahrhunderts – Konsequenzen eingreifender Handlungen verbleiben im Stub und erreichen nicht deren Erschaffer.

Bezeichnend ist hierbei, dass die Machtausübung auf die Vergangenheit, wie auch der Austausch von Informationen zurück in die Zukunft sich einzig auf digitale Transaktionen beschränkt. Angesichts der fortgeschrittenen Digitalisierung, sowohl in Hinsicht auf finanzielle aber auch auf materielle Prozesse, ist dies aber kein Hindernis für direkte Einflussnahme. Politische Gefälligkeiten lassen sich in der Vergangenheit ebenso digital erkaufen wie wirtschaftliche Macht (etwa durch Technologiepatente) oder materielle Objekte – neue Waffen können z.B. dank 3D-Druckern durch die Zeit gelangen.

Der von Gibson in den Romanen beschriebene Gegenstand ist nicht die lokale Ausprägung von Globalisierung, wie sie eine realistische Literatur darzustellen vermag. Statt ihre Auswirkungen in den Schicksalen einer durchschnittlichen, amerikanischen Familie aufzuzeigen, widmet sich Gibson der für den Einzelnen verborgenen Struktur der Globalisierung. Tatsächlich verweist Morton [1] darauf, dass Hyperobjekte „*non-local*“ sind und nicht mittels ihrer „*local manifestations*“ (1) begriffen werden können. Es bedarf also der Konstruktion von Kontinua als Miniaturwelten, um die systemische Ausbeutung ‚verschwendeter Leben‘, wie der Soziologe Zygmunt Bauman [6] sie nennt, in der Globalisierung darzustellen. Was Bauman in seiner Studie auf Länder des globalen Südens bezieht und als „*unintended and unplanned 'collateral casualties' of economic progress*“ (39) bezeichnet, trifft auch heute schon auf Teile der Bevölkerung der USA oder Europas zu, die als Kollateralschäden einer Globalwirtschaft von einer gesellschaftlichen Beteiligung ausgeschlossen sind. Das SF-Erzählen Gibsons ermöglicht hier also das abgeschieden ländlich verarmte Leben (etwa in West Virginia mit seinen geschlossenen Kohleminen, oder im Rust Belt, wo die Stahlwerke nicht mehr konkurrenzfähig sind) mit seinen ökonomischen Realitäten in die Machtstrukturen einer globalen Wirtschaft einzuordnen. Es erlaubt ein Modell zu erstellen, dass das menschenverachtende Profitstreben heutigen Handelns aufzeigt. Und es verdeutlicht dessen transformativen Einfluss, sei es durch globale Migrationsbewegungen oder in dem sozialen Aufbegehren abgehängter Teile westlicher Gesellschaften.

2 Klimawandel und soziale Ungleichheit: Neill Blomkamps *Elysium*

Natürlich muss nicht immer ein einzelnes Hyperobjekt so zentral von einem SF-Text modelliert werden. In einigen Fällen bestimmen die Hyperobjekte den Hintergrund vor dem ein Text spielt, in anderen sind verschiedene Objekte abgebildet und verbinden sich zu einem komplexen Geflecht. So verhandelt der Film *Elysium* von Neill Blomkamp [7] diverse Hyperobjekte und deren gemeinsame Wirkung auf die Menschen. Der Film spielt im Jahr 2154 und zeigt schon in seiner Anfangssequenz die Folgen heutiger gesellschaftlicher Herausforderungen.

Die den Film eröffnende Kamerafahrt führt über ein Flussgebiet, das mit Müll verseucht ist und wie eine Industriebrache wirkt: eine unwirtliche Landschaft, die thematisch bereits in den ersten Sekunden auf Umweltverschmutzungen hinweist. Es folgt ein Überflug über ein Stadtgebiet mit dicht gedrängten Gebäuden, die in keinem guten Zustand sind und eher ärmlich wirken. Die Einblendung eines Zwischentitels bestätigt die Themen, die hier verhandelt werden: die Welt ist verseucht, verschmutzt, und leidet an der Überbevölkerung. Im Hintergrund des Stadtbildes steigen schwarze Rauchsäulen auf. Das typisch aufgeräumte Grid amerikanischer Städte ist einem urbanen Chaos zusammengewürfelter Häuser gewichen. Es sind Bilder, die westliche Zuschauer*innen aus den Slums des globalen Südens kennen. Mit einem Schnitt wechselt die Szene zu Hochhäusern, zerfallen von der Zeit, zerfressen von den Umweltgiften, überwuchert, rauchend. Man erkennt die Skyline von Los Angeles mit der runden Form des US Bank Towers in der Mitte und daneben den markanten Stufen des Citibank Towers. Danach folgt ein Schnitt zur Erdsicht aus dem All und ein weiterer Zwischentitel mit der Einblendung, dass die reichsten Bürger der Erde sich vom Planeten abgewendet und ihre Art zu leben auf einer Raumstation erhalten haben. Die Raumstation Elysium ist der Rückzugsort aller, die sich die Bürgerschaft dort leisten können. Die Kamerafahrt durch die Ringstation zeigt grüne Gärten, klare Seen und große Villen, die reichlich Platz bieten. Neben der Umweltverschmutzung steht also vor allem auch die Ungleichheit des sozialen Miteinanders im Mittelpunkt des Films. Schon in dieser Eröffnungssequenz sehen wir die Extrapolation dessen, was heute dank segregierter Bebauung und der Institutionalisierung von Gated Communities Realität ist. Die Folgen einer durch Industrialisierung ausgelösten Umweltzerstörung und des Klimawandels tragen nicht alle Menschen gleichermaßen,

sondern vor allem die Nicht-Privilegierten. Reichtum und Macht helfen dabei, diesen Konsequenzen zu entgehen.

Die Handlung des Films stellt diesen Konflikt zwischen den machtlosen Massen auf der Erde und den wenigen Mächtigen in Elysium in den Mittelpunkt, konzentriert sich aber nicht auf politische Aussagen zur Demografie oder die ungleiche Wohlstandsverteilung. Viel mehr zeigt der Film eine actionlastige Handlung um den Tagelöhner Max (Matt Damon), der nach einem Arbeitsunfall einen Weg nach Elysium sucht, um sich dort – dank besserer medizinischer Versorgung – von einer tödlichen Strahlendosis heilen zu lassen. Dabei sind die sozial-kritischen Aussagen des Films deutlich spürbar und modellieren ein Bild der Zukunft, das sich klar benennen lässt. So sind schon das Setting und seine Darstellung aufschlussreich. Die den Film eröffnenden Bilder von Los Angeles stehen in Kontrast zu dem Glamour, mit dem das Publikum die Pazifikmetropole sonst im Rahmen von Filmen verbindet. Die Darstellung der Stadt wirkt farblich verwaschen, staubig, in Erdtönen gehalten und in klarem Widerspruch zu den kontrastreichen Blau- und Grüntönen Elysiums. Das Los Angeles des Jahres 2154 leidet unter Dürre, Verschmutzung, und massiver Hitze – was Filmemacher Blomkamp bildlich umsetzt, in dem die Szene in Mexiko Stadt gedreht hat. In dieser filmisch-geografischen Verschiebung liegt die Erkenntnis, dass der Klimawandel vor dem Reichtum des heutigen L.A. keinen Halt machen wird und die Mojave-Wüste weiter nach Westen vorrücken wird.

Mit dem Klimawandel verbunden ist die Überbevölkerung, die sich im Film auch konkret in der Frage um Arbeit und Lebensunterhalt zeigt. So ist Max ein Tagelöhner, der einfache, ungelernte Tätigkeiten in einer Fabrik verrichtet und sich damit schon zu den wenigen Glücklichen schätzen kann, die überhaupt einen Job haben. Die meisten Tätigkeiten sind automatisiert worden und Menschen sind nur dort noch tätig, wo sie billiger einzusetzen sind als Maschinen. Der Soziologe Peter Frase [8] argumentiert: „the existence of a large pool of unemployed and low-wage workers operates as a disincentive for employers to automate. After all, why replace a worker with a robot, if the worker is cheaper?“ (13). In der Welt von *Elysium* bedeutet dies, dass der Überhang an menschlicher Arbeitskraft nichts wert ist und Arbeiter*innen jegliche Rechte verloren haben. Erscheint Max nicht zur Arbeit oder tut nicht das, was man von ihm verlangt, gibt es jederzeit tausende Anderer, die seinen Job machen würden. Und so wird Max vom Vorarbeiter

in die Strahlenkammer geschickt, um eine Fehlfunktion zu richten, dort durch die Automatik eingeschlossen und mit einer tödlichen Dosis bestrahlt. Das Unternehmen sieht keine Verantwortung für ihn und ersetzt seine Arbeitskraft sofort. Er wird mit Hospiz-Medikation zum Sterben nach Hause geschickt. Sein Leben ist nur etwas wert, wenn er dafür arbeiten kann. Dabei gäbe es für die Mächtigen in Elysium sehr wohl medizinische Versorgung, die Max retten könnte – nur hat er in der stratifizierten Gesellschaft kein Anrecht auf eine solche Behandlung.

Der Film folgt nun Max' Bestrebungen nach Elysium zu gelangen und sich zu heilen. Dabei wird er zum Revolutionär, dem es gelingt, die Abschottung Elysiums zu durchbrechen und mit einem Sicherheitscode die Algorithmen der automatisierten Versorgung Elysiums neu zu programmieren. Durch eine Änderung im System gelingt es Max, die Rechte auf Sicherheit, Gesundheit, und Grundversorgung auf alle Menschen auf der Erde zu übertragen. Die automatisierten Systeme erkennen die Not der Erdbewohner und senden Hilfstransporte aus – ein gleichberechtigtes und vor allem grundversorgtes Leben ist möglich. Nicht ein Mangel an Ressourcen ist also der Grund für die soziale Spaltung, vielmehr ist diese politisch von den Mächtigen Elysiums gewollt, um nicht an Status zu verlieren.

Für Frase [8] stellt diese Vision einer, trotz ausreichender Ressourcen für alle, immer noch hochgradig ungleichen Gesellschaft einen möglichen Extrempunkt unserer sozialen, politischen und ökonomischen Zukunftsentwicklung dar. Er sieht darin die konsequente Weiterführung heutigen Machtdenkens: „To the extent that the rich are able to maintain their power, we will live in a world where they enjoy the benefits of automated production, while the rest of us pay the costs of ecological destruction—if we can survive at all“ (29). Mehr noch, in der logischen Konsequenz einer automatisiert betriebenen Welt, benötigen die Reichen die Armen nicht mal mehr für ihre Arbeitskraft oder ihren Konsum – Frase nennt diese Zukunftsvision „Exterminism“ (29), weil sie den Tod des größten Teils der Weltbevölkerung billigend in Kauf nimmt oder gar absichtlich vorantreibt.

In einem Film wie *Elysium* sehen wir also mögliche Szenarien einer Entwicklung gleich mehrerer Hyperobjekte – das Voranschreiten des Klimawandels, das ein Leben auf dem Planeten einschränkt und soziale Probleme verschärft; die hohe soziale Ungerechtigkeit durch die Verteilung von Geld und Macht an wenige, die damit alle politischen Entscheidungen beeinflussen können und

sich so von anderen Problemen freikaufen; und die Automatisierung von Arbeit, die ohne soziale Ausgleichsmaßnahmen die Lebensgrundlage von Millionen Menschen in Frage stellt. All diese Themen sind für uns nur schwer greifbare Hyperobjekte, aber im Film werden sie in der Darstellung von Welt modelliert und somit am Schicksal der Figuren deutlich erlebbar.

3 Pandemie: Von *28 Weeks Later* bis *Code 46*

Aber wenden wir uns doch noch mal dem Thema zu, dass uns alle in den letzten Monaten so stark beschäftigt hat und für dessen Einfluss auf uns als Gesellschaft wir bis heute keine guten Modelle haben. Oder? Ich würde zum Abschluss meiner Überlegungen provokativ behaupten, dass wir alle uns auf Corona und die daraus folgenden sozialen Aufwühlungen hätten vorbereiten können, wenn wir bei Zombie-Fiktionen besser aufgepasst hätten. Dabei geht es mir nicht um den offensichtlichen Unterschied, dass Corona zum Glück keine Toten wieder auferstehen lässt. Dennoch sind in Zombie-Fiktionen soziale Verhaltensmuster aufgegriffen worden, die uns in den Wochen des Lockdowns und während der anschließenden Lockerungen begegnet sind.

Dass Zombies ein ideales Vehikel sind, um über Maßnahmen des Seuchenschutzes aufzuklären, bewiesen die US-amerikanischen Centers for Disease Control and Prevention [9], als sie Infomaterial zur „Zombie Apocalypse“ herstellten und damit auf eine „Preparedness“ der US-Bevölkerung hinarbeiteten. Eine der Überschriften der Website lautet „Never Fear – CDC is Ready“ und versucht herauszustellen, wie wichtig die wissenschaftliche Expertise und die klare Umsetzung der Seuchenbekämpfungspläne durch die CDC ist [10]. Angesichts der heutigen Situation einer massiven Einschränkung und Manipulation der Arbeit der CDC aus politischen oder ökonomischen Gründen durch das Weiße Haus ist dies natürlich höchst ironisch.

Wer aber Filme wie *28 Weeks Later* [11] oder *Resident Evil: Apocalypse* [12] gesehen hat weiß, dass auch eine Pandemie sich machtpolitisch nutzen lässt. In *28 Weeks Later* entscheidet die NATO, obwohl noch nicht viel über das in Großbritannien ausgebrochenen Virus bekannt ist und keineswegs alle Infektionsherde unter Kontrolle sind, die britischen Inseln wieder neu zu besiedeln, um das politische Signal zu geben, man habe das Virus im Griff. Und in *Resident Evil: Apocalypse* schottet

die global agierende Umbrella Corporation eine ganze Stadt ab, um dann in einem kontrollierten Versuch die Ausbreitung des Virus und dessen Wirkung wissenschaftlich beobachten zu können. Mögliche Parallelen zu politischen Szenarien der weltweiten Corona-Pandemiebekämpfung sind zufällig: etwa, dass die britische Regierung Studierende im Herbst 2020 wieder an die Universitäten beordert hat, um der angeschlagenen Hochschul-Wirtschaft zu sicheren Mieteinnahmen zu verhelfen oder die CSU in München die Wirtschafts-Wiesn‘ erfunden hat, um der darbenenden Gastronomie zu helfen und Arbeitsplätze zu sichern. Eine Pandemie kann von bestehenden Machtstrukturen missbraucht werden, genau wie es in Zombie-Fiktionen dargestellt wird.

Doch auch auf persönlicher, privater Ebene zeigen Zombie-Fiktionen die Tücken einer Pandemie auf. Ist doch ein wesentlicher Bestandteil vieler Zombie-Erzählungen, das Motiv des „unerkannten Biss“. Es geht dabei um Familienmitglieder oder Freunde, die sich angesteckt haben, aber ihre Erkrankung ignorieren oder verheimlichen. In einer Variation auch gerne, dass Partner von der Erkrankung wissen, aber allen anderen Überlebenden Normalität vorspielen, um nicht die unvermeidlichen Konsequenzen ziehen zu müssen. In der Zombie-Fiktion wäre dies die unweigerliche Verwandlung der infizierten Person in einen Zombie. Das anzuratende Gegenmittel die sofortige Tötung der Person. Eine Pandemie, oder eben ein Zombieausbruch, birgt also das Problem, dass Kranke und Gesunde sich zumindest eine gefährliche Zeit lang nicht voneinander unterscheiden. Der bereits kranke Mitmensch wird zum Risiko, ohne dass man ihn oder sie erkennen könnte. Und ein willentliches Ignorieren von Krankheitssymptomen wird zum Risiko für alle in der Umgebung. Schon im Klassiker des Genres – *The Night of the Living Dead* [13] – findet sich dieses Motiv in der Figur der jungen Karen (Kyra Schon), die von den Zombies gebissen wurde und mit ihren Eltern im Keller eines Farmhauses ausharrt. Statt Karen zu isolieren (oder direkt zu töten), versucht die Familie sie zu pflegen und vor weiterem Schaden zu bewahren. Als sie stirbt und zum Zombie wird, wird der Keller zur Falle und Karen zum Zombie-Kind, das seine Eltern angreift und tötet.

Und letztlich zeigen Zombie-Fiktionen auch den rechtlichen Druck, den wir als Gesellschaft in einer solchen Ausnahmesituation regulieren müssen. Unser politisches System wird auf eine Probe gestellt, eine Art Sonderrecht für die Katastrophe auszurufen. Denn eine Gesellschaft, in der hochansteckende Personen sind, muss

diesen Personen ihre Rechte nehmen, sie in ihrer Bewegung einschränken und sie unter Zwänge stellen – sie also etwa zwingen, Masken zu tragen oder sich nicht in größeren Gruppen zu versammeln. Dabei stellen Zombiefiktionen unsere grundlegenden Strukturen des Zusammenlebens in Frage, was die Parodie *Shaun of the Dead* [14] wunderbar veranschaulicht. Obwohl gerade Orte, an denen sich viele Menschen versammeln vermieden werden sollten, will der Angestellte Shaun (Simon Pegg) nicht von seinem Plan abweichen, die Zombie-Apokalypse möglichst ungestört in seinem Lieblingspub zu verbringen. Das Zitat im Film lautet: „Take car. Go to mum’s. Kill Phil, grab Liz, go to the Winchester, have a nice cold pint, and wait for all of this to blow over. How’s that for a slice of fried gold?“ Natürlich ignoriert dieser Plan vollkommen das Risiko sich anzustecken und in der Tat geschieht bei der Umsetzung genau das: Shauns bester Freund Ed (Nick Frost) wird gebissen und verwandelt sich in einen Zombie. Spannend dabei ist, dass der Film einen Weg aufzeigt, wie man auf Dauer mit der Pandemie umgeht. Am Ende des Films sehen wir Shaun und Ed gemeinsam auf der Couch sitzen und weiterhin Videospiele spielen, nur das Ed jetzt ein Zombie und in Ketten gelegt ist. Selbst die Erkrankung Eds hält Shaun nicht davon ab wieder in die exakt selbe Muster zurück zu fallen, die er vor der Pandemie für normal erachtet hat. Vielleicht liegt in diesem stoischen Wunsch nach Normalität ja auch ein Hoffnungsschimmer.

Die Alternative zu einem „trotz Pandemie zurück zum alten Normal“ wäre ein „wegen der Pandemie ein neues Normal definieren“ – und hier möchte ich noch ein finales Beispiel anführen, das den Zombies den Rücken kehrt. Der Film *Code 46* [15] zeigt uns eindrücklich, wie ein neues Normal aussehen könnte, das mit erhöhtem Risiko für Viren und andere Krankheiten einhergeht. Eigentlich eine Liebesgeschichte in unsicheren Zeiten, zeigt der Film eine Welt auf, in der so starke gesundheitliche Risiken existieren, dass die Bewegungsfreiheit der Menschen stark eingeschränkt werden muss. Zum einen ist der Klimawandel massiv fortgeschritten und nur wenige Gebiete der Erde sind noch unproblematisch bewohnbar. Zum anderen existieren aber auch eine Vielzahl von Umweltgiften, Infektionen und neuen, genetisch bedingten Erbkrankheiten, die eine freie Bewegung der Menschen unmöglich machen. Wer von einer sicheren Enklave zur nächsten reisen möchte, der benötigt von einer globalen Versicherungsgesellschaft ausgestellte „Papellen“, die zugleich als Visum und als Nachweis einer

Risikobewertung gelten: Reiseversicherung und Aufenthaltsgenehmigung in einem. Angesichts von Reisewarnungen, Risikogebieten in der Pandemie und verpflichtenden Covid-Tests und Quarantäne-Maßnahmen nach Reisen erscheint diese Zukunftsvision heutzutage deutlich näher als noch vor einem Jahr. *Code 46* verbindet diese neue Form der Risiko-Beurteilung mit einer globalen Wirtschaftselite, für die Mobilität zum Zeichen einer Klassenzugehörigkeit wird. Die Ungleichheit zwischen Arm und Reich verdeutlicht sich an der Leichtigkeit, mit der Impfungen und Antigene für die richtige Summe erstanden werden können. *Code 46* könnte uns ein Modell dafür liefern, wie sich globale Mobilität in der Zukunft entwickelt, wie das Recht auf Freizügigkeit eingeschränkt werden wird, sollte sich ein Impfstoff doch als weniger effektiv herausstellen, als hoffnungsvoll angenommen wird.

Und damit möchte ich meine Überlegungen abschließen. Die hier vorgestellten Modelle sind weniger eindeutig, als die typischen Arbeiten im Rahmen der ASIM. Aber ich hoffe dennoch, aufgezeigt zu haben, dass bestimmte globale, komplexe System nicht das reine Modellieren von Daten für jeden sichtbar und erfahrbar werden; dass bestimmte Themen zu unberechenbar sind; und dass wir Modelle brauchen, die den Menschen näher rücken, die die Distanz zu Zahlen und Fakten überbrücken. Ich hoffe, dass die Science Fiction ihren Beitrag dazu leisten kann, dass wir Narrative finden, um wissenschaftliche Erkenntnisse verständlich zu verpacken.

References

- [1] Morton, T. *Hyperobjects: Philosophy and Ecology after the End of the World*. Minneapolis: U of Minnesota P, 2013.
- [2] Schmeink, L. Repräsentation des Nicht-Erfahrbaren. *Neue Rundschau*. 2019 (1): 46-56.
- [3] Chu, S.-Y. *Do Metaphors Dream of Literal Sleep? A Science-Fictional Theory of Representation*. Cambridge: Harvard UP, 2010.
- [4] Gibson, W. *The Peripheral*. New York: Berkley, 2014.
- [5] Gibson, W. *Agency*. New York: Berkley, 2020.
- [6] Bauman, Z. *Wasted Lives: Modernity and its Outcasts*. Cambridge: Polity, 2004.
- [7] *Elysium*. Regie: N. Blomkamp. Sony Pictures, 2013.
- [8] Frase, P. *Four Futures: Visions of the World After Capitalism*. Cambridge: Verso, 2016.
- [9] Centers for Disease Control and Prevention. "Zombie Preparedness" Web. 15.10.2020. <https://www.cdc.gov/cpr/zombie/index.htm>.
- [10] Khan, A. S. „Preparedness 101: Zombie Apocalypse.“ *Centers for Disease Control and Prevention*. 16.05.2011. Web. 15.10.2020. <https://blogs.cdc.gov/publichealthmatters/2011/05/preparedness-101-zombie-apocalypse/>
- [11] *28 Weeks Later*. Regie: Juan Carlos Fresnadillo. 20th Century Fox, 2007.
- [12] *Resident Evil: Apocalypse*. Regie: Alexander Witt. Constantin Film, 2004.
- [13] *The Night of the Living Dead*. Regie: George A. Romero. Continental Distributing, 1968.
- [14] *Shaun of the Dead*. Regie: Edgar Wright. Universal Pictures, 2004.
- [15] *Code 46*. Regie: Michael Winterbottom. BBC Films, 2003.

Modeling of Non-standard Queuing Policies - An Invitation to ARGESIM Benchmark C22

Peter Junglas^{1*}, Thorsten Pawletta²

¹Dep. of Engineering "Dr. Jürgen Ulderup", PHWT Vechta/Diepholz, Schlesierstr. 13a, 49356 Diepholz, Germany;
*peter@peter-junglas.de

²Wismar Univ. of Applied Sciences, Fac. of Engineering, Research Group CEA, PF 1210, 23952 Wismar, Germany

Abstract. The recently published ARGESIM benchmark C22 'Non-standard Queuing Policies' studies three queuing models, where the queues utilize more complex policies than the standards FIFO, LIFO or priority. *Jockeying queues* allow entities to switch to a shorter queue, in *reneging queues* entities leave a queue after a maximal waiting time, and *classing queues* change the entity order according to an entity attribute ("class") at the call of an external operator.

To encourage the simulation community to publish benchmark solutions this talk will explain the tasks in some detail and comment on the lessons learned from a first implementation.

Introduction

The general transition of modeling from programming or text-based modeling languages to graphical component-based methods has made possible to study highly complex models without explicit programming knowledge on the side of the modeler. On the other hand standard text books like [1] are still using explicit C code for the description of their models and algorithms – and for good reason! In standard modeling cases the set of prebuilt components is often sufficient. But for special situations it can be difficult to work around the limitations of the given building blocks. Furthermore the exact behaviour of the components is generally not defined in every detail. This can lead to all kinds of problems, ranging from awkward workarounds to unexpected behaviour [2].

Different approaches can be used to cope with this problem: One can supplement the components with an interface that allows the easy integration of callback functions or one can provide means to integrate self-programmed components. Both ways have been offered in the redesign of MathWorks' SimEvents library [3]. More in the spirit of the graphical programming paradigm, one could try to identify a set of basic components with a precisely defined behaviour, that allow

to model non-standard situations in a clear and easily understandable way.

The ARGESIM benchmark C22 'Non-standard Queuing Policies' [4] addresses this problem for the modeling of queuing systems. To this end it defines three different tasks, where a standard queuing component is not sufficient, either because some entities can leave the queue prematurely or because their queuing order can be changed dynamically. The prospective solutions will help to clarify how one can deal with such situations using common graphical modeling tools. A first implementation using the text-based MatlabGPSS language [5] gives some clues, which features could be helpful and which are still missing (in MatlabGPSS) – and it points out, where graphical tools still have problems, that should be addressed in future modeling environments.

1 Basic Queuing System

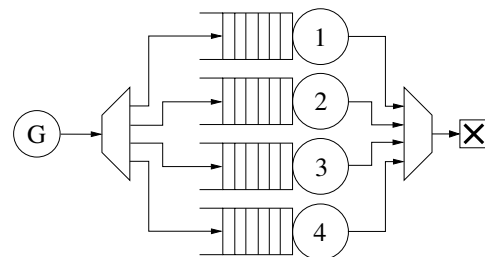


Figure 1: Basic queueing system with four queues.

A simple queueing system consisting of four queues and servers is used as a starting point and reference for all benchmark models (cf. fig. 1). It consists of a generator that creates a given number of entities with fixed or stochastic interarrival times and four FIFO queues and corresponding servers with a capacity of one. Entities

choose the shortest line (including the server allocation) and leave the system after being served. The simulation stops after all entities have left.

Two different versions have to be modelled: a small one with fixed interarrival and service times, a larger one with stochastic values. All details, such as the total number of entities and values or distributions of inter-arrival and service times, are given in the benchmark definition [4], together with the required output values and plots.

To define the models completely, particularly the deterministic version, one has to specify the order of concurrent events. This is done in the following way:

1. an entity leaves a server,
2. a queued entity enters a server,
3. a new entity enters the system and chooses a queue.

It is an interesting question, how a simulation environment allows to fix the order of concurrent events. Therefore the benchmark includes an optional variant of the deterministic model, where this order is changed to

1. a new entity enters the system and chooses a queue,
2. an entity leaves a server,
3. a queued entity enters a server.

Another problem especially for graphical environments is the modeling of large systems. To study this, the benchmark includes an optional variant of the stochastic model with 40 queues and servers.

2 Jockeying Queues

A rather common phenomenon in everyday queueing systems is *jockeying*, i.e. the process that an entity moves from the end of a queue to another shorter queue (cf. fig. 2). This not only happens, when the “entities” are humans (e.g. in supermarkets or on motorways), but it can be used to achieve a better load balancing of servers in computer networks or production lines.

The C22 benchmark asks for variants of the two basic models, where jockeying occurs, whenever a queue (incl. server) is at least shorter by 2 than another one. It defines the behaviour if there are more than one possible source or destination queues and fixes the order of concurrent events. Moreover it requires additional output describing the jockeying events.

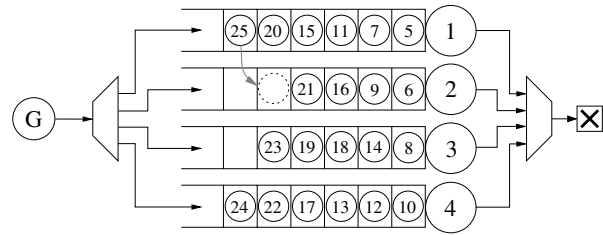


Figure 2: Queueing system with jockeying.

The main problem here is of course, how to detach the last entity of a standard FIFO queue. There are several tricks to achieve this [2], but since they complicate the model and increase the number of events considerably, a simpler solution would be preferred – if the used simulation environment allows for one.

3 Reneging Queues

In some applications an entity leaves a queue before it is served, a behaviour known as *reneging* (cf. fig. 3). This can be a customer, who has lost his patience, food, whose shelf life has been reached, or a workpiece that has to be reheated.

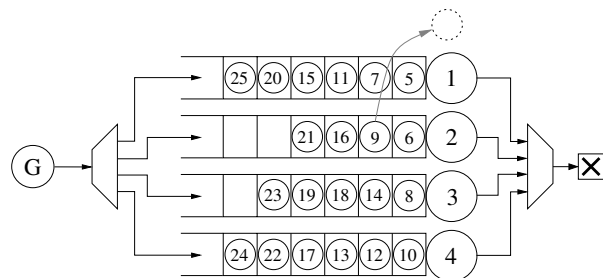


Figure 3: Queueing system with reneging.

The benchmark requests the implementation of model variants where entities renege after a fixed maximal waiting time. This problem seems to be harder than the jockeying case, because now an entity in the middle of the queue has to be released prematurely.

4 Classing Queues

The last benchmark task is inspired by a typical situation during the boarding of a plane: An operator calls “all passengers with seat numbers 15 – 30” to the front

of the queue. It assumes that entities have an additional attribute called *class*, which has a positive integer value and is assigned at entity creation in a round-robin way or stochastically.

The standard models are augmented with an operator that at certain times calls for a class number, whereupon all entities with this class proceed to the front of their queues. The relative order of the entities within this class remains intact, as does the order of the other entities among themselves (cf. fig. 4).

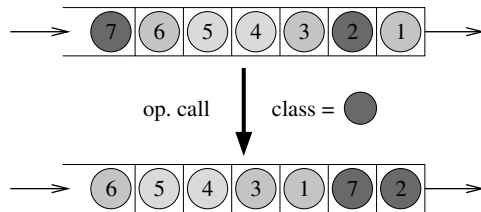


Figure 4: Result of an operator call ($\text{class} \triangleq \text{color}$).

Further practical examples could be the routing of network packets (“now all packets of a given video file”) or the loading of a truck (“now all boxes of a given size”).

As always the benchmark fixes all details such as the exact behaviour of the operator, the assignment of classes, the order of concurrent events and the output data. It is important to note that there is only one global operator that defines the current class for all queues at once. The case of individual operators for each queue is of practical interest as well, but has not been included in the benchmark, since it doesn’t add substantial difficulties to the implementation.

The classing queue model is probably the most challenging of the benchmark, since it requests a dynamical reordering of the entities within the queue. On the other hand it doesn’t require an extra queue output like the jockeying and reneging queues, which makes it similar to the standard priority queue, with the essential difference, that the meaning of “high priority” changes at runtime.

5 Benchmark Implementation using MatlabGPSS

A first implementation of the C22 benchmark has been published [5], it is based on MatlabGPSS [6]. Its basic findings may be helpful for further implementors of the benchmark and will be summarised here.

GPSS [7] is one of the oldest existing modeling languages. It uses the transaction-based paradigm to model discrete event systems, and though being somewhat outdated, it is still a good example of a simple language that uses only a few basic constructions to provide very wide modeling capabilities. The freely available implementation MatlabGPSS [8] combines GPSS statements with general Matlab code. This makes the implementation of complex control structures and the compilation of statistical and graphical results much easier than relying on pure GPSS constructs, thereby allowing to concentrate on the basic questions of queue design.

GPSS is text-based and contains statements for the generation and destruction of entities, the entering and leaving of queues, the reservation and freeing of servers and the delaying of an entity for a given (service) time. Each entity can store a set of parameters, auxiliary functions provide the current number of entities stored in a queue or a server.

For complex queueing strategies one can use so called *user chains*, which are more flexible than standard queues. Entities join a user chain with the `link` statement, entering at the front or end or according to a parameter value. The `unlink` statement allows any entity to free one or more entities from a user chain and to route them to arbitrary places. The exact possibilities of `unlink` depend on the specific GPSS implementation; in MatlabGPSS entities can only be extracted from the front or back end of a user chain.

Using these standard GPSS methods the basic model can be implemented easily. Finding the index of the shortest line is done with a Matlab function. Scaling up the model to 40 queues is then just a matter of setting a dimension parameter. And the changing of the order of concurrent events can be done by applying the GPSS `priority` command that allows to change the priority of an entity dynamically.

To implement the jockey queues, one has to move entities between different places in the model. This is done with the GPSS statement `transfer` and labelling of statements, similar to a classical *go-to*. In a graphical modeling environment similar models would use routing elements such as gates and switches. The main problem, namely to extract an entity from the back end of a FIFO queue, is trivial here, since the `unlink` command allows to extract entities from both ends of the queue.

But the slightly harder problem of the reneging queues, where one has to extract an entity from the mid-

dle of the queue, can not be handled so easily in MatlabGPSS. Therefore one has to rely to the *clone queue* trick [2]: An entity that enters a queue is cloned, one copy waits for the total reneging time, the other one tries to get the server. A bookkeeping variable is set, whenever one of the pair is ready, and checked before the action of the other one. A clone that comes late, is simply terminated.

For the implementation of the classing queues every queue is cut into two sequential queues with a gate in between, that is opened whenever the operator calls for a new class (cf. fig. 5).

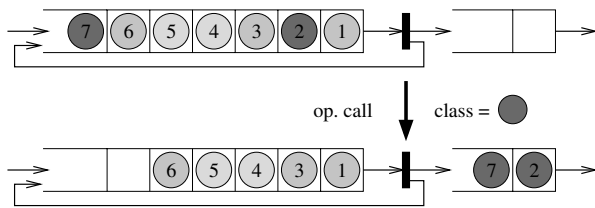


Figure 5: Implementation of the classing queue.

The first queue (at the back end) stores all incoming entities, while the second queue at the front receives all entities with the current class. Since one cannot pick only the matching entities from the first queue, one has to release all entities and route the unwanted ones back into the first queue. This scheme is a variant of the *shuffle queue* from [2].

6 Conclusions

Since all benchmark tasks are rather small and don't need a special mathematical or modeling background, the benchmark is suited for beginners in the field of modeling and simulation. Nevertheless it is probably a challenge for many of the current discrete simulation systems.

As long as perfect solutions to the benchmark problems have not been found generally or in the simulation environment used, second best solutions using special tricks are very welcome. They not only show the state of the art (and its deficiencies), but can help the practitioner to implement non-standard queueing problems in the program at hand.

References

- [1] Law AM. *Simulation Modeling and Analysis*. McGraw-Hill, New York, 5. ed. 2014.
- [2] Austermann L, Junglas P, Schmidt J, Tiekmann C. Conceptual problems of transaction-based modeling and its implementation in SimEvents 4.4. *Simulation Notes Europe SNE*. 2017; 27(3): 137–142. doi: 10.11128/sne.27.tn.10383
- [3] Li W, Mani R, Mosterman P. Extensible discrete-event simulation framework in SimEvents. *Proc. 2016 Winter Simulation Conference*; 2016 Dec; Arlington. New Jersey: IEEE. 943-954.
- [4] Junglas P, Pawletta T. Non-standard Queuing Policies: Definition of ARGESIM Benchmark C22. *Simulation Notes Europe SNE*. 2019; 29(3): 111-115. doi: 10.11128/sne.29.bn22.10481
- [5] Junglas P, Pawletta T. Solving ARGESIM Benchmark C22 'Non-standard Queuing Policies' with MatlabGPSS. *Simulation Notes Europe SNE*. 2019; 29(4): 199-205. doi: 10.11128/sne.29.bn22.10496
- [6] Pawletta T, Drewelow W, Pawletta S. Discrete Event Simulation in Interactive Scientific and Technical Computing Environments. In: *Proc. 12th European Simulation Multiconference on Simulation*; 1998 Jun; Manchester. 529-533. ISBN 1-56555-148-6.
- [7] Schriber TJ. *An introduction to simulation using GPSS/H*. New York: John Wiley & Sons, Inc.; 1991. 437 p.
- [8] Pawletta T., et al. *The MATLAB GPSS Toolbox*. Online: <http://www.cea-wismar.de/tbx/mgpss/> (called 2020-01-07).

Model Generation for Multiple Simulators Using SES/MB and FMI

Hendrik Folkerts*, Thorsten Pawletta, Christina Deatcu

Research Group Computational Engineering and Automation, University of Applied Sciences Wismar, Philipp-Müller-Straße 14, 23966 Wismar, Germany; *hendrik.folkerts@cea-wismar.de

Abstract. This paper deals with the extension of a Python-based infrastructure for studying the characteristics and behavior of families of systems. The infrastructure allows automatic execution of simulation experiments with varying system structures as well as with varying parameter sets in different simulators. Special focus is put on the support of different simulation environments by creating models implementing the Functional Mockup Interface (FMI). Possible system structures and parameterizations are defined using a System Entity Structure (SES). The SES as a high level approach for variability modeling, particularly in simulation engineering, describes a set of system configurations, i.e. different system structures and parameter settings of system components. In combination with a Model Base (MB), executable models can be generated from an SES. Based on the extended SES/MB approach, tool-supported variability modeling and automatic model generation and execution in different simulation environments using FMI is described. This is done by means of an engineering application.

Introduction

This paper is based on [1] and [2]. It focuses on the more general approach using the tools for variability modeling introduced there by integrating with the Functional Mock-up Interface (FMI) instead of just offering simulator specific solutions.

The high variant diversity with components of different application fields in today's technical systems leads to the need for variability modeling and integration of varying simulation platforms. One application area for variability modeling is e.g. the generation of software for electronic control units, which is often generated by underlying models. Those models are usu-

ally of similar type, but still differ in structure and parameterization. To handle modeling and simulation of these so called families of systems, several approaches for variability modeling exist. Most approaches make use of 150% models, which means that all possible behavior is put into just one large and complex model and functionality is then adjusted by switching off unneeded model parts. In contrast to 150% modeling, in this paper we describe a method to define, generate and simulate well-tailored and therefore lean models by making use of the System Entity Structure / Model Base (SES/MB) approach. The SES/MB approach [3] originates in the systems theory community and has undergone many extensions over the years [4, 5]. It allows platform-independent variability modeling with subsequent platform-dependent model generation of specific variants. The structures of systems are coded in an SES, while the dynamic models are organized in an MB. The SES links to these dynamic models.

For this approach, a proposal for using one MB in several simulators is detailed. This is achieved by creating an MB of models which implement the FMI. The general tool independent standard for *model exchange* and *co-simulation* FMI [6, 7] enables the exchange of models between different simulators. This makes it possible to combine models from different domains and execute them in several simulation environments.

After the extended SES/MB approach is briefly introduced, the paper presents some software tools implementing the theory. An engineering application example is then discussed in detail to clarify the process of model definition and model generation using FMI.

1 SES/MB Theory and Implementation

This section briefly discusses the general SES/MB theory and the derived extended SES/MB (eSES/MB) in-

frastructure. Subsequently, an implementation of the infrastructure is presented.

1.1 SES/MB Basics and the eSES/MB Infrastructure

An SES is represented by a tree structure comprising entity nodes, descriptive nodes and attributes. A number of different system structures can be coded in one SES tree. In the context of modeling and simulation *entity nodes* are linked to *basic models* organized in an MB. Attributes of an entity node correspond to the parameters of the associated basic model. *Descriptive nodes* describe the relations among at least two entities and are divided into *aspect*, *multi-aspect* and *specialization* nodes.

In order to derive a specific system configuration all variation points are resolved by evaluating the rules at the descriptive nodes of the SES. This procedure is called *pruning*.

The resulting *Pruned Entity Structure (PES)* represents exactly one system configuration. In conjunction with an MB, a fully configured and executable model can be generated from the PES.

The basic SES/MB framework introduced in [3] was extended by new modeling features, methods and components [4, 5], such as an *Experiment Control (EC)* and an *Execution Unit (EU)* as shown in Figure 1. In this eSES/MB infrastructure, the EC uses an interface to the SES and its methods to derive goal-driven system configurations and to generate models, which are executed by the EU. The results returned by the EU are collected and analyzed by the EC. Thus, the derivation and generation of subsequent system configurations can be controlled reactively based on experiments already carried out.

A set of variables with global scope establish the interface to the SES. They are called *SES variables (SES-var)*. *Semantic conditions* can be used to specify permitted value ranges and dependencies between SESvars. *SES functions (SESfcn)* are introduced for the specification of procedural knowledge. Complex variability can often be described more easily with SESfcns. Typical examples include the definition of varying coupling relations or the definition of variable parameter configurations in attributes. For automatic pruning, *selection rules* at descriptive nodes need to be defined, such as *aspecrules* for aspect and multi-aspect siblings or *specrules* at specialization nodes. A special mandatory attribute of multi-aspects is the attribute *number of*

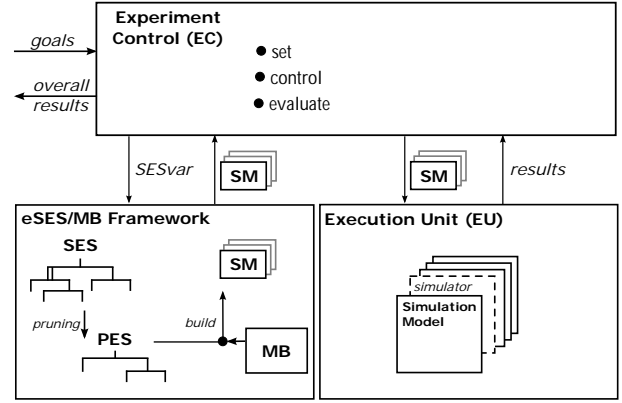


Figure 1: The eSES/MB infrastructure.

replications (numRep). The numRep attribute specifies the number of entities to create at a multi-aspect node during pruning. The *mb-attribute* of leaf entity nodes connects the entity node to a basic model in the MB. Attribute values and selection rules can be specified using SESvars or SESfcns.

1.2 Software Tools

The eSES/MB framework as presented in the lower left part of Figure 1 was implemented in a prototype software tool in MATLAB [8]. The focus of this tool is the modeling and generation of MATLAB/Simulink models. In contrast to the MATLAB prototype, the objective of the software used in this paper is to support the generation and execution of models for different simulation environments. The infrastructure in Figure 1 is implemented as a Python framework as presented in Figure 2. The tools are called *SESToPy*, *SESMoPy*, and *SESEuPy* [9].

SESToPy (System Entity Structure Tools Python) implements a graphical editor and all SES related methods. In the editor an SES tree can be specified interactively in a file browser view and attributes and rules can be defined for every node. In addition to the pruning method already mentioned, SESToPy supports some more methods such as *merging* different SES and *flattening* for removing the hierarchy information. Applying the flattening method, a *Flattened Pruned Entity Structure (FPES)* is derived.

For generating executable models, **SESMoPy (System Entity Structure Model builder Python)** was developed. SESMoPy is a model builder, which implements the *build* method in two different ways and

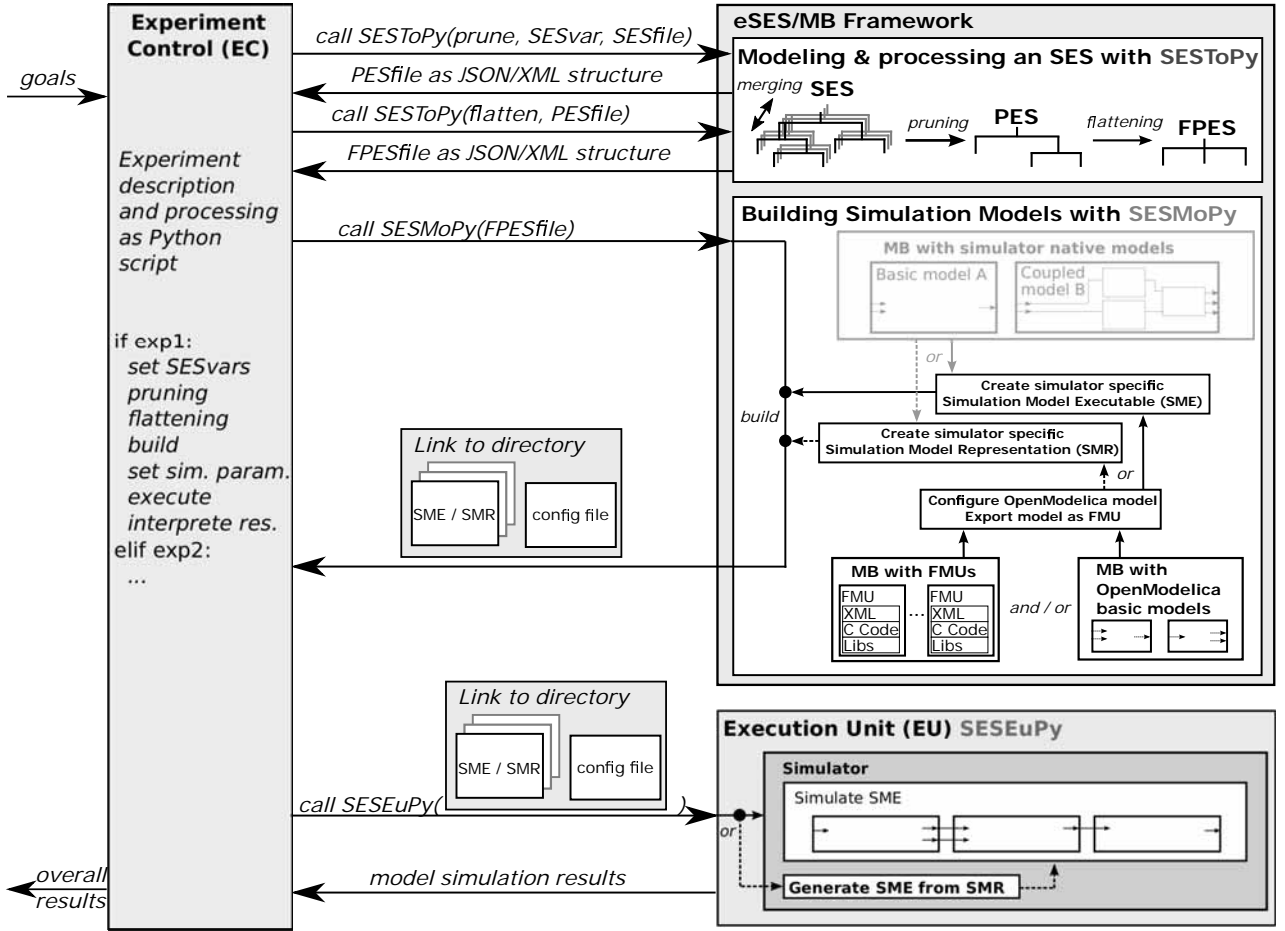


Figure 2: Python-based eSES/MB infrastructure for multiple EUs.

supports several simulation environments. For both approaches, all corresponding basic models must be organized in MBs, as shown in Figure 2. The first approach, called *native model generation*, is the generation of executables for a specific EU using a simulator specific MB. The second approach, this paper focuses at, is the *model generation based on FMI*. A Functional Mock-up Unit (FMU) is a model that implements an FMI [10]. In an FMU models are described by differential, algebraic, and discrete equations with time, state, and step events. In the scope of SESMoPy *FMI for Model Exchange* is used, which enables the simulation environment to generate C code of the FMU. *FMI for Co-simulation* is not discussed in this paper. The generalized interface FMI is supported by a number of established simulators [11], such as Simulink, OpenModelica or Dymola discussed for the use with SESMoPy. Using the FMI-based approach, an MB with basic mod-

els from the simulator OpenModelica and/or an MB with FMUs are defined. SESMoPy creates an OpenModelica model and configures it according to the information passed in the FPES. Thus FMUs in an MB need to be imported into OpenModelica. The configured OpenModelica model is exported as FMU. Depending on the target simulator a specific Simulation Model Executable (SME) or Simulation Model Representation (SMR) is created. Finally SESMoPy returns a link to a directory, where the SME or SMR is placed together with a configuration file with information on the SME or SMR.

Information about the way the model is created can be provided in the EC calling SESMoPy or at the SES level according to the SES enhancements in [4].

The Python software tool SESEuPy (System Entity Structure Execution unit Python) acts as a general EU. It implements a kind of wrapper for the integration of

different simulation environments into the framework. SESEuPy takes the link to the directory with the SME or SMR and reads the configuration file. If the model is given as SMR, an SME needs to be built. The SME then can be simulated in the target simulator and simulation results are returned.

In the next section, the components and functionality of the Python framework are explained using the example of an engineering application.

2 Engineering Application

A feedback control system can be modeled using transfer functions describing the behavior of the components in frequency domain. Controlled variables in a feedback control system are usually influenced by disturbances. A common approach for minimizing the influence of predictable disturbances is adding a feedforward control. The system can be mapped to a signal-flow oriented model. In the following paragraphs it is described how the eSES/MB infrastructure can be used to design and test such a system using the introduced tools and FMI-based model generation in combination with the simulation programs Matlab/Simulink, OpenModelica, and Dymola.

2.1 Problem Description

A process unit with a *PTI* behavior shall be controlled using a PID controller. A disturbance with a *PTI* behavior affects the output of the process unit. Different configurations of the PID controller shall be tested. If a defined regulatory goal is met, the current configuration of the PID controller is taken. Otherwise the structure is varied by adding a feedforward control to the system and different configurations of the PID controller are analyzed again. Figure 3 depicts a schematic representation of the application.

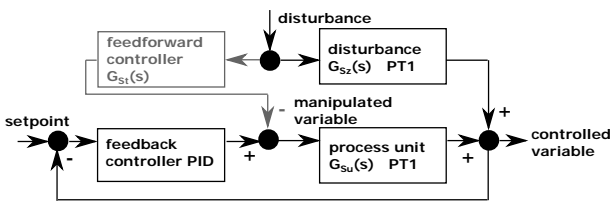


Figure 3: Structure of the feedback control system with optional feedforward control.

The system's behavior follows the *PTI* transfer

function in Equation 1 and the step-shaped disturbance affects the output of the process unit with a *PTI* behavior according to Equation 2. The optional feedforward control is realized by subtracting the disturbing signal calculated by Equation 3 from the manipulated variable. The control goals are a settling time of less than 15 seconds and a maximum overshoot of less than 5% after a disturbance.

The system has two structure variants, either without or with the feedforward control part, and a range of different configurations for the PID controller can be applied for each structure variant. In the next section, the two structure variants and their possible configurations are specified as an SES.

$$G_{Su}(s) = \frac{1}{20 \cdot s + 1} \quad (1)$$

$$G_{Sz}(s) = \frac{1}{10 \cdot s + 1} \quad (2)$$

$$G_{St}(s) = \frac{G_{Sz}(s)}{G_{Su}(s)} = \frac{20 \cdot s + 1}{10 \cdot s + 1} \quad (3)$$

2.2 Variant Modeling with SESToPy

The specification of the SES describing the feedback control system is done with the tool SESToPy. The tree and all attributes are defined via a graphical user interface. During modeling the SES with SESToPy, checks on the SES and plausibility tests are executed indicating model errors. The SES is saved as a JSON structure.

Figure 4 depicts the SES and its representation in SESToPy. The SES uses some extensions introduced in [5]. In addition to the different system configurations, essential parts for the configuration of simulation experiments are defined.

The root node *exp* of the SES and its subsequent aspect node *expDEC* describe a set of simulation based parameter studies for different system structures. The subtree of the entity node *simModel-ctrlSys* specifies the two system structures, i.e. a variant with and a variant without feedforward controller. The other two entity nodes specify experiment related information: The entity node *simMethod* specifies a target simulation environment for performing simulation runs using the SES-var *mysim*. The SESvar *myinterface* specifies whether to use the native or the FMI model generation. Other simulation execution parameters, such as the simulation period, are not specified and are set by the EC later. The entity node *expMethod* specifies the permitted value

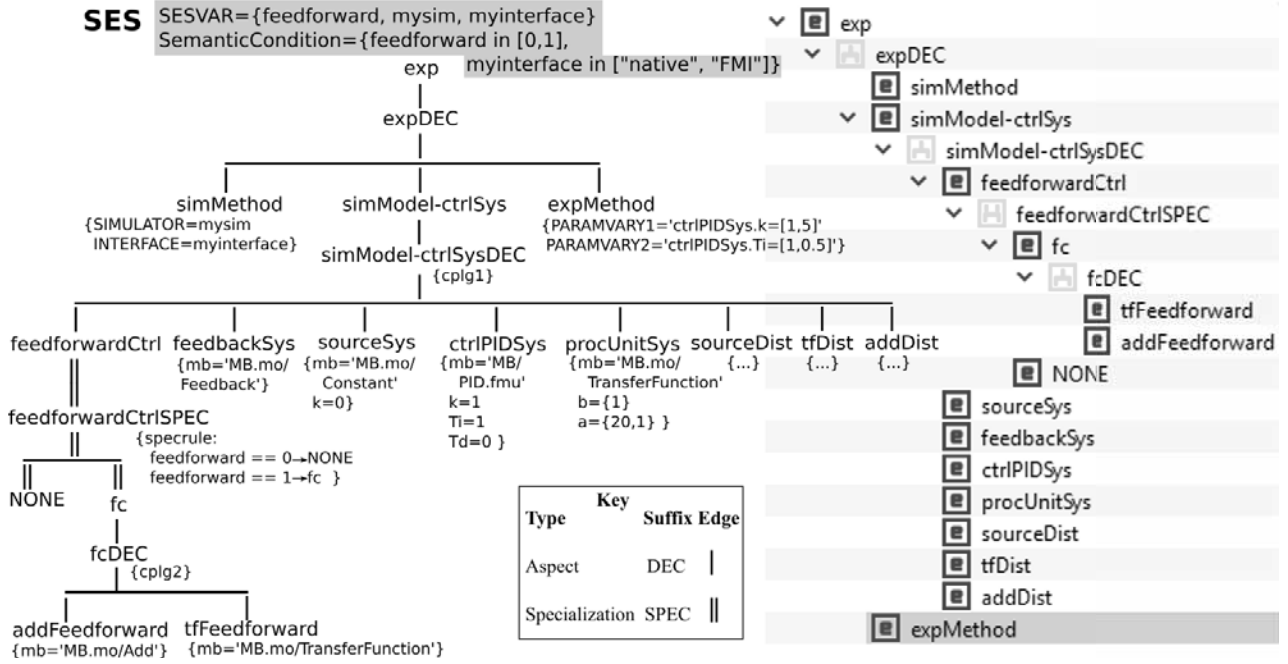


Figure 4: Left: SES specifying the feedback control system study; Right: Part of the SES representation in SESToPy.

ranges of two parameters for the PID controller. Besides the different system structures, they are the subject under study. The aspect *simModel-ctrlSysDEC* describes that each system variant consists of the following entities: *feedbackSys*, *sourceSys*, *ctrlPIDSys*, *procUnitSys*, *sourceDist*, *tfDist* and *addDist*. They are mandatory system elements. The optional feedforward control is specified by the subtree of entity *feedforwardCtrl*. The coupling relations of both structure variants are defined in the attribute *cplg1* of aspect *simModel-ctrlSysDEC*.

According to [12], optional parts in an SES are expressed by a specialization node where one of its children is a NONE element. A NONE element means that the entity is not included at all. The selection at a specialization is defined by an attribute called *specrule*. The *specrule* of the specialization *feedforwardCtrlSPEC* defines that either the entity *fc* or *NONE* is selected during pruning. The result of evaluating the *specrule* at node *feedforwardCtrlSPEC* depends on the value of the SESvar *feedforward*. The SESvar codes the two possible structure variants as values 1 or 0. Therefore, the semantic condition $feedforward \in [0, 1]$ applies to the SESvar. The entity *fc* and its subsequent aspect *fcDEC* specifies the feedforward control structure as a composition of the two entities *tfFeedforward* and *addFeed-*

forward.

Aspects and multi-aspects can define coupling relations as attribute. Couplings specify a composition of entities, which can be linked to basic models. Coupling attributes are abbreviated with *cplg* in Figure 4. Due to the varying system structures specified in the SES, the couplings in attribute *cplg1* of aspect node *simModel-ctrlSysDEC* are defined using an SESfcn. The coupling definitions in *cplg2* at node *fcDEC* are invariable and can therefore be defined without using an SESfcn.

According to Section 1, each leaf node defines an mb-attribute referring to a basic model in the MB. The basic model can be an OpenModelica component or an FMU. The other attributes of the leaf nodes define properties to configure the linked basic models. The values for *k* and *Ti* specified at node *ctrlPIDSys* are only default values, which will be overwritten because they are parameters under study.

2.3 Creating an MB

OpenModelica is an open source simulation platform and defines a set of basic models. It is widely used in different fields of engineering. In this case study OpenModelica basic models as well as FMU basic models are used. The FMUs define the FMI and can thus be

exported from any simulation environment.

For the OpenModelica MB a *package* is created that contains basic models. This package is stored as the file *MB.mo* and is referred to as *local OpenModelica library* in this paper. Furthermore the FMUs with the *fileending *.fmu* are stored in a folder on the local filesystem, which is referred to as *local FMU library* in this paper.

The local OpenModelica library is filled with the following basic models whose names correspond to the names in the mb-attributes of the leaf nodes in the SES:

- *Constant* as the setpoint for the controlled variable
- *Feedback* for closing the feedback control loop
- *TransferFunction* for representing the process, the disturbance's behavior, and the feedforward
- *Add* for adding signals

In the local FMU library FMUs are placed like listed. The names correspond to the names in the mb-attributes of the leaf nodes in the SES.

- *Step.fmu* for stimulating the disturbance
- *PID.fmu* is the controller of the feedback control system

Each basic model can be configured according to the attributes of the leaf node which they are linked to in the SES. The local OpenModelica library as well as the local FMU library act as MB for the basic models.

2.4 Experiment Execution

For executing simulation based experiments the experiment process and its goals need to be defined in a Python script. This script implements the EC according to Figure 2. The Python framework provides some EC related template scripts. The goals of the experiment were discussed in Section 2.1. The experiment should start with the study of different PID controller configurations using the control system structure without feedforward controller. The simulation is executed with the simulators OpenModelica, Dymola, and Simulink. In case that the objectives are not achieved by just varying the parameters k and T_i of the PID controller, the study shall be carried out with the additional feedforward control structure and the simulation programs OpenModelica, Dymola, and Simulink. A snippet of the EC script with essential steps of the experiment process is given next.

```
...
SESfile = ...
if conditions_for_experiment:
    #prune, flatten, build, and execute
    SESvar = [mysim = <simulator>,
               myinterface = "FMI",
               feedforward = 0]
    PESfile = SESToPy("prune", SESvar,
                      SESfile)
    FPESfile = SESToPy("flatten", PESfile)
    smHandle = SESMoPy("build", FPESfile)
    sim_param = [solver=<solver>, ...]
    results = SESEuPy("simulate", smHandle)
...
elif conditions_for_experiment:
    #prune, flatten, build, and execute
    SESvar = [mysim = <simulator>,
               myinterface = "FMI",
               feedforward = 1]
    PESfile = ...
...
...
```

The EC starts the experiment by setting the SESvars *mysim*, *myinterface*, and *feedforward*. A target simulator is set for *mysim*. Next, the EC calls SESToPy's API method for pruning with the current SESvar values and a reference to the file defining the SES as JSON structure. The pruning process results in a PES coded as JSON structure. Afterwards, the EC calls SESToPy's API method for flattening the PES. The created FPES is similar to the FPES shown in Figure 5, which represents the more complex FPES for the later SESvar assignment *feedforward* = 1. A reference to the file containing the FPES as a JSON structure is returned to the EC. The EC then calls SESMoPy's API method for the build method and passes the FPES file handle. SESMoPy determines the target simulator from the attribute at the node *simMethod* and the value ranges of the PID controller parameters under study from the attribute at node *expMethod* in the FPES.

Based on the information in the FPES and the basic models from the MB, SESMoPy creates an OpenModelica model for each configuration of the simulation model of the control system. FMU basic models need to be imported into OpenModelica. The configured OpenModelica model is exported as FMU, which is called *model FMU* in this context. This model FMU is simulator independent, since it implements the FMI. It represents an SM. Thus only one MB needs to be de-

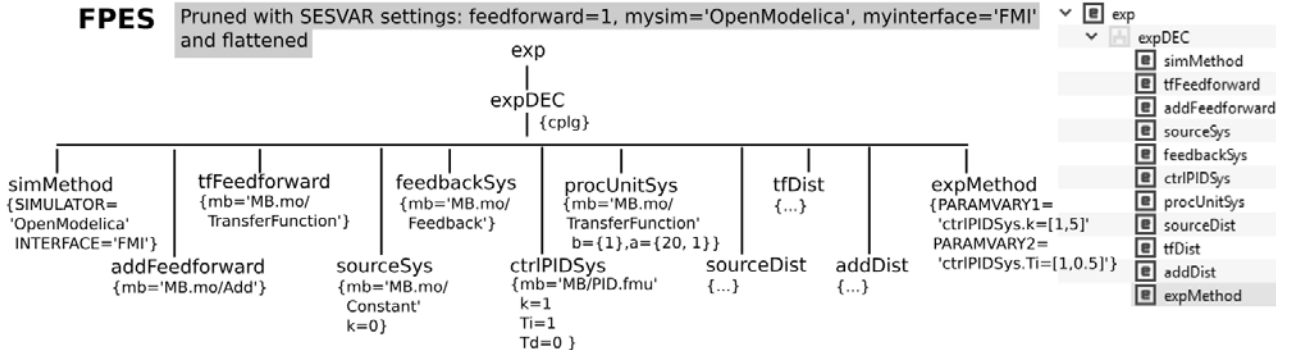


Figure 5: Left: FPES to study the feedback control system structure with feedforward; Right: FPES representation in SESToPy.

financed for use with multiple target simulators.

Depending on the target simulator different steps are necessary as discussed before. (i) A Simulation Model Executable (SME) for the target simulators OpenModelica and Dymola or (ii) a Simulation Model Representation (SMR) for the target simulator Simulink is created.

(i) The SME is built by importing the model FMU into the target simulator. Using the interface of the FMU simulator specific code is generated of the model. For execution a file with simulator specific instructions on the execution is generated. Furthermore a configuration file with information about the SME and its target simulator is created.

(ii) The SMR is a file with simulator specific instructions for the import of the FMU in the target simulator. The file is not executed yet. Furthermore a configuration file with information about the SMR and its target simulator is created.

SMs of one structure variant have different configurations of the PID controller. A handle to the directory with all SMs is returned by SESMoPy to the EC, referred to as *smHandle*. The EC extends the configuration file with simulation data, such as the solver to use or simulation start and stop time. The EC calls the tool SESEuPy and passes the *smHandle* as the link to the SMs and the configuration file. In collaboration with the target simulation environment, SESEuPy controls the execution of an SM. An SME can be executed directly, whereas during execution of an SMR an SME is built. Figure 6 shows the structure of a fully configured OpenModelica model, but with feedforward controller, i.e. for the SESvar assignment *feedforward* = 1. Finally, SESEuPy returns the simulation results to the EC.

In case the results meet the experimental goals, the

overall results are calculated and returned by the EC. In case the goals are not reached, the second system structure with the additional feedforward controller by the SESvar assignment *feedforward* = 1 is set and a new model configuration and generation is started.

If the experimental goals have been achieved, the overall results of the experiment are the necessary control structure and the appropriate PID controller parameter settings. Otherwise the failure to achieve the objectives may also be established.

In addition simulation with another simulator can be tested. In the SESvar *mysim* another simulator is set and the model generation and simulation process is started over with the structure variant without feedforward controller. In this way, model by model validation is achieved using different simulators.

3 Conclusion

In this paper the extension for working with FMI of some Python-based software tools for variant modeling are presented. The entire process of variability modeling beginning with the system specification with an SES up to automatic variant derivation, model building, and execution is described. The proposed eSES/MB infrastructure makes it possible to model and simulate engineering problems using different target simulation environments.

References

- [1] Folkerts H, Deatcu C, Pawletta T, Hartmann S. Python-based eSES/MB Framework: Model Specification and Automatic Model Generation for

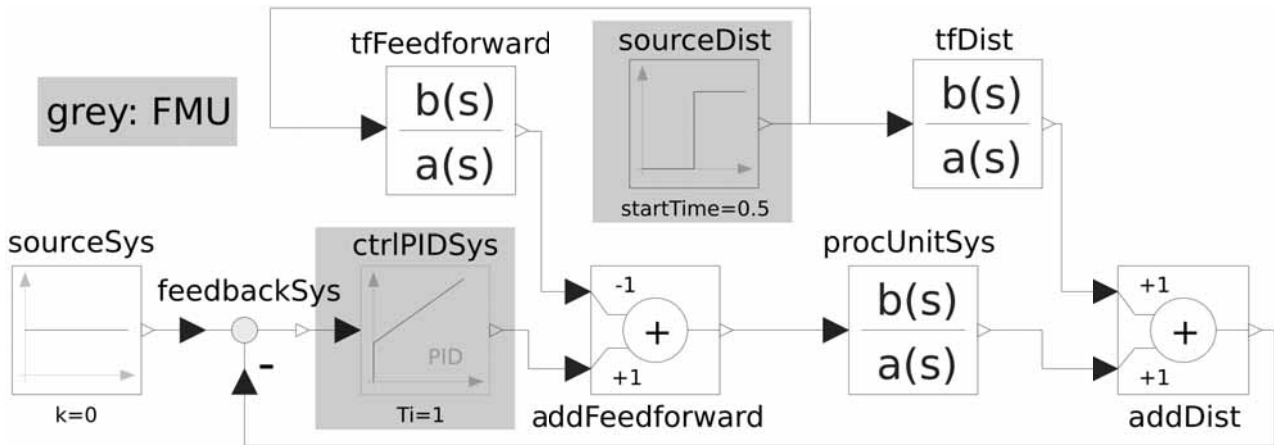


Figure 6: OpenModelica SM of the feedback control system with feedforward control.

- Multiple Simulators. *SNE Simulation Notes Europe*. 2019; 29: 207–215. doi: 10.11128/sne.29.tn.10497.
- [2] Folkerts H, Deatcu C, Pawletta T, Hartmann S. A Python Framework for Model Specification and Automatic Model Generation for Multiple Simulators. *2019 International Interdisciplinary PhD Workshop*; 2019 May; Wismar. IEEE. doi: 10.1109/IIPHDW.2019.8755423.
- [3] Zeigler BP, Kim TG, Praehofer H. *Theory of Modeling and Simulation*. 2nd ed. Cambridge: Academic Pr.; 2000. 510 p.
- [4] Schmidt A, Durak U, Pawletta T. Model-Based Testing Methodology Using System Entity Structures for MATLAB/Simulink Models. *SIMULATION: Transactions of The Society for Modeling and Simulation International*. 2016; 92(8): 729–746.
- [5] Schmidt, A. *Variant Management in Modeling and Simulation Using the SES/MB Framework* [dissertation]. University of Rostock; 2019.
- [6] Blochwitz T, Otter M, Arnold M, Bausch C, Clauß C, Elmquist H, Junghanns A, Mauss J, Monteiro M, Neidhold T, Neumerkel D, Olsson H, Peetz JV, Wolf S. The Functional Mockup Interface for Tool independent Exchange of Simulation Models. In Proc. of the 8th International Modelica Conference. *Modelica Conference*; 2011 March; Dresden, Germany. p 105–114. doi: 10.3384/ecp11063.
- [7] Blochwitz T, Otter M, Akesson J, Arnold M, Clauß C, Elmquist H, Friedrich M, Junghanns A, Mauss J, Neumerkel D, Olsson H, Viel A. Functional Mockup Interface 2.0: The Standard for Tool independent Exchange of Simulation Models. In Proc. of the 9th International Modelica Conference. *Modelica Conference*; 2012 Sept; Munich, Germany. p 173–184. doi: 10.3384/ecp12076173.
- [8] Pawletta T, Pascheka D, Schmidt A, Pawletta S. Ontology-Assisted System Modeling and Simulation within MATLAB/Simulink. *SNE Simulation Notes Europe*. 2014; 24: 59–68. doi: 10.11128/sne.24.tn.102241.
- [9] Research Group CEA. *Python-Based eSES/MB Infrastructure*. 2019. [www.github.com/hendrikfolkerts](https://github.com/hendrikfolkerts), last accessed 2020/09/10.
- [10] Modelica Association Project "FMI". *Functional Mock-up Interface for Model Exchange and Co-Simulation*. 2014. https://svn.modelica.org/fmi/branches/public/specifications/v2.0/FMI_for_ModelExchange_and_CoSimulation_v2.0.pdf, last accessed 2019/08/06.
- [11] Modelica Association Project "FMI". *Functional mock-up interface for model exchange and co-simulation*. 2019. <https://fmi-standard.org/tools/>, last accessed 2020/02/21.
- [12] Deatcu C, Folkerts H, Pawletta T, Durak U. Design Patterns for Variability Modeling Using SES Ontology. In Proc. of Spring Simulation Multi-Conference 2018. *Spring Simulation Multi-Conference*; 2018 Apr; Baltimore/MD, USA. SCS Int. p 3:1–3:12.

Generative and Modular Simulation Models for Supply and Manufacturing Networks

Pavel Gocev, Tim Hellfeuer

Siemens Energy, Huttenstraße 12, 10553 Berlin, Germany;
pavel.gocev@siemens.com, tim.hellfeuer@siemens.com

Introduction

The application of Discrete-Event Simulation (DES) models for purposes of planning and optimization of factories and supply networks is characterized with various abstraction levels and granularities of the model structure. These two aspects are dependent on the complexity of the systems to be simulated, the business goals to be achieved and the project objectives where the simulation models are deployed. This is especially intensified when different product parts and components on different levels within the supply networks are included into one model, like production lines and work centers within existing and emerging factory shop floors combined with the network of suppliers and additionally flavoured with the ramp up of new products, new work centers or both. Very often the complexity is increased due to the organizational nature of production types and different project groups with own modelling paradigms. This is particularly a characteristic of supply networks that deliver very complex commodity products like whole power plants or respective components.

The usual foundation to describe and model such complex systems is the data around the three principal consisting domains (PPR): **Products** to be delivered (raw-materials, parts, components, finished products), **Processes** that produce them (from supply chain steps down to operational steps) and **Resources** necessary to accomplish the work (suppliers, factories, production lines, work centers, machines, etc.). Yet the data is not enough to build the simulation model that, following the paradigm of digital twin, also represents its behaviour as well as the interdependencies between the consisting elements within the PPR-Domains. These interactions, behaviours, and cause-and-effect graphs are usually embodied as a procedural programming, affecting the scope and the depth of the modelled logic and therewith they

influence the abstraction levels within the model. The situation is even more complex, in a case when the simulation models represent a workshop-like production and the same simulation model is intended to be deployed for various factories and different products within one big and multifaceted company like Siemens Energy. In opposite of the typical assembly lines like in automotive or electronics industry, here we are talking about product and respective parts and components that are running through different resources in an arbitrary sequence defined by product features and manufacturing technologies available in the considered factories or within the supply network.

1 Model Structure

Whit this paper we present a kind of shell for Generative and Modular Simulation Models for Supply and Manufacturing Networks realized in Tecnomatix Plant Simulation by Siemens. The solution bears on paradigm of data driven model generation and scenario definition, as well as modular structure that offers a flexible and scalable deployment on arbitrary granularity levels with defined but extendable building blocks. The focus here is put on the data that describe the main PPR-Domains as well as scenario parameters, which variation offers two-fold controls within the simulation modle. On one side their values can be directly alternated and on the other side for each scenario they change a certain behavioural logic within the model through adjustment of some modules that are constating part of the model. During the simulation experiment the operational data is acquired and at the end of each simulation experiment the statistical data is calculated and prepared for external analysis.

2 Modules

The solution is built on modules mainly embodied in

the Plant Simulation own language SimTalk, that are interlinked through their communication and every module takes a decision that shapes and controls the behaviours during the simulation experiment.

The first aspect of the platform are the modules for data ingestion (DIM) and model generation (MGM). The external PPR-Data is imported into the Simulation Model Shell through available interfaces of Tecnomatix Plant Simulation and stored in the tables within or in a shared-memory database SQLite. The structures of data and the simulation model follow some standards like ISA-95 resp. IEC 62264 which yields is a precondition for flexible and scalable solutions. The specific and executable simulation model is generated in seconds, that offers the user an execution of arbitrary simulation experiments.

The Production Orders Module (POM) generates the necessary production orders based on the information from the Bill of Materials and the Disposition Module (DM) governs their dispatching in the supply chain and shop floor. Every Production Order has assigned at least one or more Routings that are structured around the information within the Bill of Processes. The Make or Buy Module (MBM) decides if a Production Order will be produced internally or externally at suppliers. The internal production (make) is controlled by the Scheduling Module (SM) that is founded on one of the ideas of Industrie 4.0 where for each Process the Products and Resources are communicating and take decisions regarding the context situation. Here within each Process the main function assigns a specific Production Order to a certain Resource from the alternative list considering the situation within the shop floor and the most suitable prioritization logic. The processing of the Products is monitored by the Manufacturing Module (MM) that regulates functionalities like: setting up of Resources, single part processing, batching, batch splitting, assembly, disassembly as well as an initiation for movement of the parts which is carry out by the Transportation Module (TM). As the Resources are the foundation of the supply and manufacturing networks, the Resource Module (RM) regulates the planned and stochastic events that influence their availability and therewith a numerous behaviours and decisions during the simulation experiment. Firstly, every Resource has assigned one or more Resource Life Cycle States like commissioning, ramp-up, fully allocated, ramp-down, in-relocation, etc. Within every state the Resources are further constrained with the shift model that defines working hours, breaks, shift handover, weekends

and holidays. The last limitations are the presumable outages and failures that are stochastically described for each Resource. The Simulation Analytics Module (SAM) is the crucial element that monitors the objects, acquires data, calculates some Key Performance Indicators and prepares the results for further external analysis.

Additionally there are some modules for model analysis, monitoring, optimization. Some of these modules are neither used during the business simulation experiment nor by the user but rather by developers for model improvement purposes.

3 Simulation Experiments and Results

One set of PPR-Data and defined scenario parameters are necessary input for performing a simulation experiment. The performance of the simulation model respectively the simulation time is dependant on a scenario complexity and the modules and features that are activated in a particular scenario. This possibilities to define each scenario through parameters, enables the user to carry out different experiments that can include: a combination of a long term planning with the short-time scheduling, an involvement or ignorance of the suppliers, definition of make or buy constraints, an integration of factory planning aspects like ramp-up, a relocation and ramp-down, an introduction of alternating shift models or a variation of prioritization rules.

The operational data gathered during the simulation experiment is a foundation for further analysis and evaluation. At the end of each experiment the usual production and logistics Key Performance Indicators are calculated and displayed to the user for further analysis to support the definition of new scenarios or their comparison as well as take actions in the real factory if necessary.

4 Outlook

The very complex and modularly structured shell for generation of simulation models offers a various possibility for further developments in scope and in depth. The current modules have been developed to respond on the demands within the Siemens Energy organization and emerged on the base of business requirements. Additional developments are possible and necessary to cover other requirements and functionalities that are not still covered and implemented.

Entwicklung webbasierter graphischer Benutzeroberflächen für Simulationskerne

Svenja Hilbrich^{1*}, Katharina Gerdes¹, Johannes Hinckeldeyn¹, Jochen Kreutzfeldt¹

¹Institut für Technische Logistik, Technische Universität Hamburg, Theodor-Yorck-Straße 8, 21079 Hamburg, Deutschland; *svenja.hilbrich@tuhh.de

Abstract. Simulation spielt eine wichtige Rolle in vielen Disziplinen der Wissenschaft und Praxis. Die webbasierte Simulation bietet dabei Vorzüge gegenüber Desktopanwendungen und erlaubt es die Dauer zur Durchführung einer Simulationsstudie zu verringern. Häufig werden damit Client-Server-Strukturen bezeichnet, bei denen der Simulationskern Dienste auf einem Server bereitstellt und die Benutzeroberfläche durch eine Webseite realisiert wird. Solche Systeme werden allerdings nur selten entwickelt. Daher wird mit diesem Beitrag eine Anleitung zur Entwicklung einer webbasierten Benutzeroberfläche für Simulationskerne bereitgestellt. Anhand von Designfragen und Entscheidungsbäumen wird dabei schrittweise durch den Entwicklungsprozess geleitet, welcher zusätzlich durch ein Beispiel aus der Materialflusssimulation illustriert wird.

Einleitung

Simulation ist das „Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind“ [1, S. 3]. Damit bietet eine Simulation deutliche Vorteile gegenüber Experimenten in einem realen System, zum Beispiel [2]:

- Die Erstellung eines Simulationsmodells ist kostengünstiger als die eines realen Systems.
- Das Verhalten des Systems kann ohne Auswirkungen auf das Realsystem untersucht werden.
- Die Dauer des Beobachtungszeitraums kann verlängert oder verkürzt werden.

Für Materialflusssysteme gilt zusätzlich, dass nachträgliche Veränderungen mit sehr hohen Kosten verbunden sein können. Daher ist die Simulation auch in der Planung und Dimensionierung von Materialflusssystemen ein weit verbreitetes Hilfsmittel [3].

Als webbasierte Simulation werden Simulationswerkzeuge bezeichnet, die über einen Webbrowser zugänglich sind. Vorteile dieser Form der Simulation liegen zum Beispiel in der Unabhängigkeit vom Betriebssystem und Rechenleistung des Anwenders, der Möglichkeit zur gemeinsamen Modellierung durch mehrere Anwender und der parallelen Ausführung von mehreren Simulationsläufen [4]. Diese Vorteile ermöglichen es den zeitlichen Aufwand für die Simulation zu verringern, sodass es sich bei der webbasierten Simulation um ein relevantes Forschungsthema handelt.

1 Stand der Technik

Die Verwendung eines Simulationswerkzeuges wird entscheidend durch dessen Bedienbarkeit bestimmt. Besonders für Anwender, welche keine Simulationsexperten sind bietet eine graphische Benutzeroberfläche die benötigte Unterstützung zur Erstellung und Auswertung eines Simulationsmodells. Ohne eine solche Schnittstelle würde der Modellierungsprozess vertiefende Kenntnis der für die Simulation verwendeten Programmiersprache erfordern, dies wiederum schließt Benutzer ohne entsprechendes Fachwissen aus [5].

Für Simulationswerkzeuge sind Benutzerschnittstellen traditionell in Form von Desktopanwendungen realisiert [6]. Das Thema webbasierte Simulation begann 1996 im Rahmen der Winter Simulation Conference das wissenschaftliche Interesse zu wecken [7]. Kuljis und Paul identifizierten bereits im Jahr 2000 eine Reihe an Anwendungsfällen, welche von einem webbasierten Simulationsansatz profitieren würden [8]. Syberfeldt, Karlsson et al. benennen mit den Stichworten Zugänglichkeit, Skalierbarkeit, Portabilität, Wartung, kontrolliertem Zugriff und Lizenzierung Vorteile der webbasierten Architektur gegenüber eines klassischen Ansatzes [6]. Trotz des seit Mitte der Neunzigerjahre bestehenden Interesses und der bestehenden Vorteile existieren nur wenige webbasierte Simulations-

anwendungen [4]. In [9] werden unflexible Standard-Simulationssoftware und teure Lizenzen als wichtige Gründe, die einen Einsatz von Simulation verhindern, genannt. In ihrem Lösungsansatz im Rahmen des Projektes DREAM (“simulation based application Decision support in Real-time for Efficient Agile Manufacturing”) wählen sie eine webbasierte open source Implementierung mit einem auf python aufbauenden Simulationskern. Neben dem Ansatz zur Verwendung und Adaption einer open source Simulationssoftware ist die Entwicklung eines auf allgemeinen Programmiersprachen aufbauenden individuellen Simulationskerns eine weitere Möglichkeit zur Überwindung der aufgezeigten Barrieren beim Einsatz von Simulation. In [10] wird die Entwicklung von Simulationssoftware, welche auf einer allgemeinen Programmiersprache aufbaut als von gleichbleibend signifikanter Bedeutung identifiziert. Für den Kreis der Anwender, die diesen Weg wählen stellt der vorliegende Beitrag eine strukturierte Anleitung zur Erstellung einer webbasierten Benutzeroberfläche zur Verfügung. Diese soll Entwickler individueller Simulationssoftware dabei unterstützen die Vorteile einer Benutzeroberfläche mittels einer webbasierten Implementierung für sich nutzbar zu machen. Die vorhandene Literatur beschränkt sich in diesem Bereich bisher auf die Dokumentation von individuellen prototypischen Ansätzen zur Entwicklung einer webbasierten Benutzeroberfläche. Dies zeigt sich auch in den Beiträgen von [11],[12] und [13]. Ziel dieses Beitrages ist es, die Verwendung von webbasierten Simulationsanwendungen in Wissenschaft und Praxis, durch die Beschreibung eines generischen Vorgehens zur Entwicklung einer webbasierten Benutzeroberfläche für einen Simulationskern, zu fördern.

2 Entwickler Simulationskern für die Materialflusssimulation

Die Anleitung zur Entwicklung der Benutzeroberfläche wird in diesem Beitrag zum besseren Verständnis exemplarisch an einem Beispiel illustriert. Dazu wird ein bereits entwickelter Simulationskern verwendet, welcher in diesem Abschnitt beschrieben wird. In [14] wurde eine Simulationsmethodik zur Dimensionierung von Materialflusssystemen vorgestellt. Diese basiert auf der Modellierung logistischer Systeme mithilfe von generischen Modulen. Der Ablauf der Simulation ist dabei ereignisorientiert. Für diese Methodik wurde ein Simula-

tionskern entwickelt, der die Simulation implementiert. Der Simulationskern wurde in der Programmiersprache python (Version 3.6) implementiert und zur Datenspeicherung wird eine sqlite-Datenbank verwendet. Für die Simulation müssen die folgenden Daten vom Anwender in der Datenbank gespeichert werden:

- Auftragsdaten, dienen der Erzeugung von bewegten logistischen Elementen (z. B. Paletten)
- Simulationsmodell, bestehend aus mehreren miteinander verknüpften Modulen
- Moduleigenschaften und Regeln, definieren das Verhalten der Module

Bisher wurden die Eingangsdaten manuell in der Datenbank abgespeichert. Die Ansteuerung des Simulationskerns erfolgte über das PC-Terminal.

3 Anforderungen an eine webbasierte Benutzeroberfläche für Simulationskerne

Eine Webanwendung besteht grundlegend aus drei Schichten: Die Präsentations-, Verarbeitungs- und Datenhaltungsschicht [15]. Die Präsentationsschicht entspricht in diesem Fall der Benutzeroberfläche, die Verarbeitungsschicht ist der Simulationskern zusammen mit einem Webserver und die Datenhaltungsschicht kann eine Datenbank sein. Für die Benutzeroberfläche wird eine Webseite (Client) verwendet. Der Webserver stellt Dienste für den Client zur Verfügung und realisiert somit auch die Schnittstelle zwischen dem Simulationskern und der Benutzeroberfläche.

Laut VDI 3633 ermöglichen Benutzeroberflächen den Modellaufbau, die Dateneingabe und die Ergebnisdarstellung. Der Modellaufbau kann dabei sowohl mithilfe von graphischen Elementen als auch mit Simulationssprachen erfolgen [1]. Außerdem sollte die Möglichkeit bestehen einzelne Modellbestandteile also Subsysteme zu entwickeln, zu speichern und später wiederzuverwenden. Dies kann den Prozess des Modellaufbaus verkürzen.

Die Dateneingabe umfasst sowohl Modelleigenschaften als auch Experimentdaten [1]. Diese Daten müssen entweder interaktiv in der Benutzeroberfläche eingegeben werden oder durch das Einlesen ganzer Dateien auf dem Server gespeichert werden, sodass durch den Simulationskern darauf zugegriffen werden kann.

Die Ergebnisse eines Simulationslaufes können statisch oder dynamisch dargestellt werden [1]. Die dynamische Darstellung zielt meist auf die Animation der Ereignisse während eines Simulationslaufes ab. Dies dient der Unterstützung der Ablaufkontrolle und der Validierung des Simulationsmodells. Die statische Ergebnisdarstellung dient vor Allem der Darstellung von Kennzahlen, die sich aus Analysen der Simulationsergebnisse ergeben. Die Ergebnisdarstellung kann sowohl parallel zum Simulationslauf als auch im Anschluss daran erfolgen. Außerdem sollte die Benutzeroberfläche dem Benutzer die Möglichkeit bieten die Rohdaten der Simulationsergebnisse zu beziehen.

Zusätzlich zu den genannten Punkten muss der Benutzer mithilfe der Benutzeroberfläche den Simulationskern steuern. Dies umfasst beispielsweise das Starten eines Simulationslaufes oder das Löschen von gespeicherten Daten. Insgesamt ergeben sich somit die folgenden Anforderungen für die Implementierung der Benutzeroberfläche:

Technologische Anforderungen:

- Entwicklungsmöglichkeit für eine Webseite
- Entwicklungsmöglichkeit für einen Webserver
- Realisierung des Datentransfers zwischen Webseite und Server bzw. Simulationskern

Funktionale Anforderungen:

- Steuerung des Simulationskerns
- Hochladen von Daten auf den Server
- Download von Daten auf den PC des Benutzers
- Entwicklung und Speicherung eines Simulationsmodells
- Entwicklung und Speicherung von Modellbestandteilen
- Animation des Simulationslaufes
- Visualisierung der Simulationsergebnisse

4 Anleitung zur Entwicklung einer webbasierten Benutzeroberfläche

Basierend auf den ermittelten Anforderungen, wird im Folgenden die Entwicklung einer webbasierten Benutzeroberfläche für einen bereits existierenden Simulationskern beschrieben. Dabei werden die allgemeinen

Hinweise anhand konkreter Beispiele verdeutlicht. Zunächst müssen die verwendeten Technologien ermittelt werden. Anschließend können die gewünschten Funktionalitäten entwickelt werden.

In [23] wird auszugsweise der zur Implementierung der Benutzeroberfläche entwickelte Quellcode zur Verfügung gestellt.

4.1 Auswahl eines Frameworks zur GUI-Entwicklung

Für die Entwicklung von Webseiten können so genannte Webframeworks oder Bibliotheken als Unterstützung verwendet werden. Diese geben eine gewisse Struktur für die Entwicklung vor und verringern den Implementierungsaufwand, da sie Standardfunktionalitäten bereits anbieten. Beliebte für die Entwicklung von Webseiten sind zum Beispiel das Framework Angular [16] oder die Javascript-Bibliothek React [17]. Ausführliche Vergleiche können in [18] und [19] gefunden werden. Im Folgenden werden zur besseren Lesbarkeit beide Alternativen als Framework bezeichnet.

Bei der Entscheidung zwischen den verfügbaren Frameworks sollten in einem ersten Schritt die folgenden Fragen berücksichtigt werden:

- **Rechtfertigt die Unterstützung durch das Framework den gegebenenfalls entstehenden Overhead?** Teilweise wird durch Verwendung eines Frameworks die Implementierung von sonst nicht benötigtem Code notwendig. Hier ist eine Abwägung sinnvoll, welches Framework am geeignetsten ist. Da es sich bei der Entwicklung einer Benutzeroberfläche für einen Simulationskern allerdings nicht um ein triviales Problem handelt, kann diese Frage in der Regel mit ja beantwortet werden.
- **Welcher Lizenz unterliegt das Framework?** Es ist grundsätzlich zu prüfen ob die Lizenz des Frameworks zu den Anforderungen an das Simulationswerkzeug passt.
- **Können alle erforderlichen Funktionalitäten mit dem Framework realisiert werden?** Ist dies nicht der Fall, muss ein anderes Framework gewählt werden. Die genannten Frameworks sind in der Regel aber geeignet, um alle hier benötigten Funktionalitäten zu implementieren.
- **Gibt es Vorkenntnisse in einem der Tools?** Vorkenntnisse können den zeitlichen Aufwand für die

Entwicklung der Webseite deutlich verkürzen, da die Einarbeitungszeit entfällt.

- **Wie groß ist die unterstützende Community?** Bei einer großen Community können mehr Problemlösungen online gefunden oder in Foren geteilt werden.
- **Welches Framework entspricht dem persönlichen Geschmack?** Bei der Wahl zwischen den etablierten Frameworks kann keine falsche Entscheidung getroffen werden, sodass persönliche Vorlieben berücksichtigt werden sollten.

Das Framework Angular bringt viele vordefinierte Lösungen für die Webseitenentwicklung mit, wie zum Beispiel ein Modul für den Datentransfer zwischen Client und Server. Außerdem wird durch die Aufteilung in Komponenten eine klare Struktur vorgegeben und es werden nur wenige Erweiterungen für die geplanten Funktionalitäten der Benutzeroberfläche benötigt. Da das Framework zusätzlich einer MIT-Lizenz unterliegt wurde sich in diesem Kontext für Angular als Framework für die Webseitenentwicklung entschieden. Für das Beispiel wurde Angular Version 5.2.11 verwendet.

Mit Angular können Single Page Anwendungen (SPA) mit HTML, CSS und Typescript entwickelt werden. Typescript ist ein Superset von Javascript und erlaubt die Implementierung der Funktionalität der Webseite. Angular basiert auf dem Konzept der Komponenten und der Dependency Injection. Komponenten beschreiben gekapselte wiederverwendbare Funktionalitäten bzw. Sichten (Views). Diese Komponenten können wiederum so genannte Services verwenden, die Funktionalitäten anbieten, die unabhängig von einer speziellen Sicht sind [20]. Das Angular-Tutorial „Tour of Heroes“ [21] bietet bei Bedarf einen guten Einstieg in das Framework.

Nach der Initialisierung eines neuen Angular-Projektes stehen in diesem die folgenden Dateien bereit:

- `app.module.ts`: Definition von global verwendeten Bibliotheken und Komponenten.
- `app.component.html`: Inhalt der obersten Komponente.
- `app.component.css`: Design der obersten Komponente.
- `app.component.ts`: Funktionalitäten der obersten Komponente.

- `app.component.spec.ts`: Test der obersten Komponente.

Da eine Benutzeroberfläche diverse Funktionalitäten erfüllen soll, muss diese für eine gute Übersichtlichkeit strukturierbar sein. Bei Webseiten ist die Verteilung von Inhalten auf mehrere Ansichten ein gängiges Hilfsmittel. Für die Benutzeroberfläche der Materialflusssimulation wurden der Datenaustausch, die Steuerung des Simulationskerns und die Modellierung voneinander getrennt. Bei Angular entspricht jede dieser Sichten einer eigenen Komponente, die je nach Bedarf in der Hauptkomponente angezeigt wird. Angular bietet für die Navigation zwischen den Sichten ein so genanntes `RouterModule` an (vgl. [22]). Um eine Navigationsleiste wie in Abb. 1 (oben) zu realisieren, wurde der in [23] gezeigte Code implementiert.

4.2 Aufsetzen eines Webservers

Um den Simulationskern für den Benutzer über die Webseite erreichbar zu machen, muss ein Webserver aufgesetzt werden. Analog zur Webseitenentwicklung gibt es Frameworks und Bibliotheken, die die Entwicklung des Webservers unterstützen. Für die Auswahl können die gleichen Fragen als Unterstützung beantwortet werden. Frameworks und Bibliotheken sind für alle gängigen Programmiersprachen verfügbar. Für python sind bekannte Beispiele Flask [24] und Django [25]. Ein ausführlicher Vergleich ist in [26] zu finden. Flask ermöglicht die Implementierung aller notwendigen Funktionalitäten für die hier entwickelte Anwendung. Dies ist vor Allem die Ansteuerung der Funktionen des Simulationskerns und die Abwicklung des Datentransfers zwischen Webseite und Server. Zudem wird auf Overhead im Vergleich zu Django verzichtet und unterliegt der BSD 3-Lizenz. Daher wurde Flask für die Entwicklung des Webservers verwendet.

Der Webserver wird mit dem in [23] gezeigten Code implementiert. Dabei ist die Standardeinstellung der Adresse unter der der Server erreichbar ist 'localhost', also ein lokaler Server auf dem genutzten Rechner.

4.3 Datentransfer

Die Kommunikation und somit der Datentransfer zwischen Server und Client bzw. Simulationskern und graphischer Benutzeroberfläche wird mithilfe des Hypertext Transfer Protokolls (HTTP) durchgeführt. Dieses basiert auf dem Anfrage-Antwort-Prinzip, wobei der

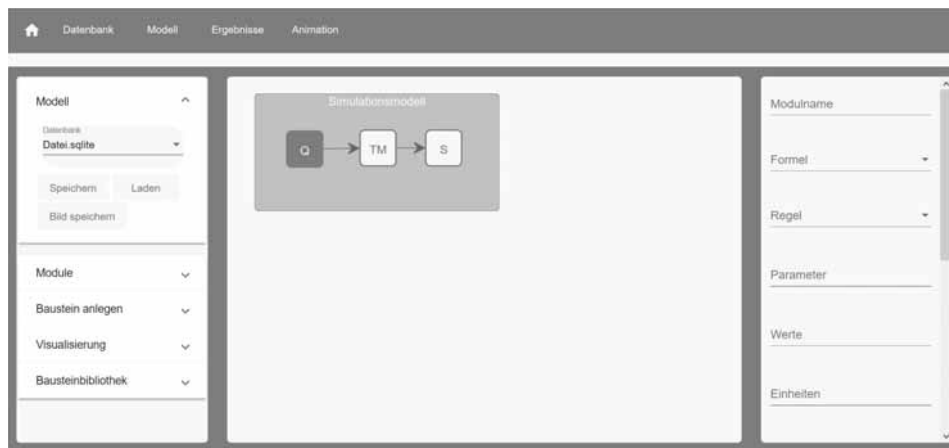


Abbildung 1: Screenshot des graphischen Modelleditors. Im oberen Bereich der Ansicht befindet sich die Navigationsleiste.

Client eine Anfrage sendet und der Server die Antwort zurück gibt. Primär werden für den Datentransfer zwei Methoden verwendet. Dies sind die GET- und die POST-Methode. Mit der GET-Methode werden Daten vom Server angefordert und mit der POST-Methode werden Daten an den Server übermittelt [27]. Nach der Bearbeitung der Anfrage des Clients sendet der Server eine Antwort an den Client. Dies können zum Beispiel vom Client angeforderte Daten sein. Werden bei der Kommunikation Anfragen vollständig nacheinander abgearbeitet, wird also auf die Antwort des Servers gewartet bevor die nächste Anfrage gesendet wird, spricht man von einer synchronen Kommunikation. Das Framework Angular verarbeitet die Anfragen asynchron, also mehrere Anfragen parallel. Bei Bedarf muss eine synchrone Verarbeitung also erzwungen werden.

Bei der reinen Kommunikation mittels HTTP ist zu beachten, dass der Server nur auf Anfragen des Clients antworten kann und die Kommunikation nicht selbstständig steuert. Ein bidirektionaler Datenaustausch ist beispielsweise mit Websockets möglich ist. Dies wird im Folgenden allerdings nicht weiter betrachtet, da diese Kommunikation für die Grundfunktionalität der Benutzeroberfläche nicht notwendig ist. Bei Bedarf finden sich weitere Informationen zu Websockets in [15]. Typischerweise wird für den Datentransfer zwischen Client und Server das JavaScript Object Notation (JSON)-Datenformat verwendet [27].

Übermittlung von Daten zum Server

Für die Übermittlung und Anforderung von Daten von bzw. auf den Server werden bei Angular Services

implementiert. Diese können anschließend von allen Komponenten verwendet werden. In [23] ist die Implementierung einer Post-Methode in Angular verfügbar. Neben Daten können auch ganze Dateien an den Server gesendet werden. Dies ist zum Beispiel notwendig, wenn der Benutzer eine ganze csv-Datei an den Server übermitteln möchte. Der Service dient auch dazu, die vom Server zurückgesendete Erfolgs- oder Fehlermeldung zu empfangen und gegebenenfalls ein entsprechendes Feedback an den Nutzer zu initialisieren.

Eine POST-Methode kann zusätzlich zum reinen Datentransfer auch als Steuerungsmethode für den Simulationskern verwendet werden. Dazu wird die Aktions-Codezeile durch den gewünschten Aufruf der Funktion des Simulationskerns ersetzt.

Anforderung von Daten

Die Anforderung von Daten vom Server erfolgt analog zur Übermittlung von Daten. Eine Besonderheit stellt aber der Download von Dateien auf den Anwender-PC dar. Aus Sicherheitsgründen ist dies nur mithilfe eines Download-Dialogs möglich. Im Kontext der Benutzeroberfläche ist der Download beispielsweise notwendig wenn dem Anwender die Rohdaten der Simulationsergebnisse als .csv-Datei zur Verfügung gestellt werden sollen. In [23] ist der entsprechende Code abrufbar.

4.4 Graphischer Modelleditor

Die Entwicklung und Speicherung eines Simulationsmodells ist ein zentraler Bestandteil einer Benutzeroberfläche für einen Simulationskern. Zur Unterstützung

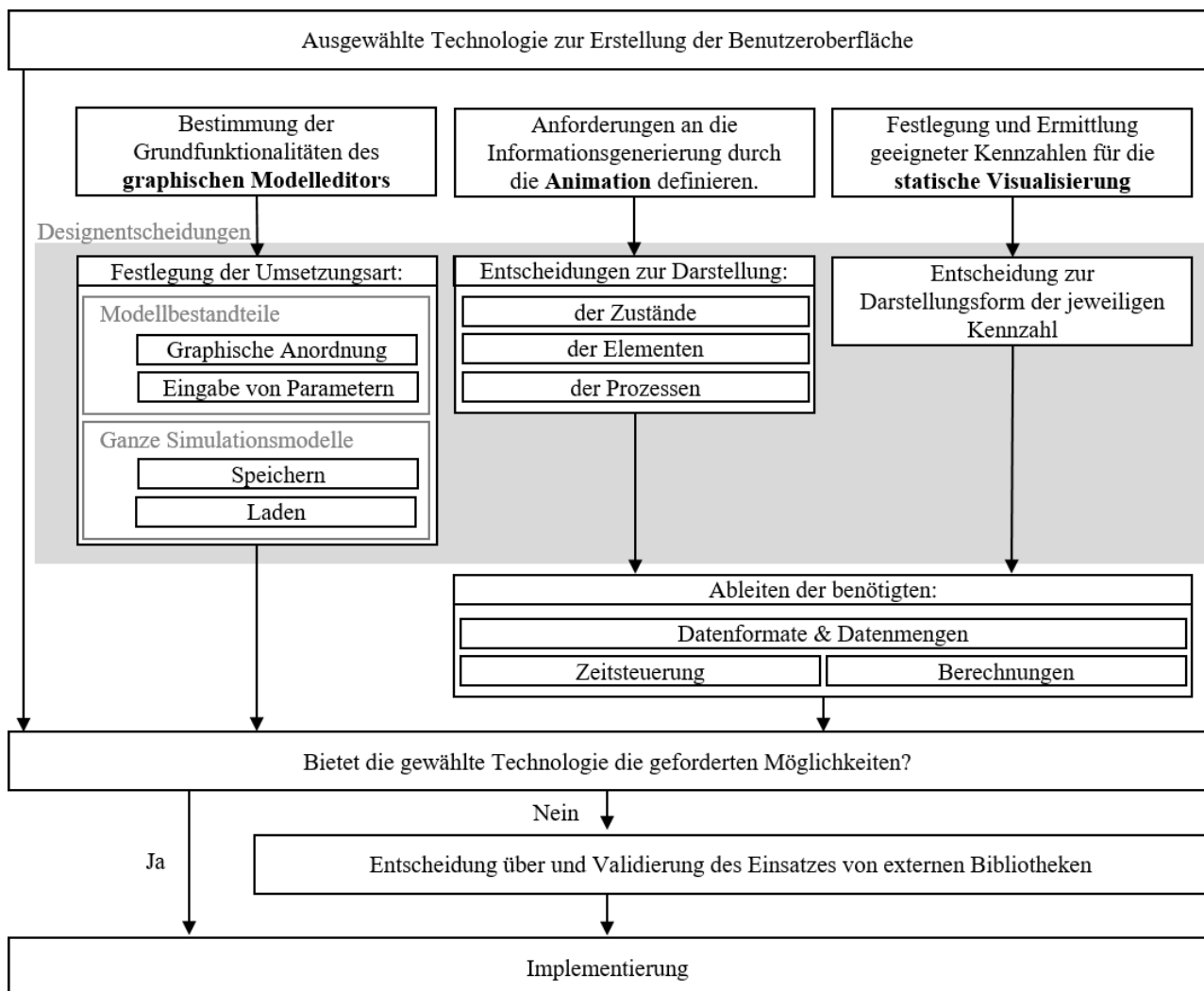


Abbildung 2: Entscheidungsbaum zur Entwicklung des Modelleditors (links), der Animation (mitte) und der statischen Visualisierung von Simulationsergebnissen (rechts).

der Modellierung haben sich graphische Modelleditor als hilfreich erwiesen. Eine Übersicht über die Entscheidungen zur Entwicklung des Modelleditors ist in Abb. 2 dargestellt. Die benötigten Grundfunktionalitäten eines solchen Modelleditors sind

- Graphische Anordnung von Modellbestandteilen
- Eingabe von Parametern der Modellbestandteile
- Speichern von ganzen Simulationsmodellen
- Laden von gespeicherten Simulationsmodellen.

In vielen Bereichen in denen Simulationswerkzeuge eingesetzt werden, werden sehr große und komplexe

Simulationsmodelle entwickelt. Um die Verständlichkeit und Übersichtlichkeit der Darstellung zu erhöhen, sollte zum Beispiel die Möglichkeit geschaffen werden, Bereiche des Simulationsmodells farblich hervorzuheben (vgl. den roten Bereich in Abb. 1). Außerdem wird empfohlen eine Zoom-Funktionalität zu implementieren, um den Gesamtmodellüberblick auch bei Modellen, welche die Größe des Editorfensters überschreiten, zu gewährleisten.

Für den in Abschnitt 2 beschriebenen Simulationskern wurde ein Modelleditor entwickelt, der in Abb. 1 dargestellt ist. Auf der rechten Seite der Seitenansicht kann die gewünschte Funktionalität ausgewählt werden,

in der Mitte werden die Simulationsmodelle zusammengestellt und rechts können die Moduleigenschaften eingestellt werden. Zusätzlich zu den genannten Funktionalitäten können so genannte Bausteine, welche aus mehreren Modulen bestehen, abgespeichert und wiederverwendet werden, rein graphische Elemente wie Rechtecke können verwendet werden, um das Modell visuell zu strukturieren und übersichtlicher zu gestalten und es besteht die Möglichkeit ein Bild des Simulationsmodells herunterzuladen. Bei der Aufteilung der Seite sollte die Modellierungsfläche möglichst groß gestaltet werden, um auch die Modelle möglichst großflächig darstellen zu können. Weitere Hinweise zur Gestaltung von Benutzeroberflächen finden sich zum Beispiel in [29] und [28].

Anordnung von Modellbestandteilen mit Drag & Drop

Modelle des entwickelten Simulationskerns bestehen aus generischen Modulen, welche verbunden werden, um ganze Materialflusssysteme abzubilden. Für die graphische Modellierung wurden daher Quadrate als Abbildung eines Moduls und gerichtete Verbindungslinien für die Darstellung der Verbindung zwischen diesem Modulen gewählt (vgl. Abb. 1). Drag & Drop erlaubt ein intuitives und visuell direkt erfassbares Anordnen von Modellbestandteilen und kann dadurch die Modellierung erleichtern. Die Verbindungen können durch Anklicken eines Moduls und Ziehen der Maus zu einem anderen Modul geschaffen werden. Für die Implementierung der Drag & Drop-Funktionalität wurde die Bibliothek `jsplumb community edition` [30] in Kombination mit `jQuery` [31] verwendet (vgl. [23]).

Die aktuelle Version `Angular 9.0.0` bietet selbst ein Modul um drag & drop zu implementieren. Für zukünftige Entwicklungen sollte daher die Eignung dieser Bibliothek für die Implementierung der Drag & Drop-Funktionalität untersucht werden.

Parametereingabe

Für die Parametereingabe können Input-, Dropdown- und autofill-Felder verwendet werden. Soll der Benutzer nur aus einer vordefinierten Menge an Attributen auswählen können, bieten sich besonders Dropdown-Felder an, da diese nur vorgegebene Attribute anzeigen und zur Auswahl stellen. Diese Attribute können entweder vom Server angefragt oder in der Benutzeroberfläche hinterlegt werden. Inputfelder sollten nur verwendet werden, wenn dem Benutzer keine Vorga-

ben bezüglich der Eingabe gemacht werden. Autofill-Felder können in beiden Fällen verwendet werden. Dabei dient die Autovervollständigung zum einen als Unterstützung, wenn lange Eingaben notwendig sind.

Im betrachteten Beispiel werden die Eingabedaten direkt als Attribute der Modul-Elemente auf der Zeichenfläche gespeichert. Für die Ansicht der Moduleigenschaften auf der rechten Seite der Benutzeroberfläche wird eine `ComponentFactory` verwendet, die bei einem Klick auf ein Modul die Eigenschaften-Komponente erzeugt (siehe [23]).

Speichern & Laden des Simulationsmodells

Beim Speichern des Simulationsmodells wird dieses zunächst in ein json-Format überführt und anschließend an den Server übermittelt. Dort kann das Modell direkt in der für den Simulationskern benötigten Form abgespeichert werden. In diesem Fall wird es in der `sqlite`-Datenbank abgelegt. Unabhängig davon ob der Simulationskern die graphischen Informationen, also Position der einzelnen Modellbestandteile im Editor benötigt, sollte auf dem Server eine json-Datei abgelegt werden, die diese Informationen enthält. Damit reicht es bei dem Ladevorgang eines Modells aus, diese Datei an die Webseite zu übermitteln und einer aufwändige Datenverarbeitung bzw. Informationsrückgewinnung wird vorgebeugt. Das Speichern und Laden von Bausteinen funktioniert analog zum Speichern eines Modells.

Bild des Simulationsmodells speichern

Häufig ist es erwünscht ein Bild eines Simulationsmodells abzuspeichern. Hierzu stehen Bibliotheken zur Verfügung, die die Aufnahme eines Screenshots unterstützen. Im Beispiel wurde die Bibliothek `domtoimage` [32] verwendet (vgl. [23]). Mithilfe der Bibliothek werden alle Elemente, die einer ausgewählten Komponente zugeordnet sind, in einem Bild gespeichert.

4.5 Animation eines Simulationslaufs

Die Animation dient im Rahmen der Materialflusssimulation dem Verständnis von komplexen Zusammenhängen [1]. Es gilt zu entscheiden wie dies im konkreten Anwendungsfall bestmöglich umgesetzt werden soll. Dazu sollten grundsätzliche Entscheidungen über die Darstellung der Elemente, der Zustände und der Prozesse des Modells getroffen werden. Aus diesen leiten sich die Anforderungen an die Datenformate und -mengen sowie die Zeitsteuerung ab. Bietet die ausge-

wählte Technologie zur Erstellung der Benutzeroberfläche bereits die Möglichkeit zur Animationserstellung so wird diese implementiert. Anderenfalls geht diesem Schritt die Entscheidung und die Validierung zum Einsatz von externen Bibliotheken zur Animationsrealisierung voraus (vgl. Abb. 2). Innerhalb der betrachteten Materialflusssimulation sind nach dem geschilderten Vorgehen zur Umsetzung der Animation folgende Entscheidungen getroffen worden. Die Form und Anordnung der Modellelemente wird dem Modell Aufbau entsprechend übernommen. Zustände werden zum einen durch „Füllstände“ und zum anderen durch sich auf den Modulen befindliche Elemente angezeigt. Die Materialflüsse zwischen den Modulen werden ihrer Richtung und ihrem mengenmäßigen Umfang gemäß durch Pfeile in unterschiedlicher Stärke angezeigt. In Abb. 3 ist eine schematische Übersicht zu den Darstellungsformen der Module gegeben. Bei der technischen Umsetzung der graphischen Darstellung wurden SVG (Scalable Vector Graphics) Elemente verwendet, da diese sich gut in das Angular Framework einfügen und manipulieren lassen, um somit die dynamischen Veränderungen abzubilden. Zur Steuerung der Systemzeit wurde die JavaScript Bibliothek `RxJS` (Reactive Extensions Library for JavaScript) [33], welche in das Angular Framework eingegliedert ist, verwendet.

4.6 Visualisierung von Simulationsergebnissen

Zur Unterstützung der Ergebnissinterpretation eines oder mehrerer Simulationsläufe sind diese aufzubereiten und graphisch darzustellen [1]. Die hierfür notwendigen Entscheidungen erfolgen analog zu denen für die Animation und sind in angepasster Form Abb. 2 zu entnehmen. Die Ermittlung geeigneter Kennzahlen und die Form ihrer graphischen Darstellung bilden die Grundlage für die abzuleitenden Datenformate und -mengen sowie die benötigten Berechnungen. Für die Umsetzung der Diagramme wurde das open source Framework `ngx-charts` eingebunden, da dieses speziell für die Verwendung in Angular-Projekten entwickelt wurde [34]. Eine exemplarische Darstellung ausgewählter Kennzahlen ist in Abb. 4 gezeigt.

5 Fazit

Jeder Entscheidung zur verwendeten Technologie oder Gestaltung einer Funktionalität der Benutzeroberfläche

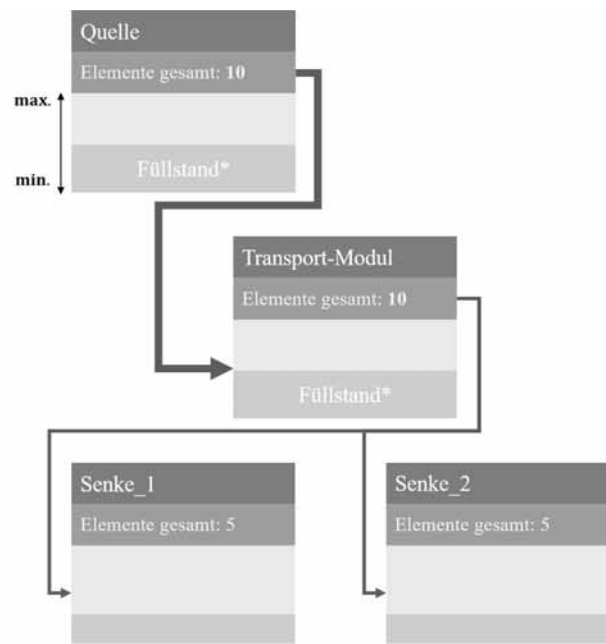


Abbildung 3: Übersicht zur gewählten Darstellungsform der Module in der Animation. * Der Füllstand zeigt den Grad der Auslastung des Moduls an.

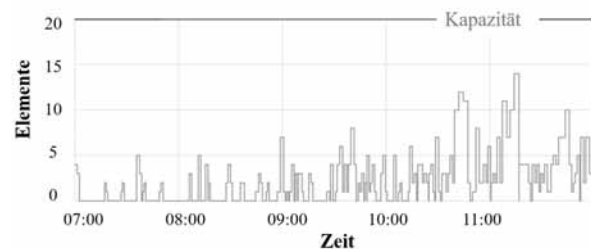


Abbildung 4: Exemplarische Ansicht der graphischen Darstellung ausgewählter Kennzahlen.

sollte diese Frage zu Grunde liegen: Wird dadurch die Bedienung des Simulationswerkzeugs erleichtert und der Aufwand für die Durchführung der Simulation verringert? Zusätzlich ergibt sich der in Abb. 5 dargestellte Entscheidungsbaum, welcher verwendet werden kann, um zu überprüfen ob eine Technologie für die Realisierung einer Funktionalität geeignet ist. Hierbei ist zu beachten, dass jeweils die Frage der Sinnhaftigkeit des Einsatzes einer Technologie vorher zu klären ist, ob also beispielsweise ein Webentwicklungsframework wie Angular überhaupt notwendig ist. Insgesamt ergibt sich damit für die Entwicklung der webbasierten Benutzeroberfläche der in Abb. 6 dargestellte Ablauf, wo-

bei Technologien für alle Funktionalitäten ausgewählt werden müssen. Der Entscheidungsbaum findet bei der Eignungsprüfung der Technologien Anwendung.

Bei der Entwicklung der Benutzeroberfläche für den in Abschnitt 2 vorgestellten Simulationskern, hat sich besonders die Wahl des Frameworks für die Entwicklung der Webseite als Entscheidung mit großer Tragweite herausgestellt. Dieses gibt die grundlegenden Rahmenbedingungen wie zum Beispiel die Programmiersprache und strukturelle Konzepte vor. Auf der einen Seite bedeutet das Framework eine Verringerung des Implementierungsaufwands, kann aber auf der anderen Seite auch die Implementierungsfreiheit einschränken, sodass dieses mit besonderer Sorgfalt ausgewählt werden sollte.

Weiterhin hat sich gezeigt, dass durch die Implementierung der webbasierten Benutzeroberfläche teilweise Mehraufwände im Vergleich zu Desktopanwendungen entstehen können. Zum Beispiel ist der Zugriff auf Daten und das zur Verfügung stellen von Dateien für den Anwender mit einem höheren Implementierungsaufwand verbunden. Allerdings übersteigen die Vorteile der WBS und die damit verbundene Zeiteinsparung diesen einmaligen zusätzlichen Aufwand, sodass webbasierte Benutzeroberflächen deutlich häufiger Anwendung finden sollten. Die hier beschriebene Anleitung kann dazu beitragen, dass zukünftig mehr webbasierte Benutzeroberflächen entwickelt werden.

Literatur

- [1] Verein Deutscher Ingenieure e.V. *VDI 3633: Simulation von Logistik-, Materialfluss- und Produktionssystemen - Grundlagen*. VDI-Richtlinien. 2014.
- [2] Bossel, H. *Systeme, Dynamik, Simulation: Modellbildung, Analyse und Simulation komplexer Systeme*. 1st ed. Norderstedt: Books on Demand GmbH; 2004. 400 p.
- [3] Wenzel, S. *Simulation logistischer Systeme*. In: Tempelmeier, H. editors. *Modellierung logistischer Systeme. Fachwissen Logistik*. 1st ed. Berlin, Heidelberg: Springer Vieweg; 2018. p 1 -34.
- [4] Byrne, J., Heavey, C., Byrne, P.J. A review of web-based simulation and supporting tools. *Simulation Modelling Practice and Theory*. 2010; 18(3): 253–276. doi: 10.1016/j.simpat.2009.09.013.
- [5] Odhabi, H.I., Paul, R.J. Macredie, R.D. Developing a graphical user interface for discrete event simulation. In Medeiros, D.J., Watson, E.F., Carson, J.S.,

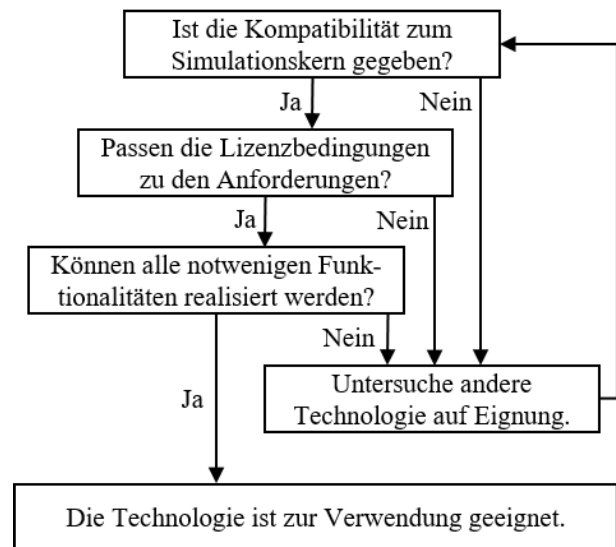


Abbildung 5: Entscheidungsbaum zur Auswahl geeigneter Technologien.

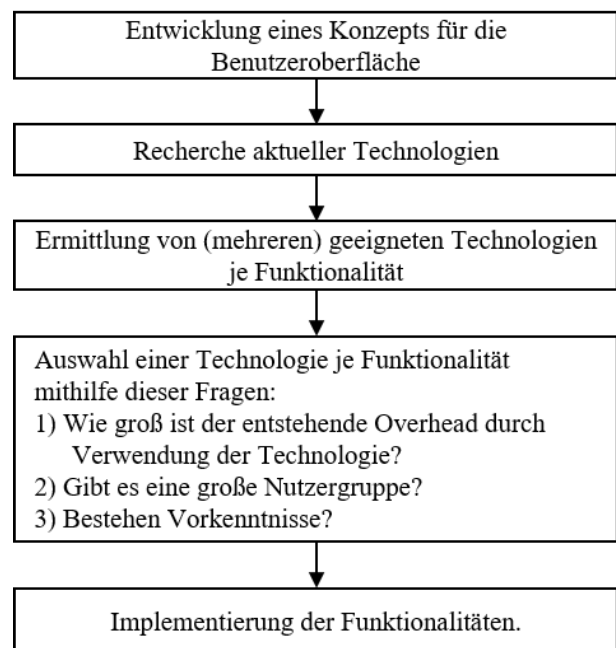


Abbildung 6: Ablauf der Entwicklung webbasierter Benutzeroberflächen für Simulationskerne.

- Manivannan, M.S., editors. Proceedings of the 1998 Winter Simulation Conference. *Winter Simulation Conference*; 1998 Dec; Washington, DC. p. 429–436.
- [6] Syberfeldt, A., Karlsson, I., Ng, A., Svantesson, J., Almgren, T. A web-based platform for the simulation –optimization of industrial problems. *Computers*

- Industrial Engineering*. 2013; 64(4): p. 987–998
- [7] Page, E.H., Buss, A., Fishwick, P.A., Healy, K.J., Nance, R.E., Paul, R.J. Web-Based Simulation: Revolution or Evolution? *ACM Transactions on Modeling and Computer Simulation*. 2000; 10(1): p. 3–17.
- [8] Kuljis, J., Paul, R.J. A review of web based simulation: whither we wander? In Joines, J.A., Barton, R.R., Kang, K., Fishwick, P.A., editors. Proceedings of the 2000 Winter Simulation Conference. *Winter Simulation Conference*. 2000 Dec; Orlando, Florida. p. 1872–1881.
- [9] Heavey, C., Dagkakis, G., Barlas, P., Papagiannopoulos, I., Robin, S., Mariani, M., Perrin, J. Development of an open-source discrete event simulation cloud enabled platform. In Tolk, A., Diallo, S.Y., Ryzhov, I.O., Yilmaz, L., Buckley, S., Miller, J.A., editors. Proceedings of the 2014 Winter Simulation Conference. *Winter Simulation Conference*; 2014 Dec; Savannah, Georgia. p. 2824–2835.
- [10] Nance, R.E., Overstreet, C.M. History of computer simulation software: an initial perspective. In Chan, W.K., D’Ambrogio, A., Zacharewicz, G., Mustafee, N., Wainer, G., Page, E.H., editors. Proceedings of the 2017 Winter Simulation Conference. *Winter Simulation Conference*. 2017 Dec; Las Vegas, Nevada. p. 243–261.
- [11] Gan, B.P., Liu, L., Ji, Z., Turner, S.J., Cai, W. Managing event traces for a web front-end to a parallel simulation. In Peters, B.A., Smith, J.S., Medeiros, D.J., Rohrer, M.W., editors. Proceedings of the 2001 Winter Simulation Conference. *Winter Simulation Conference*. 2001 Dec; Arlington, Virginia. p. 637–644.
- [12] März, L., Interaktives Montageplanungssystem zur Online- Leistungssteuerung. In Dangelmaier, W., Laroque, C., Klaas, A., editors. Simulation in Produktion und Logistik. Entscheidungsunterstützung von der Planung bis zur Steuerung. *15. ASIM Fachtagung*. 2013 Oct; Paderborn. p. 661–668.
- [13] Bergmann, S., Parzefall, F., Straßburger, S. Webbasierte Animation von Simulationsläufen auf Basis des Core Manufacturing Simulation Data (CMSD) Standards. In Wittmann, J., Deatcu, C., editors. Tagungsband ASIM 2014. 22. *Symposium Simulationstechnik*. 2014 Sep; Berlin. p. 63–70.
- [14] Hilbrich, S., Köck, H., Hinckeldeyn, J., Kreutzfeldt, J. Entwicklung einer Simulationsmethodik zur schnellen Dimensionierung komplexer Materialflusssysteme. *Logistics Journal: Proceedings*. 2017; Vol 2017. doi: 10.2195/lj_Proc_hilbrich_de_201710_01.
- [15] Abts, D. *Masterkurs Client/Server-Programmierung mit Java*. 5th ed. Wiesbaden: Springer Vieweg; 2019. 389 p.
- [16] Google. Angular Docs. <https://angular.io/docs>. 2020.
- [17] Facebook Inc. Getting Started. <https://reactjs.org/docs/getting-started.html>. 2020.
- [18] Isaak Tappert. JavaScript Frameworks im Vergleich: Vue vs. Angular vs. React. <https://entwickler.de/online/javascript/angular-vs-react-vs-vue-js-579921249.html>. 2020.
- [19] X-Company Pty Ltd. React vs angular: Their Biggest differences. <https://x-team.com/blog/react-vs-angular/>. 2019.
- [20] Google. Introduction to Angular concepts. <https://angular.io/guide/architecture>. 2020.
- [21] Google. Tour of Heroes app and tutorial. <https://angular.io/tutorial>. 2020.
- [22] Google. Add in-app navigation with routing. <https://angular.io/tutorial/toh-pt5>. 2020.
- [23] Hilbrich, S., Gerdes, K., Hinckeldeyn, J., Kreutzfeldt, J. WBS-GUI-Entwicklung. <https://collaborating.tuhh.de/w-6/forschung/wbs-gui-entwicklung.2020>.
- [24] Pallets. Flask. <https://flask.palletsprojects.com/en/1.1.x/>. 2010.
- [25] Django Software Foundation. Django. <https://djangoproject.com>. 2020.
- [26] TestDriven Labs. Django vs. Flask in 2019: Which Framework to Choose. <https://testdriven.io/blog/django-vs-flask/>. 2020.
- [27] Clow, M. *Angular 5 Projects: Learn to Build Single Page Web Applications Using 70+ Projects*. 1st ed. New York City: Apress; 2018. 458 p.
- [28] MacDonald, D. *Practical UI Patterns for Design Systems*. 1st ed. New York City: Apress; 2019. 293 p.
- [29] Shneiderman, B. et al. *Designing the user interface: strategies for effective human-computer interaction*. 6th ed. London: Pearson; 2017. 616 p.
- [30] JSPLUMB UK LTD. jsplumb. <http://jsplumb.github.io/jsplumb/home.html>. 2020.
- [31] The jQuery Foundation. jQueryAPI. <https://api.jquery.com/>. 2020.
- [32] Saienko, A. DOM to image. <https://github.com/tsayen/dom-to-image>. 2017.
- [33] Lesh, B. et al. RxJS. <https://rxjs-dev.firebaseapp.com/>. 2020.
- [34] Swimlane. ngx-charts. <https://swimlane.gitbook.io/ngx-charts/>. 2020.

NSA-DEVS: Combining Mealy behaviour and Causality

Peter Junglas*

Dep. of Engineering "Dr. Jürgen Ulderup", PHWT Vechta/Diepholz, Schlesierstr. 13a, 49356 Diepholz, Germany;
*peter@peter-junglas.de

Abstract. The RPDEVS ("Revised PDEVS") formalism has been introduced to allow for a simple description of Mealy-type components that behave consistently. This made it necessary to change the way the simulator handles event chains. Using a simple example model we show that the proposed algorithm has serious problems with the resulting sequence of concurrent events. Therefore we introduce NSA-DEVS, a variant formalism that is inspired by ideas from non-standard analysis (NSA). It uses infinitesimal time delays to make a natural ordering of concurrent events easy, without the need to fix lots of additional parameters. As proof of concept we describe the example model in NSA-DEVS and implement it in a suitably twisted RPDEVS simulator.

Introduction

More than 40 years after its invention the DEVS formalism [1] and its most popular variant PDEVS [2] are now standard tools for the mathematical modeling of discrete-event systems. If in doubt a quick search for "DEVS formalism" in Google scholar reveals over 6000 papers and that the seminal book of Zeigler et al. [3] has been cited about 7000 times.

Looking at widely-used simulation environments, the picture changes completely: Neither Arena [4] nor SimEvents [5] use DEVS internally or even mention it in their documentation. And though Mathworks has based its redesign of SimEvents on a proper modeling formalism, the developers didn't choose DEVS for this purpose [6].

On the other hand there are quite a few free simulation programs available that use DEVS or one of its variants for the definition of atomic components and the implementation of coupled systems [7]. But all of them twist the original DEVS formalism to make it a suitable foundation for a concrete simulation environment [8]. Some of the problems are just minor nuisances, like the addition of input and output ports, others are of a more fundamental nature.

Probably the most serious flaw has been named by

Preyser et al., who show in [9], that PDEVS has difficulties modeling certain Mealy-type components: The necessary introduction of transitional states leads to delays that change the expected order of concurrent events and the behaviour of subsequent components. This is a serious drawback, if one wants to define a library of reusable blocks. Therefore the PDEVS formalism has been altered in [10] to allow for Mealy-like behaviour thereby introducing the revised version RPDEVS.

To make this work, one has to change the way chains of concurrent events are handled, which is a complex and possibly dangerous endeavour. Even after the careful analysis in [10] and the formal definition of an RPDEVS simulator [11] the question remains, whether the proposed scheme is capable of handling the subtle problems that appear in practical modeling tasks.

To further investigate the status of RPDEVS we will introduce a simple example that is plagued by a complex causal structure of concurrent events, and implement it in PDEVS and RPDEVS, using freely available simulators.

Since the results show that RPDEVS has problems with the example model, we will propose a different way of how to cope with concurrent event chains, which uses concepts of non-standard analysis [12]. After a short introduction to the basic mathematical ideas, we will define the new DEVS variant NSA-DEVS, which combines the ideas of RPDEVS with a more robust method to handle concurrent events.

Finally we will implement the standard example in NSA-DEVS and find that it can handle this model in a clear-cut, easily understandable way. This supports the assumption that NSA-DEVS might be a good basis for concrete modeling and simulation environments, since it combines the security in the handling of concurrent events from PDEVS with the modeling power of RPDEVS.

1 Singleserver - a fundamental example

The singleserver example used in the following consists of a generator that creates entities in fixed time intervals $t_G = 1$ and sends them to a queue, which is connected to a simple server with fixed service time $t_S = 1.5$. Entities leaving the server are terminated (cf. Fig. 1). Additionally the queue and the server output the current number of entities stored.

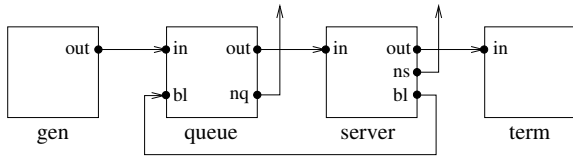


Figure 1: Example model singleserver.

Though this is probably by far the most studied system in discrete modeling, it is not trivial at all, especially if you try to model it with PDEVS. In the fundamental book of Zeigler et al. [3] a queue-server combination is modelled as one atomic component. But trying to create separate atomic models for a queue and a server is much more challenging due to the complex interaction of the two components.

The server can be implemented easily using the state diagram shown in Figure 2: When an entity E arrives, the server outputs the new blocking status and changes to the “busy” state. After the service time, it outputs the entity and the changed blocking status and returns to the “idle” state. In this and the following figures the annotation $(A)B/C$ on an arrow means: If condition A is true and input is B , then the output is C and the state changes. Any of the three parts may be missing.

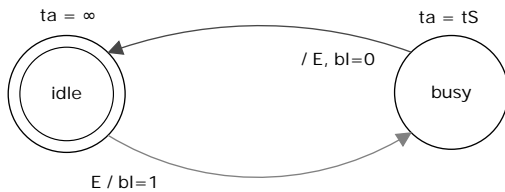


Figure 2: State diagram of the server component.

The behaviour of the queue is much more complicated, it is modelled here using the state diagram in Figure 3. The four states are distinguished by the size of the queue (“empty”, “queuing”) and the blocking status at

the output of the queue (“free”, “blocked”). The only internal transitions occur in the state “queuing free”, they output an entity and have zero transition time. All other transitions are external, triggered by an incoming entity or a new blocking status.

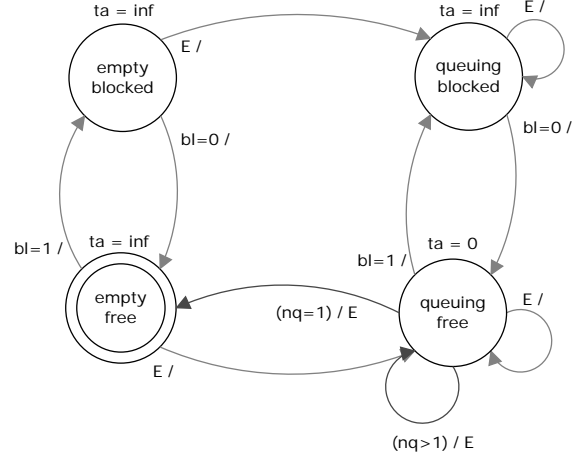


Figure 3: State diagram of the queue component.

This implementation of a queue uses a push strategy, where entities proceed as far as possible, until they are blocked. This is the basic idea behind many discrete event simulators, from GPSS to SimEvents. Alternatively, one could use a pull strategy, where entities only proceed, when they are called by a component. This would lead to a slightly simpler implementation of our example. It is an interesting question, whether pull or push strategies are better suited for complex simulation environments, but not the point of this investigation.

The simulation of the complete singleserver model leads to complicated cascades of concurrent events. For a typical example assume that the queue is in state “queuing blocked” with $nq > 1$ and the server gets ready, going from “busy” to “idle”. It sends its new blocking status $bl = 0$ to the queue, which now transitions to “queuing free”. Using an internal transition the queue outputs an entity, which arrives at the server, leading to a transition to “busy” and the sending of $bl = 1$ back to the queue. Now the queue has to change to “queuing blocked”, before another entity is output via an internal transition.

2 Implementing singleserver with PDEVS

The fundamental component in the PDEVS formalism is an *atomic PDEVS* [3]. It is formally described by an 8-tuple $\langle X^b, S, Y^b, ta, \delta_{int}, \delta_{ext}, \delta_{con}, \lambda \rangle$ with the meanings

X^b	set of possible input bags
S	set of states
Y^b	set of possible output bags
$ta : S \rightarrow [0, \infty]$	time advance function
$\delta_{int} : S \rightarrow S$	internal transition function
$\delta_{ext} : Q \times X^b \rightarrow S$	external transition function
$\delta_{con} : S \times X^b \rightarrow S$	confluent transition function
$\lambda : S \rightarrow Y^b$	output function

where an element of Q combines a state and the time since the last internal transition, i.e.

$$Q := \{(s, e) | s \in S, e \in [0, ta(s)]\}.$$

It is important to note, especially for the present discussion, that the output function λ is only called directly before an imminent internal transition.

Atomic components can be combined to form a *coupled PDEVS*, which is formally defined by the set of components and their internal and outwards connections.

To implement the singleserver example in PDEVS one has to augment the state diagrams with transitional states that allow to produce output values, when an input appears, i. e. at an external event. For the server component (Fig. 2) one additional state is sufficient, and the definition of the transition, output and time advance functions is straightforward.

The definition of the queue component (Fig. 3) is much more complicated, its extended state diagram contains five additional states and a lot of corresponding additional transitions (Fig. 4). The purpose of the four states “n out A/B/C/D” and their corresponding transitions is evident: Whenever an entity arrives, the length of the queue changes, and a corresponding output value has to be sent.

But the new states lead to further complications, in particular for the definition of the external transition function δ_{ext} : When a new entity and a new blocking status arrive at the same time, the state has to proceed two “steps” at once to reach a necessary transitional state. E. g. when the queue is in state “queueing

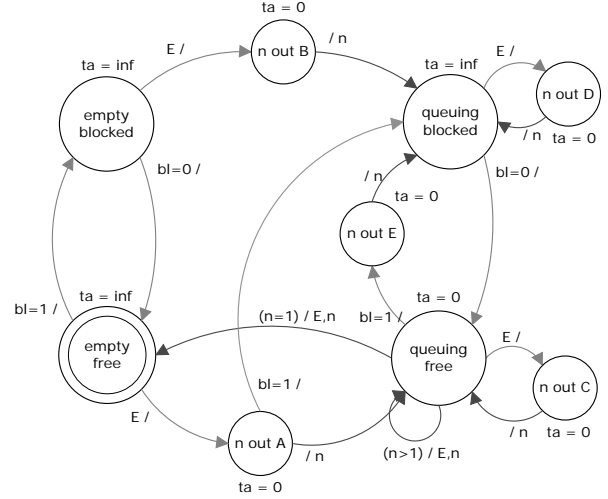


Figure 4: State diagram of the queue component with additional transient states.

blocked” and a new entity arrives together with $bl = 0$, the new entity is stored and the new state is “n out C”.

Special care is needed for the definition of the confluent transition function δ_{con} . Usually it first calls the internal, then the external transition function, so that the entity at the head of the queue leaves, before the new entity is stored. But if a new value $bl = 1$ arrives, only the external transition function is used, so that no entity leaves the queue. This leads to a change of the queue size without a call of the internal transition function, therefore one needs another transitional state “n out E” to produce the corresponding output.

If one has taken proper care of all complications the complete model can be implemented in a PDEVS simulation environment like MatlabDEVS [13], where a simulation run will produce the expected results shown in Fig. 5. The “spikes” in the plots of the queue length and the server allocation are remnants of the concurrent event chains, where state variables have different values at the same time instant.

3 Trying to implement singleserver with RPDEVS

To make the direct definition of Mealy components possible, Preyser et al. define RPDEVS in [10] as follows: An *atomic RPDEVS* is a simplified version of the atomic PDEVS, which contains only a generic transition δ . Moreover its output function λ is called at

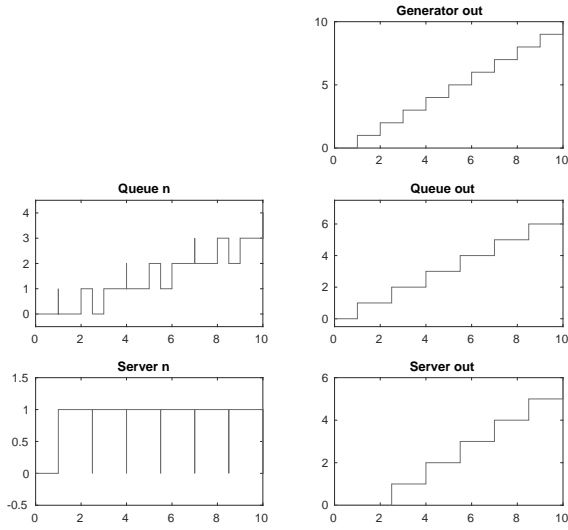


Figure 5: Simulation results of the PDEVs model.

any kind of event and depends on the state and the input. Formally an atomic RPDEVs is given by a 6-tuple $\langle X^b, S, Y^b, ta, \delta, \lambda \rangle$ with the meanings

X^b	set of possible input bags
S	set of states
Y^b	set of possible output bags
$ta : S \rightarrow [0, \infty]$	time advance function
$\delta : Q \times X^b \rightarrow S$	state transition function
$\lambda : Q \times X^b \rightarrow Y^b$	output function

and the Q defined above.

A *coupled RPDEVs* is formally defined as in PDEVs, describing the subcomponents and the internal and outward connections. But its behaviour is different, in order to cope with possible Mealy components: The call of a λ -function produces outputs that are routed to other components, where they in turn may lead to a call of their λ -functions, creating cascades of λ steps, which might change already processed input values. In such a case earlier input values are withdrawn and replaced by new ones (or cancelled completely). For models without algebraic loops these λ iterations will finally lead to a situation, where all input bags are constant. Only then a single δ call is issued.

The *singleserver* example can be formulated in RPDEVs much easier than in PDEVs, since one can stick to the simple state diagrams shown in Fig. 2 and Fig. 3. This allows to specify the atomic components in a straightforward way by identifying the event type

inside the generic δ function according to the input bag and the elapsed time. Using the PowerRPDEVs simulator [14], the components can be easily implemented in C++. Finally one can construct the complete single-server example in a graphical environment.

Though all components work in simple test models, the simulation of the *singleserver* model aborts at $t = 1$. The error message states that the maximum number of λ steps has been reached and that the model is illegitimate due to a non-resolvable algebraic loop.

The reason for this behaviour becomes clear, when one analyses the internal chain of events in the simulator at $t = 1$ (cf. Table 1): The queue starts in state “empty free” and changes to state “queuing free” in line 3, while the server remains in state “idle”. In line 4 the queue outputs its entity that is routed to the server, which now sends the blocking status $bl = 1$ to the queue. The basic problem now happens in line 6: The λ function of the queue is called again, now with $bl = 1$ in the input bag. The entity that has been sent before, is now blocked and has to be retrieved. This in turn leads to the withdrawal of the $bl=1$ message from the server, therefore the queue tries again to output its entity in line 7. The situation is now identical to line 4 and repeats, until it is stopped, when the maximal count of λ steps is reached.

No.	Block	Type	Out	Q in	Q bl	S in
1	Gen	λ	E1	E1		
2	Que	λ				
3	Que	δ				
4	Que	λ	E1			E1
5	Srv	λ	bl=1		1	
6	Que	λ				
7	Que	λ	E1			E1

Table 1: Events at $t = 1$.

The basic idea behind the state diagram in Fig. 3 was, that a component changes its state immediately after sending its output message. Therefore new input messages, that arrive due to event cascades, find the component in a changed state. But the repeated execution of λ steps without any state changing δ steps in RPDEVs leads to a completely different behaviour.

From the point of view of RPDEVs the *singleserver* is faulty, containing an algebraic loop. On the other hand the behaviour described in Fig. 3 is quite simple and can be easily implemented in PDEVs. Zeigler knew very well, why he didn’t include Mealy-type behaviour.

But if one insists on it for the sake of better modularity, one has to think over the simulator behaviour, or better: the abstract model behind it.

4 Extending the time line by infinitesimals

Modeling experience teaches us that a mathematical problem in the description or simulation of a model often has its roots in an oversimplification of the system one wants to describe. This is of course true here: In real world systems small delay times are inevitable, whenever a message is sent or a state changes. If one includes them in the model description, the problems with cascades of concurrent events disappear immediately. But the price one has to pay for this solution, is high: Such a model contains a huge amount of delay times, whose values are not known, often not even their order of magnitude. In addition, the behaviour of the model gets much more complicated on a fine time scale, though one often is not interested in these details.

What we are looking for, are time steps that are larger than zero, but so small that they can be ignored for most purposes. Furthermore their actual size should not matter, even though we need different sizes of such steps. This is actually exactly what one commonly denotes as *infinitesimals*. Hewitt [15] and Robinson [16] have shown that one can implement such ideas in a mathematically rigorous manner. Therefore we will shortly introduce the basic concepts and use them afterwards for a new definition of discrete event systems. A precise and pedagogical introduction to the mathematical ideas and applications can be found in [12].

The set ${}^*\mathbb{R}$ of *hyperreals* is a totally ordered field that includes the ordinary real numbers. In addition it contains an infinitesimal element $\varepsilon > 0$ that is smaller than any positive real number. Using the field axioms one gets additional infinitesimals like $2\varepsilon, -\varepsilon, \varepsilon^2$. Each real number r is surrounded by an infinite cloud $r + \delta$ with infinitesimal δ , its *halo*. On the other end ${}^*\mathbb{R}$ contains $\omega := 1/\varepsilon$, which is *unlimited*, i. e. larger than any real number. Again one has lots of unlimited numbers like $2\omega, -\omega, \omega^2$, which are all surrounded by clouds $\omega + r + \delta$ with real r and infinitesimal δ , called their *galaxy*. Each limited element $h \in {}^*\mathbb{R}$, i. e. an element of the galaxy of 0, can be uniquely written as $h = r + \delta$ with real r and infinitesimal δ . r is called the *standard part* $st(h)$ of h .

The actual construction of ${}^*\mathbb{R}$ relies on heavy ma-

chinery from set theory and logic, like ultrafilters and the axiom of choice. From our current point of view the main reason for an explicit construction is to convince oneself that such a set exists in a precise mathematical way. Hyperreals have been used to reformulate the usual analysis with definitions that closely mimic the original ideas of Leibniz, an endeavor commonly designated as *nonstandard analysis*. This often leads to simple and intuitive proofs – once one accepts the basic properties of ${}^*\mathbb{R}$.

It is impossible to implement real numbers in a computer, much less hyperreals. For our purposes it is sufficient to use pairs (t, r) of floating point numbers, which correspond to the hyperreal $t + r\varepsilon$, where t could be the floating point value ∞ to include infinite time delays in passive states. This implementation looks similar to the concept of *superdense time* [17] that uses a pair of a real time value and a natural number for ordering of concurrent events. But the structure of the hyperreals is much richer, and the reasoning behind their use is more intuitive and better adapted to the problems that are addressed here.

5 Definition of NSA-DEVS

We will now use non-standard analysis (“NSA”) to get rid of concurrent events by defining NSA-DEVS, a variant of RPDEVs. The basic idea is to forbid transient states, i. e. transition times are always > 0 , though they may be infinitesimal. Furthermore we assume that the transport of data between components always takes a certain amount of time. Therefore we include an input delay $\tau > 0$ between the arrival of input and the call of the output function.

An *atomic NSA-DEVS* is given by a 7-tuple $\langle X^b, S, Y^b, \tau, ta, \delta, \lambda \rangle$, where $\tau \in {}^*\mathbb{R}_{>0}$ is the *input delay time*. All other elements have the same meaning as in RPDEVs, but the definitions of the functions are changed to

$$\begin{aligned} Q &:= \{(s, e) | s \in S, e \in (0, ta(s)]\} \\ ta &: S \rightarrow (0, \omega] \\ \delta &: Q \times X^b \rightarrow S \\ \lambda &: Q \times X^b \rightarrow Y^b \end{aligned}$$

The intervals $(0, ta(s)]$ and $(0, \omega]$ are meant as subsets of the hyperreals ${}^*\mathbb{R}$.

When an external event, i. e. an input $x \in X^b$, occurs at time t , the output function λ is called at $t + \tau$,

followed by an immediate call of δ . An internal event, i.e. an imminent state change after a waiting time $ta(s)$, leads to a direct (undelayed) call of λ and δ . A concurrent incidence of a (delayed) external event and an internal event can be detected by both functions directly and doesn't need a special mechanism.

A *coupled NSA-DEVS* is defined as in RPDEVS and PDEVS, outputs are transported as usual. Due to the (infinitesimal) delays a strict Mealy-type behaviour is impossible, therefore special provisions like the iterated λ calls in RPDEVS are unnecessary, each λ call is followed immediately by the corresponding δ call.

Formally we have introduced a lot of additional (infinitesimal) parameters, but a simulator might be able to free the user from this burden by using a simple default value of $\tau = \varepsilon$ for all components. Furthermore the transition times of previously transient states could be set to ε in many cases. It remains to be seen, whether such a simple scheme actually works in standard situations. In the case of the *singleserver* example manual finetuning is necessary, as will be shown below. On the other hand, if one insists on a special ordering of (ideally) concurrent events, one can use the infinitesimal delays to achieve any order in a quite intuitive way.

Since one is generally not interested in the infinitesimal behaviour, an NSA-DEVS simulator should show state changes and output values only at the end of an infinitesimal cascade, i.e. directly before a finite (non-infinitesimal) step. All short-lived states and overwritten outputs are then internal to the simulator. This behaviour is somewhat similar to that of an ODE solver that uses microsteps internally for stepsize adaptation, but outputs only completed steps. Optionally it should be possible to replace the value ε by a user supplied small real number for debugging purposes or to analyse the behaviour at the infinitesimal scale.

6 Implementing singleserver with NSA-DEVS

To adapt the RPDEVS description of *singleserver* to NSA-DEVS, one needs two modifications: All components get an additional parameter τ with a default value of ε , and the queue component gets a further parameter t_D , which defines the transition time of the state "queuing free", again with a default value of ε .

Until a proper NSA-DEVS simulator is available, one can use the PowerRPDEVS simulator to mimic the behaviour in debug mode, where the infinitesimal value

ε is set to a small number ($\varepsilon = 10^{-4}$ in the following examples). To this end one creates a simple delay component, basically a simple server, and adds it before every input of each component. All delay times are set to the value of τ of the corresponding component. In particular, the delay times of several inputs of one component have to be identical to properly implement the NSA-DEVS behaviour defined above. These delays guarantee that no λ iterations occur, therefore the simulator works as required by the NSA-DEVS definition.

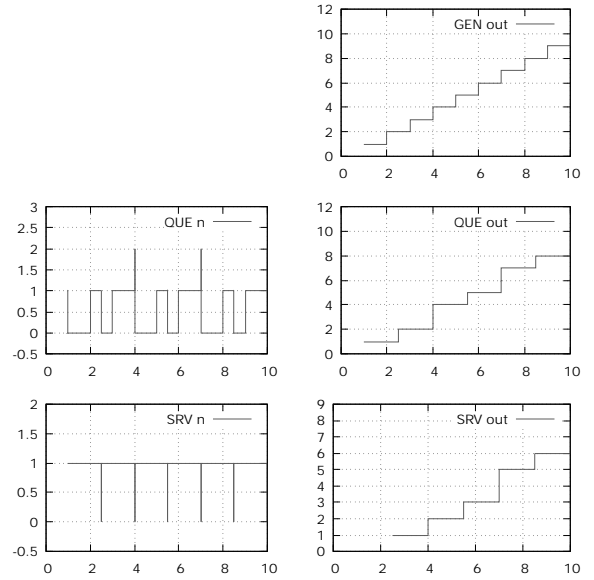


Figure 6: Simulation results of the NSA-DEVS model with default delays.

With these changes the *singleserver* model runs in "NSA-DEVS debug mode", the results are shown in Fig. 6. But they are not as expected: At $t = 4$ the entity 4 joins entity 3 in the queue and both leave the queue immediately. After the service time entity 3 leaves the server at $t = 5.5$, but entity 4 is lost. At first thought this problem seems to be related to the nullserver problem described in [18]: The delay at the input of the server acts as additional storage and accepts entity 4 though the server is busy.

While this is true, the real cause of the problem lies elsewhere. Even in a correct implementation of NSA-DEVS, where the delay is implemented directly in the simulator, entity 4 would get lost! This is due to the delay of the "busy" message from the server: Before it arrives, the queue has already output the next entity according to the internal transition shown in Fig. 3. This

is an example, where it is not sufficient to use the default value ε for all infinitesimal delays.

In order to obtain the desired causal ordering of “concurrent” events, a fine tuning of the delays is necessary. One has to guarantee that the message from the server arrives, before the queue sends a new entity. Therefore one sets the transition time t_D of the “queuing free” state to a value larger than the sum of the two transport delays at the inputs of the server and the queue. Choosing $t_d = 2.1 \varepsilon$ solves the problem and the results are as expected. They coincide with the results of the PDEVs implementation (cf. Fig. 5), including the “spikes”. Here they have a small, but finite width, like it should be in a proper NSA-DEVs simulator in debug mode. In standard mode output values that change in an infinitesimal time are suppressed and only the last values are shown.

7 Conclusions

Like RPDEVs, the variant NSA-DEVs proposed here allows for simple and consistent handling of Mealy-type behaviour. Furthermore it seems to solve the problems of RPDEVs with chains of concurrent events. Therefore it possibly could be a working basis for a concrete simulation environment.

The drastic measure of prohibiting real “concurrency” for causally ordered events generally causes serious side effects by introducing lots of additional time parameters. This is mitigated here by the introduction of infinitesimal delays used mainly internally and whose actual values do not matter. That such a scheme is mathematically sound has been shown by referring to the results of non-standard analysis.

A certain amount of fine-tuning can still be necessary to ensure a requested causal ordering. But this is not a speciality of NSA-DEVs: The proper behaviour of the system has to be modelled anyhow, in PDEVs this is done by a careful design of the confluent transition function. In case of causally unrelated events, one could twist some of the infinitesimal delay parameters to ensure a certain temporal order, if requested.

At first sight NSA-DEVs seems to destroy the potential of parallel execution. But this is not necessarily the case: For unrelated events one simply chooses identical delays – usually just ε –, so that they still occur at the same time $t \in {}^*\mathbb{R}$ and can be executed in parallel. Only causally depending events have different times, so that their order is fixed – as it should be. Moreover,

the elimination of many transitional states in RPDEVs and NSA-DEVs could provide more opportunities for parallel execution than PDEVs.

This is only a first step in the analysis of a new DEVs-based scheme that could possibly be used for simulation environments and the definition of universally applicable component libraries. The next step would be the definition and implementation of a proper simulator. This could be followed by a thorough investigation of standard examples with complex event cascades, such as a switch that routes entities according to an input value [9] or models of digital circuits containing flip-flops [19, 20]. Finally one could try to implement a complex case study like the Argesim C22 benchmark [21] as a further step to investigate the practical usefulness of the proposed scheme.

Acknowledgement

The author gratefully acknowledges clarifying discussions with Christina Deatcu and Thorsten Pawletta.

References

- [1] Zeigler BP. *Theory of Modeling and Simulation*. New York: Wiley-Interscience, 1st ed. 1976.
- [2] Chow ACH. Parallel DEVs: A Parallel, Hierarchical, Modular Modeling Formalism and its Distributed Simulators. *Transactions of The Society for Computer Simulation International*. 1996;13(2):55–67.
- [3] Zeigler BP, Praehofer H, Kim TG. *Theory of Modeling and Simulation*. San Diego: Academic Press, 2nd ed. 2000.
- [4] W David Kelton NBZ Randall P Sadowski. *Simulation with Arena*. New York: McGraw-Hill, 6th ed. 2015.
- [5] Clune MI, Mosterman PJ, Cassandras CG. Discrete Event and Hybrid System Simulation with SimEvents. In: *8th International Workshop on Discrete Event Systems*. Ann Arbor. 2006; pp. 386–387.
- [6] Li W, Mani R, Mosterman PJ. Extensible discrete-event simulation framework in SimEvents. In: *Proc. 2016 Winter Simulation Conference*. Arlington: IEEE. 2016; pp. 943–954.
- [7] Franceschini R, Bisgambiglia PA, Touraille L, Bisgambiglia P, Hill D. A survey of modelling and simulation software frameworks using Discrete Event System Specification. In: *Proc. of 2014 Imperial College Computing Student Workshop*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2014; pp. 40–49.

- [8] Goldstein R, Breslav S, Khan A. Informal DEVS conventions motivated by practical considerations. In: *Proc. of Symposium on Theory of Modeling & Simulation – DEVS Integrative M&S Symposium*. 2013; pp. 10:1–10:6.
- [9] Preyser FJ, Heinzl B, Raich P, Kastner W. Towards Extending the Parallel-DEVS Formalism to Improve Component Modularity. In: *Proc. of ASIM-Workshop STS/GMMS*. Lippstadt. 2016; pp. 83–89.
- [10] Preyser FJ, Heinzl B, Kastner W. RPDEVS: Revising the Parallel Discrete Event System Specification. In: *9th Vienna Int. Conf. Mathematical Modelling*. Wien. 2018; pp. 242–247.
- [11] Preyser FJ, Heinzl B, Kastner W. RPDEVS Abstract Simulator. *SNE Simulation News Europe*. 2019; 29(2):79–84. doi: 10.11128/sne.29.tn.10473.
- [12] Goldblatt R. *Lectures on the Hyperreals*. New York: Springer. 1998.
- [13] Pawletta T, Deatcu C, Pawletta S, Hagendorf O, Colquhoun G. DEVS-based modeling and simulation in scientific and technical computing environments. In: *Proc. of DEVS Integrative M&S Symposium (DEVS'06) - Part of the 2006 Spring Simulation Multiconference (SpringSim'06)*. Huntsville/AL, USA: D. Hamilton. 2006; pp. 151–158.
- [14] Preyser F. *PowerRPDEVS on sourceforge*. URL <https://sourceforge.net/projects/powerrpdevs/>
- [15] Hewitt E. Rings of real-valued continuous functions I. *Transactions of the American Mathematical Society*. 1948;64(1):45–99.
- [16] Robinson A. *Non-standard Analysis*. Amsterdam: North-Holland. 1966.
- [17] Sarjoughian HS, Sundaramoorthi S. Superdense time trajectories for DEVS simulation models. In: *SpringSim (TMS-DEVS)*. 2015; pp. 249–256.
- [18] Austermann L, Junglas P, Schmidt J, Tiekmann C. Conceptional problems of transaction-based modeling and its implementation in SimEvents 4.4. *SNE Simulation News Europe*. 2017;27(3):137–142. doi: 10.11128/sne.27.tn.10383.
- [19] Fiedler C, Preyser FJ, Kastner W. Simulation of RPDEVS Models of Logic Gates. *SNE Simulation News Europe*. 2019;29(2):85–91. doi: 10.11128/sne.29.tn.10474.
- [20] Junglas P. Pitfalls using discrete event blocks in Simulink and Modelica. In: *Proc. of ASIM-Workshop STS/GMMS*. Lippstadt. 2016; pp. 90–97.
- [21] Junglas P, Pawletta T. Non-standard Queuing Policies: Definition of ARGESIM Benchmark C22. *SNE Simulation News Europe*. 2019;29(3):111–115. doi: 10.11128/sne.29.bn22.10481.

Investigation on Stability Properties of Hierarchical Co-Simulation

Irene Hafner^{1*}, Niki Popper²

¹dwh GmbH, Neustiftgasse 57–59, 1070 Vienna, Austria; *irene.hafner@dwh.at

²Institute of Information Systems Engineering, Research Unit of Information and Software Engineering, TU Wien, Favoritenstraße 9–11, 1040 Vienna, Austria

Abstract. This paper introduces the concept of hierarchical co-simulation and presents an investigation on stability properties of this method. In conventional co-simulation methods, all participating simulations are executed on the same level via one co-simulation. Hierarchical co-simulation, on the other hand, enables the introduction of several levels of co-simulation by allowing participating subsystems to consist of co-simulated systems themselves, thus nesting co-simulations within co-simulations. While on the one hand, certain stability issues can arise by the introduction of more co-simulation layers, this method enables the usage of different synchronization references for parts of the overall system according to varying dependencies between the subsystems, which can increase accuracy and numerical stability.

Introduction

Co-simulation has become an important instrument to approach the simulation of large-scale heterogeneous systems in recent years. While definitions for the term co-simulation vary depending on the field of origin, in this paper we refer (in accordance with the terminology found in [1]) to co-simulation as the coupling of two or more simulations which differ in at least one aspect out of simulation tool, solver algorithm or step size. Hierarchical approaches or multi-level descriptions have already been introduced in other fields within modeling and simulation (f.i. DEVS [2], multi-level agent-based modeling [3], MPC [4] or partitioned multi-rate approaches [5]). However, hierarchical co-simulation as explained in the following has to the authors' knowledge not been investigated up to now, although several frameworks and standards do not prohibit the realization of further co-simulations within a co-simulation. The idea is illustrated in Figure 1.

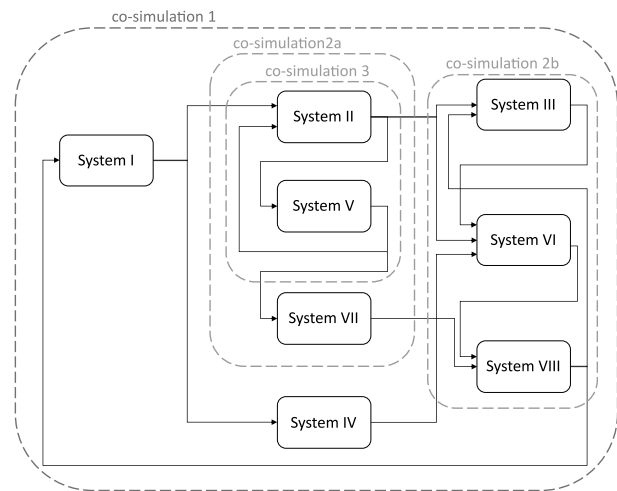


Figure 1: Schematic depiction of a hierarchical co-simulation approach.

In a traditional co-simulation approach all eight participating subsystems would have been co-simulated in one overall co-simulation, probably requiring all systems to synchronize at the same points in time. In the hierarchical approach, systems III, VI and VIII are coupled in another co-simulation (2b) as well as systems II and V, the coupled system of which (co-simulation 3) is again co-simulated with system VII, before the resulting co-simulation (2a) represents a system coupled in the top-level co-simulation 1.

A coupling structure like this could be motivated by the usage of different synchronization intervals on every co-simulation level, thus enabling frequent exchanges between subsystems which are sensitive to changes in their respective exchanged values while allowing larger communication intervals with other, slower reacting system parts, which can speed up the overall execution. In the course of this paper, we will show that these ideas are valid and that a hierarchically structured co-

simulation approach indeed allows to enhance stability at a low computational cost.

1 Consistency

A valid method to bound the global co-simulation error is local error control, which justifies to investigate the consistency error, i.e. the error of the method in one step. For traditional co-simulation, it has been shown that consistency can be maintained, but possibly reduced to the extrapolation order of values from other systems, see for example [6, 17]. Since consistency is defined locally (i.e. per step), and it is a property regarded for the limit of step size $h \rightarrow 0$, the value present at the most recent point in time where the method sets a step is considered to be the exact solution - a property that is not affected by the method used in the respective other subsystems or the time steps and further synchronizations happening there in-between. This means that consistency in hierarchical co-simulation is also maintained with its order depending on the applied extrapolation method.

2 Zero-stability

Zero-stability, i.e. convergence of a method for infinitesimal step sizes, has been analyzed for certain co-simulation approaches in [7], on which we base our investigation. The mathematical description of coupled DAEs is given in [7] as follows:

$$\dot{\mathbf{x}}^i(t) = \mathbf{f}^i(\mathbf{x}^i, \mathbf{u}^i, t), \quad \mathbf{x}^i(t_0) = \mathbf{x}_0^i \quad (1a)$$

$$\mathbf{y}^i(t) = \mathbf{g}^i(\mathbf{x}^i, \mathbf{u}^i, t) \quad (1b)$$

with $i = I, \dots, N$, $\mathbf{x}^i \in \mathbb{R}^{n_x^i}$, $\mathbf{u}^i \in \mathbb{R}^{n_u^i}$, $\mathbf{y}^i \in \mathbb{R}^{n_y^i}$ and

$$\mathbf{u}^i = \mathbf{L}^i \mathbf{y} \quad (1c)$$

where

$$\mathbf{L}^i = [\mathbf{L}^{i,I} \quad \dots \quad \mathbf{L}^{i,i-1} \quad 0 \quad \mathbf{L}^{i,i+1} \quad \dots \quad \mathbf{L}^{i,N}],$$

$$\mathbf{y} = [\mathbf{y}^I \quad \dots \quad \mathbf{y}^{i-1} \quad \mathbf{y}^i \quad \mathbf{y}^{i+1} \quad \dots \quad \mathbf{y}^N]^T$$

with $\mathbf{L}^{i,j} \in \mathbb{R}^{n_u^i \times n_y^j} \quad \forall i, j \in \{I, \dots, N\}$ and the elements of $\mathbf{L}^{i,j}$ being equal to zero or one.

Under certain assumptions (given in [7], p. 100), the outputs can be written as $\mathbf{y}^i = \bar{\mathbf{g}}^i(\mathbf{x}^i) + \mathbf{D}^i(\mathbf{x}^i) \mathbf{u}^i$, yielding

the discretized output equations

$$\mathbf{y}_{k+1}^i = \bar{\mathbf{g}}^i + \mathbf{D}^i \mathbf{u}_k^i \quad (2)$$

with constant $\bar{\mathbf{g}}^i, \mathbf{D}^i$. Using this, it holds for the outputs of global system

$$\mathbf{y}_{k+1} = \bar{\mathbf{g}} + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{D}^I \mathbf{L}^{I,I} & \dots & \mathbf{D}^I \mathbf{L}^{I,N} \\ \mathbf{D}^{II} \mathbf{L}^{II,I} & \mathbf{0} & \dots & \mathbf{D}^{II} \mathbf{L}^{II,N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}^N \mathbf{L}^{N,I} & \mathbf{D}^N \mathbf{L}^{N,II} & \dots & \mathbf{0} \end{bmatrix}}_{=\mathbf{D}} \mathbf{y}_k \quad (3)$$

that stability is guaranteed if the spectral radius ρ of \mathbf{D} is less than or equal to 1. This is fulfilled in particular if $\rho(\mathbf{D}) = 0$ which for the case of two participating subsystems means that there is no algebraic loop.

To determine zero-stability properties of hierarchical co-simulation, a co-simulation of N systems is considered, where w.l.o.g. systems M to N are combined in a second-level co-simulation as depicted in Figure 2.

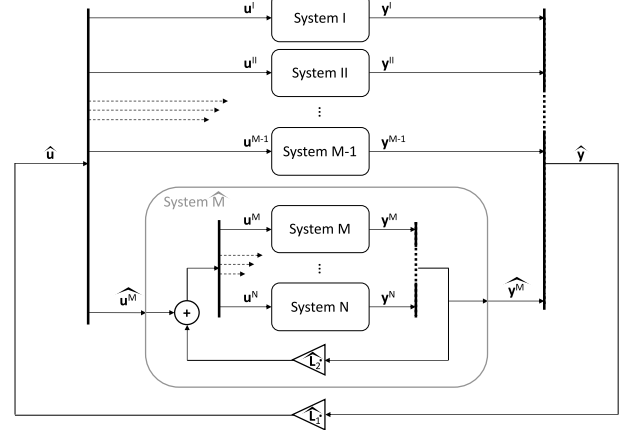


Figure 2: Hierarchical co-simulation of N systems on two levels.

System \hat{M} replaces systems M to N of the original co-simulation on one level (coupled via 1c and called CS_0 henceforth). We obtain coupling equations (4) for the upper co-simulation level CS_1 :

$$\begin{bmatrix} \mathbf{u}^I \\ \mathbf{u}^{II} \\ \vdots \\ \mathbf{u}^{M-1} \\ \widehat{\mathbf{u}}^M \end{bmatrix} = \widehat{\mathbf{L}}_1 \begin{bmatrix} \mathbf{y}^I \\ \mathbf{y}^{II} \\ \vdots \\ \mathbf{y}^{M-1} \\ \widehat{\mathbf{y}}^M \end{bmatrix} \quad (4)$$

with

$$\widehat{\mathbf{L}}_1 = \begin{bmatrix} \mathbf{0} & \mathbf{L}^{I,II} & \dots & \mathbf{L}^{I,M-1} & \widehat{\mathbf{L}}^{I,M} \\ \mathbf{L}^{II,I} & \mathbf{0} & \dots & \mathbf{L}^{II,M-1} & \widehat{\mathbf{L}}^{II,M} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{L}^{M-1,I} & \mathbf{L}^{M-1,II} & \dots & \mathbf{0} & \widehat{\mathbf{L}}^{M-1,M} \\ \widehat{\mathbf{L}}^{M,I} & \widehat{\mathbf{L}}^{M,II} & \dots & \widehat{\mathbf{L}}^{M,M-1} & \mathbf{0} \end{bmatrix}$$

and $\widehat{\mathbf{u}}^M$ as input to the new subsystem \widehat{M} , $\widehat{\mathbf{y}}^M$ as its output and

$$\widehat{\mathbf{L}}^{i,M} = [\mathbf{L}^{i,M} \quad \mathbf{L}^{i,M+1} \quad \dots \quad \mathbf{L}^{i,N}], i = I \dots M-1$$

$$\widehat{\mathbf{L}}^{M,i} = \begin{bmatrix} \mathbf{L}^{M,i} \\ \mathbf{L}^{M+1,i} \\ \vdots \\ \mathbf{L}^{N,i} \end{bmatrix}, i = I \dots M-1.$$

Thus, the only difference between \mathbf{L} and $\widehat{\mathbf{L}}_1$ is the increased number of zeroes in the lower right corner. The discretized output equations of CS_1 are (5):

$$\begin{aligned} \mathbf{y}_{k+1}^I &= \bar{\mathbf{g}}^I + \mathbf{D}^I \mathbf{u}_k^I \\ \mathbf{y}_{k+1}^{II} &= \bar{\mathbf{g}}^{II} + \mathbf{D}^{II} \mathbf{u}_k^{II} \\ &\vdots \\ \widehat{\mathbf{y}}_{k+1}^M &= \widehat{\mathbf{g}}^M + \widehat{\mathbf{D}}^M \widehat{\mathbf{u}}_k^M \end{aligned} \quad (5)$$

While $\widehat{\mathbf{y}}^M$ in general corresponds to the stacked output vectors $\mathbf{y}^M \dots \mathbf{y}^N$ of CS_0 , the input vectors don't as the coupling with the outputs of systems M to N is considered within the new system \widehat{M} , cf. Figure 2 and (6). The outputs of the global system can with (5) be written as

$$\begin{bmatrix} \mathbf{y}_{k+1}^I \\ \vdots \\ \mathbf{y}_{k+1}^{M-1} \\ \widehat{\mathbf{y}}_{k+1}^M \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{g}}^I \\ \vdots \\ \bar{\mathbf{g}}^{M-1} \\ \widehat{\mathbf{g}}^M \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{D}^I & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & \mathbf{D}^{M-1} & \widehat{\mathbf{D}}^M \end{bmatrix}}_{=:\mathbf{D}_{CS_1}} \widehat{\mathbf{L}}_1 \begin{bmatrix} \mathbf{y}_k^I \\ \vdots \\ \mathbf{y}_k^{M-1} \\ \widehat{\mathbf{y}}_k^M \end{bmatrix}$$

In analogy to the case of one co-simulation level, the co-simulation of the upper level is stable if $\rho(\mathbf{D}_{CS_1}) \leq 1$. The only unknown in comparison to \mathbf{D} of CS_0 is $\widehat{\mathbf{D}}^M$, for which we have to take a look at the second-level co-simulation CS_2 . The coupling equations within this system can be written (cf. Figure 2) as follows:

$$\begin{bmatrix} \mathbf{u}_k^M \\ \mathbf{u}_k^{M+1} \\ \vdots \\ \mathbf{u}_k^{N-1} \\ \mathbf{u}_k^N \end{bmatrix} = \widehat{\mathbf{L}}_2 \cdot \begin{bmatrix} \mathbf{y}_k^M \\ \mathbf{y}_k^{M+1} \\ \vdots \\ \mathbf{y}_k^{N-1} \\ \mathbf{y}_k^N \end{bmatrix} + \widehat{\mathbf{u}}_k^M \quad (6)$$

where

$$\widehat{\mathbf{L}}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{L}^{M,M+1} & \dots & \mathbf{L}^{M,N-1} & \mathbf{L}^{M,N} \\ \mathbf{L}^{M+1,M} & \mathbf{0} & \dots & \mathbf{L}^{M+1,N-1} & \mathbf{L}^{M+1,N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{L}^{N-1,M} & \mathbf{L}^{N-1,M+1} & \dots & \mathbf{0} & \mathbf{L}^{N-1,N} \\ \mathbf{L}^{N,M} & \mathbf{L}^{N,M+1} & \dots & \mathbf{L}^{N,N-1} & \mathbf{0} \end{bmatrix}.$$

The discretized output equations are

$$\mathbf{y}_{k+1}^i = \bar{\mathbf{g}}^i + \mathbf{D}^i \mathbf{u}_k^i, i = M \dots N. \quad (7)$$

Thus follows for the global output of CS_2

$$\widehat{\mathbf{y}}_{k+1}^M = \begin{bmatrix} \mathbf{y}_{k+1}^M \\ \vdots \\ \mathbf{y}_{k+1}^N \end{bmatrix} = \widehat{\mathbf{g}}^M + \begin{bmatrix} \mathbf{D}^M & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{D}^N \end{bmatrix} \widehat{\mathbf{u}}_k^M \quad (8)$$

with

$$\widehat{\mathbf{g}}^M = \begin{bmatrix} \bar{\mathbf{g}}^M \\ \vdots \\ \bar{\mathbf{g}}^N \end{bmatrix} + \begin{bmatrix} \mathbf{D}^M & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{D}^N \end{bmatrix} \widehat{\mathbf{L}}_2 \begin{bmatrix} \mathbf{y}_k^M \\ \vdots \\ \mathbf{y}_k^N \end{bmatrix}.$$

The part containing \mathbf{y}_k^i , $i = M \dots N$ can be included in $\widehat{\mathbf{g}}^M$ as these are only internal states of CS_2 which are unknown in CS_1 . Hence we obtain

$$\widehat{\mathbf{D}}^M = \begin{bmatrix} \mathbf{D}^M & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{D}^N \end{bmatrix}, \quad (9)$$

which yields

$$\mathbf{D}_{CS_1} = \begin{bmatrix} \mathbf{0} & \mathbf{D}^I \mathbf{L}^{I,II} & \dots & \widehat{\mathbf{D}^I \mathbf{L}^{I,M}} \\ \mathbf{D}^{II} \mathbf{L}^{II,I} & \mathbf{0} & \dots & \widehat{\mathbf{D}^{II} \mathbf{L}^{II,M}} \\ \vdots & \ddots & \dots & \vdots \\ \mathbf{D}^{M-1} \mathbf{L}^{M-1,I} & \dots & \mathbf{0} & \mathbf{D}^{M-1} \widehat{\mathbf{L}^{M-1,M}} \\ \widehat{\mathbf{D}^M \mathbf{L}^{M,I}} & \dots & \widehat{\mathbf{D}^M \mathbf{L}^{M,M-1}} & \mathbf{0} \end{bmatrix}.$$

Due to

$$\widehat{\mathbf{D}^M \mathbf{L}^{M,i}} = \begin{bmatrix} \mathbf{D}^M & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{D}^N \end{bmatrix} \cdot \begin{bmatrix} \mathbf{L}^{M,i} \\ \mathbf{L}^{M+1,i} \\ \vdots \\ \mathbf{L}^{N,i} \end{bmatrix} = \begin{bmatrix} \mathbf{D}^M \mathbf{L}^{M,i} \\ \mathbf{D}^M \mathbf{L}^{M+1,i} \\ \vdots \\ \mathbf{D}^N \mathbf{L}^{N,i} \end{bmatrix}$$

and

$$\begin{aligned} \mathbf{D}^i \widehat{\mathbf{L}^{i,M}} &= \mathbf{D}^i \cdot [\mathbf{L}^{i,M} \quad \mathbf{L}^{i,M+1} \quad \dots \quad \mathbf{L}^{i,N}] \\ &= [\mathbf{D}^i \mathbf{L}^{i,M} \quad \mathbf{D}^i \mathbf{L}^{i,M+1} \quad \dots \quad \mathbf{D}^i \mathbf{L}^{i,N}], \end{aligned}$$

the only difference compared to matrix \mathbf{D} of system CS_0 is the increased number of zeroes in the lower right corner. In the following, we try to use this to gain information on the properties of the spectral radius of \mathbf{D}_{CS_1} using knowledge on $\rho(\mathbf{D})$.

We know that for every matrix norm $\|\cdot\|$ and arbitrary matrix $\mathbf{A} = (a_{ij}); i = 1, \dots, m; j = 1, \dots, n; m, n \in \mathbb{N}$

$$\rho(\mathbf{A}) \leq \|\mathbf{A}\| \quad (10)$$

holds ([8], Thm. 5.6.9).

If we consider $\|\cdot\|_\infty$ given as

$$\|\mathbf{A}\|_\infty = \max_{i=1, \dots, m} \sum_{j=1}^n |a_{ij}|$$

we see at once that $\|\mathbf{D}_{CS_1}\|_\infty \leq \|\mathbf{D}\|_\infty$. Unfortunately, this does *not* imply $\rho(\mathbf{D}_{CS_1}) \leq \rho(\mathbf{D})$, see f.i. the following example: Let matrices \mathbf{A}_1 and \mathbf{A}_2 given as

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0.1 & 0.5 & 0 \\ 0.1 & 0 & 0 & 0.5 \\ 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 0.1 & 0.5 & 0 \\ 0.1 & 0 & 0 & 0.5 \\ 0.2 & 0 & 0 & -0.1 \\ 0 & 0.2 & -0.1 & 0 \end{bmatrix}.$$

Here $\|\mathbf{A}_1\|_\infty = \|\mathbf{A}_2\|_\infty = 0.6$ but

$\rho(\mathbf{A}_1) \approx 0.3702 > \rho(\mathbf{A}_2) \approx 0.3317$. This means that in general, stability for hierarchical co-simulation has to be determined anew, even if the starting point is a zero-stable co-simulation on one level. An exception is the case where not only $\rho(\mathbf{D}) \leq 1$ but also $\|\mathbf{D}\|_\infty \leq 1$, as from

this follows further

$$\rho(\mathbf{D}_{CS_1}) \leq \|\mathbf{D}_{CS_1}\|_\infty \leq \|\mathbf{D}\|_\infty \leq 1 \quad (11)$$

which ensures zero-stability of the co-simulation on the upper level CS_1 .

For the stability properties of the coupling in CS_2 , we are interested in the input-output dependencies within the system only, thus we need to look at the spectral radius of \mathbf{D}_{CS_2} , which results from (8):

$$\mathbf{D}_{CS_2} = \begin{bmatrix} \mathbf{D}^M & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{D}^N \end{bmatrix} \widehat{\mathbf{L}}_2 \quad (12)$$

Since we see that \mathbf{D}_{CS_2} is composed of a submatrix of \mathbf{D} , here again $\|\mathbf{D}_{CS_2}\|_\infty \leq \|\mathbf{D}\|_\infty$ holds, and thus $\rho(\mathbf{D}_{CS_2})$ has to be determined separately only if $\|\mathbf{D}\|_\infty > 1$.

To sum up, we can conclude that zero-stability of hierarchical co-simulation can be determined analogously to customary co-simulation on one level. To this end, the matrices referring to the global system outputs on every co-simulation level have to be examined - except for the cases where the origin is a stable co-simulation with matrix \mathbf{D} fulfilling $\|\mathbf{D}\|_\infty \leq 1$, which is satisfied in particular for couplings where no feed-through occurs in at least one system, so $\|\mathbf{D}\|_\infty = \rho(\mathbf{D}) = 0$. These considerations, of course, can directly be taken further and applied to more than two levels of co-simulation, as well.

3 Numerical stability

Depending on the coupling method, instabilities can still occur for zero-stable coupling methods due to the errors introduced by extra- or interpolation. A weak coupling approach is called *numerically stable* if it yields a stable solution for a finite macro-step size $H > 0$ [9].

To investigate stability properties for finite communication step sizes, we consider a three-mass oscillator as benchmark example, which is illustrated in Figure 3.

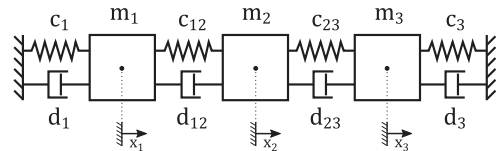


Figure 3: Illustration of a three-mass oscillator.

The underlying equation system can be interpreted as coupled Dahlquist equations, which can be solved analytically and thus provide an eminently suitable test case. The oscillator with two masses has been taken into consideration in numerous investigations on stability of conventional, single-level co-simulation approaches, where it proves highly sensitive to the choice of parameters and macro step size [9].

For the intended co-simulation, the system is split along the individual masses and coupled via force-displacement-coupling (cf. f.i. [10] for further information on the coupling approach), as illustrated in Figure 4.

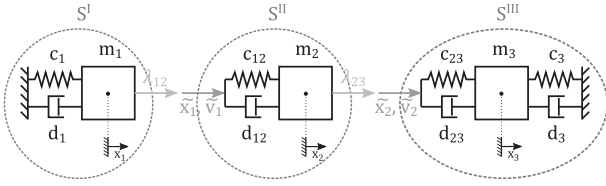


Figure 4: Force-displacement coupling of the three-mass oscillator.

By this coupling approach we obtain the subsystem equations for systems S^I , S^{II} and S^{III} :

$$S^I: \quad \begin{aligned} \dot{x}_1 &= v_1 \\ m_1 \dot{v}_1 &= -c_1 x_1 - d_1 v_1 + \lambda_{12} \end{aligned} \quad (13a)$$

$$S^{II}: \quad \begin{aligned} \dot{x}_2 &= v_2 \\ m_2 \dot{v}_2 &= -c_{12}(x_2 - \tilde{x}_1) - d_{12}(v_2 - \tilde{v}_1) + \lambda_{23} \end{aligned} \quad (13b)$$

$$S^{III}: \quad \begin{aligned} \dot{x}_3 &= v_3 \\ m_3 \dot{v}_3 &= -c_{23}(x_3 - \tilde{x}_2) - d_{23}(v_3 - \tilde{v}_2) \\ &\quad + c_3(-x_3) + d_3(-v_3) \end{aligned} \quad (13c)$$

With the coupling conditions

$$\lambda_{12} - c_{12}(x_2 - x_1) - d_{12}(v_2 - v_1) = 0 \quad (14a)$$

$$\tilde{x}_1 - x_1 = 0 \quad (14b)$$

$$\tilde{v}_1 - v_1 = 0 \quad (14c)$$

$$\lambda_{23} - c_{23}(x_3 - x_2) - d_{23}(v_3 - v_2) = 0 \quad (14d)$$

$$\tilde{x}_2 - x_2 = 0 \quad (14e)$$

$$\tilde{v}_2 - v_2 = 0 \quad (14f)$$

Following the considerations from section 2, we obtain for matrix D in (3):

$$D = \begin{bmatrix} \mathbf{0} & D^I L^{I,II} & D^I L^{I,III} \\ D^{II} L^{II,I} & \mathbf{0} & D^{II} L^{II,III} \\ D^{III} L^{III,I} & D^{III} L^{III,II} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -c_{12} & -d_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & -c_{23} & -d_{23} & 0 & 0 \end{bmatrix} \quad (15)$$

whence follows $\rho(D) = 0$, thus guaranteeing zero-stability.

For the hierarchical co-simulation approach, systems S^{II} and S^{III} are combined in a second-level co-simulation.

As expected (cf. section 2), we obtain $\rho(D_{CS_1}) = \rho(D_{CS_2}) = 0$ for

$$D_{CS_1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -c_{12} & -d_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (16)$$

and

$$D_{CS_2} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -c_{23} & -d_{23} & 0 & 0 \end{bmatrix}, \quad (17)$$

thus the conditions for zero-stability are satisfied for both levels of co-simulation.

In the following, several scenarios are performed for both co-simulation approaches to compare numerical stability properties. For all settings, explicit Euler methods are used to solve the individual subsystems. These simple methods have been chosen to enable the focus on the different methods of co-simulation without additional corrections (f.i. by step size control). As synchronization method, Jacobi-type coupling without iteration using zero-order extrapolation for external variables has been used. The initial conditions for all scenarios have been chosen as $x_1 = 1$, $x_2 = 2$, $x_3 = 3$ and $v_1 = v_2 = v_3 = 0$.

Scenario 1. The parameters for the first scenario to be considered are given in Table 1.

c_1	c_{12}	c_{23}	c_3	d_1	d_{12}	d_{23}	d_3	m_1	m_2	m_3
1E-02	1E-01	1	10	0.1	0.4	1	2	10	10	10

Table 1: Parameter settings for Scenario 1.

As can be seen, the spring stiffnesses are chosen to increase from left to right (cf. Figure 3) to result in slower and faster varying subsystems. The step sizes for the individual subsystem solvers are chosen accordingly with $h_1 = 0.005$, $h_2 = 0.0025$, $h_3 = 0.00125$. The monolithic reference system is of the form $\dot{\mathbf{y}} = \mathbf{A} \cdot \mathbf{y}$ and can thus be solved analytically. In addition to the analytical solution, the results of the hierarchical co-simulation are compared to a conventional single-level co-simulation. For the latter, a macro step size H of 0.1 seconds is chosen. The results in Figure 5 show that even if the overall communication step size H_1 is doubled in comparison to the traditional co-simulation, the hierarchical approach yields significantly more accurate results for systems S^{II} and S^{III} if the step size for the second-level co-simulation is chosen adequately ($H_1 = 0.2s$, $H_2 = 0.05s$).

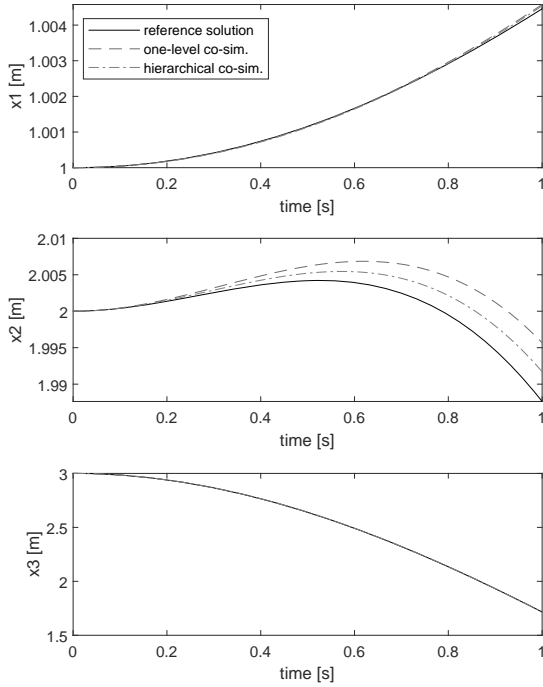


Figure 5: Trajectories of x_1 , x_2 and x_3 for Scenario 1 with $H = 0.1s$, $H_1 = 0.2s$ and $H_2 = 0.05s$.

In spite of plainly distinct errors in specific phases, both approximations remain stable, as can be seen in simulations over a longer period of time.

The maximum absolute errors and elapsed time for several different settings are given in Table 2.

appr.	t_{end}	H/H_1	H_2	err_{x_1}	err_{x_2}	err_{x_3}	err_{x_1}	err_{x_2}	err_{x_3}	el. time
trad.	1	0.1		1.21E-04	5.30E-04	8.05E-03	1.53E-02	9.51E-04	6.32E-04	0.0066
hier.	1	0.1	0.025	4.52E-05	2.89E-04	2.10E-03	3.67E-03	8.22E-04	1.38E-03	0.0161
hier.	1	0.2	0.05	8.26E-05	5.34E-04	4.08E-03	7.44E-03	8.51E-04	1.06E-03	0.0099
trad.	25	0.1		3.96E-02	7.13E-03	9.35E-02	3.68E-02	1.78E-02	1.24E-02	0.1845
hier.	25	0.1	0.025	1.76E-02	3.29E-03	2.14E-02	8.75E-03	6.63E-03	5.48E-03	0.4018
hier.	25	0.2	0.05	3.53E-02	6.64E-03	4.28E-02	1.75E-02	9.67E-03	7.20E-03	0.2514

Table 2: Maximum error and elapsed time for the traditional and hierarchical co-simulation approach in Scenario 1.

We see that while the execution time is significantly higher in case of the same step size on the upper level and the traditional co-simulation - which has to be expected due to the additional synchronization on the lower level - the high difference can be overcome while still maintaining better accuracy by increasing both macro step sizes in the hierarchical approach.

Scenario 2. In Scenario 2, the stiffnesses differ to a greater extent (see parameters in Table 3), which can lead to stability issues if communication step sizes are chosen too large. The conventional co-simulation already yields unstable results for the same step size as in Scenario 1 ($H = 0.1$). The solution obtained by the hierarchical approach with the same upper-level communication step size but additional synchronization between subsystems S^{II} and S^{III} remains stable.

c_1	c_{12}	c_{23}	c_3	d_1	d_{12}	d_{23}	d_3	m_1	m_2	m_3
1E-03	1E-01	10	100	0.1	0.4	1	2	10	10	10

Table 3: Parameter settings for Scenario 2.

Even for a larger communication step size on the upper level ($H_1 = 0.2$), stability is maintained with the hierarchical approach, as the coupling between systems S^{II} and S^{III} is the crucial one (cf. Figure 6).

If the synchronization time on the second level is also increased (to $H_2 = 0.05$), qualitative behavior is still maintained but errors are too high to consider the solution still acceptable (cf. Table 4).

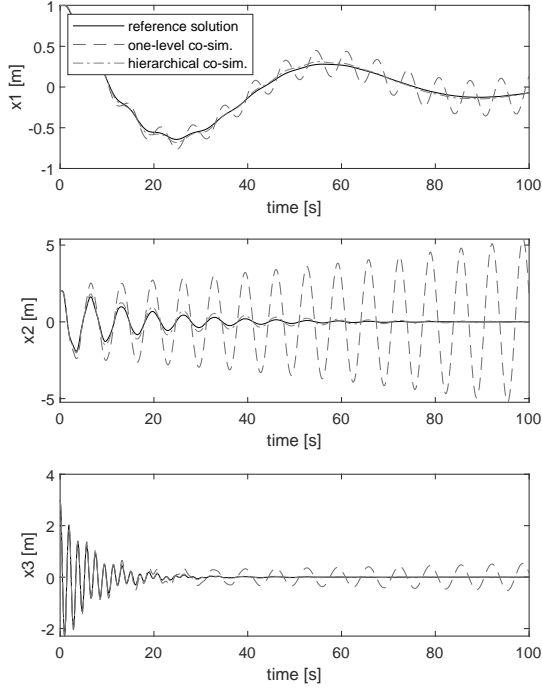


Figure 6: Trajectories of x_1 , x_2 and x_3 for Scenario 2 with $H = 0.1s$, $H_1 = 0.2s$ and $H_2 = 0.025s$ from $t_{start} = 0s$ to $t_{end} = 100s$.

Scenario 3. In Scenario 3, the stiffnesses for the springs attached to mass m_1 are increased, too (see Table 5), which leads to unstable results for the traditional as well as hierarchical approach with step sizes $H = 0.1s$, $H_1 = 0.1s$ and $H_2 = 0.025s$, see Figure 7. This makes sense as the increased stiffness for System *I* can not be compensated by closer communication of Systems *II* and *III*.

The macro step sizes H and H_1 would have to be chosen as low as 0.03 to keep the error in bounds at all, even though results are still too far from the reference solution to be of use.

4 Conclusion

In this paper, the method of hierarchical co-simulation has been presented and investigated with respect to stability properties. In comparison to hierarchical multirate approaches as presented in [5, 12], the application of the hierarchical co-simulation method presented in this paper

appr.	t_{end}	H/H_1	H_2	err_{x_1}	err_{x_2}	err_{x_3}	err_{x_4}	err_{x_5}	err_{x_6}	el. time
trad.	3	0.1		1.19E-02	1.56E-02	4.03E-01	4.61E-01	9.78E-02	2.37E-01	0.0202
hier.	3	0.1	0.025	7.59E-03	7.01E-03	9.54E-02	1.10E-01	5.01E-02	1.33E-01	0.0376
hier.	3	0.2	0.025	1.54E-02	1.44E-02	1.94E-01	2.24E-01	6.56E-02	1.66E-01	0.0372
hier.	3	0.2	0.05	1.46E-02	1.29E-02	9.48E-02	1.10E-01	5.00E-02	1.33E-01	0.0258
trad.	100	0.1		2.37E-01	2.28E-01	5.38E+00	5.06E+00	5.35E-01	5.06E-01	2.2885
hier.	100	0.1	0.025	2.57E-02	1.30E-02	2.90E-01	2.79E-01	6.86E-02	1.98E-01	6.4143
hier.	100	0.2	0.025	5.81E-02	3.08E-02	6.96E-01	6.64E-01	1.06E-01	2.53E-01	5.3847
hier.	100	0.2	0.05	3.90E-02	1.47E-02	2.88E-01	2.77E-01	6.85E-02	1.98E-01	2.5811

Table 4: Maximum error and elapsed time for the traditional and hierarchical co-simulation approach in Scenario 2.

c_1	c_{12}	c_{23}	c_3	d_1	d_{12}	d_{23}	d_3	m_1	m_2	m_3
1	10	10	100	0.1	0.4	1	2	10	10	10

Table 5: Parameter settings for Scenario 3.

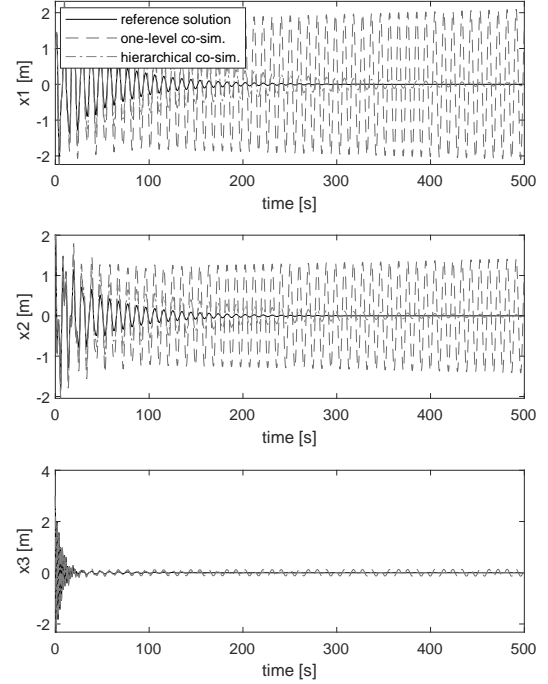


Figure 7: Trajectories of x_1 , x_2 and x_3 for Scenario 3 with $H = 0.1s$, $H_1 = 0.1s$ and $H_2 = 0.025s$ from $t_{start} = 0s$ to $t_{end} = 100s$.

does not require any knowledge on the underlying system per se. The subsystems can, as in common co-simulation methods, be treated as black boxes with information on the input and output dependencies without interfering

with the subsystem solvers. This can be beneficial when using co-simulation platforms like the BCVTB [11] or standards like the FMI¹, and in particular for interdisciplinary collaborative projects where partial systems are developed independently and possibly protected by company-specific privacy agreements.

While in the experiments above, quite simple subsystem solvers and coupling methods are chosen, improvement methods commonly used in single-level co-simulation approaches like variations of extrapolation order, coupling methods (sequential or mixed algorithms and waveform iteration) and stabilization techniques can of course be utilized in hierarchical co-simulation as well. Detailed studies on the advantages of said techniques for traditional co-simulation are ample in the literature (see for example [13, 14, 15, 16]). In addition, the results from section 3 show that stability issues can be tackled by introducing another layer of communication instead of having to decrease the overall communication step size, thus providing an innovative method for stabilization.

References

- [1] Hafner I, Niki P. On the Terminology and Structuring of Co-simulation Methods. In Proceedings of the 8th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools. *EOOLT '17*; 2017; Weßling, Germany. New York, USA: ACM. 67-76. doi: 10.1145/3158191.3158203.
- [2] Zeigler BP. *Object-Oriented Simulation with Hierarchical, Modular Models: Intelligent Agents and Endomorphic Systems*. Saint Louis: Elsevier Science; 2014.
- [3] Morvan G. Multi-level agent-based modeling - A literature survey. *arXiv:1205.0561 [cs]*. 2013.
- [4] Scattolini R. Architectures for distributed and hierarchical Model Predictive Control - A review. *Journal of Process Control*. 2009; 19(5): 723–731. doi: 10.1016/j.jprocont.2009.02.003.
- [5] Günther M, Rentrop P. Partitioning and Multirate Strategies in Latent Electric Circuits. In: Bank RE, Gajewski H, Bulirsch R, Merten K. *Mathematical Modelling and Simulation of Electrical Circuits and Semiconductor Devices*. Basel: Birkhäuser; 1994. p 33–60. doi: 10.1007/978-3-0348-8528-7_3
- [6] Trcka, M. *Cosimulation for Performance Prediction of Innovative Integrated Mechanical Energy Systems in Buildings* [dissertation]. Technische Universiteit Eindhoven; 2008.
- [7] Kübler R, Schiehlen W. Two Methods of Simulator Coupling. *Mathematical and Computer Modelling of Dynamical Systems*. 2000; 6(2): 93–113. doi: 10.1076/1387-3954(200006)6:2;1-M;FT093.
- [8] Horn RA, Johnson CR. *Matrix analysis*. 23. print. Cambridge: Cambridge Univ. Press; 2010.
- [9] Busch, M. *Zur effizienten Kopplung von Simulationsprogrammen*. Kassel: Kassel University Press; 2012.
- [10] Schweizer B, Lu D. Semi-implicit co-simulation approach for solver coupling. *Archive of Applied Mechanics*. 2014; 84(12): 1739–1769. doi: 10.1007/s00419-014-0883-5.
- [11] Wetter M. Co-simulation of building energy and control systems with the Building Controls Virtual Test Bed. *Journal of Building Performance Simulation*. 2011; 4(3): 185–203. doi: 10.1080/19401493.2010.518631.
- [12] Striebel, M. *Hierarchical Mixed Multirate for Distributed Integration of DAE Network Equations in Chip Design* [dissertation]. Bergische Universität Wuppertal; 2006.
- [13] White J, Odeh F, Sangiovanni-Vincentelli AL, Ruehli AE. *Waveform Relaxation: Theory and Practice* [Technical Report]. [EECS Department]. University of California; 1985.
- [14] Larsson J, Krus P. Stability Analysis of Coupled Simulation. In Dynamic Systems and Control, Volumes 1 and 2. *ASME International Mechanical Engineering Congress and Exposition*. 2003 Nov; Washington DC, USA. 861–868. doi: 10.1115/IMECE2003-41192.
- [15] Arnold M. Stability of Sequential Modular Time Integration Methods for Coupled Multibody System Models. *Journal of Computational and Nonlinear Dynamics*. 2010; 5(3): 031003. doi: 10.1115/1.4001389.
- [16] Schweizer B, Li P, Lu D, Meyer T. Stabilized implicit co-simulation methods: solver coupling based on constitutive laws. *Archive of Applied Mechanics*. 2015; 85(11): 1559–1594. doi: 10.1007/s00419-015-0999-2.
- [17] Arnold M, Clauss C, Schierz T. Error Analysis and Error Estimates for Co-Simulation in FMI for Model Exchange and Co-Simulation V2.0. *Archive of Mechanical Engineering*. 2013; LX(1). doi: 10.2478/meceng-2013-0005.

¹<https://fmi-standard.org/>

A Stable but Explicit Cosimulation Coupling Method

Thilo Moshagen

Hochschule Wismar, Fakultät Technik, Bereich Maschinenbau

Abstract. The term *co-simulation* denotes the coupling of some simulation tools for dynamical systems into one big system by having them exchange data at points of a fixed time grid and extrapolating the received data into the interval, while none of the steps is repeated for iteration. From the global perspective, the simulation thus has a strong explicit component. Frequently, among the data passed across subsystem boundaries there are flows of conserved quantities, and as there is no iteration of steps, system-wide balances may not be fulfilled: the system is not solved as one monolithic equation system. If these *balance errors* accumulate, simulation results become inaccurate. Balance correction methods which compensate these errors by adding corrections for the balances to the signal in the next coupling time step have been considered in past research. But establishing the balance of one quantity *a posteriori* due to the time delay in general cannot establish the balances of quantities that depend on the exchanged quantities, usually energy. In most applications from physics, the balance of energy is equivalent to stability. In this paper, a method is presented which allows users to choose the quantity that should be balanced to be that energy, and to accurately balance it. This establishes also numerical stability for many classes of stable problems.

Co-simulation, coupled problems, simulator coupling, explicit coupling, stability, convergence, balance correction

1 Introduction

With the rise of simulation software for technical systems emerged the desire to couple those simulations in order to take into account the influence the systems exercise onto each other. In other words, these systems are now viewed as subsystems which form one big system.

One now wants to simulate this large system, using the subsystems' simulator software and coupling it by sharing data. What used to be a parameter when the systems were calculated separately is given now by a state variable of the other subsystem, reading:

$$S_1 : \dot{x}_1 = f_1(x_1, x_2, z_1, z_2) \quad (1)$$

$$0 = g_1(x_1, x_2, z_1, z_2) \quad (2)$$

$$S_2 : \dot{x}_2 = f_2(x_1, x_2, z_1, z_2) \quad (3)$$

$$0 = g_2(x_1, x_2, z_1, z_2). \quad (4)$$

Here, the (x_1, x_2) are the differential, the (z_1, z_2) are the algebraic states. The setting generalizes to n subsystems in a straightforward way, and it includes parabolic par-

tial differential equations. We require that the derivatives $d_{z_i} g_i$ have full rank. Such each of the S_i is an index-1 differential-algebraic system if the $(x_{k \neq i}, z_{k \neq i})$ are seen as parameters of it. The influence of x_2, z_2 in a split setting is therefore modeled by parameters u_{12} in S_1 and x_1, z_1 as parameters u_{21} :

$$S_1 : \dot{x}_1 = f_1(x_1, z_1, u_{12}) \quad (5)$$

$$0 = g_1(x_1, z_1, u_{12}) \quad (6)$$

$$S_2 : \dot{x}_2 = f_2(x_2, z_2, u_{21}) \quad (7)$$

$$0 = g_2(x_2, z_2, u_{21}). \quad (8)$$

When coupled, the u_{ij} are determined by the coupling conditions

$$0 = h_{21}(x_1, z_1, u_{21}) \quad (9)$$

$$0 = h_{12}(x_2, z_2, u_{12}) \quad (10)$$

that have to be fulfilled, and exchanged at fixed time nodes T_k . Between them, the u_{ij} are extrapolated. To

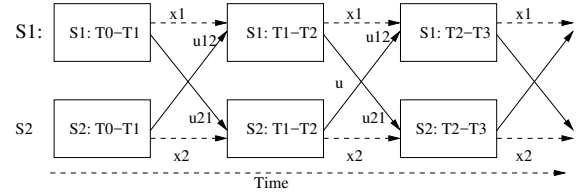


Figure 1: Explicit co-simulation scheme.

establish coupling, the h_{ik} must be solvable with respect to the u_{ik} . The $d_{z_i} g_i$ have full row rank, too. Such, the differential-algebraic system given by Equations (5) - (10) is again of index 1.

This description of the setting is widespread [2].

It is commonly said that the coupling is done by a *co-simulation scheme* if the u_{ij} are calculated from Equations (9) and (10) at exchange time nodes T_k and then passed on to S_2 and S_1 , respectively. Of course, some extrapolation of u_{ij} into $[T_k, T_{k+1})$ is required. Considerable research has been done on coupling [6, 7, 5]. A lot of methods repeat the timestep after the calculation with an extrapolation that has been improved with respect to some objective. Thus, they are implicit, see e.g. [5]. For

a convergence proof, see [2]. The convergence of explicit co-simulation methods for ODE and index one DAE not surprisingly as well improves with the extrapolation order of subsystems input [8, 2]. The situation here in simulator coupling mirrors the one in ODE solvers: The explicit solvers are quick in each step but not stable [12, 8], while the implicit ones require iterations within each step but usually ensure some stability. When used for stiff problems, the explicit schemes require such small stepwidths that the implicit schemes are finally cheaper. Also, implicit algorithms for coupled solvers require additional programming and storage. Therefore, the co-simulation scheme, where one just proceeds to the next timestep (Figure 1), is still popular.

So far, it has been common sense that the usual stability classifications like A - and B-Stability cannot be achieved with explicit algorithms [14]. A solution for these stability issues would be helpful in many applications and is the subject of this contribution.

It is important to note that all results and figures herein have been published before in [1]. This contribution is a highly condensed presentation of that content for the purpose of reaching the engineering community rather than novelty.

2 The lack of stability

2.1 Stability classifications

For readability, we present the concepts of stability classifications of methods.

Definition 2.1 (Stable points of ODE) Let x^* be an equilibrium point of the ODE $\dot{x} = f(x)$ and $\phi^t x$ the solution for the initial value $x(t_0) = x$. Then x^* is

- stable if $\forall \varepsilon > 0 \exists \delta > 0 : \|x - x^*\| < \delta \Rightarrow \|\phi^t x - x^*\| < \varepsilon \forall t \in [t_0, T]$
- asymptotically stable if $\exists r > 0 : \|x - x^*\| < r \Rightarrow \lim_{t \rightarrow \infty} \phi^t x = x^*$.

Definition 2.2 (Stable Point of Difference equation)

Let x^* be an equilibrium point of the k -th order difference equation $x^{n+1} = f(x^n, \dots, x^{n-k})$. Then x^* is classified as in Definition (2.1) where x is replaced by x_n and $\forall t \in [t_0, T]$ by $\forall n \in \{1, \dots, N\}$ and furthermore $t \rightarrow \infty$ by $n \rightarrow \infty$.

Using these two definitions, stability classifications like zero-, A- or B-stability are defined: The respective stability of a method is the inheritance of the stability of an equilibrium point of a certain ODE class to the equilibrium point of the difference equation yielding from the application of the numerical scheme.

2.1.1 Stability, consistency and convergence

In this framework, zero stability of a numerical method means that the difference equation that one gets by applying the method to $\dot{x} = 0$ is stable. It is well-known that this is a necessary condition for convergence [13, 14]. But this condition is fulfilled by all one-step methods¹ as $x_{n+1} = x_n + 0$ is a stable equation. So unlike for multistep methods, there is no need here to examine zero-stability when one examines convergence of one-step methods. It frequently causes confusion that zero stability in the original paper [13] was labeled *stability* only, and with this nomenclature Lax's and Richtmyers' theorem is given in an equation-like form *stability + consistency = convergence*.

2.2 Stability

These results were confirmed numerically in [8, Sec.3.2] using the two-dimensional linear problem

$$\dot{x} = Ax, \quad (11)$$

which with

$$A = \begin{pmatrix} 0 & 1 \\ -\frac{c}{m} & 0 \end{pmatrix}, \quad x = \begin{pmatrix} x \\ \dot{x} \end{pmatrix} \quad (12)$$

can be interpreted as linear spring-mass oscillator with mass m and spring constant c . This problem is the most simple problem possible that is linear and can be splitted. The original problem is marginally stable, so stable, as its spectrum is purely imaginary.

Written as a co-simulation problem, Problem (11) with (12) yields Table 1. In [8] and [12] it is shown that co-simulation schemes are not stable for linear problems, even not for stable subsystem solvers. The stability for linear problems replaces the notion of A-stability, as the one-component equation used there cannot be split. When treated with a co-simulation scheme (output of the spring is the force $f = -cx$, that of the mass is the velocity $v = \dot{x}$) the emerging (method-induced) ODE

$$\begin{cases} \dot{x}_1 = a_{1,1}x_1 + a_{1,2}\text{Ext}(x_2) \\ \dot{x}_2 = a_{2,2}x_2 + a_{2,1}\text{Ext}(x_1) \end{cases} \quad (15)$$

is obviously unstable [8, Section 2.5].

Its numerical solution is shown in Figure 2, – there is no linear stability for general step sizes. This means $\|x\| \rightarrow \infty$ for $t \rightarrow \infty$. The energy of our system is $E = \frac{1}{2}mv^2 + \frac{1}{2}cs^2 = \langle x, \frac{1}{2} \text{diag}(m, c)x \rangle = \langle x, x \rangle_{\frac{1}{2} \text{diag}(m, c)} =$

¹One-step methods can be written as $x_{n+1} = x_n + h\psi(x_n, t_n, h_n)$, and $\psi(x_n, t_n, 0) = f$, where f is the ODE's right hand side.

Spring	System States	Mass
$x_1 := s = x$		$x_2 := v = \dot{x}$
	Outputs	
$u_{21} := F = -cx$		$u_{12} := v = \dot{x}$
	Inputs	
u_{12}		u_{21}
	Equations	
$\dot{x}_1 = \text{Ext}(u_{12}) = v$		$\dot{x}_2 = -\frac{1}{m} \text{Ext}(u_{21})$ $= -\frac{F}{m}$

Spring	System States	Mass
...		...
	Outputs	
$u_{21} := (f, \dot{f})$ $= (-cx, -cv)$	(13)	$u_{12} := (v, a)$ $= (\dot{x}, \dot{f}/m)$
	Inputs	
u_{12}		u_{21}
	Equations	
\vdots		\vdots

Table 1: Standard Co-simulation schemes for the spring-mass system, top constant, down linear extrapolation. When there is no difference, dots have been used.

$\|x\|_{\frac{1}{2} \text{diag}(m,c)}$, which is an equivalent norm, so lack of stability is equivalent to energy augmentation.

Using piecewise constant extrapolation of inputs, the force, as it is seen by the mass, is effectively shifted to later times: a value from time T_i is used for all future times $t \in (T_i, T_{i+1})$. The analogy with the reactive power and the real power of an electrical network is apparent. Work from oscillating systems with phase shift contains an integral over a constant and thus grows unbounded (Figure 2). Similar arguments hold for linear extrapolation co-simulation.

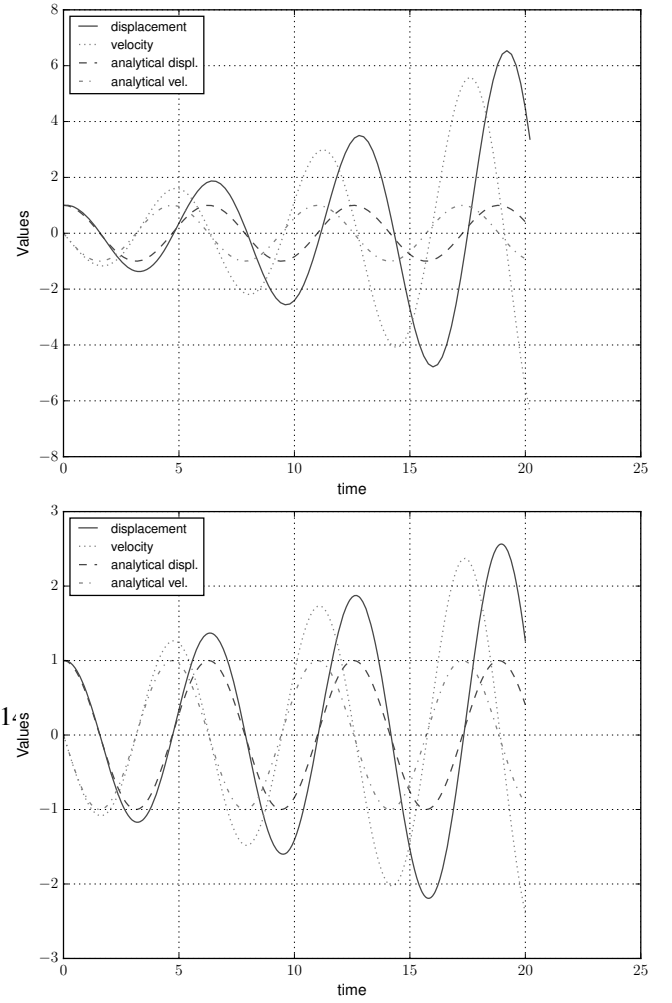


Figure 2: Simulation of the system (11)-(12) in the co-simulation scheme with constant extrapolation, varying the exchange step size H . Upper row, left: $H = 0.2$, right: $H = 0.1$. Previously published in [12].

3 Enforcing balance by sharing the view on potential flow

3.1 The proposed method

The key feature to establish energy balance is exchanging the value of power and calculating the variable of interest from that power. Consider a co-simulation problem with subsystems S_1 and S_2 as given by Equations (5) - (10) with states x_1 and x_2 respectively and inputs u_{21} and u_{12} . We suggest the following procedure to enforce energy balance between subsystems S_1 and S_2 :

1. At data exchange timepoint T_n the powers P_{ij} as the

flux of energy are calculated in both subsystems, using up-to-date input u_{ji}^n . In general $P_{ij} \neq P_{ji}$.

Applied to the $S_1 - S_2$ setting, P_{21} (the power calculated in S_1 for passing to S_2), is calculated using fresh u_{12} , the input into S_1 . Now in the input vectors u_{ij} one component is replaced by P_{ij} and that new vector $u_{21, \text{bal}}$ is exchanged between the subsystems. Applied to $S_1 - S_2$ setting, the value P_{21} replaces one component $(u_{21})_m$ of u_{21} , and respectively, P_{12} replaces $(u_{12})_n$ of u_{12} .

This means S_1 's point of view about the power has been passed on to S_2 and vice versa.

2. Now both subsystems have the same information and thus the opportunity to draw the same conclusion on what energy exchange should be assumed. We denote this assumed energy exchange as

$$\hat{P}_{12}(P_{21}, P_{12}) = -\hat{P}_{21}, \quad (16)$$

a straightforward choice is $\hat{P}_{21} = (P_{12} - P_{21})/2 = -\hat{P}_{12}$, where now it is necessary to define flow directions: P_{ij} shall be negative if it leaves S_j , so it is counted with opposite sign in S_i .

Again remember that P_{21} is the power calculated in S_1 for passing to S_2 , calculated using u_{12} , the input into S_1 . The former input $(u_{12})_n(t)$ now is calculated subject to

$$P_{21}(x_1(t), u_{12 \setminus n}, (u_{12})_n(t)) = \text{Ext}(\hat{P}_{12}). \quad (17)$$

Analogously $(u_{21})_m(t)$ s.t. $P_{12}(x_2, u_{21 \setminus m}, (u_{21})_m) = \text{Ext}(\hat{P}_{21})$ is calculated. The expression $12 \setminus k$ in subscript is to say that the k -th component of the vector is left out. For the unique inversion of P_{ij} it is required that the maps $(u_{ji})_k \rightarrow P_{ij}(\cdot, \cdot, (u_{ji})_k)$ are strictly monotone.

As $\text{Ext}(\hat{P}_{12}) = -\text{Ext}(\hat{P}_{21})$, now it is established that the inputs of S_1 and S_2 are consistent in terms of energy conservation for all t .

3.2 Example

To apply the scheme given in Section 3.1 above to a spring-mass system (12), replacing the standard co-simulation scheme from Table 1, one first calculates the energies of the systems parts, powers acting on subsystems boundaries, and their derivatives. As $P_i = \dot{W}_i$, $P_i < 0$ indicates that energy leaves S_i .

Spring	Energy	Mass
$W = \int -f ds$ $= \int -f v dt$		$W = \int f ds$ $= \int m a v dt$
Power		
$P = \dot{W}$ $= -f v = c x v$		$P = \dot{W}$ $= m a v = f v$
Derivative of Power		
$\dot{P} = c(v^2 + s a)$		$\dot{P} = m(a^2 + v \dot{a})$ $= m(a^2 + v \frac{\dot{f}}{m})$

The derivative of force \dot{f} is available as output of spring, as it is usually needed for linearly extrapolating the input. With this, the scheme yields Table 2.

4 Stability of power balanced schemes

As discussed in Section 2.2 and shown in [8], stability for linear systems of a partly explicite scheme is not given. This section shall relate energy conservation of our method to stability. The class of problems under consideration are all stable *gradient flow problems*

$$\dot{x} = -M \nabla_x \mathcal{P}^T, \quad (18)$$

which is a huge class, containing entropy driven and energy conserving problems. The *mobility Matrix* M determines the systems stability - it is positive definite if the system is dissipative and skew if energy conserving. This behavior must be inherited to the ODE that is induced by our splitting method. We give an outline of the arguments:

1. Switch to gradient flow view. In this, inserting (18) into the time derivative of the respective potential $\dot{\mathcal{P}}(x)$ yields

$$\dot{\mathcal{P}}(x) = \langle \nabla_x \mathcal{P}(x), \dot{x} \rangle = \langle \nabla_x \mathcal{P}(x), -M \nabla_x \mathcal{P}(x)^T \rangle \quad (19)$$

with the scalar product $\langle \cdot, \cdot \rangle$.

2. Introduce split system

Spring	System States	Mass
$x_1 := s = x$		$x_2 := v = \dot{x}$
Outputs		
$(u_{21,\text{Std}})_1 := f = -cx$		$(u_{12,\text{Std}})_1 := v = \dot{x}$
$(u_{21,\text{Std}})_2 := \dot{f} = -cv$		$(u_{12,\text{Std}})_2 := \dot{v} = f/m$
(intermediately exchanging $u_{ij,\text{Std}}$)		
$(u_{21})_1 = P(x_1, u_{12})$ $= cxv = cx_1(u_{12})_1$		$(u_{12})_1 = P(x_2, u_{21})$ $= fv = (u_{21})_1 x_2$
$(u_{21})_2 = \dot{P}(x_1, u_{12})$ $= c(v^2 + xa)$ $= c((u_{12})_1^2 + x_1(u_{12})_2)$		$(u_{12})_2 = \dot{P}(x_2, u_{21})$ $= m(a^2 + v\frac{\dot{f}}{m})$ $= m\left(\frac{(u_{21})_1^2}{m} + x_2\frac{(u_{21})_2}{m}\right)$
Inputs		
$(u_{12})_1 := \hat{P}$		$(u_{21})_1 := -\hat{P}$
$(u_{12})_2 := \hat{P}$		$(u_{21})_2 := -\hat{P}$
Input variables of standard method u_{std} depending on Power		
$v = \frac{\text{Ext}(\hat{P})}{cs} = \frac{\text{Ext}(u_{12})_1}{cx_1}$		$f = -\frac{\text{Ext}(\hat{P})}{v} = \frac{\text{Ext}(u_{21})_1}{x_2}$
Equations		
$\dot{x}_1 = v$		$\dot{x}_2 = \frac{f}{m}$

Table 2: Method form Section 3.1 applied to the spring-mass system

- Identify coupling contributions
- Characterize potential conservation/dissipation properties (see below)

3. See method as decoupling ODE – Insert calculation

of inputs from power into original equations

4. Relate decoupled ODEs stability properties to stability of original systems

- Show that negotiated exchange conserves $\dot{\mathcal{P}} \leq 0$. It can be shown and there are straightforward arguments that there is no unphysical power production when sharing subsystems agree on the exchanged energy
- Use Lyapunov's direct method on the decoupled system.
- Additionally, one can argue that maximum stable stepwidth for dissipative systems is augmented (method is closer to B-stability than extrapolation of inputs method).

5. If such stable subsystems ODEs are solved with methods preserving that stability, overall solution will be stable.

Items (2) and also (4) need closer consideration. The split systems potential production $\dot{\mathcal{P}}(x)$ according to Eq. (19) in subsystem-wise block matrix form reads

$$\dot{\mathcal{P}}(x) = P_k + P_l + \dots \quad (20)$$

$$= \begin{pmatrix} (\nabla_x \dot{\mathcal{P}}(x))_{I_k} \\ \vdots \\ (\nabla_x \dot{\mathcal{P}}(x))_{I_l} \end{pmatrix} \cdot \dots \quad (21)$$

$$\begin{pmatrix} \dots & -(M)_{I_k, I_k} & \dots & -(M)_{I_k, I_l} & \dots \\ \dots & -(M)_{I_l, I_k} & \dots & -(M)_{I_l, I_l} & \dots \end{pmatrix} \begin{pmatrix} (\nabla_x \dot{\mathcal{P}}(x))_{I_k} \\ \vdots \\ (\nabla_x \dot{\mathcal{P}}(x))_{I_l} \end{pmatrix} \quad (22)$$

$$= \underbrace{\left\langle \nabla_{x_{I_k}} \dot{\mathcal{P}}(x), -M_{I_k, I_k} \nabla_{x_{I_k}} \dot{\mathcal{P}}(x)^T \right\rangle}_{P_{kk}} \quad (23)$$

$$+ \underbrace{\left\langle \nabla_{x_{I_k}} \dot{\mathcal{P}}(x), -M_{I_k, I_l} \nabla_{x_{I_l}} \dot{\mathcal{P}}(x)^T \right\rangle}_{P_{kl}} \quad (24)$$

$$+ \underbrace{\left\langle \nabla_{x_{I_l}} \dot{\mathcal{P}}(x), -M_{I_l, I_l} \nabla_{x_{I_l}} \dot{\mathcal{P}}(x)^T \right\rangle}_{P_{ll}} \quad (24)$$

we identify

$$P_{kl} := \left\langle (\nabla_x \dot{\mathcal{P}}(x))_{I_k}, -(M)_{I_k, I_l} (\nabla_x \dot{\mathcal{P}}(x)^T)_{I_l} \right\rangle \quad (25)$$

as the *potential production* in S_k by S_l 's variables, or power acting from subsystem l onto subsystem k . Item (4) now means that those eliminate in the suggested scheme, as the exchanging subsystems agree on their value. So, there is no contribution to \mathcal{P} by the extrapolation during coupling.

Theorem 4.1 *For a Lyapunov stable (asymptotically stable) gradient flow initial value problem (IVP), the IVP resulting from the energy balancing method as described in Section 3.1 is also stable (asymptotically stable).*

5 Discussion, conclusion and future work

The suggested method overcomes the decade-old issue of stability in coupled simulation for a huge class of problems.

Moreover, the method has a clear interpretation in physics: the enforcement of the power balance in systems interactions. It can therefore be implemented by anyone with understanding of the systems they want to couple, without deep knowledge of numerical analysis. For simulations in industrial research and development, the new method enables stable calculations with big timesteps and few programming effort and is thus a big step forward.

References

- [1] Moshagen, Thilo, *On Meeting Energy Balance Errors in Co-Simulations*, Mathematical and Computer Modelling of Dynamical Systems, 25, pp 139-166, Publisher: Taylor & Francis, 2019, Available at doi:10.1080/13873954.2019.1595667.
- [2] M. Arnold and M. Günther, *Preconditioned Dynamic Iteration for Coupled Differential-Algebraic Systems*, BIT Numerical Mathematics 41 (2001), pp. 1–25, Available at <http://dx.doi.org/10.1023/A3A1021909032551>.
- [3] M. Arnold, C. Clauss, and T. Schierz, *Error Analysis and Error Estimates for Co-Simulation in FMI for Model Exchange and Co-Simulation v2.0*, Progress in Differential-Algebraic Equations 60.1 (2013), pp. 75–94, Available at doi:10.2478/meceng-2013-0005.
- [4] M. Arnold, C. Bausch, T. Blochwitz, C. Clauss, M. Monteiro, T. Neidhold, J.V. Peetz, and S. Wolf, *Functional Mock-up Interface for Co-Simulation* (2010), Available at https://svn.modelica.org/fmi/branches/public/specifications/v1.0/FMI_for_CoSimulation_v1.0.pdf.

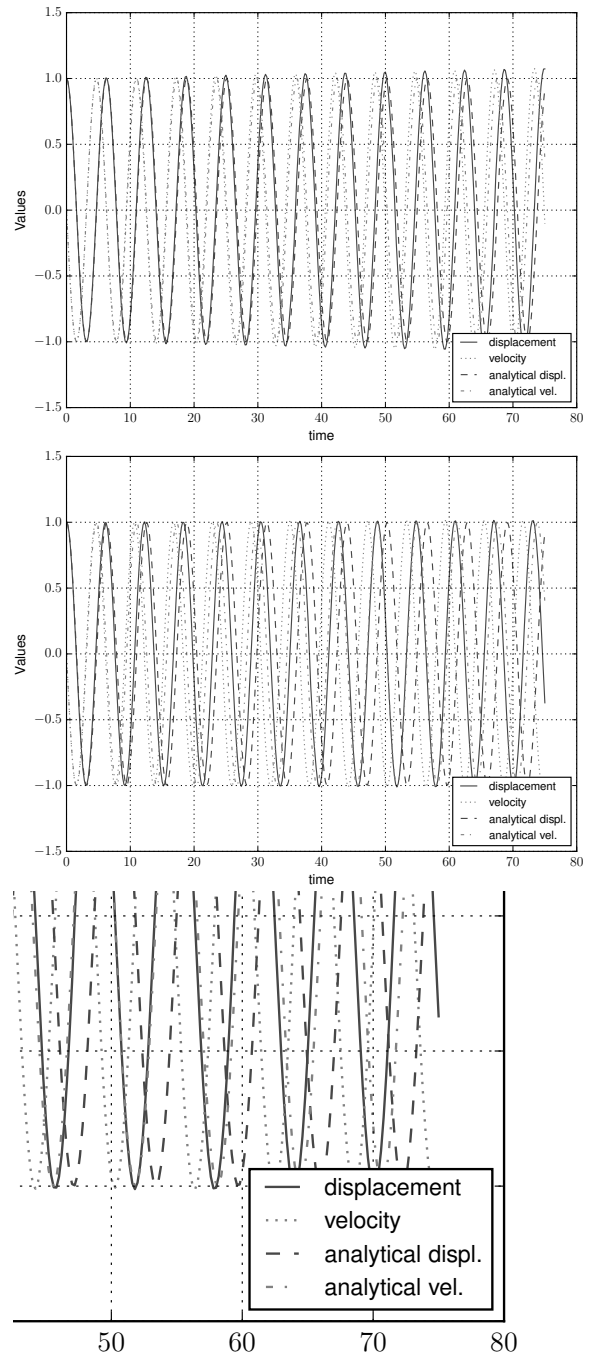


Figure 3: Stability of cosimulation schemes applied to spring-mass system: Top: Linear extrapolation, middle and bottom: Power balanced scheme. The image on bottom is an amplification of the lower right corner of the image above to confirm that there is no energy gain. $T_{\text{end}} = 75$, exchange stepwidth $H = 0.2$, subsystems refinement decisions left to subsystems solvers, stable vode used on subsystems.

- [5] M. Busch, *Zur effizienten Kopplung von Simulationsprogrammen*, Ph.D. diss., Universität Kassel, 2012, Available at <http://books.google.de/books?id=0qBpXp-f2gQC>.
- [6] R. Kübler and W. Schiehlen, *Modular Simulation in Multibody System Dynamics*, Multibody System Dynamics 4 (2000), pp. 107–127, Available at <http://dx.doi.org/10.1023/A:1009810318420>.
- [7] S.B.E. Elmqvist, *Interface Jacobian-based Co-Simulation*, International Journal for Numerical Methods in Engineering 98 (2014), pp. 418–444, Available at <http://dx.doi.org/10.1002/nme.4637>.
- [8] T. Moshagen, *Convergence of explicitly coupled Simulation Tools (Co-Simulations)*, Journal of Numerical Mathematics (2018), p. 27.
- [9] R. Kossel, *Hybride Simulation thermischer Systeme am Beispiel eines Reisebusses*, Ph.D. diss., Techn. Univ. Braunschweig, 2012.
- [10] D. Scharff, C. Kaiser, W. Tegethoff, and M. Huhn, *Ein einfaches Verfahren zur Bilanzkorrektur in Kosimulationsumgebungen*, in *SIMVEC - Berechnung, Simulation und Erprobung im Fahrzeugbau*. 2012.
- [11] M. Wells J.; Hasan and C. Lucas, *Predictive Hold with Error Correction Techniques that Maintain Signal Continuity in Co-Simulation Environments*, SAE Int. J. Aerosp (2012), pp. 481–493.
- [12] D. Scharff, T. Moshagen, and J. Vondřejc, *Treating Smoothness and Balance during Data Exchange in Explicit Simulator Coupling or Cosimulation* (2017), p. 30, Available at <https://arxiv.org/abs/1703.05522>.
- [13] P.D. Lax and R.D. Richtmyer, *Survey of the stability of linear finite difference equations*, Communications on Pure and Applied Mathematics 9 (56), pp. 267–293, Available at <https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160090206>.
- [14] P. Deuflhard and F.A. Bornemann, *Numerische Mathematik II*, de Gruyter, 1994.
- [15] *Scipy Integration and ODEs Web Documentation*. Available at <https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.ode.html>.
- [16] K. Brenan, S. Campbell, and L. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, Society for Industrial and Applied Mathematics, 1995, Available at <http://epubs.siam.org/doi/abs/10.1137/1.9781611971224>.
- [17] T. Moshagen, *Diffuse Grenzflächen thermodynamisch scharf - ein voll physikalisch eingebettetes Multiphasenfeldmodell*, Ph.D. diss., Universität Bremen, 2011, Available at <http://nbn-resolving.de/urn:nbn:de:gbv:46-00101865-17>, Online-Ressource (PDF: 208 S., 22,5 MB).
- [18] A.V.D. Schaft, *Port-Hamiltonian Systems: an introductory Survey* (2006). Available at http://www.icm2006.org/proceedings/Vol_III/contents/ICM_Vol_3_65.pdf.
- [19] A. van der Schaft and B.M. Maschke, *Port-hamiltonian Systems: a Theory for Modeling, Simulation and Control of Complex Physical Systems* (2003). Available at http://www-lar.deis.unibo.it/euron-geoplex-sumsch/files/lectures_1/Van%20Der%20Schaft/VDSchaft_01_PCHS.pdf.
- [20] J. Dormand and P. Prince, *A Family of embedded Runge-Kutta Formulae*, Journal of Computational and Applied Mathematics 6 (1980), pp. 19 – 26, Available at <http://www.sciencedirect.com/science/article/pii/0771050X80900133>.

Model Order Reduction of Deterministic Microscopic Models - A Case Study

Matthias Rößler^{1*}, Niki Popper^{1,2}

¹dwh GmbH, dwh simulation services, Neustiftgasse 57-59, 1070 Vienna, *matthias.roessler@dwh.at

²DEXHELPP Society of Decision Support for Health Policy and Planning, Neustiftgasse 57-59, 1070 Vienna, Austria

Abstract. In this paper we present a method for model order reduction of microscopic models, i.e. models that consist of a high number of entities that can interact and cooperate with each other. Due to this high numbers of entities such models are often highly computationally expensive. But classic model order reduction techniques often use the equations the models are based on to simplify the model and make it more performant. These approaches are not applicable for microscopic models. We present a data-based approach for model order reduction using radial basis functions and analyze the specifics and opportunities of model reduction for microscopic models. As a case study Conway's Game Of Life is used.

Introduction

Microscopic models are typically comprised of a high number of entities that interact with each other in a certain way. In contrast to macroscopic models where the global behavior of the system is described, the dynamics of a microscopic model emerge through the definition of the single entities and their interaction. On the one hand the high number of entities results in computationally expensive simulations on the other hand this usually leads to a high number of parameters that define the behavior of the model. Different modeling approaches lead to microscopic models, for example cellular automata (CA) or agent based modeling approaches (ABM). For this paper we chose to study CAs, specifically Conway's Game of Life.

Parametrized model order reduction (PMOR) aims at reducing the computation time of a parametrized models. Application fields include control theory, optimization or statistical analysis. There are many different approaches that use the underlying model equations (for example see [1], [2], or [3]), which is not suitable for microscopic models as the underlying equations are

not available directly. The most promising approaches that are based on interpolation and are independent of the availability of model equations are techniques that use radial basis functions (RBF) [4]. This approach is based on already available simulation results at other parameter constellations and uses interpolation to approximate the simulation result at a given parameter set.

In this paper the different possibilities of applying RBF interpolation on data generated from microscopic models is investigated. Conway's game of life is used as a stand-in for a population model. The analysis focuses on different possibilities to use the generated simulation results in order to create a suitable interpolant and the results are compared to each other.

1 RBF Interpolation

A closer view of the theory of radial basis functions can be found in [5] or [6]. We summarise some important results.

A function $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$ is called radial, if there exists a univariate function $\phi : \mathbb{R} \rightarrow \mathbb{R}$ with

$$\psi(x) = \phi(\|x\|) \quad \forall x \in \mathbb{R}^d \quad (1)$$

where $\|\cdot\|$ is a norm on \mathbb{R}^d , usually the euclidian norm.

An interpolation problem using radial basis functions can be formulated as follows:

Given a set of points $\{x_1, \dots, x_n\}$ (called centers) and a function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, for which the function evaluations $f(x_k)$, $k = 1, \dots, n$ at the centers are known, an interpolant s_f of f is given by

$$s_f(x) := \sum_{k=1}^n a_k \cdot \phi(\|x - x_k\|). \quad (2)$$

s_f must fulfill the interpolation conditions $s_f(x_k) = f(x_k)$, $k = 1, \dots, n$

The problem leads to a linear system for a_k :

$$A_\varphi \cdot \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix} \quad (3)$$

with

$$A_\varphi := \begin{pmatrix} \varphi(\|x_1 - x_1\|) & \cdots & \varphi(\|x_1 - x_n\|) \\ \vdots & \ddots & \vdots \\ \varphi(\|x_n - x_1\|) & \cdots & \varphi(\|x_n - x_n\|) \end{pmatrix} \quad (4)$$

Obviously A_φ is symmetric. It can be shown that the matrix is positive definite for arbitrary, distinct $x_1, \dots, x_n \in \mathbb{R}^d$ for a certain group of functions. These functions are called (conditionally) positive definite, examples include:

- Linear:

$$\varphi(\|x\|) = \|x\| \quad (5)$$

- Gaussian:

$$\varphi(\|x\|) = e^{-\frac{\|x\|^2}{\varepsilon^2}} \quad (6)$$

- Multiquadric:

$$\varphi(\|x\|) = \sqrt{1 - \frac{\|x\|^2}{\varepsilon^2}} \quad (7)$$

While gaussian functions are positive definite i.e.

$$c^\top A_\varphi c > 0 \quad (8)$$

for arbitrary $c \neq 0 \in \mathbb{R}^n$, linear and multiquadric functions are conditionally positive definite of order 1. This means that c has to additionally fulfill

$$\sum_{k=1}^n c_k = 0. \quad (9)$$

This can be directly incorporated in (3) by adding an extra equation, which leads to

$$\begin{pmatrix} A_\varphi & \mathbf{1}_{n \times 1} \\ \mathbf{1}_{1 \times n} & 0 \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ \vdots \\ a_n \\ d \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \\ 0 \end{pmatrix} \quad (10)$$

The shape parameter ε for the gaussian and multiquadric functions has to be chosen carefully, because

the quality of the approximation is highly dependent on this parameter.

2 Case Study

Conway's Game of Life [7] is one of the most famous cellular automata. It is defined on a rectangular grid with 2 possible states $\{dead, alive\}$. The update rules are defined as follows:

- A cell that is *alive* stays alive if 2 or 3 neighboring cells are alive otherwise it dies.
- A cell that is *dead* is brought to life if there are exactly 3 living cells in its neighborhood. Otherwise it stays dead.

As neighborhood the Moore-neighborhood (8 neighboring cells, the 4 directly adjacent and the 4 diagonal adjacent cells) is used. The behavior of the model is dependent on the initial states of the cells and it could be called very chaotic as a change of the initial state in a single cell can lead to huge differences after several time steps.

Typically the results of the game of life model are analysed based on the spatial distribution of the living cells over time. But the model can also be viewed as a population model, where the number of living entities is of interest. So the analysis of simulation results can be performed on an aggregated level by counting the living cells in every time step, a similar look at the game of life was done in [8]. For the experiments the game of life is simulated on a 50×50 grid. The time evolution is observed over 50 time steps. For the initial conditions each cell is given a probability of 0.3 to be alive at $t = 0$, an overview is given in Table 1.

grid ($m \times n$)	50×50
time steps (t_{end})	50
init. prob. living (p)	0.3

Table 1: Basic parameters for game of life.

Figure 1 depicts the time evolution of the number of living cells for various simulation runs using the parameters given in Table 1. While the chaotic behavior of the game of life is evident in the evolution as well, it can be seen that the basic behavior of the evolution is similar for most runs. Throughout the paper the evolution (*pop*) will be given as relative frequency of living

cells against the number of total cells in the cellular automaton ($m \cdot n = 2500$). It can be seen as timeseries with 51 entries corresponding to times $t = 0, \dots, 50$.

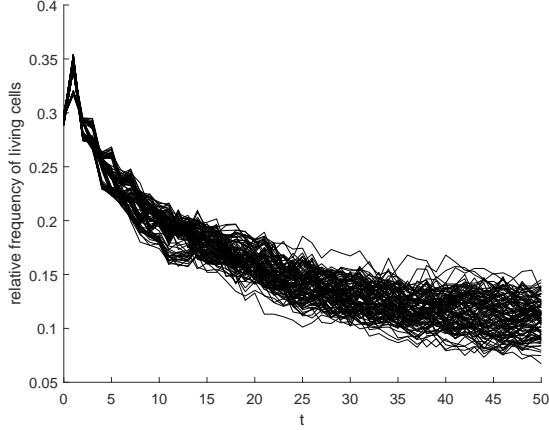


Figure 1: Time evolution of living cells for various simulation runs.

As a measure for the distance between simulation runs s_1, s_2 the Frobenius norm is used:

$$d(s_1, s_2) = \|IC_1 - IC_2\|_F \quad (11)$$

where $IC_1, IC_2 \in \{0, 1\}^{50 \times 50}$ are the initial condition matrices of the respective simulation runs.

3 Experiments

During the experiments two aspects were investigated:

- influence of number of used interpolation points
- influence of minimal distance between an evaluation point and an interpolation point.

A single experiment was built the following way:

1. Fix the minimal distance between an evaluation point and an interpolation point ($d_{min} \in \{1, \dots, 20\}$).
2. Randomly create 10 evaluation points (ep_i) and the evolution of the number of living cells $pop_i(t)$ as reference.
3. Create an interpolation point ($ip_{i,j}$) for each ep_i and add them, as well as their corresponding evolutions $pop_{i,j}(t)$, to the set of interpolation points.

4. Perform the interpolation on the current set of interpolation points and calculate the errors between the interpolated population evolution ($\overline{pop}_i(t)$) and its reference $pop_i(t)$.

5. Repeat steps 1-4 20 times.

The presented experiment setup results in a sequence of interpolations that use more and more simulation results as interpolation points (10 in the first iteration and 200 in the last one). Additionally, it is ensured that for each evaluation point exactly one simulation result with the given distance is added at every iteration.

The error at an evaluation point is calculated as

$$err_i = \frac{\sum_{t=0}^{t_{end}} \|pop_i(t) - \overline{pop}_i(t)\|_2}{t_{end} + 1} \quad (12)$$

and the error of an iteration of the experiment is given as the mean error over all evaluation points.

3.1 Direct Interpolation

For direct interpolation, i.e. directly calculating the resulting evolutions at the given evaluation points, 2 approaches can be distinguished:

- The first idea is to directly interpolate the population evolutions. This means to directly take the initial conditions of the simulation runs as input of the interpolation and the evolutions as the output.

$$s_f : \{0, 1\}^{50 \times 50} \rightarrow \mathbb{R}^{51} \quad (13)$$

- The second idea is to take the initial conditions and the points in time as input for the interpolation and getting the population size (number of living cells) at a specific point in time as an output.

$$s_f : \{0, 1\}^{50 \times 50} \times \mathbb{N} \rightarrow \mathbb{R} \quad (14)$$

For the direct interpolation the use of linear functions and multiquadric functions yielded the best results. So they are presented here. For the multiquadric function $\varepsilon = 1$ was chosen.

Figures 3 and 4 show the errors of the presented approaches using linear and multiquadric RBF-functions. While the errors for the time series approach are basically the same for the different used functions, the errors for the multiquadric pointwise approach are higher than for the linear pointwise approach. It is also more

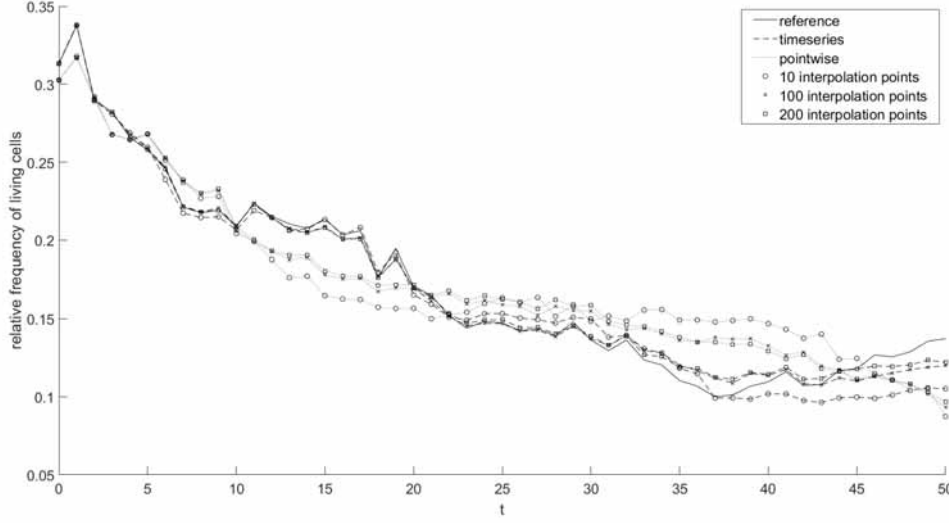


Figure 2: Interpolated time series of a single evaluation point using interpolation data with $d_{min} = 1$ and linear function.

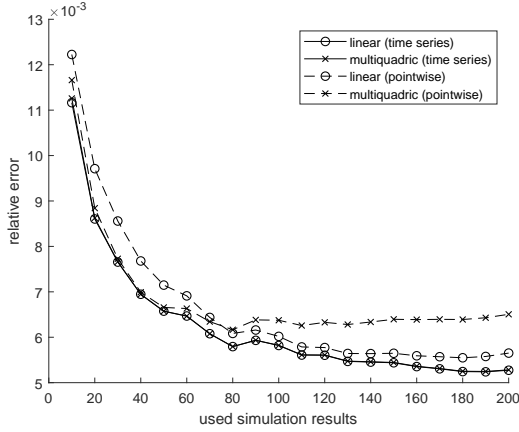


Figure 3: Error of interpolation over used simulation results for $d_{min} = 1$

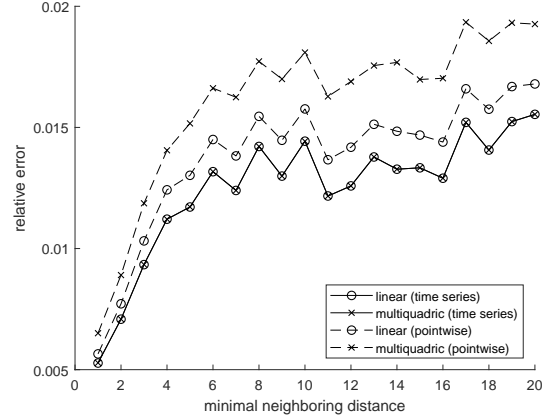


Figure 4: Error of interpolation over minimal neighboring distance for 200 used interpolation points

dependent on the minimal distance to the next interpolation point. In order to gain a closer look at the emergence of the errors Figure 2 shows the results of a single interpolation using the linear function. Additionally, the results of using different numbers of interpolation points are depicted, as the minimal neighboring distance $d_{min} = 1$, i.e. the best possible data, was used. It can be seen that the time series interpolation follows the chaotic behavior of the underlying evolution much more closely than the pointwise interpolation. One possible explanation is that if all points in time are used as

interpolation points separately they have a much higher impact on the interpolation result, especially if the distance to the other data points is relatively high.

It is to note, that the pointwise approach results in much more interpolation points as input data. This leads to a significantly bigger interpolation matrix A_ϕ for which the solution of (3) and (10), respectively, are much more computationally expensive. So even in the case where the pointwise interpolation results in better approximations than the time series approach it has to be assessed if the higher accuracy outweighs the signif-

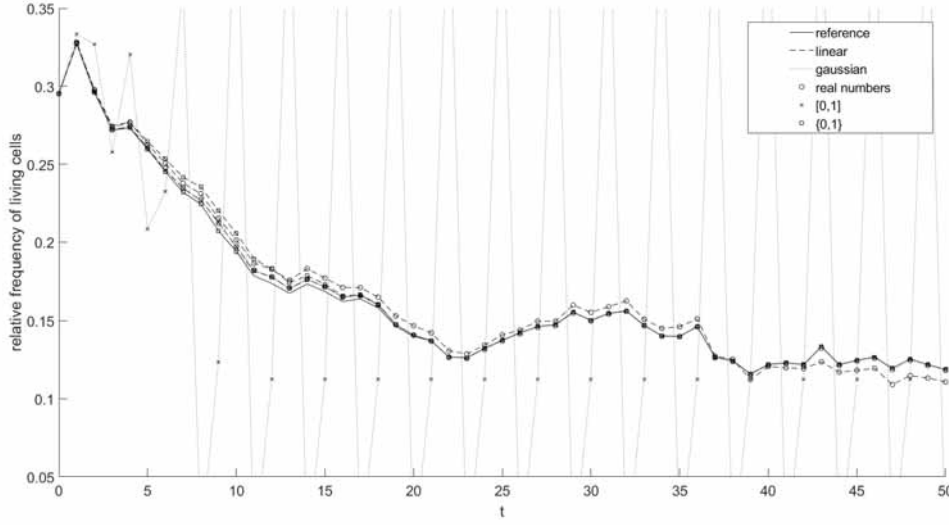


Figure 5: Iteratively interpolated time series of a single evaluation point using interpolation data with $d_{min} = 1$.

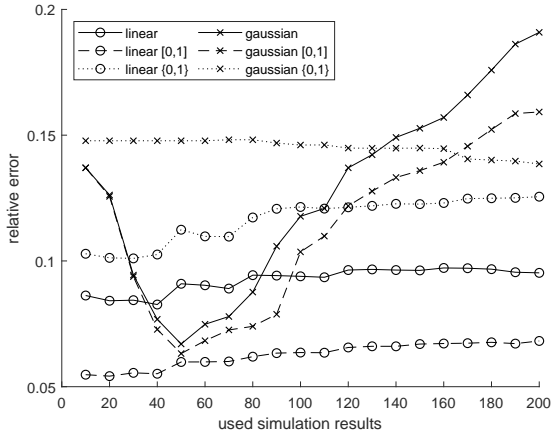


Figure 6: Error of iterative interpolation over used simulation results for $d_{min} = 1$

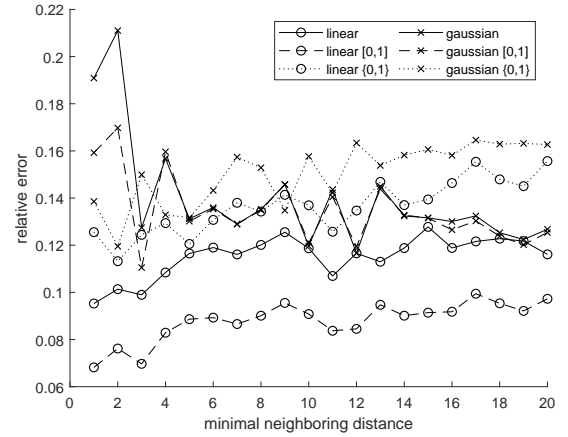


Figure 7: Error of iterative interpolation over minimal neighboring distance for 200 used interpolation points

icantly higher computational costs.

3.2 Iterative Interpolation

Another possible ansatz is to not directly interpolate the time evolution of the living cells, but to interpolate a single time step of the game of life, i.e. interpolate every single resulting state on the grid separately:

$$s_f : \{0, 1\}^{50 \times 50} \rightarrow \{0, 1\}^{50 \times 50} \quad (15)$$

The time evolution results from the repeated evaluation of the interpolation at the result of the previous interpolation and adding up the states of the resulting state matrix. Again it can be differentiated between different approaches. $x(t) \in \{0, 1\}^{50 \times 50}$ is used as the state matrix at time $t = 1, \dots, 50$

- Using the output of the interpolation directly.

$$x(t) = s_f(x(t-1)) \quad (16)$$

- Projecting the result to the interval $[0, 1]$.

$$x(t) = \max\{\min\{s_f(x(t-1)), 1\}, 0\} \quad (17)$$

- Rounding the result to 0 or 1. ($[\cdot]$ stands for the rounding operator)

$$x(t) = [\max\{\min\{s_f(x(t-1)), 1\}, 0\}] \quad (18)$$

For the iterative interpolation the use of linear functions and gaussian functions yielded the best results. For the gaussian function $\varepsilon = 5$ was chosen.

Figures 6 and 7 show the error curves for the presented approaches. The graphs show that there is no convergent behavior for the error. Even worse especially the use of more simulation results for the interpolation can lead to worse results. There are two explanations for this. First RBF-interpolation often leads to ill-conditioned problems, and second the present interpolation is very sensitive to single data points. This results, in addition to the previously mentioned chaotic behavior of the game of life, to the observed behavior of the error. Generally the magnitude of error is about 2 orders higher than the error of the direct interpolation.

Despite these discouraging results, Figure 5 shows that the gaussian results that are projected to $[0, 1]$ are oscillating and don't predict the population evolution of the game of life. The rest of the results, on the other hand, can indeed approximate the trajectory of the simulation result.

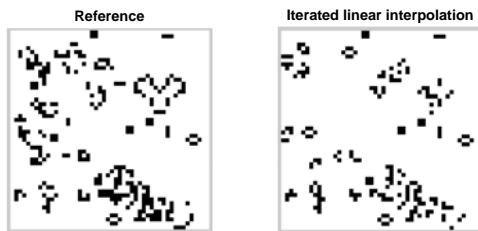


Figure 8: Spatial distribution of living cells at $t = 50$ for the reference simulation (left) and the linear interpolation with rounding (right)

Another characteristic of this approach is, that the spatial distribution of the living cells is approximately preserved during interpolation as can be seen in Figure 8. This characteristic could lead to new ways how microscopic models could be analysed.

4 Conclusion

We presented different approaches to interpolate simulation results of microscopic population models. As a test case Conway's Game of Life was used. The results show, that each of the presented approaches has its own perks and problems, but the overall conclusion is that the methods lead to promising results that should be further investigated.

Especially the iterative approach, that interpolates every state separately, seems promising as it not only approximates the time evolution of the population but can also approximate the spatial distribution within the model.

Future work will focus on error prediction and on ways to automatically adjust the shape parameter ε .

References

- [1] Benner P, Gugercin S, Willcox K. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review*. 2015; 57(4): 483–531. doi: 10.1137/130932715
- [2] Ionita AC, and Antoulas AC. Data-Driven Parametrized Model Reduction in the Loewner Framework. *SIAM Journal on Scientific Computing*. 2014; 36(3): 984–1007. doi: 10.1137/130914619
- [3] Constantine P, Wang Q. Residual Minimizing Model Interpolation for Parameterized Nonlinear Dynamical Systems. *SIAM Journal on Scientific Computing* 2012; 34(4): 2118–2144. doi: 10.1137/100816717.
- [4] Haasdonk B, Santin G. Greedy Kernel Approximation for Sparse Surrogate Modeling. In: Keiper W, Milde A, Volkwein S, editors. *Reduced-Order Modeling (ROM) for Simulation and Optimization*. Cham: Springer; 2018. p 21–46
- [5] Wendland H. *Scattered Data Approximation*. (Cambridge Monographs on Applied and Computational Mathematics). Cambridge university press, 2004. doi: 10.1017/CBO9780511617539
- [6] Fasshauer, GE. *Meshfree Approximation Methods with MATLAB*. World Scientific, 2007.
- [7] Berlekamp ER. *Winning ways for your mathematical plays*. 2nd Edition. Natick, Mass: A.K. Peters Ltd; 2001.
- [8] Bicher M, Popper N. Spatial Effects in Stochastic Microscopic Models - Case Study and Analysis. *IFAC-PapersOnLine*. 2015; 48(1): 153–158

Analyse, Simulation und optimale Steuerung eines Dengue-Fieber-Modells mit temporärer Kreuzimmunität

Mark Herath¹, Kurt Chudej^{1,2*}

¹Lehrstuhl für Wissenschaftliches Rechnen, Universität Bayreuth, 95440 Bayreuth, Germany; *kurt.chudej@uni-bayreuth.de

²Forschungszentrum Modellierung und Simulation (MODUS), Universität Bayreuth, 95440 Bayreuth, Germany

Abstract. Das Dengue-Fieber ist eine Virenerkrankung, welche durch den Stich von weiblichen, infizierten Moskitos der Spezies *Aedes* auf den Menschen übertragen wird. Diese Krankheit tritt vorwiegend in tropischen und subtropischen Gebieten auf, circa 40 % der Weltbevölkerung ist davon betroffen. Im Zuge des Klimawandels breitet sich die Asiatische Tigermücke (*Aedes albopictus*) immer weiter in Europa und insbesondere im Süden Deutschlands aus. Durch die Rückkehr von Fernreisenden wird dieser Effekt zusätzlich verstärkt. Wir untersuchen ein Kompartimentmodell mit zwei Serotypen, das eine nach einer Erstinfektion temporäre Kreuzimmunität gegen alle auftretenden Serotypen berücksichtigt. Daneben werden unterschiedliche Kontrollstrategien in das Modell integriert, welche die Anzahl der Stechmücken minimieren.

einer erstmaligen Infektion gebildeten Antikörper zusammen mit dem Virus des neuen Serotypen Antigen-Antikörper-Viren-Komplexe bilden [2]. Infolgedessen sind Zweitinfektionen mit deutlich mehr Komplikationen verbunden, symptomatisch sind Atemnot oder Magen-Darm-Blutungen. Heutzutage ist die Krankheit in 128 Ländern als endemisch eingeordnet, wobei insbesondere die Region Asien/Pazifik betroffen ist [3]. Im Jahr 2014 wurden erstmalig Funde erwachsener Tigermücken im Süden Deutschlands nachgewiesen [4, 5]. In Deutschland wurden im Jahr 2018 über 600 Infektionen mit Dengue registriert, die durch Rückreisende aus Risikogebieten der Tropen verursacht wurden [6]. Das benutzte Modell für Dengue-Fieber basiert auf Untersuchungen in [7, 8, 9]. In dieser Ausarbeitung wird ein Zwei-Serotypen-Modell für Dengue-Fieber untersucht, welches auf mehreren Überlegungen beruht [10, 11, 12, 13].

1 Einleitung

Das Dengue-Fieber ist eine infektiöse Virenerkrankung, die durch den Stich von weiblichen Moskitos der Spezies *Aedes* auf den Menschen übertragen wird [1]. Eine direkte Mensch-zu-Mensch-Infektion ist nicht möglich. Aus serologischer Sicht gehört der Dengue-Virus zur Familie des Flavivirus und es sind vier verschiedene Serotypen DENV-1, DENV-2, DENV-3 und DENV-4 bekannt. Bei einer Infektion werden nur Antikörper gegen den verursachenden Serotypen gebildet. Es entsteht dadurch keine lebenslange, sondern nur eine temporäre Kreuzimmunität gegen die anderen Dengue-Serotypen [1]. Eine Erstinfektion verursacht meist nur grippeähnlichen Symptome, wie Schüttelfrost oder Gliederschmerzen. Eine Erkrankung an einem weiteren Serotypen führt zu einer höheren Viruslast, da die nach

2 Modell mit temporärer Kreuzimmunität und Kontrollmaßnahmen

Mit 1 und 2 werden im folgenden zwei verschiedene, aber fest gewählte, der vier möglichen Dengue-Serotypen DENV-1, DENV-2, DENV-3 und DENV-4 beschrieben. Für die beiden Indizes i, j , mit $i \neq j$, gelte stets $i, j \in \{1, 2\}$.

Das SIR-Modell stellt die menschliche Population, welche in zehn Kompartimente unterteilt wird, dar. S_h entspricht den für beide Serotypen anfälligen Individuen, I_h^i den Erstinfizierten mit dem Serotyp i , R_h^i den gegen den Serotyp i resistenten Individuen, S_h^i entspricht den für Serotyp i anfälligen Individuen, I_h^{ji} den Zweitinfizierten mit Serotyp i und R_h den gegen beide Seroty-

pen resistenten Individuen. Die Moskitopopulation, untergliedert in vier Kompartimente, wird durch das ASI-Modell repräsentiert. A_m beschreibt die Vektoren in der aquatischen Phase, S_m die gegen den Virus anfällige Stechmücken und I_m^i die mit Serotyp i infizierten Moskitos, welche infizierend sind.

Für die Modellierung der Differentialgleichungen müssen noch weitere Annahmen getroffen werden [7, 9]. Beide Populationen werden als homogen angenommen, die räumlich gleichmäßig verteilt sind. Somit ist die Wahrscheinlichkeit, sich mit einem Serotypen zu infizieren, für jedes Individuum gleich groß. Bereits infizierte Individuen können nicht gleichzeitig an einem weiteren Serotypen erkranken. Des Weiteren wird vorausgesetzt, dass nur zwei der vier Dengue-Serotypen sowohl in der Bevölkerung als auch in der Vektorpopulation präsent sind. Da für die menschliche Population $N_h(t)$ keine Migration berücksichtigt wird, ist diese konstant für alle Zeiten t und die einzelnen Kompartimente sind als stetige Größen zu betrachten. Insofern ist die Größe der Bevölkerung unabhängig von t und es muss gelten:

$$N_h = S_h(t) + I_h^1(t) + I_h^2(t) + R_h^1(t) + R_h^2(t) + S_h^1(t) + S_h^2(t) + I_h^{12}(t) + I_h^{21}(t) + R_h(t) \quad \forall t$$

Durch den Proportionalitätsfaktor μ_h wird die natürliche Sterberate der Menschen beschrieben, der dazugehörige Kehrwert stellt die durchschnittliche Lebenserwartung der menschlichen Population dar. Bei einer Erkrankung mit einem Dengue-Virus wird keine erhöhte Sterblichkeitsrate angenommen. Damit die Bevölkerungszahl konstant bleibt, gibt es zum Kompartiment S_h einen konstanten Zufluss $\mu_h N_h$, der die Neugeborenen repräsentiert. Diese sind gesund, aber anfällig für eine Infektion mit einem der beiden Serotypen. Die Parameter, welche die Übertragung des Dengue-Virus beeinflussen, werden nachfolgend beschrieben. Die durchschnittliche Stechrate der Moskitos pro Tag ist definiert durch den Parameter B . Die Wahrscheinlichkeit einer Übertragung beim Stich eines Menschen, verursacht durch Aedes-Stechmücken, wird durch β_{mh} beschrieben. Bei einer Infektion mit dem Dengue-Virus wird die Genesungsrate durch den Faktor η_h definiert. Dessen Kehrwert charakterisiert in Tagen die durchschnittliche Erkrankungsdauer der Menschen. Der Kehrwert des neu hinzugefügten Parameters ζ stellt die Dauer der temporären Kreuzimmunität nach der Erkrankung mit einem Serotypen dar. Analog zum Parameter μ_h

wird die natürliche Sterberate der ausgewachsenen Vektorpopulation durch den Proportionalitätsfaktor μ_m dargestellt. Ihr Kehrwert entspricht der durchschnittlichen Lebenserwartung der Moskitos. Die Sterberate der nicht ausgewachsenen Moskitos wird durch den Parameter μ_A beschrieben. Analog zu den Neugeborenen sind diese bei Geburt gesund. η_A beschreibt in Tagen die durchschnittliche Dauer der Vektoren in der aquatischen Phase. Der Kehrwert des Faktors ist als Reifungsrate der Larven zu ausgewachsenen Moskitos zu verstehen. Es wird vorausgesetzt, dass jeder ausgewachsene Moskito im Durchschnitt täglich ϕ Eier in einen Brutplatz legt. Diese Brutplätze sind bezüglich der Anzahl der Eier beschränkt, was durch die Trägerkapazität kN_h ersichtlich wird. Der Faktor k beschreibt die durchschnittliche Anzahl an nicht-ausgewachsenen Stechmücken pro Mensch. Sind die Moskitos ausgereift, so verlassen diese die aquatische Phase und können sich nun mit einem Dengue-Virus infizieren. Analog zur aquatischen Phase ist auch die Anzahl der ausgewachsenen Moskitos beschränkt, dargestellt durch den Faktor m . Die Übertragungswahrscheinlichkeit von Mensch zu Moskitos ist durch den Parameter β_{hm} definiert. Die zuvor beschriebenen Parameter sind positiv und die Eintrittswahrscheinlichkeiten sind zusätzlich nach oben durch 1 beschränkt.

Weiterhin werden in das Modell drei Kontrollmöglichkeiten eingefügt:

Anteil an Larvizid:	$0 \leq c_A(t) \leq 1$
Anteil an Adultizid:	$0 \leq c_m(t) \leq 1$
Anteil an mech. Kontrolle:	$0 < \alpha_{\min} \leq \alpha(t) \leq 1$

Mit mechanischer Kontrolle wird die Anzahl der Brutplätze der Moskitos reduzieren, beispielsweise Wasseransammlungen in Blumenuntersetzern, Moskitonetze über Regentonnen, Reduzierung von Altreifen zur Abdeckung von Silofolien in der Landwirtschaft usw. Zudem versteht man darunter auch die persönliche Vorbeugung, wie das Tragen von langen Hosen und Ärmeln, und das Anbringen von Fliegengittern an Fenstern und Türen. Das Verteilen von Larviziden wird zur Bekämpfung der aquatischen Stechmücken eingesetzt, das Sprühen von Adultiziden zur Reduzierung der erwachsenen Moskitopopulation.

Unter Berücksichtigung der beschriebenen Sachverhalte entsteht ein Kompartiment-Modell, das in Abbildung 1 skizziert ist. Die Kompartimente der anfälligen Population sind blau, die der Infizierten rot und die der resistenten Population grün dargestellt.

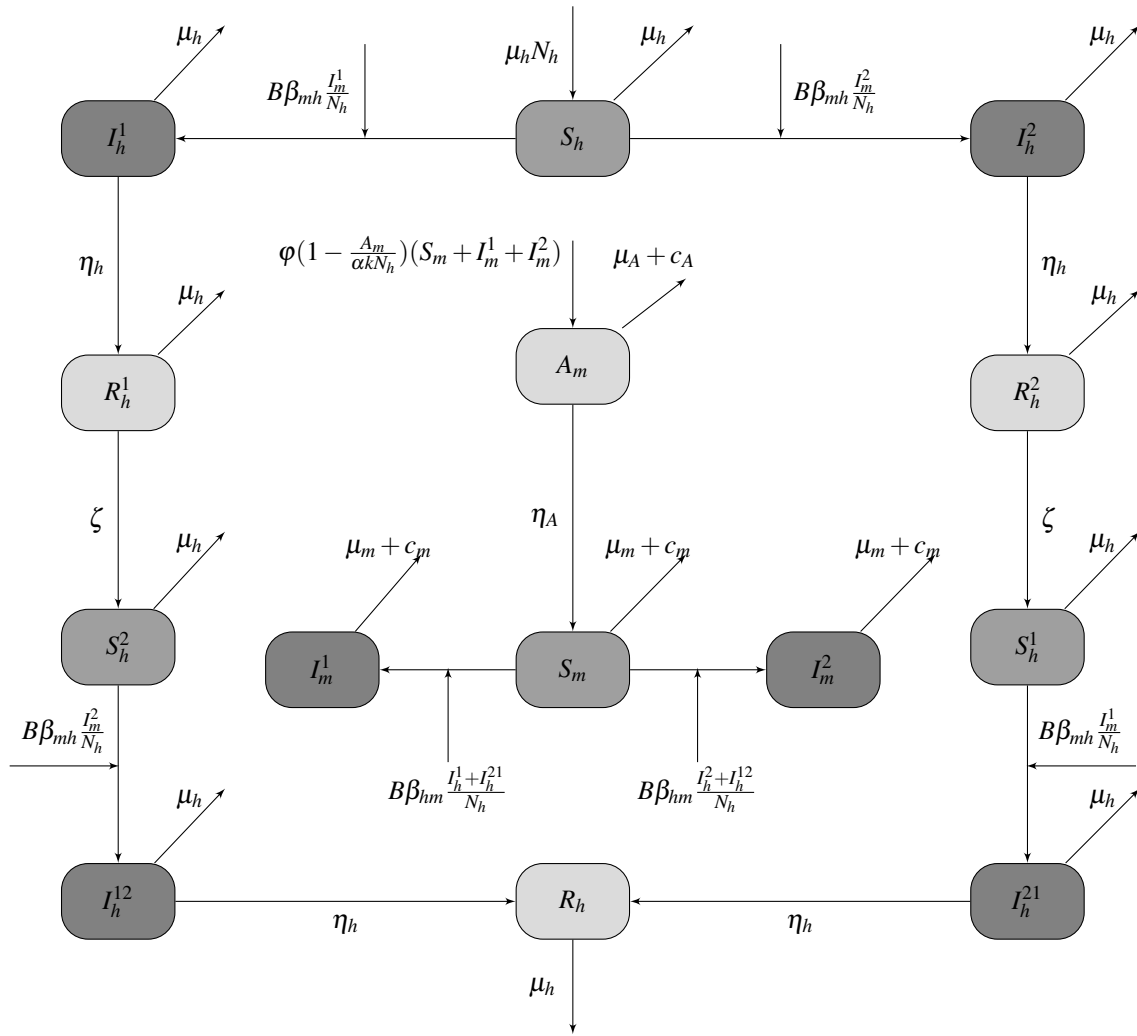


Abbildung 1: Kompartimentmodell mit temporärer Kreuzimmunität

Zusammenfassend ergibt sich ein System von nicht-linearen Differentialgleichungen. Dabei gilt für die menschliche Population

$$\begin{cases}
 \dot{S}_h = \mu_h N_h - \left(B\beta_{mh} \frac{I_m^1 + I_m^2}{N_h} + \mu_h \right) S_h \\
 \dot{I}_h^i = B\beta_{mh} \frac{I_m^i}{N_h} S_h - (\eta_h + \mu_h) I_h^i \\
 \dot{R}_h^i = \eta_h I_h^i - (\zeta + \mu_h) R_h^i \\
 \dot{S}_h^i = \zeta R_h^i - \left(B\beta_{mh} \frac{I_m^i}{N_h} + \mu_h \right) S_h^i \\
 \dot{I}_h^{ij} = B\beta_{mh} \frac{I_m^j}{N_h} S_h^i - (\eta_h + \mu_h) I_h^{ij} \\
 \dot{R}_h = \eta_h (I_h^{12} + I_h^{21}) - \mu_h R_h
 \end{cases} \quad (1)$$

und für die Moskitopopulation

$$\begin{cases}
 \dot{A}_m = \varphi \left(1 - \frac{A_m}{\alpha k N_h} \right) (S_m + I_m^1 + I_m^2) - (\eta_A + \mu_A + c_A) A_m \\
 \dot{S}_m = \eta_A A_m - \left(B\beta_{hm} \frac{I_h^1 + I_h^2 + I_h^{12} + I_h^{21}}{N_h} + \mu_m + c_m \right) S_m \\
 \dot{I}_m^i = B\beta_{hm} \frac{I_h^i + I_h^{ji}}{N_h} S_m - (\mu_m + c_m) I_m^i
 \end{cases} \quad (2)$$

mit $i, j \in \{1, 2\}$, $i \neq j$.

3 Theoretische Analyse

Die Steuerungen c_A , c_m und α werden für die theoretische Analyse als konstante Größen angenommen. Diese

wird auf der nachfolgenden Menge durchgeführt:

$$\Omega = \{(S_h, I_h^1, I_h^2, R_h^1, R_h^2, S_h^1, S_h^2, I_h^{12}, I_h^{21}, R_h \mid A_m, S_m, I_m^1, I_m^2) \in \mathbb{R}_+^{14} \mid S_h + R_h + \sum_i (I_h^i + R_h^i + S_h^i) + \sum_{i,j} I_h^{ij} \leq N_h, A_m \leq kN_h, S_m + I_m^1 + I_m^2 \leq mN_h\}$$

Die einzelnen Kompartimente, die die menschliche Population beschreiben, dürfen in der Summe die Gesamtbevölkerung nicht überschreiten. Zudem sind die Anzahl der Larven und die Summe der ausgewachsenen Moskitos durch die menschliche Bevölkerungszahl beschränkt.

Satz 1 : Für das DGL-System (1), (2) ist die Menge \mathbb{R}_+^{14} und insbesondere auch Ω positiv invariant.

Beweis: Das DGL-System lässt sich für $X \in \mathbb{R}_+^{14}$, $X = (S_h, I_h^1, I_h^2, R_h^1, R_h^2, S_h^1, S_h^2, I_h^{12}, I_h^{21}, R_h \mid A_m, S_m, I_m^1, I_m^2)$, in ein Metzler-System $\dot{X} = A(X)X + F$ umschreiben. Es gilt $F = (\mu_h N_h, 0, 0, 0, 0, 0, 0, 0, 0, 0 \mid 0, 0, 0, 0)^T$ und $A(X) = \begin{pmatrix} A_h(X) & 0 \\ 0 & A_m(X) \end{pmatrix}$, Details siehe Figur 2.

Die Blockdiagonalmatrix $A(X)$ ist eine Metzler-Matrix, da die Werte auf der Diagonalen negativ sind und die Einträge außerhalb der Diagonalen nur nicht-negativ sind. Ferner gilt $F \geq 0$. Nach den Ausführungen in [14, 15] sind \mathbb{R}_+^{14} und Ω positiv invariant. \square

Nach diesem Resultat lässt sich festhalten, dass die Lösungen für alle Startwerte aus Ω innerhalb der Menge Ω verlaufen. Dementsprechend können bei der Wahl eines biologisch sinnvollen Startwerts keine biologisch nicht-relevanten Lösungen entstehen.

Die nachfolgenden Lösungen wurden mit dem Computeralgebrasystem Maple berechnet. Zur besseren Darstellung werden folgende Hilfsgrößen eingefügt:

$$\begin{aligned} \mathcal{A}_{\text{kontroll}} &= \varphi \eta_A - (\mu_m + c_m)(\eta_A + \mu_A + c_A), \\ \mathcal{B}_{\text{kontroll}} &= \alpha k B^2 \beta_{hm} \beta_{mh} \mathcal{A}_{\text{kontroll}}, \\ \mathcal{C}_{\text{kontroll}} &= \varphi (\mu_m + c_m)^2 (\eta_h + \mu_h), \\ \mathcal{D}_{\text{kontroll}} &= B \beta_{hm} (\alpha B k \beta_{mh} \mathcal{A}_{\text{kontroll}} + \varphi \mu_h (\mu_m + c_m)). \end{aligned}$$

Satz 2 : In Ω besitzt das DGL-System (1), (2) bis zu sechs Gleichgewichtspunkte, davon vier Gleichgewichte mit aussterbenden Serotypen. Maximal zwei sind krankheitsfrei (E_1^* , E_2^*) und höchstens zwei sind endemische Randgleichgewichte (E_3^* , E_4^*). Diese nehmen folgende Werte an:

$$\begin{aligned} \bullet E_1^* &= (N_h, 0, 0, 0, 0, 0, 0, 0, 0, 0 \mid 0, 0, 0, 0) \\ \bullet E_2^* &= (N_h, 0, 0, 0, 0, 0, 0, 0, 0, 0 \mid \frac{kN_h \mathcal{A}_{\text{kontroll}}}{\varphi \eta_A}, \frac{kN_h \mathcal{A}_{\text{kontroll}}}{\varphi (\mu_m + c_m)}, 0, 0) \end{aligned}$$

$$\begin{aligned} \bullet E_3^* &= (S_h^{**}, I_h^{1**}, 0, R_h^{1**}, 0, 0, S_h^{2**}, 0, 0, 0 \mid A_m^{**}, S_m^{**}, I_m^{1**}, 0) \\ \bullet E_4^* &= (S_h^{**}, 0, I_h^{2**}, 0, R_h^{2**}, S_h^{1**}, 0, 0, 0, 0 \mid A_m^{**}, S_m^{**}, 0, I_m^{2**}), \text{ mit} \\ - S_h^{**} &= \frac{N_h (B \beta_{hm} \varphi \mu_h (\mu_m + c_m) + \mathcal{C}_{\text{kontroll}})}{\mathcal{D}_{\text{kontroll}}} \\ - I_h^{1**} &= \frac{\mu_h N_h (\mathcal{B}_{\text{kontroll}} - \mathcal{C}_{\text{kontroll}})}{(\eta_h + \mu_h) \mathcal{D}_{\text{kontroll}}} \\ - R_h^{1**} &= \frac{\eta_h \mu_h N_h (\mathcal{B}_{\text{kontroll}} - \mathcal{C}_{\text{kontroll}})}{(\zeta + \mu_h) (\eta_h + \mu_h) \mathcal{D}_{\text{kontroll}}} \\ - S_h^{i**} &= \frac{\zeta \eta_h N_h (\mathcal{B}_{\text{kontroll}} - \mathcal{C}_{\text{kontroll}})}{(\zeta + \mu_h) (\eta_h + \mu_h) \mathcal{D}_{\text{kontroll}}} \\ - A_m^{**} &= \frac{k N_h \mathcal{A}_{\text{kontroll}}}{\varphi \eta_A} \\ - S_m^{**} &= \frac{(\mu_h + \eta_h) N_h (\alpha k B \beta_{mh} \mathcal{A}_{\text{kontroll}} + \varphi \mu_h (\mu_m + c_m))}{B \beta_{mh} \varphi (B \beta_{hm} \mu_h + (\eta_h + \mu_h) (\mu_m + c_m))} \\ - I_m^{i**} &= \frac{\mu_h N_h (\mathcal{B}_{\text{kontroll}} - \mathcal{C}_{\text{kontroll}})}{B \beta_{mh} (\mathcal{C}_{\text{kontroll}} + B \beta_{hm} \varphi \mu_h (\mu_m + c_m))} \end{aligned}$$

Bemerkungen: Der Punkt E_1^* beschreibt einen trivialen Gleichgewichtspunkt, während E_2^* ein nichttriviales, krankheitsfreies Equilibrium (kurz: DFE) darstellt. In beiden Punkten gibt es keine infizierte Menschen und Moskitos sowie keine gegen mindestens einen Serotypen resistente Menschen.

Satz 3 : Die Basisreproduktionszahl \mathcal{R}_0 des DGL-Systems (1), (2) erfüllt:

$$\mathcal{R}_0 = \sqrt{\frac{\mathcal{B}_{\text{kontroll}}}{\mathcal{C}_{\text{kontroll}}}} = \sqrt{\frac{\alpha k B^2 \beta_{hm} \beta_{mh} \mathcal{A}_{\text{kontroll}}}{\varphi (\mu_m + c_m)^2 (\eta_h + \mu_h)}} \quad (3)$$

Beweis: Zur Berechnung der Basisreproduktionszahl (kurz: BRN) wird das Next-Generation-Verfahren angewandt [16, 17, 18, 19] zu finden. Mit dieser Methode nach [18] lässt sich für jeden Serotypen i , mit $i \in \{1, 2\}$, eine BRN bestimmen. In die Berechnungen fließen nur Kompartimente mit Neuinfektionen ein. Die Rate von Neuinfektion mit Serotyp i wird durch den Vektor \mathcal{F}_i dargestellt, \mathcal{V}_i umfasst die restlichen Transferterme bezüglich Serotyp i , d.h. Tode, Geburten, Krankheitsveränderungen sowie Heilungen. Die in dieser Ausarbeitung vorgenommene Zerlegung ist nach [16, 18] gewählt, aufgrund verschiedener Deutungen im Krankheitsverlauf ist diese aber nicht zwingend eindeutig. Von den zuvor beschriebenen Vektoren werden die Jacobi-Matrizen berechnet, die mit F_i beziehungsweise V_i bezeichnet werden. Diese werden am nichttrivialen, krankheitsfreien Gleichgewicht E_2^* ausgewertet. Dabei ist F_i eine nicht-negative Matrix und V_i eine invertierbare M-Matrix. Abschließend wird das Matrixprodukt $F_i V_i^{-1}$ betrachtet, welches als Next-Generation-Matrix bezeichnet wird. Dessen Spektralradius entspricht der

$$A_h(X) = \begin{pmatrix} -(B\beta_{hm} \frac{I_h^1 + I_h^2}{N_h} + \mu_h) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ B\beta_{mh} \frac{I_h^1}{N_h} & -(\eta_h + \mu_h) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ B\beta_{mh} \frac{I_h^2}{N_h} & 0 & -(\eta_h + \mu_h) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \eta_h & 0 & -(\zeta + \mu_h) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \eta_h & 0 & -(\zeta + \mu_h) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \zeta & -(B\beta_{hm} \frac{I_h^1}{N_h} + \mu_h) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \zeta & 0 & 0 & -(B\beta_{hm} \frac{I_h^2}{N_h} + \mu_h) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & B\beta_{hm} \frac{I_h^2}{N_h} & -(\eta_h + \mu_h) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & B\beta_{hm} \frac{I_h^1}{N_h} & 0 & 0 & -(\eta_h + \mu_h) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \eta_h & \eta_h & -\mu_h \end{pmatrix}$$

$$A_m(X) = \begin{pmatrix} -(\varphi \frac{S_m + I_m^1 + I_m^2}{\alpha k N_h} + \eta_A + \mu_A + c_A) & \varphi & \varphi & \varphi \\ \eta_A & -(B\beta_{hm} \frac{I_h^1 + I_h^2 + I_h^{12} + I_h^{21}}{N_h} + \mu_m + c_m) & 0 & 0 \\ 0 & B\beta_{hm} \frac{I_h^1 + I_h^{21}}{N_h} & -(\mu_m + c_m) & 0 \\ 0 & B\beta_{hm} \frac{I_h^2 + I_h^{12}}{N_h} & 0 & -(\mu_m + c_m) \end{pmatrix}$$

Abbildung 2: Teilmatrizen von $A(X)$

BRN des Serotypen i , d.h. es gilt: $\mathcal{R}_i = \rho(FV^{-1})$. Untersuchungen in [19] haben gezeigt, dass das DFE nicht eindeutig sein muss.

Für Serotyp 1 mit den Kompartimenten I_h^1 , I_h^{21} und I_m^1 lassen sich die folgenden Vektoren bilden:

$$\mathcal{F}_1 = \begin{pmatrix} B\beta_{mh} \frac{I_m^1}{N_h} S_h \\ B\beta_{mh} \frac{I_m^1}{N_h} S_h^1 \\ B\beta_{hm} \frac{I_h^1 + I_h^{21}}{N_h} S_m \end{pmatrix}, \quad \mathcal{V}_1 = \begin{pmatrix} (\eta_h + \mu_h) I_h^1 \\ (\eta_h + \mu_h) I_h^{21} \\ (\mu_m + c_m) I_m^1 \end{pmatrix}.$$

Nach Einsetzen der Werte des Gleichgewichts E_2^* sehen die jeweiligen Jacobi-Matrizen F_1 und V_1 wie folgt aus:

$$F_1 = \begin{pmatrix} 0 & 0 & B\beta_{mh} \frac{S_h^*}{N_h} \\ 0 & 0 & B\beta_{mh} \frac{S_h^{1*}}{N_h} \\ B\beta_{hm} \frac{S_m^*}{N_h} & B\beta_{hm} \frac{S_m^*}{N_h} & 0 \end{pmatrix},$$

$$V_1 = \begin{pmatrix} \eta_h + \mu_h & 0 & 0 \\ 0 & \eta_h + \mu_h & 0 \\ 0 & 0 & \mu_m + c_m \end{pmatrix}$$

Die Next-Generation-Matrix $F_1 V_1^{-1}$ nimmt somit folgende Gestalt an:

$$F_1 V_1^{-1} = \begin{pmatrix} 0 & 0 & \frac{B\beta_{mh}}{\mu_m + c_m} \\ 0 & 0 & 0 \\ \frac{\alpha k B\beta_{hm} \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)(\eta_h + \mu_h)} & \frac{\alpha k B\beta_{hm} \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)(\eta_h + \mu_h)} & 0 \end{pmatrix}$$

Die BRN des Serotyps 1 berechnet sich zu:

$$\mathcal{R}_1 = \rho(F_1 V_1^{-1}) = \sqrt{\frac{\alpha k B^2 \beta_{hm} \beta_{mh} \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)^2 (\eta_h + \mu_h)}}$$

Aufgrund der symmetrischen Betrachtungsweise ergibt die Berechnung für die BRN des Serotyps 2 das gleiche Ergebnis. Somit folgt:

$$\mathcal{R}_2 = \rho(F_2 V_2^{-1}) = \sqrt{\frac{\alpha k B^2 \beta_{hm} \beta_{mh} \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)^2 (\eta_h + \mu_h)}}$$

Es gilt $\mathcal{R}_1 = \mathcal{R}_2$ und die Behauptung in (3) folgt aus $\mathcal{R}_0 = \max\{\mathcal{R}_1, \mathcal{R}_2\}$. \square

Analog zu [20] kann die BRN wie folgt umgeschrieben werden:

$$\mathcal{R}_0^2 = \frac{\mathcal{B}_{kontroll}}{\mathcal{C}_{kontroll}} = \frac{\alpha k B^2 \beta_{hm} \beta_{mh} \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)^2 (\eta_h + \mu_h)} =$$

$$= B^2 \beta_{hm} \beta_{mh} \cdot \frac{\mathcal{E}_m}{N_h} \cdot \frac{\alpha}{\mu_m + c_m} \cdot \frac{1}{\eta_h + \mu_h},$$

wobei $\mathcal{E}_m = S_m^* = \frac{\eta_A}{\mu_m + c_m} A_m^* = \frac{k N_h \mathcal{A}_{kontroll}}{\varphi(\mu_m + c_m)}$ gewählt wurde.

Die ersten drei Terme repräsentieren die Stichintensität. Der zweite Ausdruck gibt die Anzahl der erwachsenen Stechmücken pro Mensch an. Der anschließende Ausdruck ist abhängig von den Parametern der Moskitos sowie der Kontrollmaßnahmen. Abschließend folgt ein Term in Abhängigkeit der menschlichen Parameter.

Anhand dieser Kennzahl ist zu beobachten, dass eine erhöhte Infektionsrate bei einem Stich, sowohl von

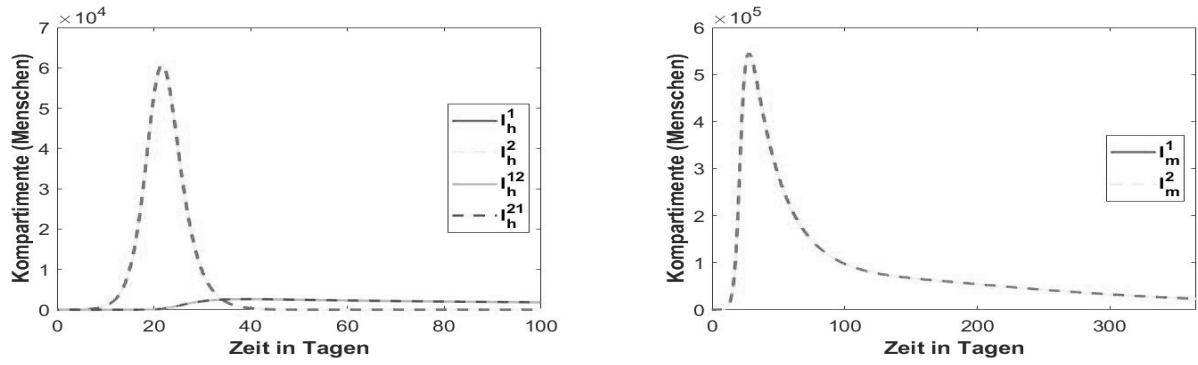


Abbildung 3: Lösungsverlauf für die infizierten Kompartimente der Menschen (links) und der Vektoren (rechts)

Kostenfunktional

$$K = \int_0^{t_f} [\gamma_D \cdot i(t)^2 + \gamma_A \cdot c_m(t)^2 + \gamma_L \cdot c_A(t)^2 + \gamma_M \cdot (1 - \alpha(t))^2] dt$$

gilt $i(t) = [I_h^1(t) + I_h^2(t) + I_h^{12}(t) + I_h^{21}(t)]/N_h$ und es beinhaltet die Gewichte γ_j . Die obere Integralgrenze t_f beträgt 365 [Tage] und die Anfangsbedingungen entsprechen denjenigen aus dem vorherigen Abschnitt. Die Kosten der verschiedenen Strategien zur Eindämmung der Moskitopopulation wurden mit der Modellierungssprache AMPL und dem Solver IPOPT berechnet [23, 24]. Um numerisch stabile Ergebnisse zu erhalten, wurden das Zielfunktional sowie das System von Differentialgleichungen entdimensionalisiert. Drei verschiedene Fälle werden betrachtet: Ein mittleres Szenario (Fall A), teure Behandlungskosten für die infizierte Population (Fall B) sowie hohe Kosten für die Eindämmung der Stechmücken (Fall C).

Wie sich zeigt, führt eine längere, temporäre Kreuzimmunität und somit ein kleinerer Wert von ζ zu geringeren Kosten. Fall B, das eine möglichst geringe Anzahl an Infizierten priorisiert, ist mit relativ hohen Kosten verbunden. Fall C, in dem die Eindämmung der Moskitos einen hohen Stellenwert besitzt, bildet den kostengünstigsten der drei Szenarien. In Fall A wird ein mittleres Szenario simuliert, die dazugehörigen Werte des Zielfunctionals liegen zwischen den beiden anderen Fällen.

In zwei Grafiken wird in Figur 4 der prozentuale Anteil an Erst- und Zweitinfektionen mit Serotyp 1 dargestellt. Aufgrund der Symmetrie im Modell gelten die nachfolgenden Zahlen für beide Serotypen und es genügt daher, nur einen Serotypen (hier: Serotyp 1) zu betrachten. Im mittleren Szenario (Fall A) treten ma-

ximal 8.04% Erstinfektionen mit Serotyp 1 auf. Ohne Kontrollstrategien lagen diese Zahlen bei 15.7% für beide Serotypen, sodass die Anzahl der Primärinfizierten durch das Einbringen von Kontrolle nahezu halbiert werden konnte. Wird wieder Fall A (mittleres Szenario) betrachtet, so können bis zu 0.72% Zweitinfektionen registriert werden. Ohne Kontrolle lag diese Zahl bei 0.68%, sodass die Anzahl der Sekundärinfizierten trotz Einsatz von Kontrollmaßnahmen einen leichten Anstieg verzeichnete.

In der Figur 5 ist der Einsatz der drei Kontrollstrategien im Betrachtungszeitraum abgebildet. Bei der mechanischen Kontrolle wird dabei die Differenz betrachtet, da für einen Wert nahe der 0 besonders viel Kontrolle ausgeübt wird. In allen drei Grafiken zeigt sich, dass im Szenario von teuren Behandlungskosten (Fall B) am meisten Kontrolle erbracht wird. Hier wird maximal bis zu 0.16% Larvizid, bis zu 12.83% Adultizid und bis zu 1.04% mechanische Kontrolle eingesetzt. Im Szenario mit teuren Kontrollmaßnahmen (Fall C) wird dementsprechend auch am wenigsten Kontrolle in einem kürzeren Zeitraum eingesetzt. Dabei werden bis zu 0.02% Larvizid, bis zu 3.74% Adultizid und bis zu 0.17% mechanische Kontrolle verwendet. Im mittleren Szenario (Fall A) liegen die Lösungskurven zwischen den zuvor beschriebenen Lösungen. Höchstens bis zu 0.05% Larvizid, bis zu 8.17% Adultizid und bis zu 0.50% mechanische Kontrolle wird eingesetzt. Nach spätestens 70 Tagen nehmen alle Lösungskurven ein relativ geringes konstantes Niveau, sodass nur noch sehr wenig Kontrolle ausgeübt wird.

	Gewichte	Kosten		
		$\zeta = \frac{1}{365}$	$\zeta = \frac{2}{365}$	$\zeta = \frac{4}{365}$
A	$\gamma_D = \gamma_A = \gamma_L = \gamma_M = 0.25$	0.119041	0.127501	0.142727
B	$\gamma_D = 0.55, \gamma_A = \gamma_L = \gamma_M = 0.15$	0.177034	0.196673	0.230429
C	$\gamma_D = 0.10, \gamma_A = \gamma_L = \gamma_M = 0.30$	0.057502	0.060873	0.066516

Tabelle 1: Gewichte und Kosten bei der Optimalen Steuerung

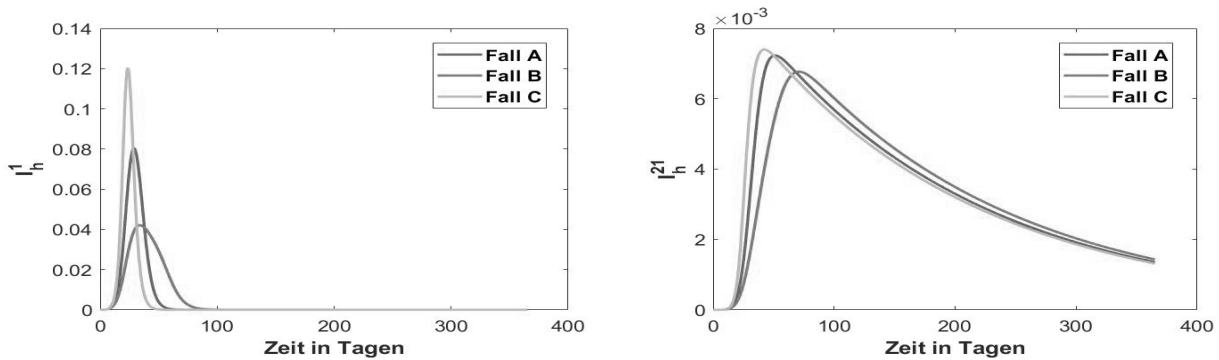


Abbildung 4: Erst- (links) und Zweitinfektionen (rechts) mit Serotyp 1

6 Fazit

Wie sich zeigt, ist das Einbringen von Kontrollstrategien eine sinnvolle Gegenmaßnahme, um die Anzahl der erstmalig infizierten Individuen zu minimieren und somit die Verbreitung des Virus einzudämmen. Als besonders effektiv stellt sich der Einsatz von Adultizid heraus, das zur Bekämpfung der erwachsenen Moskitopopulation eingesetzt wird. Allerdings erhöht sich durch diese Gegenmaßnahmen die maximale Anzahl der Zweitinfizierten, sodass in diesem Fall weitere Anpassungen der Strategien zwingend erforderlich sind. Zur Minimierung dieser Fallzahlen können auch zusätzliche geeignete Impfmaßnahmen in Betracht gezogen werden.

Literatur

- [1] WHO. Dengue and severe dengue. September 2020.
- [2] Recker M, Blyuss K, Simmons C, Tinh Tran H, Wills B, Farrar J, Gupta S. Immunological serotype interactions and their effect on the epidemiological pattern of dengue. *Proceedings. Biological sciences / The Royal Society* 05 2009; **276**:2541–2548.
- [3] Brady O, Gething P, Bhatt S, Messina J, Brownstein J, Hoen A, Moyes C, Farlow A, Scott T, Hay S. Refining the global spatial limits of dengue virus transmission by evidence-based consensus. *PLoS neglected tropical diseases* 08 2012; **6**:e1760.
- [4] Becker N, Schön S, Klein A, Ferstl I, Kizgin A, Tanich E, Kuhn C, Pluskota B, Jöst A. First mass development of *Aedes albopictus* (Diptera: Culicidae)—its surveillance and control in Germany. *Parasitology research* 2017; **116**(3):847–858.
- [5] Kampen H, Schäfer M, Scheuch D, Werner D. Further specimens of the Asian tiger mosquito *Aedes albopictus* (Diptera, Culicidae) trapped in southwest Germany. *Parasitology research* 09 2012; **112**(2):905–907.
- [6] Zur Situation bei wichtigen Infektionskrankheiten - Reiseassoziierte Krankheiten 2018. *Epidemiologisches Bulletin - Robert Koch-Institut* 2019; (1):513–524.
- [7] Rodrigues HS. Optimal Control and Numerical Optimization Applied to Epidemiological Models. PhD Thesis, University Aveiro, Portugal 2012.
- [8] Rodrigues HS, Monteiro MTT, Torres DF. Bioeconomic perspectives to an optimal control dengue model. *International Journal of Computer Mathematics* 2013; **90**(10):2126–2136.
- [9] Fischer A, Chudej K, Pesch HJ. Optimal vaccination and control strategies against dengue. *Mathematical Methods in the Applied Sciences* 2019; **42**(10):3496–3507.
- [10] Castillo-Chavez C, Hethcote HW, Andreasen V, Levin SA, Liu WM. Epidemiological models with age structure, proportionate mixing, and cross-immunity. *Journal of Mathematical Biology* 1989; **27**(3):233–258.
- [11] Feng Z, Velasco-Hernández JX. Competitive exclusion

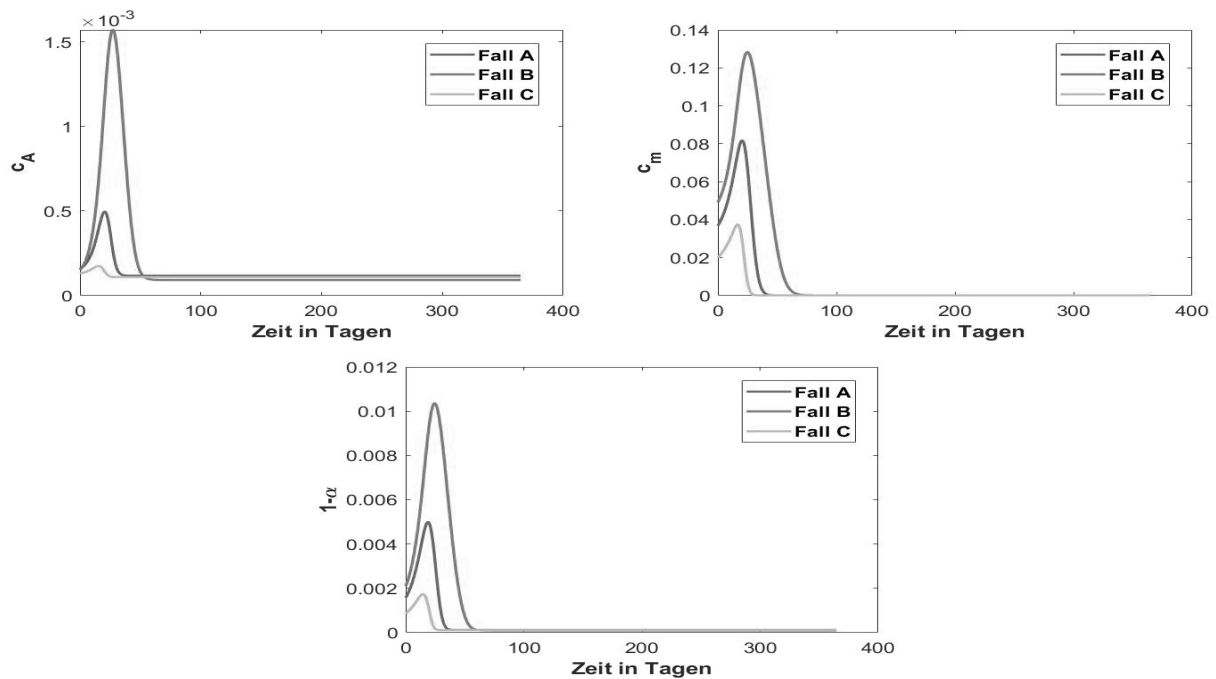


Abbildung 5: Einsatz von Larvizid (links), Adultizid (rechts) und mechanischer Kontrolle (unten)

- in a vector-host model for the dengue fever. *Journal of Mathematical Biology* 1997; **35**(5):523–544.
- [12] Esteva L, Vargas C. Coexistence of different serotypes of dengue virus. *Journal of Mathematical Biology* 2003; **46**(1):31–47.
- [13] Chudej K, Fischer A. Optimal Vaccination Strategies for a new Dengue Model with two Serotypes. *IFAC-PapersOnLine* 2018; **51**(2):13–18.
- [14] Mitkowski W. Dynamical properties of Metzler systems. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 2008; **56**(4):309–312.
- [15] Abate A, Tiwari A, Sastry S. Box invariance in biologically-inspired dynamical systems. *Automatica* 2009; **45**(7):1601–1610.
- [16] Van den Driessche P, Watmough J. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences* 2002; **180**(1-2):29–48.
- [17] Heffernan J, Smith R, Wahl L. Perspectives on the basic reproductive ratio. *Journal of the Royal Society* 10 2005; **2**:281–93.
- [18] van den Driessche P, Watmough J. *Further Notes on the Basic Reproduction Number*. Springer: Berlin, 2008; 159–178.
- [19] Van den Driessche P. Reproduction numbers of infectious disease models. *Infectious Disease Modelling* 2017; **2**(3):288–303.
- [20] Albrecht G, Fischer A, Chudej K. Analyse, Simulation und Optimale Steuerung eines mathematischen Dengue-Fieber Modells mit Impfung. *Tagungsband Workshop Heilbronn 2018 ASIM/GI Fachgruppen, ARGESIM Report*, vol. 54. ARGESIM Verlag: Wien, Österreich, 2018; 223–229.
- [21] Mordecai EA, Cohen JM, Evans MV, Gudapati P, Johnson LR, Lippi CA, Miazgowicz K, Murdock CC, Rohr JR, Ryan SJ, *et al.*. Detecting the impact of temperature on transmission of Zika, dengue, and chikungunya using mechanistic models. *PLoS Neglected Tropical Diseases* 2017; **11**(4):e0005 568.
- [22] Herath M, Albrecht G, Chudej K. Ein asymmetrisches zwei Serotyp Dengue Fieber Modell mit Kontrollmaßnahmen. *Simulation in Umwelt- und Geowissenschaften: Workshop 2020, ASIM Mitteilung*, vol. 173. Shaker Verlag: Düren, 2020; 191–202.
- [23] Fourer R, Gay D, Kernighan B. *AMPL: A Modeling Language for Mathematical Programming*, vol. 36. Duxbury Press: Pacific Grove, 2002.
- [24] Wächter A, Biegler L. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming* 03 2006; **106**:25–57.

Entwicklung und Analyse einer stufigen SVIR-Baummodellierung

Amelie Flothow¹, Kurt Chudej^{1,2*}

¹Lehrstuhl für Wissenschaftliches Rechnen, Universität Bayreuth, 95440 Bayreuth, Germany; *kurt.chudej@uni-bayreuth.de

²Forschungszentrum Modellierung und Simulation (MODUS), Universität Bayreuth, 95440 Bayreuth, Germany

Abstract. Eine neue Baummodellierung für imperfekte zeitlich nachlassende Impfungen bei SIR-Krankheitsmodellen wird entwickelt. Theoretische und numerische Ergebnisse werden präsentiert.

1 Einleitung

Mathematische Krankheitsmodelle mit Impfungen werden seit langem untersucht [1]. Bei mathematischen Modellen von imperfekten und zeitlich nachlassenden Massenimpfungen zur Bekämpfung von durch Vektoren verursachten Krankheiten gibt es mathematische Probleme, die Gleichgewichte symbolisch für beliebige Parameterwerte zu berechnen [2].

Dies motiviert die Untersuchung von neuartigen Baummodellierungen, zunächst für mathematische SIR-Krankheitsmodelle mit imperfekten und zeitlich nachlassenden Massenimpfungen für Mensch-zu-Mensch übertragene Krankheiten. Dies geschieht um die Vor- und Nachteile der neuen Modellierung besser herauszuarbeiten.

2 Standard-SVIR-Modellierung

Die konstante Gesamtbevölkerung N wird aufgeteilt in die Gesunden, aber für die Krankheit empfänglichen Menschen S , die infizierten und infizierenden Menschen I , die wieder gesunden und für die Krankheit immunen Menschen R sowie die geimpften (und dadurch gegen die Krankheit immunen) Menschen V . Mit μ^{-1} wird die Sterbe- und Geburtenrate bezeichnet. Der Krankheitsübertragungsparameter ist mit β bezeichnet. Mit η^{-1} wird die (mittlere) Dauer der Infektiosität bezeichnet. Die Größe ψ gibt die Impfrate der empfänglichen Menschen S an (Massenimpfung). Mit θ^{-1} wird die (mittlere) Wirkungsdauer der Impfung bezeichnet. Danach fällt man aus dem Kompartiment V wieder ins

Kompartiment S zurück. Obwohl man sich im Kompartiment der geimpften Menschen V befindet, kann man mit der um σ verringerten Wahrscheinlichkeit gegenüber den Menschen in S trotzdem erkranken.

Das SVIR-Modell in Abbildung 1 wird durch das nichtlineare Differentialgleichungssystem mathematisch dargestellt

$$\begin{aligned}\dot{S} &= N\mu + \theta V - (\beta \frac{I}{N} + \psi + \mu)S \\ \dot{V} &= \psi S - (\theta + \sigma \beta \frac{I}{N} + \mu)V \\ \dot{I} &= \beta \frac{I}{N} S + \sigma \beta \frac{I}{N} V - (\eta + \mu)I \\ \dot{R} &= \eta I - \mu R.\end{aligned}\tag{1}$$

Sei die Menge

$$\Omega = \{(S, V, I, R) \in \mathbb{R}_{\geq 0}^4 | S + V + I + R \leq N\}.\tag{2}$$

definiert.

Satz 1. Die konvexe Menge Ω und $\mathbb{R}_{\geq 0}^4$ sind positiv invariant bzgl. des Differentialgleichungssystems (1). Zudem besitzt das Differentialgleichungssystem (1) die Invariante

$$N = S(t) + V(t) + I(t) + R(t) \quad \forall t.\tag{3}$$

Um die folgenden Rechnungen einfacher darzustellen, benutzen wir die Entdimensionalisierung

$$s = S/N, \quad i = I/N, \quad v = V/N, \quad r = R/N\tag{4}$$

die zum entdimensionalisierten Differentialgleichungssystem

$$\begin{aligned}\frac{d}{dt}s &= \mu + \theta v - (\beta i + \psi + \mu)s \\ \frac{d}{dt}v &= \psi s - (\theta + \sigma \beta i + \mu)v \\ \frac{d}{dt}i &= \beta i s + \sigma \beta i v - (\eta + \mu)i\end{aligned}\tag{5}$$

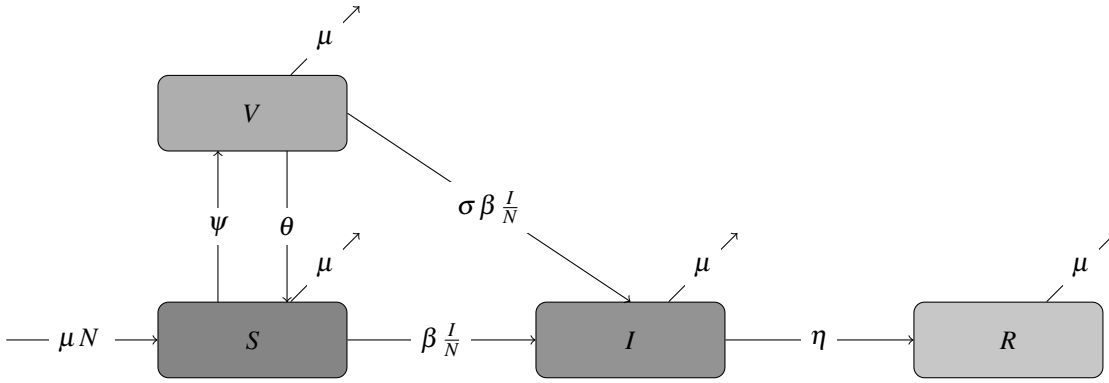


Abbildung 1: Graphische Darstellung der Standard-SVIR-Modellierung.

$$\frac{d}{dt}r = \eta i - \mu r,$$

mit der Invariante

$$1 = s(t) + v(t) + i(t) + r(t) \quad \forall t. \quad (6)$$

führt.

Uns interessieren nun die Gleichgewichtspunkte des Differentialgleichungssystems (5). Aufgrund der Invariante (6) reicht es für die Gleichgewichtslösungen, dass in drei der vier Kompartimente keine Änderung zu verzeichnen ist. Bei der Berechnung der Gleichgewichte treten erste Schwierigkeiten auf. So lässt sich mit Maple zunächst nur das krankheitsfreie Gleichgewicht (DFE) berechnen. Durch das Hinzufügen der Bedingung, dass keines der Kompartimente negativ sein darf, werden auch zwei krankheitsbehaftete Gleichgewichte berechnet. Diese sind von der Form

$$e^* = (s^*, v^*, i^*). \quad (7)$$

Das krankheitsfreie Gleichgewicht (DFE) lautet

$$e_1^* = \left(\frac{\theta + \mu}{\mu + \psi + \theta}, \frac{\psi}{\mu + \psi + \theta}, 0 \right). \quad (8)$$

Die krankheitsbehafteten Gleichgewichte sind nur definiert, sofern die Kompartimente nicht negativ sind. Die Bedingungen dazu sind schreibintensiv und werden nicht explizit ausformuliert. Zur Darstellung der krankheitsbehafteten Gleichgewichte werden einige Abkürzungen eingeführt, siehe Abbildung 2. Falls durch die gewählten Parameter die Existenz der beiden Gleichge-

wichte oder eines Gleichgewichts gegeben ist, gilt

$$e_2^* = \left(\frac{1}{2} \frac{\mathcal{C} - \mathcal{B}}{\mu(\sigma - 1)\beta}, \frac{1}{2} \frac{\mathcal{D} + \mathcal{B}}{\mu\sigma(\sigma - 1)\beta}, \frac{-(\psi\sigma + \mu + \theta)\mathcal{B} + \mathcal{E}}{\beta\sigma(\mathcal{D} + \mathcal{B})} \right), \quad (13)$$

$$e_3^* = \left(\frac{1}{2} \frac{\mathcal{C} + \mathcal{B}}{\mu(\sigma - 1)\beta}, \frac{1}{2} \frac{\mathcal{D} - \mathcal{B}}{\mu\sigma(\sigma - 1)\beta}, \frac{(\psi\sigma + \mu + \theta)\mathcal{B} + \mathcal{E}}{\beta\sigma(\mathcal{D} - \mathcal{B})} \right). \quad (14)$$

Zur Berechnung von \mathcal{R}_0 wird der Ansatz mit der Next-Generation Matrix angewendet [3]. Dazu wird das infizierende Kompartiment i der SVIR-Modellierung in Formel (5) selektiert und in die passende Form gebracht.

$$\mathcal{F} = (\beta s i + \sigma \beta v i), \quad \mathcal{V} = ((\eta + \mu) i). \quad (15)$$

Dann werden die Jacobimatrizen berechnet.

$$F = (\beta s^* + \sigma \beta v^*), \quad V = (\eta + \mu). \quad (16)$$

Mit der Inversen von V ,

$$V^{-1} = \left(\frac{1}{\eta + \mu} \right) \quad (17)$$

ergibt sich

$$K = FV^{-1} = \frac{\beta s^* + \sigma \beta v^*}{\eta + \mu}. \quad (18)$$

Einsetzen des krankheitsfreien Gleichgewichts (DFE) ergibt die Basisreproduktionszahl

$$\mathcal{R}_0 = \rho(K) = \frac{\beta s^* + \sigma \beta v^*}{\eta + \mu} = \frac{1}{\eta + \mu} \frac{\beta(\theta + \mu + \sigma \psi)}{\theta + \mu + \psi}. \quad (19)$$

Gäbe es keinen Impfstoff, wäre $\psi = 0$ und $\theta = 0$. Dann geht die Formel (19) der Basisreproduktionszahl in die bekannte Formel des SIR-Modells über [1].

$$\mathcal{B} := \sqrt{\begin{aligned} &(\sigma-1)^2\mu^4 + 2((\eta+\psi-\beta)\sigma - \eta - \theta)(\sigma-1)\mu^3 \\ &+ ((\eta^2 + (4\psi-2\beta)\eta + (\psi+\beta)^2)\sigma^2 + (-2\eta^2 + (-4\psi+2\beta-4\theta)\eta \\ &\quad + 2\theta(\psi+\beta))\sigma + \eta^2 + 4\eta\theta + \theta^2)\mu^2 + 2\eta(\psi(\eta+\psi+\beta)\sigma^2 \\ &\quad + ((-\psi-\theta)\eta + 2\theta(\psi+\beta/2))\sigma + \theta(\eta+\theta))\mu + \eta^2(\psi\sigma + \theta)^2 \end{aligned}} \quad (9)$$

$$\mathcal{C} := ((\mu+\psi)\sigma - \mu + \theta)\eta + \mu(\mu+\psi+\beta)\sigma - \mu^2 + \mu\theta \quad (10)$$

$$\mathcal{D} := ((\mu-\psi)\sigma - \mu - \theta)\eta + \mu(\mu-\psi-\beta)\sigma - \mu^2 - \mu\theta \quad (11)$$

$$\mathcal{E} := (-\sigma+1)\mu^3 + (\sigma^2\psi + (-\eta+\beta-\theta)\sigma + \eta+2\theta)\mu^2 + (\psi(\eta+\psi+\beta)\sigma^2 - \theta(\eta-2\psi-\beta)\sigma + 2\eta\theta + \theta^2)\mu + \eta(\psi\sigma + \theta)^2 \quad (12)$$

Abbildung 2: Formeln

3 Neue Baummodellierung

Die neuartige Baummodellierung bricht den Zyklus im Standard-SVIR-Impfmodell auf. Lässt die Impfwirkung nach einiger Zeit nach, da $\theta \neq 0$ ist, so wird in der üblichen SVIR-Modellierung ein Mensch wieder dem ursprünglichen Kompartiment S zugeordnet. In der neu strukturierten stufigen Baummodellierung wird er dem Kompartiment S_2 der nächsthöheren Stufe zugeordnet. Das ursprüngliche Kompartiment wird durch S_1 gekennzeichnet. Auch der Transfer eines Menschen, der trotz einer aktiven Impfung infiziert wird, da $\sigma \neq 0$ gilt, transferiert nicht in das Kompartiment I_1 , sondern wird einem neuen Kompartiment \tilde{I}_1 zugeordnet. Nach einer infektiösen Phase der Dauer η^{-1} gehen Menschen in das, ebenfalls neue, Kompartiment \tilde{R}_1 über. Weitere Stufen werden analog hinzugefügt. In Abbildung 3 ist eine 2-stufige SVIR-Modellierung dargestellt. Durch die gestrichelte Linie ist die erste Stufe von der zweiten Stufe unterteilt. Die Kompartimente erhalten die Stufenzahl als Index. Für die 2-stufige SVIR-Baummodellierung entsteht das folgende Differentialgleichungssystem:

Erste Stufe:

$$\begin{aligned} \dot{S}_1 &= \mu N - (\beta \frac{f(I)}{N} + \psi + \mu) S_1 \\ \dot{V}_1 &= \psi S_1 - (\sigma \beta \frac{f(I)}{N} + \theta + \mu) V_1 \\ \dot{I}_1 &= \beta \frac{f(I)}{N} S_1 - (\eta + \mu) I_1 \\ \dot{R}_1 &= \eta I_1 - \mu R_1 \end{aligned} \quad (20)$$

$$\begin{aligned} \dot{\tilde{I}}_1 &= \sigma \beta \frac{f(I)}{N} V_1 - (\eta + \mu) \tilde{I}_1 \\ \dot{\tilde{R}}_1 &= \eta \tilde{I}_1 - \mu \tilde{R}_1 \end{aligned}$$

Zweite Stufe:

$$\begin{aligned} \dot{S}_2 &= \theta V_1 - (\beta \frac{f(I)}{N} + \psi + \mu) S_2 \\ \dot{V}_2 &= \psi S_2 - (\sigma \beta \frac{f(I)}{N} + \theta + \mu) V_2 \\ \dot{I}_2 &= \beta \frac{f(I)}{N} S_2 - (\eta + \mu) I_2 \\ \dot{R}_2 &= \eta I_2 - \mu R_2 \\ \dot{\tilde{I}}_2 &= \sigma \beta \frac{f(I)}{N} V_2 - (\eta + \mu) \tilde{I}_2 \\ \dot{\tilde{R}}_2 &= \eta \tilde{I}_2 - \mu \tilde{R}_2 \end{aligned} \quad (21)$$

Aufgrund der Umstrukturierung der Modellierung zu einer Modellierung mit einer gerichteten Baumstruktur gibt es nicht mehr „das eine“ Kompartiment der infektiösen Menschen I bzw. i . Durch die Funktion $f(I)$ werden im folgenden *verschiedene* Varianten definiert, wie infektiöse Menschen bei der Übertragung des Erregers (näherungsweise) berücksichtigt werden. Die vier betrachteten Fälle sind für n -stufige Baummodellierungen definiert durch

$$f(I) = \begin{cases} I_1 & \text{Fall 1,} \\ I_1 + \tilde{I}_1 & \text{Fall 2,} \\ \sum_{j=1}^n I_j & \text{Fall 3,} \\ \sum_{j=1}^n I_j + \tilde{I}_j & \text{Fall 4.} \end{cases} \quad (22)$$

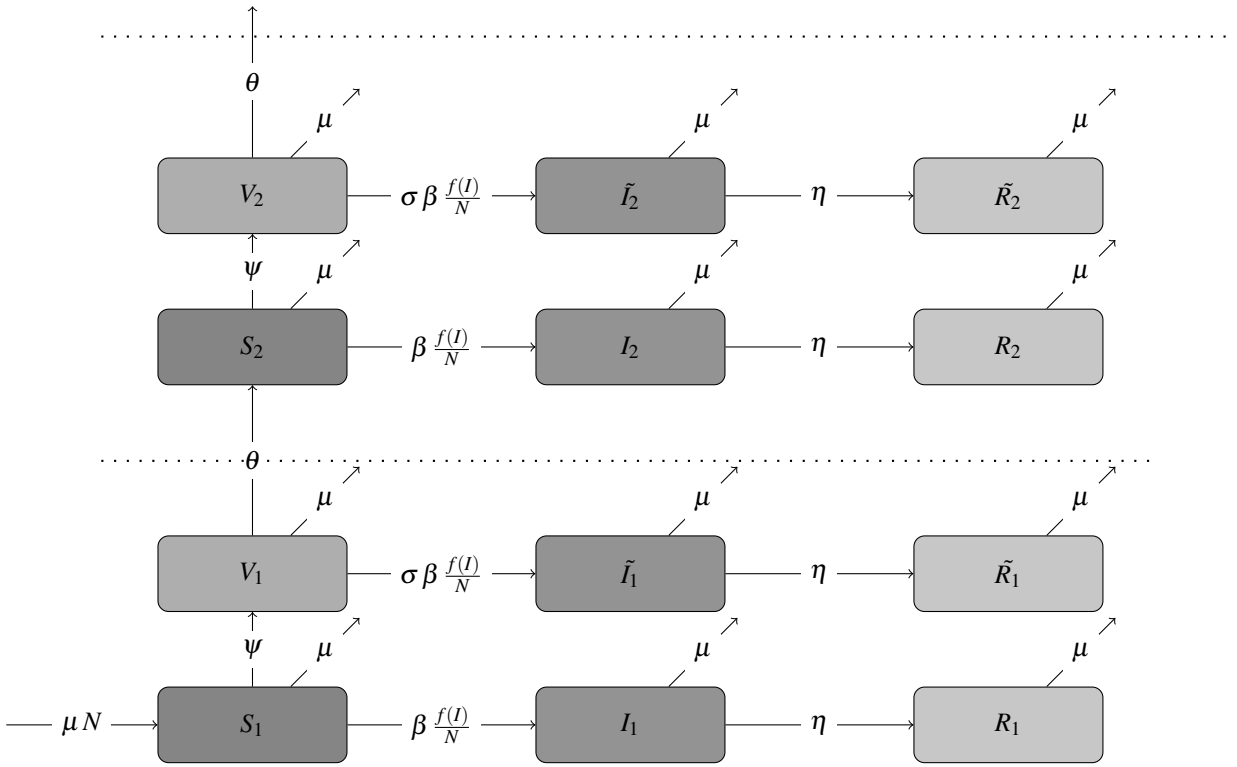


Abbildung 3: Graphische Darstellung der 2-stufigen SVIR-Baummodellierung.

In den verschiedenen Versionen werden unterschiedliche infizierende Kompartimente berücksichtigt. So enthält Fall 1 nur das Kompartiment I_1 . Ist ein ausreichend großer Anteil der Menschen (aus I im Standard-SVIR-Modell) in I_1 , könnte es ausreichen dieses Kompartiment für die Beschreibung des Übergangs in infektiöse Kompartimente zu verwenden. In Fall 2 werden die beiden infektiösen Kompartimente $I_1 + \tilde{I}_1$ der ersten Stufe verwendet und in Fall 3 werden die infektiösen Kompartimente I_j aller Stufen $j \in \{1, \dots, n\}$ verwendet. Der Fall 4 berücksichtigt alle infektiösen Kompartimente der Modellierung. Es ist zu erwarten, dass mit $f(I)$ im Fall 4 die besten Annäherungsergebnisse an die Kompartimentsgrößen im zeitlichen Verlauf resultieren. Die verschiedenen Möglichkeiten der Wahl von $f(I)$ und verschieden-stufige SVIR-Baummodellierungen werden verglichen.

Auch hier werden die Rechnungen an einem analog entdimensionalisierten Differentialgleichungssystem vorgenommen. Die Funktion $f(i)$ sei analog zu $f(I)$ definiert, in dem I_k, \dots jeweils durch i_k, \dots ersetzt wird.

Erste Stufe:

$$\begin{aligned}
 \frac{d}{dt}s_1 &= \mu - (\beta f(i) + \psi + \mu)s_1 \\
 \frac{d}{dt}v_1 &= \psi s_1 - (\sigma \beta f(i) + \theta + \mu)v_1 \\
 \frac{d}{dt}i_1 &= \beta f(i)s_1 - (\eta + \mu)i_1 \\
 \frac{d}{dt}r_1 &= \eta i_1 - \mu r_1 \\
 \frac{d}{dt}\tilde{i}_1 &= \sigma \beta f(i)v_1 - (\eta + \mu)\tilde{i}_1 \\
 \frac{d}{dt}\tilde{r}_1 &= \eta \tilde{i}_1 - \mu \tilde{r}_1
 \end{aligned} \tag{23}$$

Zweite Stufe:

$$\begin{aligned}
 \frac{d}{dt}s_2 &= \theta v_1 - (\beta f(i) + \psi + \mu)s_2 \\
 \frac{d}{dt}v_2 &= \psi s_2 - (\sigma \beta f(i) + \theta + \mu)v_2 \\
 \frac{d}{dt}i_2 &= \beta f(i)s_2 - (\eta + \mu)i_2 \\
 \frac{d}{dt}r_2 &= \eta i_2 - \mu r_2
 \end{aligned} \tag{24}$$

$$\begin{aligned}\frac{d}{dt}\tilde{i}_2 &= \sigma\beta f(i)v_2 - (\eta + \mu)\tilde{i}_2 \\ \frac{d}{dt}\tilde{r}_2 &= \eta\tilde{i}_2 - \mu\tilde{r}_2\end{aligned}$$

Zunächst werden einige theoretischen Analysen an der n -stufigen SVIR-Baummodellierung durchgeführt. Sei

$$\Omega = \{(S_1, V_1, I_1, R_1, \tilde{I}_1, \tilde{R}_1, \dots, \quad (26)$$

$$S_n, V_n, I_n, R_n, \tilde{I}_n, \tilde{R}_n) \in \mathbb{R}_+^{6n} \mid \quad (27)$$

$$\sum_{j=1}^n (S_j + V_j + I_j + R_j + \tilde{I}_j + \tilde{R}_j) \leq N\}.$$

Satz 2. Die konvexe Menge $\mathbb{R}_{\geq 0}^{6n}$ und Ω sind für das Differentialgleichungssystem der n -stufigen SVIR-Baummodellierung (20, 21) positiv invariant.

Beweis. Das zugehörige entdimensionalisierte n -stufige SVIR-System (23, 24) kann in die Form $\frac{dx}{dt} = A(x)x + b$ gebracht werden mit

$$x = (x_1, \dots, x_n)^T \text{ mit } x_j = (s_j, v_j, i_j, r_j, \tilde{i}_j, \tilde{r}_j) \quad (28)$$

für $j = 1, \dots, n$.

Für eine übersichtlichere Darstellung wird die Übertragungsfunktionen durch $g(i) := \beta f(i)$ abgekürzt. Für jede der n Stufen gilt, siehe Abbildung 4.

Bei der Definition der additiven Faktoren wird zwischen Stufe eins und den restlichen Stufen unterschieden. Es gilt

$$b_1 = (\mu, 0, 0, 0, 0, 0), \quad (29)$$

$$b_j = (0, 0, 0, 0, 0, 0) \quad \forall j \geq 2. \quad (30)$$

Die Matrix $A(x)$ für die n -stufige SVIR-Modellierung ist zusammenfassend definiert durch

$$\begin{aligned}A(x) &= \text{diag}(A_j(x_j)) \text{ für } j = 1, \dots, n, \\ \text{mit } a_{(6(j-1)+1, 6(j-2)+2)} &= \theta \quad \forall j \geq 2 \quad (31) \\ \text{die restlichen Einträge sind null.}\end{aligned}$$

An der Struktur der Matrizen $A_j(x_j)$ ist erkennbar, dass diese wegen $x \in \mathbb{R}_+^{6n}$ Metzler-Matrizen sind. Durch die vorliegende Verknüpfung der Matrizen ist auch $A(x)$ eine Metzler-Matrix. Auch die Bedingung $b \geq 0$ ist erfüllt. Somit sind die Bedingungen von [4, 5] erfüllt. \square

Unter Einführung der vordefinierten Größen

$$\begin{aligned}\mathcal{T} &:= \mu\beta - \mu\eta - \psi\eta - \mu^2 - \psi\mu \\ \mathcal{S} &:= (\beta\mu\sigma - \eta\mu\sigma - \eta\psi\sigma - \\ &\quad - \mu^2\sigma - \mu\psi\sigma + \eta\mu + \mu^2)\end{aligned}$$

und der definierenden Sequenzen

$$\begin{aligned}\bar{e}_1 &= \frac{\mu}{\mu+\psi}, \frac{\mu\psi}{(\mu+\theta)(\mu+\psi)}, 0, 0 \\ \hat{e}_1 &= \frac{\psi\theta\mu}{(\mu+\psi)^2(\mu+\theta)}, \frac{\mu\psi^2\theta}{(\mu+\theta)^2(\mu+\psi)^2}, 0, 0 \\ \bar{e}_2 &= \frac{\mu+\eta}{\beta}, \frac{\psi(\eta+\mu)^2}{\beta(\mathcal{S}+\theta(\eta+\mu))}, \frac{\mathcal{T}}{(\mu+\eta)\beta}, \frac{\sigma\psi\mathcal{T}}{\beta(\mathcal{S}+\theta(\eta+\mu))} \\ \hat{e}_2 &= \frac{\theta\psi(\mu+\eta)^3}{\mu\beta^2(\mathcal{S}+\theta(\mu+\eta))}, \frac{\psi^2\theta(\mu+\eta)^4}{\mu\beta^2(\mathcal{S}+\theta(\mu+\eta))^2}, \\ &\quad \frac{\mathcal{T}\theta\psi(\mu+\eta)}{\mu\beta^2(\mathcal{S}+\theta(\mu+\eta))}, \frac{\mathcal{T}\theta\psi^2(\mu+\eta)^2\sigma}{\mu\beta^2(\mathcal{S}+\theta(\mu+\eta))^2}\end{aligned}$$

lassen sich die Gleichgewichte der 1- bis 4-stufigen SVIR-Baummodellierung übersichtlich darstellen. Sie haben die Form $e^* = (s_1, v_1, i_1, \tilde{i}_1; s_2, v_2, i_2, \tilde{i}_2)$. In der Tabelle wird angegeben, ob die restlichen Einträge null sind (r.e.n.) oder falls ein Gleichgewicht mit *Maple* nicht berechenbar ist (m.M.n.b.).

In Tabelle 1 ist zu erkennen, dass mit *Maple* in Fall 1 und Fall 2 krankheitsfreie Gleichgewichte berechnet werden. Die krankheitsfreien Gleichgewichte der n -stufigen SVIR-Modellierungen stimmen in Fall 1 und Fall 2 und in den Kompartimenten der ersten Stufe im Gleichgewicht überein. Die Kompartimente der zweiten Stufe der krankheitsfreien Gleichgewichte stimmen auch in den mehrstufigen Modellierungen überein. Endemische Gleichgewichte werden nur in Fall 1 explizit berechnet. Auch hier stimmen die Kompartimente der erste Stufe aller stufigen SVIR-Modellierungen überein und die Kompartimente der zweiten Stufe im Gleichgewicht stimmen in den mehrstufigen Modellierungen überein.

Um Auskunft über eine Verbreitung oder eine Eindämmung der Krankheit zu erhalten, wird die Basisreproduktionszahl \mathcal{R}_0 errechnet. Falls $\mathcal{R}_0 < 1$ ist, so ist der krankheitsfreie Gleichgewichtspunkte (DFE) lokal asymptotisch stabil, für $\mathcal{R}_0 > 1$ ist er instabil [3]. Da für Fall 3 und Fall 4 von $f(I)$ die krankheitsfreien Gleichgewichte nicht explizit angegeben sind, wird \mathcal{R}_0 für die 2-stufige Modellierung für alle vier Fälle von $f(I)$ durch ein krankheitsfreies Gleichgewicht der Form $e^* = (s_1^*, v_1^*, i_1^*, \tilde{i}_1^*, s_2^*, v_2^*, i_2^*, \tilde{i}_2^*)$ angegeben. Die Berechnung von \mathcal{R}_0 folgt dem Vorgehen in [3].

Satz 3. Sofern ein krankheitsfreies Gleichgewicht der

$$A_j(\vec{x}_j) = \begin{pmatrix} -(g(i) + \psi + \mu) & 0 & 0 & 0 & 0 & 0 \\ \psi & -(\sigma g(i) + \theta + \mu) & 0 & 0 & 0 & 0 \\ g(i) & 0 & -(\eta + \mu) & 0 & 0 & 0 \\ 0 & 0 & \eta & -\mu & 0 & 0 \\ 0 & \sigma g(i) & 0 & 0 & -(\eta + \mu) & 0 \\ 0 & 0 & 0 & 0 & \eta & -\mu \end{pmatrix} \quad (25)$$

Abbildung 4: Blockmatrix

Fall	1 stufig	2 stufig	3 stufig	4 stufig
1	(\bar{e}_1) (\bar{e}_2)	$(\bar{e}_1; \hat{e}_1)$ $(\bar{e}_2; \hat{e}_2)$	$(\bar{e}_1; \hat{e}_1; r.E.n.)$ $(\bar{e}_2; \hat{e}_2; r.E.n.)$ $e_3^*, e_4^* \text{ m.M.n.b.}$	$(\bar{e}_1; \hat{e}_1; r.E.n.)$ $(\bar{e}_2; \hat{e}_2; r.E.n.)$ $e_3^* - e_8^* \text{ m.M.n.b.}$
2	(\bar{e}_1) $e_2^* \text{ m.M.n.b.}$	$(\bar{e}_1; \hat{e}_1)$ $e_2^* \text{ m.M.n.b.}$	$(\bar{e}_1; \hat{e}_1; r.E.n.)$ $e_2^* - e_4^* \text{ m.M.n.b.}$	Ausgabelänge
3	analog Fall 1	Ausgabelänge	Ausgabelänge	Ausgabelänge
4	analog Fall 2	Dauer Rechnung	Dauer Rechnung	Dauer Rechnung

Tabelle 1: Symbolische Gleichgewichte stufiger SVIR-Baummodellierungen.

Form $e^* = (s_1^*, v_1^*, i_1^*, \tilde{i}_1^*, s_2^*, v_2^*, i_2^*, \tilde{i}_2^*)$ für das 2-stufige SVIR-Baummodell für $f(i)$ in Gleichung (23, 24) existiert, ist \mathcal{R}_0 mit dem Algorithmus aus [3] in den Fällen von $f(i)$ der Gleichung (22) definiert durch

$$\begin{aligned} f(i) \text{ in Fall 1: } \quad \mathcal{R}_0 &= \frac{\beta s_1^*}{\eta + \mu}, \\ f(i) \text{ in Fall 2: } \quad \mathcal{R}_0 &= \frac{\beta (s_1^* + \sigma v_1^*)}{\eta + \mu}, \\ f(i) \text{ in Fall 3: } \quad \mathcal{R}_0 &= \frac{\beta (s_1^* + s_2^*)}{\eta + \mu}, \\ f(i) \text{ in Fall 4: } \quad \mathcal{R}_0 &= \frac{\beta (\sigma (v_1^* + v_2^*) + s_1^* + s_2^*)}{\eta + \mu}. \end{aligned}$$

4 Numerische Simulation

Jetzt werden numerische Simulationen des SVIR-Original-Modells und der vier Fälle der neuen SVIR-Baummodellierung untersucht. Die folgenden Daten für Parameter und Anfangswerte werden benutzt (angelehnt an Daten des RKI zu Covid-19 von Ende März 2020)

$$\begin{aligned} N &= 83\,149\,300[-], \quad \beta = 0.31666 \left[\frac{1}{\text{Tag}} \right], \\ \eta &= \frac{1}{9} \left[\frac{1}{\text{Tag}} \right], \quad \mu = \frac{1}{80 \cdot 365} \left[\frac{1}{\text{Tag}} \right]. \end{aligned}$$

$$S(0) = S_1(0) = 83\,099\,118, \quad R(0) = R_1(0) = 1\,600,$$

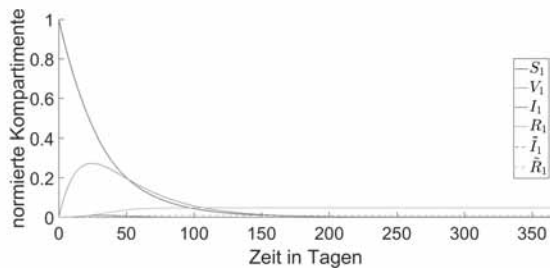
und alle übrigen Anfangswerte der Kompartimente sind zu Null gesetzt. Für ein *mathematisches Szenario* einer fiktiven Impfung wird angenommen, das ambitioniert täglich 3 % der empfänglichen Menschen geimpft werden und (leider) der Impfschutz bereits nach 20 Tagen endet. Damit sind die Kompartimente der späteren Stufen wichtig. Wenn eine Person aus S mit Wahrscheinlichkeit p erkrankt, dann erkrankt auch eine Person aus V mit Wahrscheinlichkeit σp .

$$\sigma = 0.3[-], \quad \theta = 0.05[d^{-1}], \quad \psi = 0.03[d^{-1}]$$

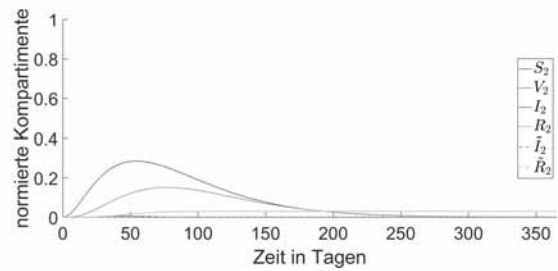
Die numerischen Simulationen werden mit *MATLAB* und dessen Routine *ode15s* erstellt. Zunächst werden die Zusammensetzung der Kompartimente aus Stufe eins und Stufe zwei der 2-stufigen SVIR-Baummodellierung gezeigt.

Die Verlaufskurven der 1- bis 4-stufigen SVIR-Baummodellierung zeigen deutlich, dass mit steigender Stufenanzahl die Approximationsgüte zunimmt. Beispielhaft sind $f(I)$ in Fall 1 und $f(I)$ in Fall 4 graphisch dargestellt.

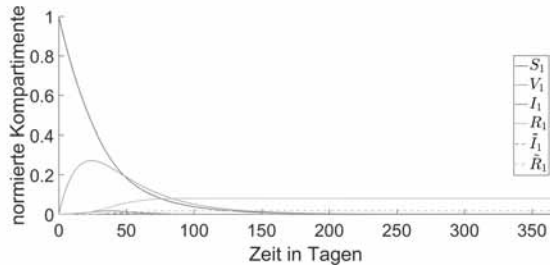
Ebenfalls die Verbesserung der Annäherungen der Kompartimentsverläufe in Fall 1 im Vergleich zu denen in Fall 4 ist deutlich zu erkennen. Insbesondere die langfristigen Anteile der gesunden Menschen (Kompartiment R) in Abbildung 6 und Abbildung 7 zeigen, dass mit steigender Stufenanzahl die Verlaufskurve besser



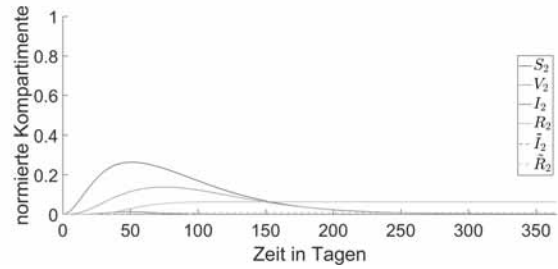
a) Stufe eins - Fall 1



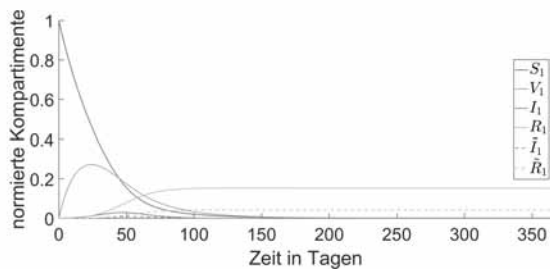
b) Stufe zwei - Fall 1



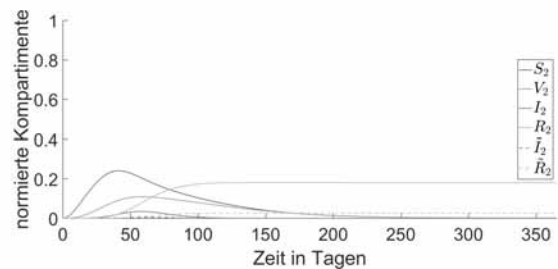
c) Stufe eins - Fall 2



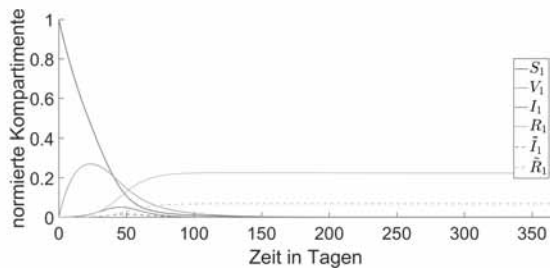
d) Stufe zwei - Fall 2



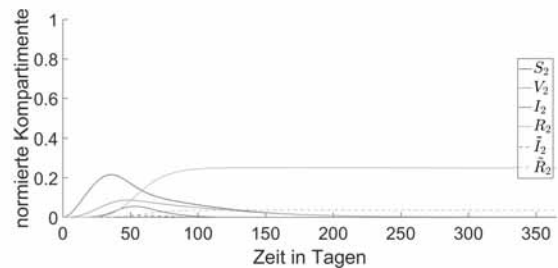
e) Stufe eins - Fall 3



f) Stufe zwei - Fall 3



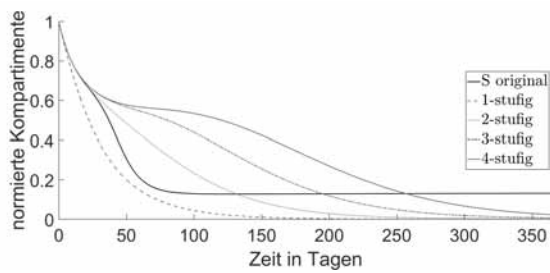
g) Stufe eins - Fall 4



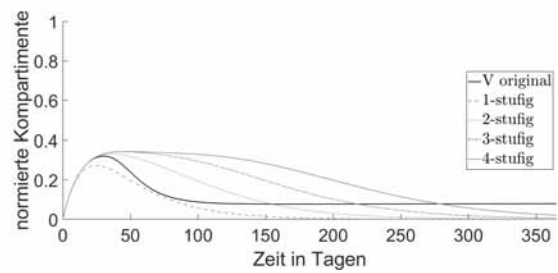
h) Stufe zwei - Fall 4

Abbildung 5: Vergleich der Verlaufskurven der ersten und zweiten Stufe der 2-stufige SVIR-Modellierung mit $f(I)$ in Fall 1 bis Fall 4.

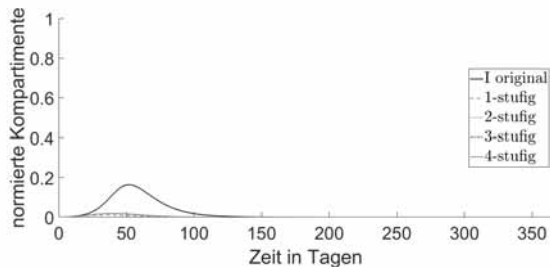
angenähert wird. Deutliche Unterschiede zeigen sich insbesondere bei der besonders wichtigen Kurve für die infierten und infizierenden Kompartimente bei den vier Fällen. Aber auch die Endwerte, die Approximationen des endemischen Gleichgewichts darstellen, unterscheiden sich z.T. erheblich.



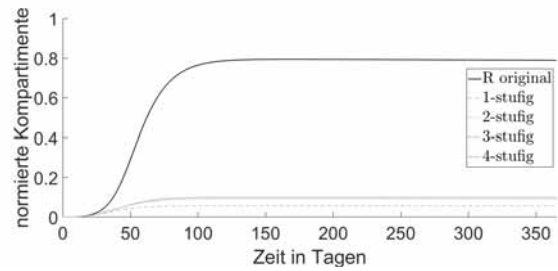
a) Kompartiment S



b) Kompartiment V

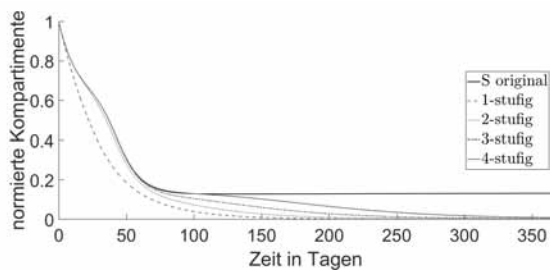


c) Kompartiment I

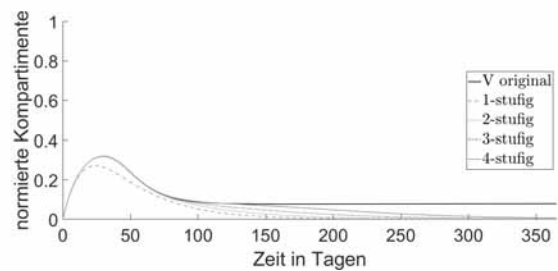


d) Kompartiment R

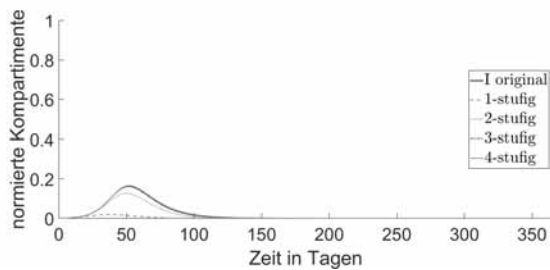
Abbildung 6: Vergleich der n -stufigen SVIR-Baummodellierung für $f(I)$ in Fall 1.



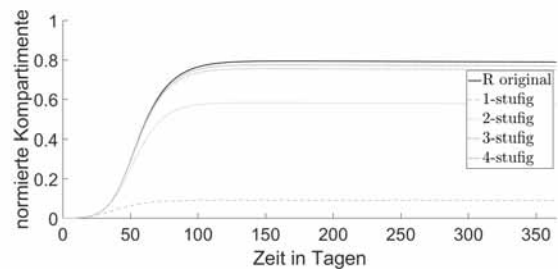
a) Kompartiment S



b) Kompartiment V



c) Kompartiment I



d) Kompartiment R

Abbildung 7: Vergleich der n -stufigen SVIR-Baummodellierung für $f(I)$ in Fall 4.

5 Fazit und Ausblick

Erste Untersuchungen ein SVIR-Kompartimentmodell mit zyklischer Dynamik, durch ein stufiges SVIR-

Kompartimentmodell mit gerichteter Baumstruktur zu ersetzen wurden dargestellt. Dazu wurden die Auswirkungen verschiedener Konstruktion von stufigen SVIR-Modellierungen ausgetestet und bewertet.

1- bis 4-stufige SVIR-Baummodellierungen wurden verglichen. Es ist zu erkennen, dass hierbei die 4-stufige SVIR-Baummodellierung in den ersten Tagen die beste Approximation liefert. Zusätzlich wurden vier verschiedene Konstellationen untersucht das infektiöse Kompartiment in der Übertragung der Infektion zu approximieren. Die Vergleiche zeigen, dass durch eine 4-stufige SVIR-Baummodellierung mit der Verwendung von $f(I)$ in Fall 4 in der ersten Zeit die beste Approximation resultiert.

Bei der hier besprochenen Mensch-zu-Mensch übertragenen Krankheit mit Impfung hat die Approximation der Gesamt-Infizierten bei der ungünstigen Wahl der kurzen Schutzdauer der Impfung noch nicht überzeugt. Bei längeren Schutzdauern der Impfung ist vermutlich die Approximationsgüte besser.

In einem nächsten Schritt wird die Idee auf eine durch Mücken übertragene Krankheit übertragen.

Literatur

- [1] Martcheva M. *An Introduction to Mathematical Epidemiology*. Springer, 2015.
- [2] Albrecht G, Fischer A, Chudej K. Analyse, Simulation und Optimale Steuerung eines mathematischen Dengue-Fieber Modells mit Impfung. *Tagungsband Workshop Heilbronn 2018 ASIM/GI Fachgruppen, ARGESIM Report*, vol. 54. ARGESIM Verlag: Wien, Österreich, 2018; 223–229.
- [3] Van den Driessche P, Watmough J. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences* 2002; **180**(1-2):29–48.
- [4] Abate A, Tiwari A, Sastry S. Box invariance in biologically-inspired dynamical systems. *Automatica* 2009; **45**(7):1601–1610.
- [5] Mitkowski W. Dynamical properties of Metzler systems. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 2008; **56**(4):309–312.

Simulation of a discharge electrode needle for particle charging in an electrostatic precipitator

Sebastian C.-Beckers^{1*}, Julian Pawlik², Hikmet Eren¹, Adam Sanaf¹, Jürgen Kiel¹

¹FMDauto-Institut, Düsseldorf University of Applied Sciences, Münsterstr. 156, 40476 Düsseldorf, Germany;

*sebastian.beckers@hs-duesseldorf.de

²getAir GmbH, Krefelder Str. 670, 41066 Mönchengladbach, Germany

Abstract. This paper describes a model approach for the simulation of a discharge electrode (DE) needle to charge particles using positive ions in an electrostatic precipitator. This includes the simulation of the electrostatic field, the space charge field of the ions and the flow field at the DE needle. The interactions of the fields, e.g. the reaction of the space charge on the electrostatic field or the electric wind are also considered in the model. To simplify and accelerate the simulation, a radial symmetry around the DE needle is partly assumed. The results of the simulation are validated by comparing the experimentally determined current-voltage characteristic with the simulated one, which show a satisfying correlation. Therefore, this model can be used as a basis for future particle flight simulation and further investigations.

Introduction

In residential applications, two-stage electrostatic precipitators (ESPs) are mainly used to separate harmful particles from the air. Particles entering the filter are first charged in the ioniser by an ion field based on a corona discharge and then separated in a subsequent filter stage by an electrostatic field (Coulomb's law) on the electrodes of the collector.

Although this filtering process is very efficient, it has the major disadvantage that it generates ozone during operation [1] [2]. Ozone can be harmful to human health when inhaled, therefore the WHO (Air Quality Guidelines Global Update 2005) sets a limit value of 50 ppb (parts per billion) for an average exposure of eight hours.

A very effective method to minimize the ozone concentration is to reduce the corona plasma region at the discharge electrode (DE) within the ioniser [3] [4], where the ozone production process takes place. Consequently, the development of DEs is geared towards ever smaller dimensions [5]. The shape and arrangement of these DEs can be very different for each application, which makes a generally valid analytical mathematical description difficult and therefore requires numerical modelling.

Experimental studies on particle separation and ozone

generation have shown good results with particle charging by a DE needle [6]. Therefore, the modelling of this approach is described in the following.

1 Experimental setup

The experimental setup used in this study consists of a stainless-steel DE needle with a radius of curvature of 55 μm at the tip and a round grid arranged at a distance of 50 mm as a ground electrode with a diameter of 85 mm, as shown in Fig. (1). The DE needle is centered by a holder on the rotation axis and protrudes 4 mm from it.

Furthermore, the DE needle is raised to a positive voltage potential by a high-voltage source of the company FUG (HCP35-20000) and the grid is connected to an electrical grounding. By using this configuration, it is possible to set and measure voltages as well as currents. Thus, the voltage-current characteristic of the DE needle can be analysed.

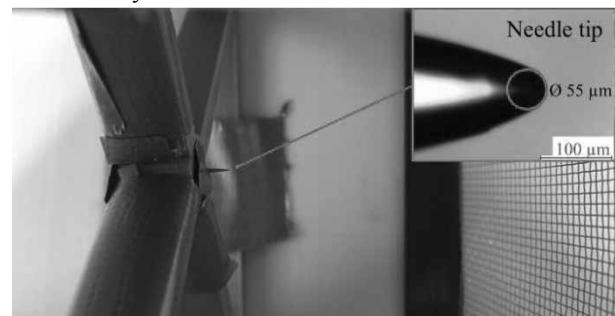


Figure 1: Test setup with the stainless-steel discharge electrode needle in the middle.

2 Model

2.1 Model approach

In order to implement the simulation of a DE needle, not only the electrostatic field, but also the flow and space charge field must be modelled.

Furthermore, the interactions of the different fields are considered in the model. For example, the reaction of the space charge density to the electrostatic field as well as the electric wind as an impact of the electrohydrodynamic (EHD) effect are taken into consideration. Fig. (2) gives an overview of the model approach.

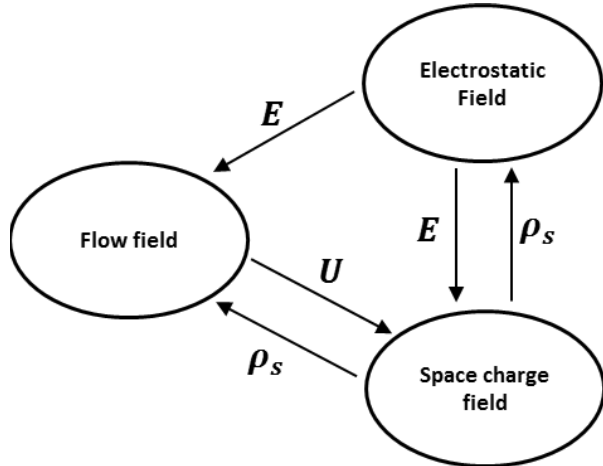


Figure 2: Overview of the simulated fields and their interactions.

To simplify the simulation, the geometry of the DE needle is modelled as a simple composition of a truncated cone with an outer diameter of 0.6 mm and a length of 4 mm and a semi-sphere with a diameter of 55 μm as needle tip, see Fig. (3).

The holder of the DE needle is also simplified as a cylinder with a diameter of 11.4 mm and a length of 10 mm, as is the measuring chamber with a diameter of 85 mm and a length of 64 mm. The geometry of the grid at the exit of the measuring chamber is neglected.

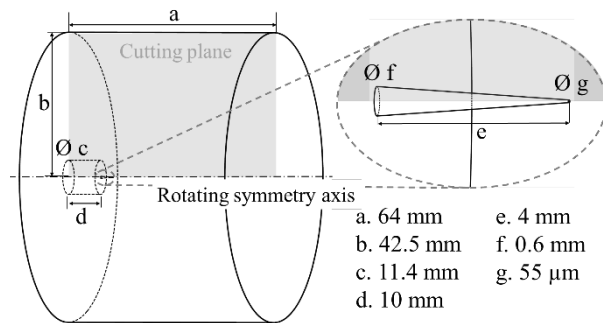


Figure 3: Simplified geometric model of the experiment.

In addition, a radial symmetry around the DE needle is assumed for the simulation of the space charge field and the flow field respectively, as shown in Fig. (4).

The micromechanisms of the corona plasma region are not simulated but the resulting convection current of

the space charges are. As a further simplification of the procedure, the corona plasma region is placed on the DE needle tip surface. A positive corona and thus a positive convection current (positive ions) are assumed.

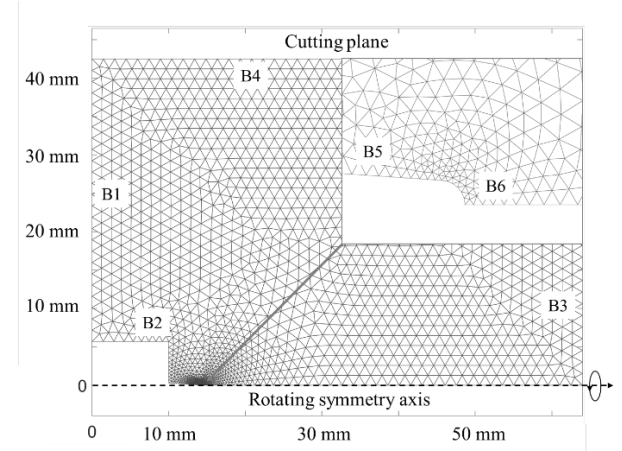


Figure 4: Geometry and meshing of the cutting plane with the boundary numbers.

2.2 Electrostatic field

The electrostatic field can be described mathematically using the following equation of Poisson [7]:

$$\Delta V = - \frac{\rho_s}{\epsilon_0 \epsilon_r} \quad (1)$$

where V is the voltage potential (V), ρ_s the space charge density ($A s m^{-3}$), ϵ_0 the permittivity of the vacuum ($8.85 \cdot 10^{-12} A s V^{-1} m^{-1}$) and ϵ_r the relative permittivity (approx. 1 for air).

Due to the very fine DE electrode tip, which is strongly curved in contrast to the grid ground electrode, a very inhomogeneous electrostatic field is created. In order to cope with this and take all effects into account, it is simulated in three dimensions. For the simulation itself, as well as the meshing, the Partial Differential Equation Toolbox (PDE-Tool) in MATLAB® is used. The total amount of tetrahedral cells used in the simulation mesh is 48811.

The tool's integrated solver calculates the solution using an FEM algorithm, assuming the following boundary conditions, where the locations can be obtained from Fig. (4).

Boundary	Description	Value
B5, B6	Potential at DE needle	$V_{SE} = V_0 + \Delta V$
B3	Potential at grid	$V_{grid} = 0$ (ground)

Table 1: Boundary conditions of the electrostatic field.

The voltage potential at the DE needle V_{SE} is composed

of the breakdown voltage V_0 which corresponds to the initial voltage of the corona discharge, and a correction value ΔV which is described in detail in Chpt. (2.5).

2.3 Space charge field

The simulation of the space charge field is based on the formula of White [8], which describes the current density \mathbf{J} ($A m^{-2}$) considering the convection and diffusion charge transport components.

$$\nabla \cdot \mathbf{J} = 0 \quad (2)$$

$$\mathbf{J} = (b_i \mathbf{E} + \mathbf{U}) \rho_s - D_i \nabla \rho_s \quad (3)$$

The convection part of Eq. (3) shows the coupling to the electrostatic field \mathbf{E} ($V m^{-1}$) and to the velocity field \mathbf{U} ($m s^{-1}$). The quantities ρ_s and b_i represent the space charge density ($A s m^{-3}$) and the ion mobility ($m V^{-1} s^{-1}$) respectively. The latter is assumed as a constant with the value $b_i = 1.85 \cdot 10^{-4} m V^{-1} s^{-1}$ [9].

The diffusion part of Eq. (3) consists of the local gradient of the space charge $\nabla \rho_s$ ($A s m^{-4}$), and the ionic diffusion coefficient D_i ($m^2 s^{-1}$), which can be estimated using the following formula [10]:

$$D_i = (b_i k T) / e \quad (4)$$

where k is the Boltzmann's constant ($1.38 \cdot 10^{-23} J K^{-1}$), e the elementary charge ($1.6 \cdot 10^{-19} As$) and T the temperature (K).

Since the geometry can be assumed to be approximately rotationally symmetrical, the simulation area for modelling the space charge density can be reduced to a two-dimensional cutting plane, see Fig. (3) and (4). As with the electrostatic field, the automatic mesher of the PDE-Tool is used for the grid generation of the two-dimensional solution area. The two-dimensional grid used has 4512 triangular cells.

The solution of Eq. (2) which describes the space charge transport is achieved by using the Finite Volume Method (FVM) in MATLAB®. In order to accomplish that, the solution area (Ω) is divided into many subareas (Ω_i) (finite volumes) and the current density at the interfaces is balanced:

$$\int_{\Omega_i} \nabla \cdot ((b_i \mathbf{E} + \mathbf{U}) \rho_s - D_i \nabla \rho_s) d\Omega_i = 0 \quad (5)$$

Due to the Gaussian integral theorem and the assumption that the values on the cell-face are uniform over the entire face, Eq. (5) can be brought into a discrete form [11]:

$$\sum_f \left[[(b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n}]_f \rho_{sf} - \left(D_i \frac{\partial \rho_s}{\partial n} \right)_f \right] A_f = 0 \quad (6)$$

where the index f represents the face, \mathbf{n} the normal vector and A_f the area of the face.

The convection term of Eq. (6) is calculated according to Long [12] using the second order Upwind Difference Method (2nd UDM). In this method, a Taylor series approach is used to project the respective space charge density onto the center of the intersection face (F), see also Fig. (5).

$$\sum_f [(b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n}]_f \rho_{sF,f} A_f \quad (7)$$

The projected space charge density $\rho_{sF,f}$ can be calculated using the 2nd UDM with the following case distinction:

$$\rho_{sF,f} = \begin{cases} \rho_{sC} + \nabla \rho_{sC} \cdot \mathbf{d}_{CF} & \text{if } ((b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n})_{F,f} > 0 \\ \rho_{sN} + \nabla \rho_{sN} \cdot \mathbf{d}_{NF} & \text{if } ((b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n})_{F,f} < 0 \end{cases} \quad (8)$$

where $\nabla \rho_{sC}$ is the local gradient of space charge densities of the cell and $\nabla \rho_{sN}$ is the one of the neighbouring cell.

In this equation, the vectors \mathbf{d}_{CF} and \mathbf{d}_{NF} represent the distance vectors between the centers of the particular cell (C and N) and the center point of the intersection face (F).

The diffusion term in Eq. (6) is implicitly calculated in this study using the space charge field.

$$\sum_f - \left(D_i \frac{\partial \rho_s}{\partial n} \right)_f A_f \quad (9)$$

Following the approach of Long [12], the gradient of space charge density in the diffusion term is determined by projected substitute points for the space charge density of the cell $\rho_{sC'}$ and the neighboring cell $\rho_{sN'}$ as well as the projected substitute point on the intersection face $\rho_{sF'}$. These three substitute points are determined by the following equations:

$$\rho_{sC'} = \rho_{sC} + \nabla \rho_{sC} \cdot \mathbf{d}_{CC'} \quad (10)$$

$$\rho_{sN'} = \rho_{sN} + \nabla \rho_{sN} \cdot \mathbf{d}_{NN'} \quad (11)$$

and

$$\rho_{sFC} = \rho_{sC} + \nabla \rho_{sC} \cdot \mathbf{d}_{CF} \quad (12)$$

$$\rho_{sFN} = \rho_{sN} + \nabla \rho_{sN} \cdot \mathbf{d}_{NF} \quad (13)$$

$$\rho_{sF} = \frac{\rho_{sFC} + \rho_{sFN}}{2} \quad (14)$$

where $\nabla \rho_s$ is the local gradient of the respective cell and \mathbf{d} is the respective difference vector between the corresponding points in the indices. ρ_{sFC} and ρ_{sFN} represent space charge density values projected from the centers of the cell (C) and the neighbouring cell (N) to the center point of the intersection face (F).

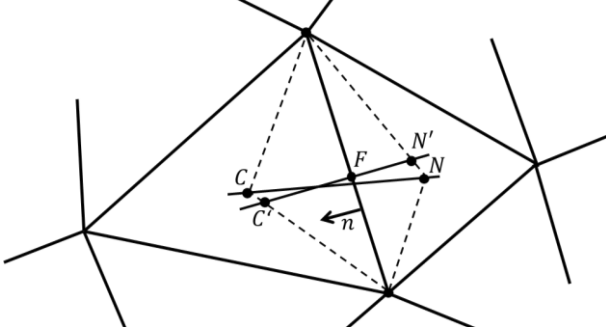


Figure 5: Demonstration of the gradient calculation between adjacent mesh cells.

Based on these three substitute points of the space charge density (Eq. (12) to Eq. (14)), the mean slope can be determined by linear interpolation. The mean slope then corresponds to the gradient at the intersection face of both cells.

The convection and diffusion term of space charge transport shown in Eq. (7) and Eq. (9) can then be expressed in a simple form:

$$\sum_f a_{conv_f} \rho_{s,f} = \sum_f (b_{conv_f} - b_{diff_f}) \quad (15)$$

where

$$a_{conv_f} = [(b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n}]_f A_f \quad (16)$$

$$b_{conv_f} = -a_{conv_f} (\nabla \rho_{s_i} \cdot \mathbf{d}_{ij})_f \quad (17)$$

$$b_{diff_f} = -\left(D_i \frac{\partial \rho_s}{\partial n}\right)_f A_f \quad (18)$$

The quantities $\nabla \rho_{s_i}$ and \mathbf{d}_{ij} refer to the case distinction of the 2nd UDM in Eq. (8).

If Eq. (15) is applied to all cells in the solution area it yields a linear system of equations in the form:

$$\mathbf{A} \cdot \boldsymbol{\rho}_s = \mathbf{B} \quad (19)$$

which is then solved using the method of least squares (*lqslin* function) in MATLAB®.

The boundary conditions used for the simulation of the space charge field are given below.

Boundary	Description	Value
B6	Current density input	$\mathbf{J} \cdot \mathbf{n} = \frac{I_0}{A_{out}}$
B1, B3	Current density output	$\mathbf{J} \cdot \mathbf{n} = [(b_i \mathbf{E} + \mathbf{U}) \cdot \mathbf{n}] \rho_s$
B2, B4, B5	Wall	$\mathbf{J} \cdot \mathbf{n} = 0$

Table 2: Boundary conditions of the space charge field.

The current value I_0 in the boundary condition of the input current density represents an input parameter of the model and must be distributed over the entire outlet surface A_{out} of the DE needle tip.

For the boundary condition of the output current density, only the convection component is taken into account, due to the assumption that the change of space charge density near the surface of the output is neglectable.

2.4 Flow field

The flow field in an electrostatic precipitator which can be modelled according to [13 - 15] by the time-averaged Navier-Stokes equation for incompressible fluids with the standard k - ϵ turbulence model [16]:

$$\nabla \cdot \mathbf{U} = 0 \quad (20)$$

$$\rho_F (\mathbf{U} \cdot \nabla) \mathbf{U} - (\mu + \mu_T) \Delta \mathbf{U} = -\nabla p + \rho_F \mathbf{g} + \mathbf{F}_{EHD} \quad (21)$$

where ρ_F is the fluid density ($kg\ m^{-3}$), μ the laminar viscosity ($kg\ m^{-1}\ s^{-1}$), μ_T the turbulent viscosity of the k - ϵ turbulence model ($kg\ m^{-1}\ s^{-1}$), p the fluid pressure (Pa) and \mathbf{g} the body accelerations acting on the continuum ($m\ s^{-2}$). \mathbf{F}_{EHD} represents the electrical body force term of the EHD-effect ($N\ m^{-3}$), which appears in form of electric wind in the flow field and is determined as follows:

$$\mathbf{F}_{EHD} = \mathbf{E} \rho_s \quad (22)$$

The flow field is simulated with the flow simulation software OpenFOAM® based on the Finite Volume Method. A program interface between MATLAB® and OpenFOAM® was developed to exchange input and output parameters in form of geometry and mesh data, boundary and start conditions, material and substance values as well as field data.

Geometry and mesh data are created in MATLAB® by the PDE Tool's automatic mesher and the finished mesh is transferred to the OpenFOAM® software. As a simplification, a two-dimensional geometry with a radial

symmetry is assumed, see Fig. (4). The number of triangular cells of the mesh is also 4512.

For the implementation of the EHD effect the *simpleFoam* solver was modified. Flow simulations of stationary and incompressible Newtonian and turbulent fluids can be performed with the *simpleFoam* solver (OpenFOAM® User Guide), in which the standard k- ϵ model was used as turbulence model. The modified solver considers the influence of the electric wind as a body source term in the Navier-Stokes equation based on the current fields of \mathbf{E} and ρ_s , also see Eq. (21) and Eq. (22). As \mathbf{F}_{EHD} is a spatial volume force in the flow field, it must be projected onto a two-dimensional geometry.

The resulting simulated flow field is then returned to the other MATLAB® models via the programmed interface.

The following boundary conditions are used for the flow and pressure field of the model:

Boundary	Description	Value
B1	inlet flow	$\mathbf{U} = \mathbf{U}_{in}$ $\nabla p = 0$
B3	outlet flow	$\nabla \mathbf{U} = 0$ $p = p_0$
B2, B4, B5	Wall	$\mathbf{U} = 0$ (no slip) $\nabla p = 0$

Table 3: Boundary conditions of the flow and pressure field.

In Tab. (3), U_{in} corresponds to the inlet flow ($m\ s^{-1}$) and p_0 to the ambient pressure (Pa).

2.5 Calculation sequence

The calculation sequence shown in Fig. (6) starts with an input current I_0 and an input start voltage V_0 of the DE needle. This input voltage can be determined experimentally or estimated by using empirical formulas (e.g. according to Peek [8]). Next, the three model fields are calculated until convergence is achieved. After convergence, a correction value ΔV is determined for the voltage potential of the DE needle via the resulting electrostatic field. Afterwards, the potential V_0 is adjusted accordingly with $V_0 = V_0 + \Delta V$ and the calculation of the fields is started again.

The calculation sequence ends as soon as the correction value ΔV runs towards zero and no voltage potential change can be observed anymore.

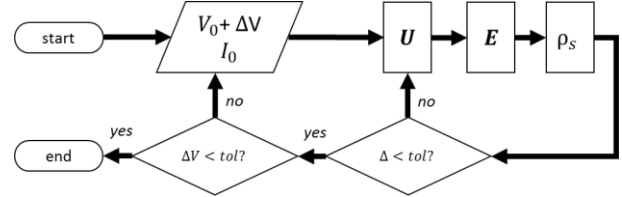


Figure 6: Schematic representation of the calculation sequence.

3 Model validation

The model is validated by comparing the experimentally determined and simulated voltage-current characteristics of the DE needle, which are shown in Fig. (7).

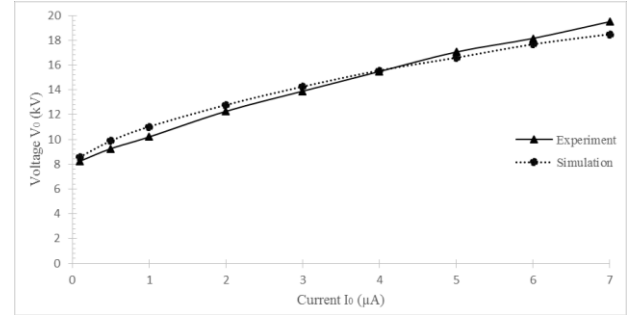


Figure 7: Voltage-current characteristics of the DE needle.

The breakdown voltage of the DE needle was approximately $V_0 = 8\ kV$ in the experiment. This value was used as the starting value for the simulation according to the calculation sequence. As can be seen in Fig. (7), the curve progression of the simulation largely complies well with the experimentally determined curve progression, whereby the simulation slightly exceeds the voltage potential below $4\ \mu A$ and slightly falls below it above $4\ \mu A$.

These deviations can probably be explained by inaccuracies in geometric modelling (e.g. the shape of the DE needle) and by the model simplifications of rotational symmetry that were applied.

4 Conclusion and outlook

Due to the various simplifications in geometry and symmetry assumptions, an efficient DE needle model could be developed, which provides fast and sufficiently good results with regard to validation.

Based on this, the particle flight can then be modelled using the Discrete Element Method (DEM) to analyse the particle separation behaviour by the DE needle in the electrostatic precipitator.

If the accuracy of the model is to be improved, a three-dimensional model approach to the space charge

density as well as the flow field should be used. It would also be advisable to use a more precise geometric model. However, these improvements would be accompanied by an increased computing time.

Based on the model presented, the modelling of ozone production at the DE needle would also be an interesting topic for future studies.

References

- [1] Caste GSP. *Electrostatic Precipitation in Electrified Media and Positive Corona Ozone Generation in the Design of High Efficiency Air Cleaners* [dissertation]. [Faculty of Engineering Science, CAN]. University of Western Ontario; 1969.
- [2] Boelter KJ, Davidson JH. Ozone Generation by Indoor, Electrostatic Air Cleaners. *Aerosol Sci. and Technology*. 1997; 27(6), 689–708. doi: 10.1080/02786829708965505.
- [3] Chen J, Davidson JH. Ozone production in the positive DC corona discharge: Model and comparison to experiments. *Plasma Chem. and Plasma Proc.* 2002; 22 (4), 495–522. doi: 10.1023/A:10231315412208
- [4] Chen J, Davidson JH. Ozone production in the negative DC corona: the dependence of discharge polarity. *Plasma Chem. and Plasma Proc.* 2003; 23 (3), 501–518. doi: 10.1023/A:1022468803203
- [5] Bo Z, Yu K, Lu G, Mao S, Chen J, Fan F-G. Nanoscale discharge electrode for minimizing ozone emission from indoor corona devices. *Env. science & technology*. 2010; 44 (16), 6337–6342. doi: 10.1021/es903917f.
- [6] Hak-Joon K, Myungjoon K, Bangwoo H, Chang GW, Ayyoub Z, Noureddine Z, Yong-Jin K. Fine particle removal by a two-stage electrostatic precipitator with multiple ion-injection-type prechargers. *J. of Aerosol Scienc.* 2019; 130, 61–75. doi: 10.1016/j.jaerosci.2019.01.004.
- [7] Leuchtmann P. *Einführung in die elektrische Feldtheorie*. 1. Auflage. Freising: Pearson Studium; 2007. 602 p.
- [8] White HJ. *Entstaubung industr. Gase mit Elektrofiltern*. Leipzig: D. Verlag für Grundstoffindustrie; 1969. 336 p.
- [9] McDonald JR, Smith WB, Spencer, Herbert W, Sparks, Leslie E. A mathematical model for calculating electrical conditions in wire-duct electrostatic precipitation devices. *J. of Appl Phys.* 1977; 48 (6), 2231–2243. doi: 10.1063/1.324034.
- [10] Abdel-Salam M, Nakano M, Mizuno A. Corona-induced pressures, potentials, fields and currents in electrostatic precipitator configurations. *J. Phys. D: Appl. Phys.* 2007; 40 (7), 1919–1926. doi: 10.1088/0022-3727/40/7/014.
- [11] Patankar SV, Minkowycz WJ, Sparrow EM. *Series in computational methods in mechanics and thermal sciences*. New York: McGraw-Hill Book Company; 1980. 197 p.
- [12] Long Z, Yao Q, Song Q, Li S. A second-order accurate finite volume method for the computation of electrical conditions inside a wire-plate electrostatic precipitator on unstructured meshes. *Journal of Electrostatics*. 2009; 67 (4), 597–604. doi: 10.1016/j.elstat.2008.12.006.
- [13] Chun YN, Chang J-S, Berezin AA, Mizeraczyk J. Numerical modeling of near corona wire electrohydrodynamic flow in a wire-plate electrostatic precipitator. *IEEE Trans. Dielect. Electr. Insul.* 2007; 14 (1), 119–124. doi: 10.1109/TDEI.2007.302879.
- [14] Long Z, Yao Q. Evaluation of various particle charging models for simulating particle dynamics in electrostatic precipitators. *J. of Aer. Sci.* 2010; 41 (7), 702–718. doi: 10.1016/j.jaerosci.2010.04.005.
- [15] Chang JS, Dekowski J, Podlinski J, Brocilo D, Urashima K, Mizeraczyk J. Electrohydrodynamic gas flow regime map in a wire-plate electrostatic precipitator. *Fourtieth IAS Ann..Meet. C. Record of the 2005 Ind. Appl. Conf.* 2005; 4, 2597–2600. doi: 10.1109/IAS.2005.1518826.
- [16] Launder BE, Spalding DB. The numerical computation of turbulent flows. *C. Meth.in Appl. Me. and En.* 1974; 3(2), 269–289. doi: 10.1016/0045-7825(74)90029-2.

Analyse und Simulation des Kraftübertragungsverhaltens von Mecanum-Rädern

Marian Göllner^{1*}, Xiaobo Liu-Henke¹, Ludger Frerichs²

¹Institut für Mechatronik, Ostfalia Hochschule für angewandte Wissenschaften, Salzdahlumer Str. 46/48, 38302 Wolfenbüttel; *mar.goellner@ostfalia.de

²Institut für mobile Maschinen und Nutzfahrzeuge, Technische Universität Braunschweig, Langer Kamp 19a, 38106 Braunschweig

Abstract. Im vorliegenden Beitrag wird exemplarisch eine Methode zur Auslegung eines Mecanum-Rades unter Berücksichtigung der Abrollgeometrie und der wirkenden Kräfte aufgezeigt. Die Berechnungsergebnisse werden anschließend auf Grundlage von Erkenntnissen der Tribologie mit den zu erwartenden Reibeigenschaften des Rades zum Grund verknüpft, um Erkenntnisse zur Modellierung von Reibung in der Simulation dynamischer Systeme zu gewinnen.

Einleitung

Um in engen Arealen und streng vorgegebenen Bahnen, bspw. Intralogistiksysteme in einem Fabrikumfeld, effizient manövrieren zu können, bedarf es omnidirektionaler Antriebe, die auf Grundlage von Allseitenrädern funktionieren, welche über einen passiven Freiheitsgrad verfügen und daher keine zusätzlichen nicht-holonomen Bindungen bei der Integration in ein Fahrwerk verursachen. Der passive Freiheitsgrad wird bei diesen sog. Allseitenrädern durch Montage von frei drehbaren, tonnenförmigen Rollen (Radtonnen) auf dem Radumfang realisiert [1]. Man unterscheidet nach dem Winkel Γ zwischen Rollrichtung der umlaufenden Tonne und der horizontalen Achse nach Mecanum-Rädern ($\Gamma = 45^\circ$) und Omniwheels ($\Gamma = 90^\circ$). Dabei weisen Mecanum-Räder nach Untersuchungen von [2] bessere Eigenschaften bei der Höhe der zu transportierenden Last, der Kraftübertragung und der Flächenpressung im Bezug zu ihrer Größe und ihrem Gewicht auf. Aufgrund dessen haben sich für den Einsatz in Intralogistiksystemen Mecanum-Räder durchgesetzt. Die hier hergeleiteten Gleichungen sind also gerade in dieser Domäne von besonderer Relevanz.

Die Fachgruppe für Regelungstechnik und Fahrzeugmechatronik des Instituts für Mechatronik

der Ostfalia Hochschule hat zwei Funktionsträger im Bezug zur Intralogistik aufgebaut, welche beide über Mecanum-Radkonfigurationen angetrieben werden. Nachfolgend sind beide Forschungsträger in Abbildung 1 dargestellt.

1 Motivation

Die Analyse und Simulation des Kraftübertragungsverhaltens der Mecanum-Räder gestaltet sich aufgrund der komplexen Beschaffenheit der interagierenden Oberflächen als technisch anspruchsvoll, weshalb in der vorliegenden Arbeit die Zusammenhänge zwischen den kinematischen und geometrischen Abrollbedingungen und den daraus folgenden Kontaktpunkten zwischen Rad und Fläche genutzt werden, um die Flächenpressung zwischen diesen herzuleiten und auf dieser Basis eine Aussage über das reibbedingte Kraftübertragungsverhalten zu treffen. Weiterhin sollen diese Erkenntnisse in ein Simulationsmodell einfließen, in dem das Übertragungsverhalten dynamisch rückgekoppelt wird.

So lassen sich Aussagen über Optimierungsansätze der konstruktiven Gestaltung der Mecanum-Räder bezüglich ihrer Abrolleigenschaften und der Linearität der Kraftübertragung treffen. Eine optimale Kraftübertragung stellt sich durch einen im Idealfall konstanten, in der Praxis zumindest linearen Verlauf bezüglich des Radumfangs dar, sodass diese unter Echtzeitanforderungen in einem Regelsystem berücksichtigt werden kann. Dazu werden exemplarisch an den beiden Forschungsträgern AGV und S-Mobile die zur Untersuchung notwendigen Zusammenhänge hergeleitet. Das AGV repräsentiert hier den Stand der Technik, da seine Antriebskonfiguration der in [3] hergeleiteten üblichen Konfiguration entspricht. Das S-Mobile hingegen stellt einen Sonderfall dar, da hier die Mecanum-Räder auf einer Kugel ablaufen. In [4]

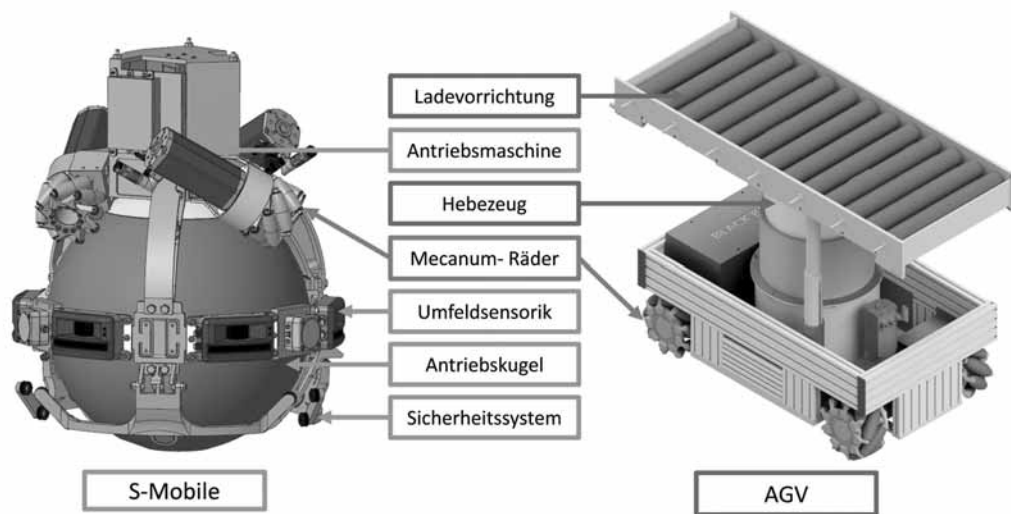


Abbildung 1: Das S-Mobile und das AGV, Forschungsträger der Fachgruppe für Regelungstechnik und Fahrzeugmechanik

wurde diesbezüglich bereits eine geometrische Methode hergeleitet mit der dieser Umstand berücksichtigt und ein ruhiges und gleichmäßiges Abrollen der Räder auf der Kugeloberfläche sichergestellt und mithilfe von CAE Methoden bewiesen werden konnte.

Da das Kraftübertragungsverhalten weiterhin nicht kontinuierlich ist und sich die Übertragungspunkte ungleichmäßig auf dem Umfang der Radtonnen verteilen, sollen im Folgenden die Kraftübertragungsverhältnisse untersucht und Möglichkeiten der Simulation sowie späteren Nutzung der Ergebnisse zur Kompensation der gefundenen Effekte hergeleitet werden. Ansatz zur Untersuchung ist dabei zunächst die Analyse der interagierenden Oberflächen und deren geometrischer Eigenschaften. Mithilfe dieser lassen sich Aussagen über die inneren Spannungszustände der Materialien der interagierenden Oberflächen wie in [5] herleiten. Eine Verknüpfung der Materialzustände zu deren Reibverhalten und somit direkt zu deren Kraftübertragungseigenschaften ist aus der Tribologie in Form von Reibkennwertkurven nach [6] bekannt.

Folgende sollen nach dem etablierten Modellbildungsprozess zunächst anhand der realen Gestalt der Mecanum- Räder Modelle der Oberfläche und deren Material-bedingter Spannungszustände hergeleitet und in mathematische Gleichungen überführt werden.

2 Modellbildungsprozess

Im Rahmen dieser Arbeit wird unter anderem die Modellbildung des Kraftübertragungsverhaltens durchgeführt. Der dazu notwendige Modellbildungsprozess (vgl. Abbildung 2) basiert auf dem realen System, welches gemäß der Aufgabenstellung reduziert bzw. vereinfacht wird, sodass sich ein physikalisches Modell ergibt. Dieses wird mithilfe physikalischer Gesetzmäßigkeiten in ein mathematisches Modell überführt, welches wiederum bspw. in Form von Signalflussplänen im Rechner abgebildet und mithilfe von CAE-Werkzeugen und entsprechender Numerik simuliert werden kann.

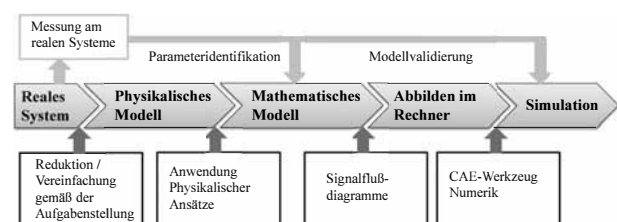


Abbildung 2: Modellbildungsprozess

Der Modellbildungsprozess umfasst zudem Messungen am realen System, um zum einen die Parameter des mathematischen Modells zu identifizieren und zum anderen die Simulation zu validieren.

2.1 Beschreibung der geometrischen Abrollbedingungen

Die essentiell zur Berechnung notwendige kartesische Beschreibung der realen Oberfläche der Mecanum-Radtonne ergibt sich aus den Abrollbedingungen des Rades auf der Kugel.

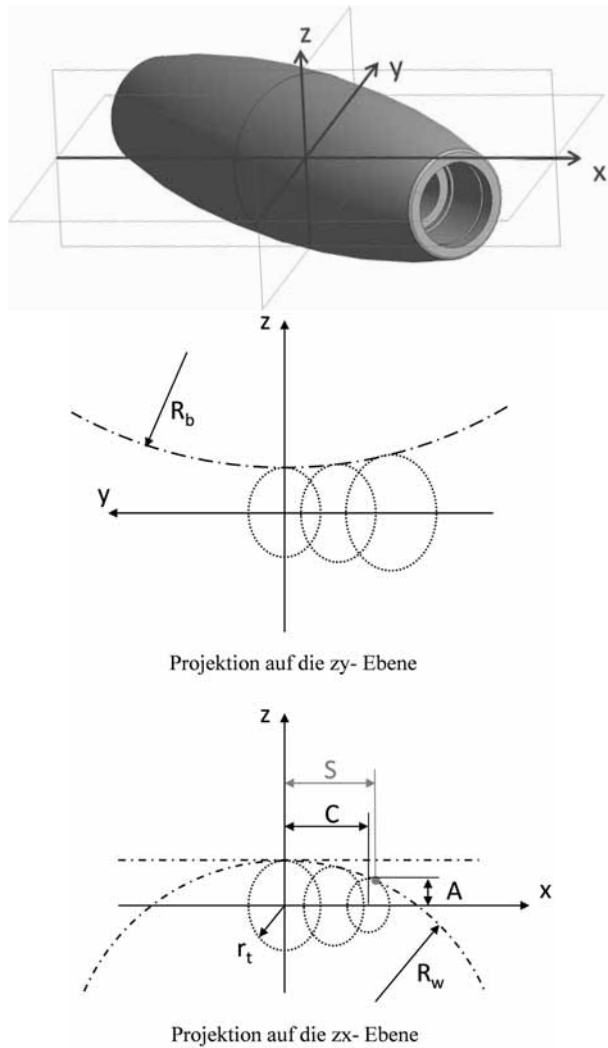


Abbildung 3: notwendigen Projektionen der Mecanum-Radtonne

Um eine ruhiges und gleichmäßiges Abrollen zu ermöglichen wurde bereits in [4] ein geometrischer Zusammenhang zwischen dem axialen und radialen Krümmungsverlauf der Tonne zu dem Radius der Antriebskugel R_b und dem Schnittradius des Mecanum-Rades r_t über eine Hilfsgröße S errechnet. Diese Hilfsgröße S beschreibt wie in Abbil-

dung 3 zu sehen den idealen Kontaktpunkt zwischen Kugel und Radtonne. An diesem Punkt liegen die approximierten Oberflächengeometrien der Antriebsskugel, des (virtuellen) Umfangs des Mecanum-Rades R_w und des Radius der Mecanum Tonne r_t als auch die Ableitungsfunktionen der Oberflächengeometrie übereinander. Die geschlossenen dreidimensionale Oberfläche entsteht durch Drehung mit dem Winkel β . Der folgende formale Zusammenhang gilt entsprechend für die Kontur der auf einer Kugel abrollenden Tonne in radialer und axialer Richtung.

$$\begin{aligned} \Xi(S, \beta) &= \begin{bmatrix} x_S \\ y_b \\ z_b \end{bmatrix} \\ &= \begin{bmatrix} S \cdot \sqrt{2} + \frac{\sqrt{2}}{2} \cdot ((R_A - (R_w - r_t)) \cdot \frac{-S}{R_A}) \\ y_s \cdot \sin \beta + z_s \cdot \cos \beta \\ y_s \cdot \cos \beta + z_s \cdot \sin \beta \end{bmatrix} \end{aligned} \quad (1)$$

Mit dem Hilfskoordinatensystem:

$$\begin{aligned} y_s &= \sqrt{(R_A - (R_w - r_t))^2 + 2 \cdot \left(\frac{1}{2} (R_A - (R_w - r_t)) \cdot \frac{-S}{R_A} \right)^2} \\ z_s &= \left(R_b - \sqrt{R_b^2 - \left(S + \frac{1}{2} (R_A - (R_w - r_t)) \cdot \frac{-S}{R_A} \right)^2} \right) \end{aligned}$$

bezogen auf den absoluten Abstand:

$$R_A = \sqrt{R_w^2 - S^2}$$

Die gefundenen Gleichungen sind allgemein für Mecanum-Räder jeglicher Gestalt, sowohl auf sphärischen als auch planaren Flächen (als Sonderfall mit unendlich großem Radius) abrollend, gültig. Nachfolgend sind in Tabelle 1 die für den hier vorgestellten Anwendungsfall genutzten Parameter aufgelistet.

Tabelle 1: Parameter der Versuchsträger

Par	Bedeutung	S-Mob.	AGV
R_w	abs. Radradius	60 mm	60 mm
r_t	max. Tonnenradius	13 mm	13 mm
R_b	Radius Sphaere	236 mm	∞

Mit diesen ergeben sich die in der folgenden Abbildung 4 gezeigten Ergebnisse für die Oberflächenkontur und den Verlauf der Oberflächenkrümmung über die

Längsachse der Radtonne. Ein ausführlicher Vergleich ist in Kapitel 4 zu finden.

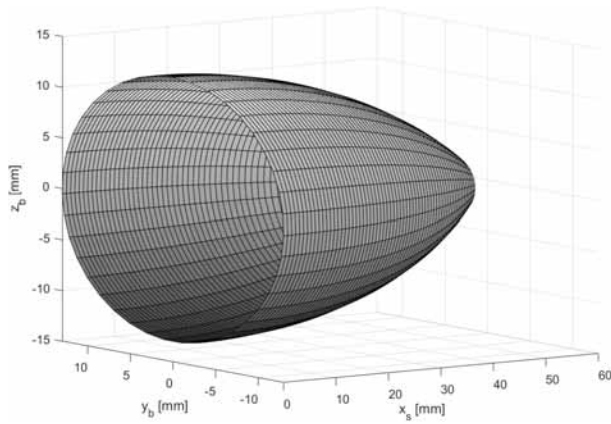


Abbildung 4: Oberflächenkrümmung über die Längsachse der Radtonne

2.2 Herleitung der Kontaktgeometrie

Um die Kontaktverhältnisse der beiden interagierenden Oberflächen herzuleiten, bietet es sich an diese im Vektorraum abzubilden, um die komplexen geometrischen Funktion über differentialgeometrische Methoden zugänglich zu machen. Dazu ist notwendig zunächst das Einheitsnormalenvektorfeld \vec{n} der Oberflächengeometrie Ξ herzuleiten. Aus der Differentialgeometrie folgt:

$$\vec{n}(S, \beta) = \frac{\frac{\partial \Xi(S, \beta)}{\partial S} \times \frac{\partial \Xi(S, \beta)}{\partial \beta}}{\left| \frac{\partial \Xi(S, \beta)}{\partial S} \times \frac{\partial \Xi(S, \beta)}{\partial \beta} \right|} \quad (2)$$

Aus den partiellen Differentialen ergeben sich die erste Fundamentalform I_p welche die innere Geometrie der Fläche beschreibt:

$$I_p(S, \beta) = \begin{bmatrix} \mathfrak{E}(S, \beta) & \mathfrak{F}(S, \beta) \\ \mathfrak{F}(S, \beta) & \mathfrak{G}(S, \beta) \end{bmatrix} \quad (3)$$

$$= \begin{bmatrix} \frac{\partial \Xi(S, \beta)}{\partial S} \cdot \frac{\partial \Xi(S, \beta)}{\partial S} & \frac{\partial \Xi(S, \beta)}{\partial S} \cdot \frac{\partial \Xi(S, \beta)}{\partial \beta} \\ \frac{\partial \Xi(S, \beta)}{\partial S} \cdot \frac{\partial \Xi(S, \beta)}{\partial \beta} & \frac{\partial \Xi(S, \beta)}{\partial \beta} \cdot \frac{\partial \Xi(S, \beta)}{\partial \beta} \end{bmatrix}$$

sowie die zweite Fundamentalform II_p der Oberflächengeometrie welche direkt aus partiellen Ableitungsfunktionen der ersten Fundamentalform

sowie dem Einheitsnormalenvektorfeld hervorgeht:

$$II_p(S, \beta) = \begin{bmatrix} \mathfrak{L}(S, \beta) & \mathfrak{M}(S, \beta) \\ \mathfrak{M}(S, \beta) & \mathfrak{N}(S, \beta) \end{bmatrix} \quad (4)$$

$$= \begin{bmatrix} \vec{n}(S, \beta) \cdot \frac{\partial^2 \Xi(S, \beta)}{\partial S \cdot \partial S} & \vec{n}(S, \beta) \cdot \frac{\partial^2 \Xi(S, \beta)}{\partial S \cdot \partial \beta} \\ \vec{n}(S, \beta) \cdot \frac{\partial^2 \Xi(S, \beta)}{\partial S \cdot \partial \beta} & \vec{n}(S, \beta) \cdot \frac{\partial^2 \Xi(S, \beta)}{\partial \beta \cdot \partial \beta} \end{bmatrix}$$

Über eine einfache Division der zweiten Fundamentalform mit der ersten erhält man die negative Weingartenabbildung der Fläche.

$$L_p(S, \beta) = I_p(S, \beta)^{-1} \cdot II_p(S, \beta) \quad (5)$$

$$= \frac{1}{\mathfrak{E} \cdot \mathfrak{G} - \mathfrak{F}^2} \begin{bmatrix} \mathfrak{L} \cdot \mathfrak{G} - \mathfrak{M} \cdot \mathfrak{F} & \mathfrak{M} \cdot \mathfrak{G} - \mathfrak{N} \cdot \mathfrak{F} \\ \mathfrak{M} \cdot \mathfrak{E} - \mathfrak{L} \cdot \mathfrak{F} & \mathfrak{N} \cdot \mathfrak{E} - \mathfrak{M} \cdot \mathfrak{F} \end{bmatrix}$$

Diese selbstadjungierte lineare Vektorraumabbildung des dreidimensionalen Raumes ist diagonalisierbar, d.h. es gibt eine Basis aus Eigenvektoren und somit Eigenwerte [7]. Berechnet man nun die Eigenwerte dieser Matrix, so ergeben diese, aus den Grundlagen der differentiellen Geometrie ersichtlich, die Krümmungen in Richtung der beiden Raumparameter. Da die Hauptkrümmungen die Kehrwerte der Radien an einer Stelle sind, lässt sich die Hypothese durch Umrechnung und Vergleich überprüfen. Das Vorzeichen folgt der Nomenklatur der Gauß'schen Differentialgeometrie:

$$\underline{\Xi}_2(S, \beta) = \begin{bmatrix} \rho_{21}(S, \beta) \\ \rho_{22}(S, \beta) \end{bmatrix} = \det(L_p(S, \beta) - \lambda I_n) \quad (6)$$

Eine Anwendung auf ein Mecanum-Rad ist indes nicht bekannt, weshalb hier über eine zweite Methode, der Kreissegmentnäherung in Abbildung 5, der Anwendungsfall validiert werden soll. Die Krümmung kann

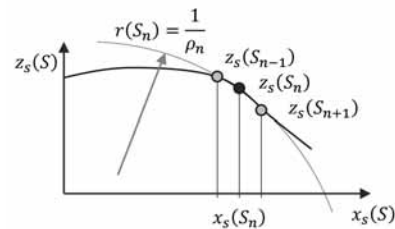


Abbildung 5: Kreissegmentnäherung

allgemein über ein Kreissegment mit einem konstanten Radius und dem einfachen Zusammenhang über folgen-

des Gleichungssystem hergeleitet werden:

$$\begin{aligned} a_1 - x_n(S_{n-1}) \cdot a_2 - z_s(S_{n-1}) \cdot a_3 &= x_s(S_{n-1})^2 + z_s(S_{n-1})^2 \\ a_1 - x_n(S_n) \cdot a_2 - z_s(S_n) \cdot a_3 &= x_s(S_n)^2 + z_s(S_n)^2 \\ a_1 - x_n(S_{n+1}) \cdot a_2 - z_s(S_{n+1}) \cdot a_3 &= x_s(S_{n+1})^2 + z_s(S_{n+1})^2 \end{aligned}$$

In Matrixschreibweise dargestellt als lineare Funktion ergibt sich:

$$\underbrace{\begin{bmatrix} 1 & -x_n(S_{n-1}) & -z_s(S_{n-1}) \\ 1 & -x_n(S_n) & -z_s(S_n) \\ 1 & -x_n(S_{n+1}) & -z_s(S_{n+1}) \end{bmatrix}}_{\underline{m}} \cdot \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} x_s(S_{n-1})^2 + z_s(S_{n-1})^2 \\ x_s(S_n)^2 + z_s(S_n)^2 \\ x_s(S_{n+1})^2 + z_s(S_{n+1})^2 \end{bmatrix}}_{=\underline{b}} \quad (7)$$

Ansatz wäre nun die Kontur in diskreten Teilstücken, wie in Abbildung 5 dargestellt über Kreis-segmente zu nähern und so die Krümmungen an den Diskretisierungspunkten zu errechnen. Diese wären einfach über

$$\underline{x} = \underline{m}^{-1} \cdot \underline{b} \quad (8)$$

zu bestimmen. Ein Vergleich beider Methoden zeigt in Abbildung 10 des Kapitels 4 das die Weingarten Abbildung mit der Methode der diskreten Kreissegment-näherung sehr gut übereinstimmt, weshalb diese für die weitere Betrachtung des Problems genutzt werden soll.

2.3 Hertz'sche Pressung im Kraftübertragungselement

Laut Originalveröffentlichung Heinrich Hertz [8] ergibt sich die Pressung in einem Kontaktpunkt zweier Bauteile über dessen gegenseitige Einwölbung der Oberfläche in Form von Verformungsellipsen. Diese Verformung ist neben den Materialeigenschaften (siehe Tabelle 2) stark abhängig von der jeweiligen Krümmung der beiden Oberflächenpaare am Kontaktpunkt. Während die Krümmungsverläufe bei gleichmäßig geformten Oberflächen, wie einer Kugel, durch Invertierung der Radien zu berechnen sind, ist diese bei den betrachteten Oberflächen der Mecanum- Räder nur über die im vorherigen Kapitel vorgestellte Vektor-raumdarstellung zugänglich.

Der Ersatzfaktor W_{ers} errechnet sich als quasi Durchschnitt der Materialeigenschaften der Paarung,

Tabelle 2: Materialeigenschaften der Mecanum Räder

Par	Bedeutung	S-Mobile	AGV
F_N	Normalkraft	976 N	976 N
E_1	E-Modul Rad	210 Pa	5 Pa
E_2	E-Modul Grund	13 Pa	30 Pa
ν	Poisson Zahl	0,3	0,3

falls diese aus unterschiedlichen Materialien bestehen. In dem hier vorliegenden Fall wird über

$$W_{ers} = \frac{1}{2} \left(\frac{1 - \nu^2}{E_1} + \frac{1 - \nu^2}{E_2} \right) \quad (9)$$

errechnet wie die Paarung zum einen beim S-Mobile aus stählernem Mecanum- Rad und GFK- Kugel und zum anderen beim AGV aus gummierten Radtonnen und Beton- Hallenboden zusammenwirken. Die folgende Abbildung 6 zeigt die ideelle Verformungsellipse mit Randpunkt $P(x|y)$ und ihren Halbachsen d und e .

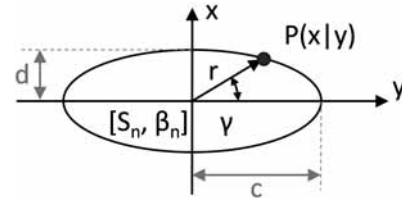


Abbildung 6: Verformungsellipse mit Halbachsen

$$P(x|y) = \begin{bmatrix} x_s(S, \beta) + c(S, \beta) \cdot \cos(\gamma) - d(S, \beta) \cdot \sin(\gamma) \\ y_b(S, \beta) + c(S, \beta) \cdot \sin(\gamma) + d(S, \beta) \cdot \cos(\gamma) \end{bmatrix}$$

Die Halbachsen ergeben sich nach Hertz zu:

$$c(S, \beta) = \zeta \cdot \sqrt[3]{\frac{3 \cdot F_N \cdot W_{ers}}{\Sigma \rho(S, \beta)}} \quad (10)$$

$$d(S, \beta) = \eta \cdot \sqrt[3]{\frac{3 \cdot F_N \cdot W_{ers}}{\Sigma \rho(S, \beta)}} \quad (11)$$

Die Verformungsellipse ist also neben den Materialeigenschaften auch von den Krümmungen ρ und zwei Beiwerten ζ und η abhängig. Die Krümmungen gehen

dabei über die Krümmungssumme

$$\begin{aligned}\sum \rho(S, \beta) &= \sum_{i=1}^2 \sum_{j=1}^2 \rho_{ij} \\ &= \rho_{11} + \rho_{12} + \rho_{21}(S, \beta) + \rho_{22}(S, \beta)\end{aligned}\quad (12)$$

in die Gleichung ein. Dies entspricht bei Krümmungsdifferenzbetrachtung über beide Kontaktflächen in den jeweiligen projizierten zweidimensionalen Raumrichtungen der Krümmungsdifferenz $F(\rho)$

Die beiden Beiwerte ergeben sich anhand flacher Geometrie nach:

$$\zeta(S, \beta) = \sqrt[3]{\frac{t \cdot E(t) - t^3 \cdot K(t)}{\frac{\pi}{4}(1-t^2) \cdot (1+F(\rho))}} \quad (13)$$

$$\eta(S, \beta) = \sqrt[3]{\frac{K(t) - E(t)}{\frac{\pi}{4}(1-t^2) \cdot (1+F(\rho))}} \quad (14)$$

Diese sind wie ersichtlich nur durch das Lösen der elliptischen Integral 1. und 2. Ordnung zu berechnen. Diese sind aufgrund ihrer nicht-Darstellbarkeit durch elementare Funktionen und dem damit nötigen iterativen rekursiven Lösungsverfahren nicht direkt lösbar. Sie können zum einen über Korrespondenztabelle nach Legendre bestimmt werden, welche aus der Legendre-Form derselben hergeleitet wurden. Zum anderen wurden in [9] algebraische Näherungsfunktionen welche ohne Iterationen zu einer Näherungslösung führen. Hier sollen die elliptischen Integrale aber direkt über den Explizitätsratio t gelöst werden.

$$K(t) = \int_0^{\pi/2} \frac{1}{\sqrt{1 - (1-t^2) \sin^2 \phi}} dx \quad (15)$$

$$E(t) = \int_0^{\pi/2} \sqrt{1 - (1-t^2) \sin^2 \phi} dx \quad (16)$$

Mit dem Explizitätsratio:

$$t(S, \beta) \equiv \frac{c(S, \beta)}{d(S, \beta)} = \frac{\zeta(S, \beta)}{\eta(S, \beta)} \leq 1 \quad (17)$$

Da dieser, wie nun ersichtlich ist, aus den eigentlich erst noch zu berechnenden Beiwerten bzw. Halbachsen gebildet werden muss, ist eine explizite Berechnung nicht möglich. Die Gleichungen müssen numerisch iterativ gelöst werden. Dazu ist es notwendig eine Abbruchbedingung zu definieren. Die folgende Abbildung 7 zeigt den Iterativen Prozess.

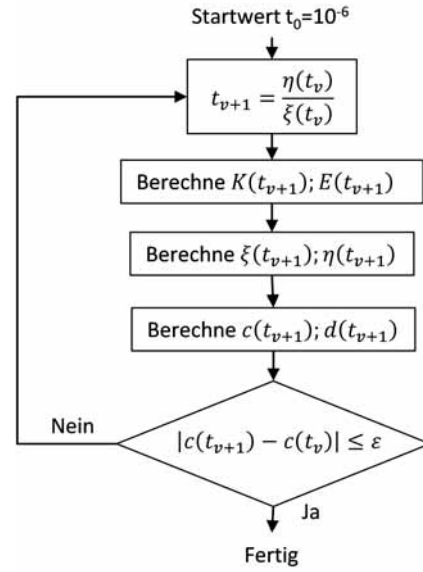


Abbildung 7: Flussdiagramm (PAP) der Berechnungsmethode

Mit Hilfe der nun vorliegenden Krümmungswerte kann die Druckverteilung im inneren der Kontaktflächen berechnet werden.

$$p_h = p_m(S, \beta) \sqrt{1 - \left(\frac{x}{d(S, \beta)}\right)^2 - \left(\frac{y}{c(S, \beta)}\right)^2} \quad (18)$$

Der Maximale Druck, der materialschädigend wirken kann sollte er den Grenzwert überschreiten, ergibt sich immer im Achsenmittelpunkt der Ellipse bei $x = y = 0$ und errechnet sich zu:

$$\begin{aligned}p_m(S, \beta) &= \frac{3 \cdot F_n}{2\pi \cdot c(S, \beta) \cdot d(S, \beta)} \\ &= \frac{3 \cdot F_n}{2\pi \cdot \zeta(S, \beta) \cdot \eta(S, \beta) \cdot \left(\frac{3 \cdot F_n \cdot W_{ers}}{\sum \rho(S, \beta)}\right)^{2/3}} \\ &= \frac{F_n}{\pi \cdot \zeta(S, \beta) \cdot \eta(S, \beta)} \sqrt{\left(\frac{3 \cdot F_n \cdot \sum \rho(S, \beta)^2}{8 \cdot W_{ers}^2}\right)}\end{aligned}\quad (19)$$

Aus diesen Erkenntnissen lassen sich nun Aussagen über die Reibung zwischen der Paarung treffen.

2.4 Herleitung der Kraftübertragungsverhältnisse

Nach den allgemeinen Untersuchungen der Materialwissenschaften zum Reibverhalten von Stoffen

wie in [10] besteht ein nicht-linearer Zusammenhang zwischen der untersuchten Hertz'schen Pressung im Kontaktpunkten des Reibpaares, der Relativgeschwindigkeit der zueinander bewegten Flächen und einem resultierenden Reibkoeffizienten. Nachfolgend ist dieser Zusammenhang wie gefunden in 8 dargestellt.

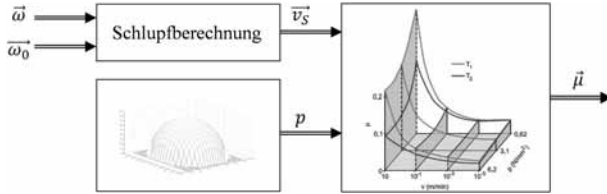


Abbildung 8: Reibkoeffizient aus Pressung und Relativgeschwindigkeit

Es ist nun möglich über eine Näherungsfunktion oder eine Look-Up Tabelle diesen Zusammenhang sowohl im Modell als auch als Inverse in einer Reibkompensation den Reibungseinfluss zu verarbeiten.

$$\underline{\mu} = f(v_s(\omega, \omega_0), p_m(S, \beta)) \quad (20)$$

Weiterhin kann dies auch zur Optimierung genutzt werden um die Konstruktive Gestaltung des Mecanum-Rades unter Berücksichtigung der Kugelgeometrie zu optimieren. Das hier verfolgte Optimierungsziel ist die Nicht-Linearität zu linearisieren und somit den Berechnungsaufwand der Reibkompensation zu senken und gleichzeitig dessen Gültigkeitsbereich auszuweiten ohne einen Verlust an Genauigkeit hinnehmen zu müssen. Im Kapitel 4 ist das Optimierungsergebnis dargestellt.

3 Integration zu Dynamikmodell

Auf Basis der Kinematik und Dynamik des jeweiligen zu modellierenden System sollen die Dynamikfunktionen in Form von verkoppelten, nicht-linearen Differentialgleichungen hergeleitet werden. Ein Bereits in [11] für das S-Mobile auf Basis des Euler-Lagrange Ansatzes hergeleitetes Dynamikmodell wurde in die allgemeine Matrixschreibweise überführt und für die Modellierung der Kraftübertragungsverhältnisse nach dem hier gezeigten Ansatz erweitert. Die nachfolgende Gleichung zeigt das darauf basierende, verallgemein-

erte Dynamikmodell mit Reibmodellierung zur Berücksichtigung der Kraftübertragung.

$$\underline{\underline{M}}(\underline{q}) \cdot \underline{\dot{q}} + \underline{C}(\underline{q}, \underline{\dot{q}}) \cdot \underline{\dot{q}} + \underline{G}(\underline{q}) = \underline{Q}(\underline{q}) \cdot \underline{u} \quad (21)$$

Der Eingangsvektor \underline{u} wird über eine Minimalbedingung jedes einzelnen Vektorelements i definiert, sodass entweder die volle Antriebskraft \underline{u} oder aber die reduzierte, maximal übertragbare Kraft $\underline{\tau}$ in das Modell weitergeleitet wird.

$$\forall (u_i, \tau_i) \in (\underline{u} | \underline{\tau}) : \quad \underline{u} = \min_{i=1 \dots n} [u_i, \tau_i] = \begin{cases} \underline{u}(i) & u_i \leq \tau_i \\ \underline{\tau}(i) & u_i > \tau_i \end{cases} \quad (22)$$

Der Kraftübertragungsvektor $\underline{\tau}$ bildet sich aus den Anpresskräften \underline{F}_N und dem Reibbeiwertsvektor $\underline{\mu}$. Es soll hierbei beachtet werden, dass durch mechanische Vorspannung der Räder die Anpresskräfte einstellbar und unabhängig von der Kinematik des Systems sein können.

$$\underline{\tau} = \underline{F}_N \cdot \underline{\mu} \quad (23)$$

Die folgenden Abbildung 9 zeigt die allgemeine Struktur des Dynamikmodells der Strecke. Dieses Modell besitzt die Eingänge:

- $\underline{u} \in \mathbb{R}^n$ Krafteingangsvektor
- $\underline{p} \in \mathbb{R}^n$ Vektor der Pressungen

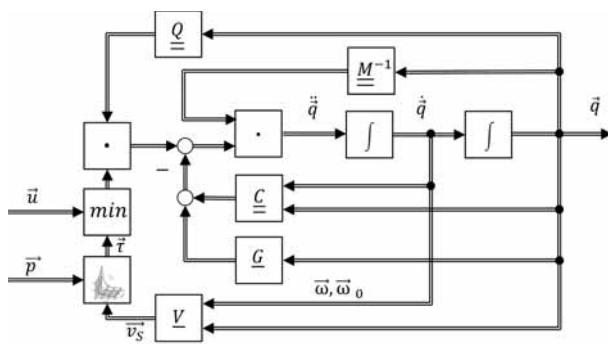
die Zustandsvektoren:

- $\underline{q} \in \mathbb{R}^m$ Zustandsvektor
- $\underline{v}_s \in \mathbb{R}^n$ Schlupfvektor
- $\underline{\tau} \in \mathbb{R}^n$ Kraftübertragungsvektor

und die charakterisierenden Matrizen und Vektoren:

- $\underline{\underline{M}} \in \mathbb{R}^{m \times m}$ Massenmatrix
- $\underline{C} \in \mathbb{R}^{m \times m}$ Coriolis- oder Zentrifugalkraftmatrix
- $\underline{G} \in \mathbb{R}^{m \times m}$ Gravitationsmatrix
- $\underline{V} \in \mathbb{R}^{m \times m}$ Matrize der Geschwindigkeitsvektoren
- $\underline{Q} \in \mathbb{R}^{m \times n}$ Eingangsmatrix der eingepprägten Kräfte

Durch den neu geschaffenen Eingang über den die Pressung am Kontaktpunkt in das Modell extern eingeleitet und den Zugang über die Matrize der



Geschwindigkeitsvektoren mit der dieser Eingang am Modell angekoppelt wird, ist es nun möglich online eine veränderliche Reibung zu modellieren und simulieren.

4 Validierung

Abschließend sollen die Simulationsergebnisse analysiert und validiert werden. Dazu werden zunächst die geometrischen Oberflächeneigenschaften auf Plausibilität überprüft. In der Abbildung 10 sind die in

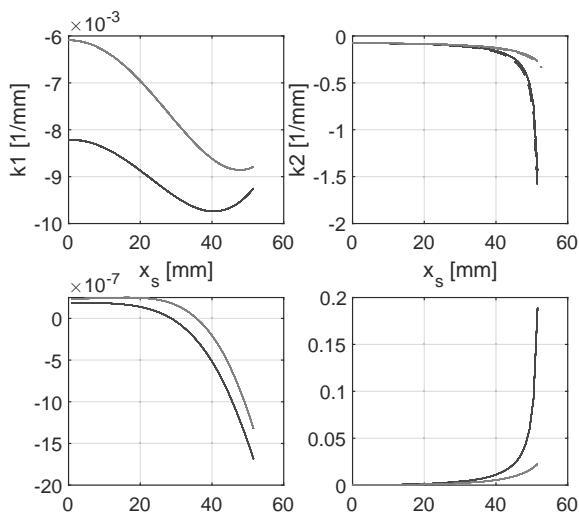


Abbildung 10: Krümmungsverläufe der Oberflächen der Mecanum- Räder an S-Mobile und AGV

Kapitel 2.2, auf Basis der in [4] hergeleiteten Verläufe der Oberflächenkontur, gefundenen Krümmungsverläufe dargestellt. der Blaue Verlauf bezieht sich auf die Radtonne des AGV, der rote Verlauf auf die Radtonne

des S-Mobile. Es sind jeweils die Krümmungsverläufe in die jeweiligen Achsenrichtungen des in 3 dargestellten Koordinatensystems als k_1 und k_2 über x_s verlaufend dargestellt. Dabei sind die durchgezogenen Verläufe mittels der differentialgeometrischer Methode (Eigenwerte der Weingartenabbildung) und die gestrichelten Verläufe mit der Kreissegmentnäherung errechnet worden. Jeweils darunter ist die Abweichung beider Methoden zueinander dargestellt. Es ist ersichtlich, dass diese in einem akzeptablen Bereich liegen. Da k_1 über den Umfang der Tonne in y_b Koordinatenrichtung liegt, nähert die Kreisnäherung entsprechend sehr gut an, die Abweichungen sind prozentual gering. Über k_2 ist zu sehen, dass eine größere prozentuale Abweichung vorhanden ist, die Näherung ist nur beschränkt gültig.

5 Zusammenfassung und Fazit

Im vorliegenden Beitrag wurden zunächst geometrische Untersuchungen zum Abrollverhalten von Mecanum-Rädern am Beispiel der beiden Forschungsträger AGV und S-Mobile durchgeführt. Auf Basis dieser wurden die Kraftübertragungsverhältnisse hergeleitet und gezeigt wie das Reibverhalten in ein mehrdimensionales, nicht-lineares mathematisches Modell eingebaut werden kann. Es wurden gezeigt das der hergeleitete Ansatz mit einem üblichen Näherungsverfahren approximiert werden konnte und somit Gültigkeit hat. Das aufgezeigte Verfahren kann als allgemeingültige Methode zur Auslegung von Mecanum-Rädern jeder Art gewertet werden. Weiterhin ist es als Grundlage allgemeiner Untersuchungen von Reibpaarungen und deren Integration in mathematische Simulationsmodelle geeignet.

Acknowledgement

Der vorliegende Beitrag wurde im Rahmen des Verbundprojektes MiMec als Teil des Verbundprojekts Synus unter dem Förderkennzeichen ZW6-85012454 gefördert. Die Verantwortung für den Inhalt liegt bei den Autoren. Für die Förderung bedanken sich diese herzlichst.



References

- [1] Ilon BE. Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base. 13.11.1972.
- [2] Connette C. *Kinematische Modellierung und Regelung omnidirektionaler, nicht-holonomer Fahrwerke*. Zugl.: Stuttgart, universität stuttgart, diss., 2013, Universitätsbibliothek der Universität Stuttgart, Stuttgart. 2013.
- [3] Siegwart R, Nourbakhsh IR. *Introduction to autonomous mobile robots*. Intelligent robotics and autonomous agents. Cambridge, Mass: MIT Press. 2004.
- [4] Göllner M, Liu-Henke X. Mathematical derivation of the geometry of a Mecanum-wheel for a size exact roll off on a spherical surface. In: *Mechatronic systems and materials 2014*, Herausgegeben durch Pawliczek R, Robak G. Opole: Opole University of Technology Faculty of Mechanical Engineering Department of Mechanics and Machine Design. 2015;.
- [5] Harris TA, Kotzalas MN. *Rolling bearing analysis*. Boca Raton, Fla.: CRC, Taylor & Francis, 5th Ausg. 2007.
- [6] Boresi AP, Schmidt RJ. *Advanced mechanics of materials*. New York, NY: Wiley, 6th Ausg. 2003.
- [7] Eschenburg JH, Jost J. *Differentialgeometrie und Minimalflächen*. Springer-Lehrbuch Masterclass. Berlin: Springer Spektrum, 3rd Ausg. 2014.
- [8] Hertz H. Über die Berührung fester elastischer Körper. *Journal für die reine und angewandte Mathematik*. 1881;(92):156–171.
- [9] Brewe DE, Hamrock BJ. Simplified solution for point contact deformation between two elastic solids. 1976;.
- [10] Popov VL. *Kontaktmechanik und Reibung: Ein Lehr- und Anwendungsbuch von der Nanotribologie bis zur numerischen Simulation*. Berlin, Heidelberg: Springer Berlin Heidelberg. 2009.
- [11] Liu-Henke X, Göllner M, Tao H. An intelligent control structure for highly dynamic driving of a spherical electrical drive. In: *2017 Twelfth International Conference on Ecological Vehicles and Renewable Energies (EVER)*. Piscataway, NJ: IEEE. 2017; pp. 1–10.

Modellreduktion für hochviskose, nicht-isotherme Fluide mit freier Oberfläche

Edmond Skeli¹, Dirk Weidemann¹, Klaus Panreck¹,

¹Institut für Systemdynamik und Mechatronik, FH Bielefeld, Interaktion 1, 33619 Bielefeld;
{edmond.skeli,dirk.weidemann,klaus.panreck}@fh-bielefeld.de

Kurzfassung. Mit dem Ziel effiziente Steuerungs-, Regelungs- und/oder Diagnoseverfahren nutzen zu können, greift man auch in der verfahrenstechnischen Industrie vermehrt auf mathematische Modelle, die die zugrundeliegenden physikalischen Prozesse beschreiben, zurück. Hierfür ist es notwendig, mathematische Modelle zu bestimmen, die einerseits hinreichend präzise sind, andererseits aber einen nicht zu hohen Rechenaufwand erfordern. Vor diesem Hintergrund wird die Reduzierung eines Modells, welches das Verhalten eines hochviskosen, nicht-isothermen Fluids mit einer freien Oberfläche beschreibt, erörtert. Das Verhalten des Fluids genügt einem Modell, das aus einem System partieller Differentialgleichungen besteht und neben den zweidimensionalen Navier-Stokes Gleichungen auch die thermische Energiegleichung, die das Temperaturverhalten beschreibt, umfasst. Mit Hilfe der Störungstheorie kann gezeigt werden, dass das Verhalten der Geschwindigkeit und der Temperatur des Fluids durch zwei einfachere Teilmodelle beschrieben werden kann. Das erste Teilmodell dient zur Berechnung der Strömungsdynamik, während das zweite Teilmodell die Berechnung des thermischen Verhaltens ermöglicht.

Einleitung

In vorliegenden Beitrag wird das Einlaufen eines hochviskosen, nicht-isothermen Fluids in den Spalt zwischen zwei gegenläufig rotierenden Zylindern betrachtet. Hierbei gilt es zu berücksichtigen, dass zum einen der Spalt zu Beginn leer ist und sich erst im Laufe der Zeit mit dem Fluid füllt, sodass das Einlaufen des Fluids als instationäres Prozessverhalten zu beschreiben ist. Zum anderen entsteht während des Füllens vor dem Spalte ein Wulst, dessen Größe sich über der Zeit solange verändert, bis ein stationärer Arbeitspunkt erreicht ist. Da die zeitliche Entwicklung der Wulst und damit die zeitliche Veränderung der Fluidgrenze a-priori unbekannt ist, bedarf es hinsichtlich der Bestimmung des Fluidverhaltens nicht nur der numerischen Lösung eines Systems von partiellen Differentialgleichungen, das aus den inkompressiblen Navier-Stokes Gleichungen und der thermischen Energiegleichung besteht, son-

dern auch der Berechnung der Fluidgrenze. Im Weiteren wird die Fluidgrenze, d.h. die an die umgebende Luft angrenzende Oberfläche des Fluids auch als freie Oberfläche bezeichnet.

Einen geeigneten Ansatz zur Berechnung der instationären, inkompressiblen Navier-Stokes-Gleichungen mit freier Oberfläche bietet die Marker- und Cell (MAC)-Methode, die von Harlow und Welch in [1] eingeführt wurde. Amsden und Harlow vereinfachten die MAC-Methode in [2], indem sie die Geschwindigkeits- und Druckberechnungen entkoppelten. Darüber hinaus wird in [3, 4] die MAC-Methode für drei räumliche Dimensionen angepasst. Auf der Grundlage der MAC-Methode wird in [5] ein Ansatz zur Bestimmung der freien Oberfläche eines hochviskosen, nicht-isothermen Fluids, das in den Spalt zwischen zwei gegensinnig rotierenden Zylindern eintritt, vorgeschlagen. Dieser Ansatz erlaubt zwar die Simulation des instationären Verhaltens des Fluids, hat allerdings den Nachteil, dass er einen hohen Rechenaufwand erfordert, sodass der Ansatz nicht für modellbasierte Steuerungs-, Regelungs- und/oder Diagnosezwecke verwendet werden kann.

Um den Rechenaufwand zu verringern, erweist es sich als sinnvoll, das mathematische Modell geeignet zu reduzieren. Hinsichtlich der Charakterisierung des zeitlichen Verhaltens werden daher die partiellen Differentialgleichungen normiert, sodass die einzelnen Zeitkonstanten bestimmt werden können. Diese Zeitkonstanten erlauben nach [6, 7] eine qualitative Beurteilung des transienten Verhaltens der physikalischen Größen, d.h. der Geschwindigkeiten und der Temperatur. Mit Hilfe der Störungstheorie (vgl. [8, 9]) lässt sich zeigen, dass sich die Geschwindigkeiten und die Temperatur des Fluids auf verschiedenen Zeitskalen entwickeln, was darauf hinweist, dass zwei reduzierte Modelle, d.h. ein schnelles und ein langsames Teilmodell verwendet werden können. Mit Hilfe des schnellen Teilmodells erfolgt die Berechnung der Geschwindigkeiten des Fluids, wobei davon ausgegangen wird, dass sich die Fluidtem-

peratur während dieser Berechnung nicht verändert. Demgegenüber wird mit dem langsamen Teilmodell die Temperatur berechnet.

Im Weiteren werden zunächst das mathematische Fluidmodell in normalisierter Form sowie die schnellen und langsamen Teilmodelle, die mit Hilfe der Störungstheorie bestimmt werden, eingeführt. Im Anschluss erfolgt eine kurze Darstellung der räumliche Diskretisierung der Modellgleichungen und der Marker- und Cell (MAC)-Methode zur Bestimmung der freien Oberfläche. Abschließend werden die numerischen Simulationsergebnisse, die mit Hilfe des reduzierten Modells berechnet worden sind, präsentiert und mit den Ergebnissen des Vollmodells (s. [5]) verglichen.

1 Normalisierung der Modellgleichungen

Das zeit- und örtliche Verhalten der Geschwindigkeiten und des Drucks eines hochviskosen, nicht-isothermen Fluids genügt den inkompressiblen Navier-Stokes Gleichungen

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\eta}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (1)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\eta}{\rho} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right), \quad (2)$$

$$0 = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (3)$$

mit den Anfangs- und Randbedingungen

$$\begin{aligned} \mathbf{u}(\zeta, 0) &= \mathbf{u}_0(\zeta) & \forall \zeta \in \Gamma, \\ \mathbf{u}(\zeta, t) &= \mathbf{h}(\zeta, t) & \forall (\zeta, t) \in \partial\Gamma \times [0, t_e], \end{aligned}$$

wobei u, v Geschwindigkeiten in x -, y -Richtung repräsentieren. Im Weiteren kennzeichnet $\mathbf{u} = (u, v)^T : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}^2$ den Vektor der Fluidgeschwindigkeiten, $\mathbf{u}_0(\zeta) \in \mathbb{R}^2$ die Anfangsbedingungen und $\mathbf{h} : \partial\Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}^2$ die Randbedingungen, wobei $\Gamma \subset \mathbb{R}^2$ die Definitionsmenge und $\partial\Gamma$ den Rand der Definitionsmenge darstellt. Ferner ist $p : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}$ der Druck, $\rho \in \mathbb{R}$ die Dichte und $\eta(T) \in \mathbb{R}$ die Viskosität.

Des Weiteren genügt die zeit- und örtliche Entwicklung der Fluidtemperatur der partiellen DGL

$$\rho C_p \left(\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} \right) = \lambda \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right)$$

$$+ 2\eta \left(\frac{\partial u}{\partial x} \right)^2 + \eta \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + 2\eta \left(\frac{\partial v}{\partial y} \right)^2, \quad (4)$$

wobei $T : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}$ die Temperatur darstellt und $C_p, \lambda \in \mathbb{R}$ spezifische Wärmekapazität und Wärmeleitfähigkeit sind. Die zugehörigen Anfangs- und Randbedingungen seien durch

$$\begin{aligned} T(\zeta, 0) &= T_0(\zeta) & \forall \zeta \in \Gamma, \\ T(\zeta, t) &= d(\zeta, t) & \forall (\zeta, t) \in \partial\Gamma \times [0, t_e] \end{aligned}$$

mit $T_0(\zeta) \in \mathbb{R}$ gegeben.

Normalisierung der Variablen ergibt

$$\begin{aligned} u_n &= \frac{u}{\bar{u}}, & v_n &= \frac{v}{\bar{u}}, & \Pi_{px} &= \frac{h^2}{\bar{u}\eta} \frac{\partial p}{\partial x}, \\ \Pi_{py} &= \frac{h^2}{\bar{u}\eta} \frac{\partial p}{\partial y}, & x_n &= \frac{x}{L}, & y_n &= \frac{y}{h}, \\ t_n &= \frac{t}{\bar{t}}, & T_n &= \frac{T}{\bar{T}}, & \eta_n &= \frac{\eta}{\bar{\eta}}, \end{aligned} \quad (5)$$

wobei die Größen $u_n, v_n, \Pi_{nx}, \Pi_{ny}, x_n, y_n$ die normalisierten Geschwindigkeiten, Druckgradienten, Koordinaten und t_n, T_n, η_n die normalisierte Zeit, Temperatur und Viskosität repräsentieren.

Unter Verwendung der normalisierten Variablen aus (5) lässt sich das System aus partiellen Differentialgleichung, welches sowohl die Navier-Stokes Gleichungen als auch die thermische Energiegleichung umfasst, in die Form

$$\frac{t_\eta}{\bar{t}} \frac{\partial u_n}{\partial t_n} = h_1(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (6)$$

$$\frac{t_\eta}{\bar{t}} \frac{\partial v_n}{\partial t_n} = h_2(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (7)$$

$$0 = h_\nabla(\mathbf{r}_n, \mathbf{u}_n), \quad (8)$$

$$\frac{\tau_\lambda}{\bar{t}} \frac{\partial T_n}{\partial t_n} = h_3(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) \quad (9)$$

mit $\mathbf{r}_n = [x_n, y_n]^T, \mathbf{u}_n = [u_n, v_n]^T$ und

$$\begin{aligned} h_1(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) &= -Re \left(\frac{h}{L} u_n \frac{\partial u_n}{\partial x_n} + v_n \frac{\partial u_n}{\partial y_n} \right) \\ &\quad - \Pi_{px} + \eta_n \left(\left(\frac{h}{L} \right)^2 \frac{\partial^2 u_n}{\partial x_n^2} + \frac{\partial^2 u_n}{\partial y_n^2} \right), \end{aligned}$$

$$h_2(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) = -Re \left(\frac{h}{L} u_n \frac{\partial v_n}{\partial x_n} + v_n \frac{\partial v_n}{\partial y_n} \right)$$

$$-\Pi_{py} + \eta_n \left(\left(\frac{h}{L} \right)^2 \frac{\partial^2 v_n}{\partial x_n^2} + \frac{\partial^2 v_n}{\partial y_n^2} \right)$$

$$h_{\nabla}(\mathbf{r}_n, \mathbf{u}_n) = \frac{\bar{u}}{h} \left(\frac{h}{L} \frac{\partial u_n}{\partial x_n} + \frac{\partial v_n}{\partial y_n} \right),$$

$$\begin{aligned} h_3(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) = & -Gr \left(u_n \frac{\partial T_n}{\partial x_n} + v_n \frac{L}{h} \frac{\partial T_n}{\partial y_n} \right) \\ & + \left(\left(\frac{h}{L} \right)^2 \frac{\partial^2 T_n}{\partial x_n^2} + \frac{\partial^2 T_n}{\partial y_n^2} \right) + 2\eta_n Br \left(\frac{h}{L} \right)^2 \left(\frac{\partial u_n}{\partial x_n} \right)^2 \\ & + \eta_n Br \left(\frac{\partial u_n}{\partial y_n} + \frac{h}{L} \frac{\partial v_n}{\partial x_n} \right)^2 + 2\eta_n Br \left(\frac{\partial v_n}{\partial y_n} \right)^2 \end{aligned}$$

überführen. Hierbei stellen

$$Re = \frac{\rho h \bar{u}}{\bar{\eta}}, \quad Br = \frac{\bar{\eta} \bar{u}^2}{\lambda \bar{T}}, \quad Gr = \frac{\bar{u} h^2}{a L} \quad (10)$$

die Reynolds-, Brinkmann-, and Graetz-Zahlen und

$$t_\eta = \frac{\rho h^2}{\bar{\eta}}, \quad \tau_\lambda = \frac{h^2 \rho C_p}{\lambda} \quad (11)$$

die viskose Relaxionszeit bzw. die konduktive thermische Ausgleichszeit dar.

2 Modellreduktion

Im Folgenden wird das Verhältnis der beiden in (11) gegebenen Zeitkonstanten näher betrachtet, da diese Aufschluss über das transiente Verhalten der Fluidgeschwindigkeiten und -temperatur geben. Ein Vergleich der beiden Zeitkonstanten

$$\frac{t_\eta}{\tau_\lambda} = \frac{\frac{\rho h^2}{\bar{\eta}}}{\frac{h^2 \rho C_p}{\lambda}} = \frac{\lambda}{\bar{\eta} C_p} \quad (12)$$

zeigt, dass in Folge der hohen Viskosität

$$\tau_\lambda \gg t_\eta \quad (13)$$

gilt. Physikalisch gesehen zeigt (13), dass das transiente Verhalten der Geschwindigkeiten sehr viel schneller ist als das der Temperatur.

Da entweder $\bar{t} = t_\eta$ oder $\bar{t} = \tau_\lambda$ als Normalisierungskonstante verwendet werden kann, wird in den beiden folgenden Unterabschnitten diskutiert, welchen Effekt die Wahl von $\bar{t} = t_\eta$ bzw. $\bar{t} = \tau_\lambda$ auf die Gleichungen (6)-(9) hat.

2.1 Viskose Relaxionszeit als Normalisierungskonstante

Wählt man t_η als Normalisierungskonstante, folgt für die Faktoren auf der linken Seite der Gleichungen (6), (7) und (9) unmittelbar $t_\eta/\bar{t} = t_\eta/t_\eta = 1$ und $\tau_\lambda/\bar{t} = \tau_\lambda/t_\eta$, sodass die normalisierten Navier-Stokes Gleichungen die Form

$$\frac{\partial u_n}{\partial t_n} = h_1(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (14)$$

$$\frac{\partial v_n}{\partial t_n} = h_2(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (15)$$

$$0 = h_{\nabla}(\mathbf{r}_n, \mathbf{u}_n) \quad (16)$$

annehmen. Wählt man darüber hinaus $t_\eta/\tau_\lambda = \varepsilon$ mit $\varepsilon \ll 1$ als Störungsparameter und multipliziert die thermische Energiegleichung (9) mit ε ergibt sich

$$\frac{\partial T_n}{\partial t_n} = \varepsilon h_3(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) = 0. \quad (17)$$

Das durch die Gleichungen (14)-(16) beschriebene Modell wird als schnelles Teilmodell bezeichnet. Es wird zur Berechnung der Geschwindigkeiten bei Impulsänderungen verwendet. Da sich die Temperatur über t_η nicht signifikant ändert, muss die partielle Differentialgleichung (9) nicht gelöst werden. Vielmehr kann die Temperatur, wie in (17) angegeben, als konstante Größe angenommen werden.

2.2 Konduktive thermische Ausgleichszeit als Normalisierungskonstante

Im Unterschied zu der in Abschnitt 2.1 beschriebenen Vorgehensweise führt die Wahl von τ_λ als Normalisierungskonstante zu einem System

$$\varepsilon \frac{\partial u_n}{\partial t_n} = h_1(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (18)$$

$$\varepsilon \frac{\partial v_n}{\partial t_n} = h_2(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (19)$$

$$0 = h_{\nabla}(\mathbf{r}_n, \mathbf{u}_n), \quad (20)$$

$$\frac{\partial T_n}{\partial t_n} = h_3(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n) \quad (21)$$

von singular gestörten partiellen Differentialgleichungen, wobei ε wie oben beschrieben definiert ist. Nimmt man an, dass $\varepsilon \rightarrow 0$ gilt, gehen die instationären Navier-

Stokes Gleichungen in die stationäre Form

$$0 = h_1(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (22)$$

$$0 = h_2(t_n, \mathbf{r}_n, \mathbf{u}_n, p, T_n), \quad (23)$$

$$0 = h_\nabla(\mathbf{r}_n, \mathbf{u}_n). \quad (24)$$

über.

Das durch (21)-(24) gegebene Modell wird als langsames Teilmodell bezeichnet und dient der Berechnung der Fluidtemperatur. Während dieser Berechnung sind die Geschwindigkeiten nicht als differentielle, sondern vielmehr als algebraische Zustände zu interpretieren, die so angepasst werden müssen, dass die algebraischen Bedingungen (22)-(24) erfüllt sind.

2.3 Hybrides Modell

Im Gegensatz zum Ansatz in [5], bei dem das Verhalten des in den Spalt einlaufenden Fluids mit Hilfe des vollständigen Modells, d.h. mit Hilfe des aus den partiellen Differentialgleichungen (1)-(4) bestehenden Systems bestimmt wird, basiert der in diesem Beitrag vorgestellte Ansatz darauf, das Verhalten mit Hilfe der reduzierten Modellgleichungen (14)-(16) und (21)-(24) zu berechnen. Es zeigt sich, dass sich die Rechenzeit durch Verwendung der reduzierten Modelle signifikant reduzieren lässt.

Solange keine Impulsänderungen auftritt, erfolgt die Nutzung des langsamen Teilmodells. Ein Impulsänderung tritt auf, wenn das Fluid in eine Zelle des Diskretisierungsgitters (vgl. Abschnitt 3) eintritt, die zum einen noch nicht mit Fluid gefüllt war und zum anderen mit einem der beiden Zylinder in Kontakt steht. Unmittelbar nach dem Erkennen einer Impulsänderung wird das schnelle Teilmodell verwendet. Das schnelle Teilmodell wird so lange genutzt, bis sich die Anzahl der belegten Zellen nicht verändert und die Geschwindigkeiten des Fluids in diesen Zellen stationär sind. Interpretiert man eine Impulsänderung als Ereignis e_{im} und das Auftreten der lokal stationären Geschwindigkeiten als Ereignis e_{st} , lässt sich das Verhalten des Fluids durch den in Abb. 1 dargestellten hybriden Automaten modellieren.

3 Ortsdiskretisierung

Hinsichtlich des numerischen Lösens der partiellen Differentialgleichungen ist eine geeignete räumliche Diskretisierung erforderlich. Wie in Abb. 2 dargestellt, werden die Geschwindigkeiten in der Mitte der verti-

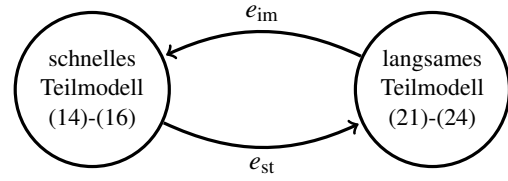


Abb. 1: Reduziertes Modell als hybrider Automat

kalen und horizontalen Kanten des Diskretisierungsgitters berechnet. Im Gegensatz dazu werden sowohl der Druck als auch die Temperatur in der Zellmitte des Gitters berechnet. Eine derartiges Gitter wird auch als *staggered grid* bezeichnet.

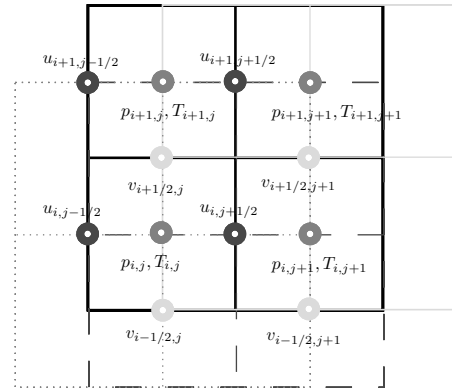


Abb. 2: Diskretisierungsgitter mit Berechnungsknoten

Die partiellen Ableitungen erster Ordnung können entsprechend

$$D^x f_{i,j} = \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta x}, \quad D_-^x f_{i,j} = \frac{f_{i,j} - f_{i,j-1}}{\Delta x},$$

$$D_+^x f_{i,j} = \frac{f_{i,j+1} - f_{i,j}}{\Delta x}$$

durch Vorwärts-, Rückwärts- und Zentrale-Differenzenquotienten und partielle Ableitungen zweiter Ordnung durch

$$K^x f_{i,j} = \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{(\Delta x)^2} \quad (25)$$

mit $i = 1, 2, \dots, n_1$ und $j = 1, 2, \dots, n_2$ approximiert werden, wobei f wahlweise u , v , p oder T repräsentiert, $n_1, n_2 \in \mathbb{R}$ die Anzahl der Diskretisierungspunkte in x - und y -Richtung darstellen und Δx die Schrittweite der Diskretisierung in x -Richtung. Die Operati-

on D^y, D_-^y, D_+^y und K^y sind in analoger Weise definiert, wobei $\Delta y = f_{\Delta y}(x)$ die variable Diskretisierungsschrittweite in y -Richtung ist, vgl. [5]. Es sei angemerkt, dass in diesem und in Abschnitt 5 infolge der Ortsdiskretisierung $\mathbf{u} : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}^{2(n_1+n_2)}$, $p : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}^{(n_1+n_2)}$ und $T : \Gamma \times \mathbb{R}_{(+)} \rightarrow \mathbb{R}^{(n_1+n_2)}$ gilt.

Die Diskretisierung überführt das schnelle Teilsystem (14)-(16) in die Form

$$I\dot{\mathbf{u}}(t) = K(\mathbf{u})\mathbf{u}(t) - Bp(t) + \mathbf{f}(\mathbf{u}(t), p(t)), \quad (26)$$

$$0 = B^T \mathbf{u}(t) \quad (27)$$

und das langsame Teilsystem (21)-(24) in die Form

$$0 = K(\mathbf{u})\mathbf{u}(t) - Bp(t) + \mathbf{f}(\mathbf{u}(t), p(t)), \quad (28)$$

$$0 = B^T \mathbf{u}(t), \quad (29)$$

$$I\dot{T}(t) = K_T(\mathbf{u})T(t) + D(\mathbf{u}) + g(T(t)) \quad (30)$$

mit $B = [D_+^x, D_+^y]^T$. Hierbei stellt I die Einheitsmatrix dar und

$$K(\mathbf{u}) = \begin{bmatrix} K_1 + N_1(\mathbf{u}) & 0 \\ 0 & K_2 + N_1(\mathbf{u}) \end{bmatrix}$$

mit $K_1 = K_2 = K^x + K^y$ repräsentiert den linearen Diffusionsterm und

$$N(\mathbf{u}) = \begin{bmatrix} N_1(\mathbf{u}) \\ N_2(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} u_{i,j+1/2} D^x + v_{i,j+1/2}^* D^y \\ u_{i+1/2,j}^* D^x + v_{i+1/2,j} D^y \end{bmatrix}$$

den nichtlinearen Konvektionsterm mit

$$u_{i+1/2,j}^* = \frac{1}{4} (u_{i,j-1/2} + u_{i,j+1/2} + u_{i+1,j+1/2} + u_{i+1,j-1/2}),$$

$$v_{i,j+1/2}^* = \frac{1}{4} (v_{i-1/2,j} + v_{i+1/2,j} + v_{i+1/2,j+1} + v_{i-1/2,j+1}).$$

Es gilt zu berücksichtigen, dass $\mathbf{f}(\mathbf{u}(t), p(t))$ und $g(T(t))$ von den zeitveränderlichen Randbedingungen abhängen, sodass diese Funktionen entsprechend der freien Oberfläche (vgl. Abschnitt 4) anzupassen sind. Ferner sind die Operatoren für die Berechnung der Temperatur durch

$$D(\mathbf{u}) = 2\eta((D_+^x u)^2 + \frac{1}{\eta}(D_+^y u + D_+^x v)^2 + (D_+^y v)^2).$$

und $K_T(\mathbf{u}) = (K_1 + N_T(\mathbf{u}))$ mit

$$N_T(\mathbf{u}) = \frac{u_{i,j-1/2} + u_{i,j+1/2}}{2} D^x + \frac{v_{i-1/2,j} + v_{i+1/2,j}}{2} D^y$$

gegeben.

4 Bestimmung der freien Oberfläche

Zur Bestimmung der freien Oberfläche findet die von Harlow und Welch in [1] eingeführte MAC-Methode Verwendung. Bei der MAC-Methode werden masselose Partikel zur Markierung der Zellen des Diskretisierungsgitters genutzt, die mit dem Fluid gefüllt sind. D.h., jede Zelle des Diskretisierungsgitters, die mindestens ein masseloses Partikel enthält, ist Teil des Bereichs, der mit Fluid gefüllt ist. Vor diesem Hintergrund werden die masselosen Partikel als Marker bezeichnet. Wenn eine oder mehrere leere Zellen des Diskretisierungsgitters an eine mit Fluid gefüllte Zelle angrenzen, durchläuft die freie Oberfläche die Diskretisierungszelle. Ein beispielhafte Konfiguration findet sich in Abb. 3. Entsprechend Abb. 3 sind die obere linke, die obere rechte und die untere rechte Zelle leer, während die untere linke Zelle mit Fluid gefüllt ist. Folglich durchläuft die freie Oberfläche, die in Abb. 3 als gepunktete Linie dargestellt ist, die untere linke Zelle.

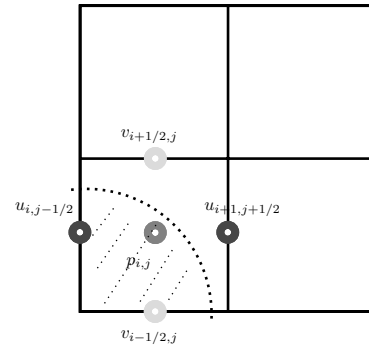


Abb. 3: Markierte Zelle mit drei freien angrenzenden Zellen

Hinsichtlich der Bestimmung der freien Oberfläche müssen die Geschwindigkeiten und der Druck berücksichtigt werden. Da bei einem inkompressiblen Fluid die Normal- und Tangentialspannungen an einer freien Oberfläche gleich Null sind (vgl. [10, 11]), müssen die Randwerte der Geschwindigkeiten und des Drucks auf

der freien Oberfläche den Gleichungen

$$\frac{p}{\rho} = 2 \frac{\eta}{\rho} \left[n_x n_x \frac{\partial u}{\partial x} + n_x n_y \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + n_y n_y \frac{\partial v}{\partial y} \right],$$

$$\begin{aligned} \left[2n_x m_x \frac{\partial u}{\partial x} + 2n_y m_y \frac{\partial v}{\partial y} \right] = \\ - (n_x m_y + n_y m_x) \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \end{aligned}$$

genügen, wobei $\mathbf{n} = (n_x, n_y)$ den Normalenvektor und $\mathbf{m} = (m_x, m_y) = (n_y, -n_x)$ den Tangentialvektor darstellt.

5 Berechnungsschema

Schnelles Teilmodell. Die Berechnung des schnellen Teilmodells (26)-(27) erfolgt unter Nutzung der Projektionsmethode von Chorin, wobei die umgesetzte Implementierung dem in [12] vorgestellten Algorithmus folgt. Man beachte, dass $T^k = T^{k-1}$ für alle Integrationszeitpunkte gilt. Der Algorithmus umfasst die folgenden Schritte:

- Zeitdiskretisierung von (26)-(27) ergibt

$$\begin{aligned} \frac{\mathbf{u}^k - \mathbf{u}^{k-1}}{\Delta t} &= K(\mathbf{u}^{k-1})\mathbf{u}^{k-1} - Bp^k + \mathbf{f}^k, \quad (31) \\ 0 &= B^T \mathbf{u}^k \end{aligned}$$

mit der Integrationsschrittweite Δt und dem aktuellen Integrationszeitpunkt k .

- Entkopplung von Druck und Geschwindigkeiten in der Impulsgleichung (31) ermöglicht das Bestimmen der Pseudogeschwindigkeiten $\tilde{\mathbf{u}}$ durch Lösen des Gleichungssystems

$$\frac{\tilde{\mathbf{u}} - \mathbf{u}^{k-1}}{\Delta t} = K(\mathbf{u}^{k-1})\mathbf{u}^{k-1} + \mathbf{f}^k.$$

- Der Druck wird durch Lösung von

$$\Delta t B^T B p^k = B^T \tilde{\mathbf{u}}$$

bestimmt und zur Korrektur der Geschwindigkeiten gemäß

$$\mathbf{u}^k = \tilde{\mathbf{u}} - \Delta t B p^k.$$

genutzt.

Langsames Teilmodell. Die Berechnung des langsamen Teilsystems (28)-(30) umfasst die folgenden Schritte:

- Zeitdiskretisierung von (30) ergibt

$$\frac{T^k - T^{k-1}}{\Delta t} = K_T(\mathbf{u}^{k-1})T(k) + D(\mathbf{u}^{k-1}) + g(T^{k-1}), \quad (32)$$

sodass die Temperatur T^k durch Lösen von (32) bestimmt werden kann.

- Entkopplung von Druck und Geschwindigkeiten ermöglicht das Bestimmen der Pseudogeschwindigkeiten $\tilde{\mathbf{u}}$ durch Lösen der stationären, diskretisierten Navier-Stokes Gleichungen

$$\mathbf{0} = K(\tilde{\mathbf{u}})\tilde{\mathbf{u}} + \mathbf{f}^k.$$

- Bestimmung des Drucks p^k durch Lösen von

$$\Delta t B^T B p^k = B^T \tilde{\mathbf{u}}.$$

- Korrektur der Geschwindigkeiten gemäß

$$\mathbf{u}^k = \tilde{\mathbf{u}} - \Delta t B p^k.$$

Bemerkung: Die Bedingungen, wann von einem Teilmodell zum anderen gewechselt werden muss, sind im Abschnitt 2.3 beschrieben.

6 Simulationsergebnisse

Obleich der komplette zeitliche Verlauf der Geschwindigkeiten, des Drucks und der Temperatur berechnet wurde, werden aus Platzgründen im Folgenden nur die Temperatur und die Geschwindigkeit in x -Richtung zum Zeitpunkt des Erreichens des stationären Zustands dargestellt. Man beachte, dass die Berechnungen bis Erreichen des stationären Zustands ca. 10h bei Verwendung des vollständigen Modells (s. [5]) und ca. 10min bei Verwendung des reduzierten Modells dauern.

Abb. 5 zeigt das mit Hilfe des vollständigen Modells berechnete Temperaturfeld des Fluids in einem Querschnitt, der sich in der Mitte der Zylinder befindet¹. Im Gegensatz dazu ist das mit dem reduzierten Modell berechnete Temperaturfeld in Abb. 6 dargestellt.

¹Diese Querschnittsfläche wird in den Abb. 4-8 einfachheitshalber als mittlere Querschnittsfläche bezeichnet.

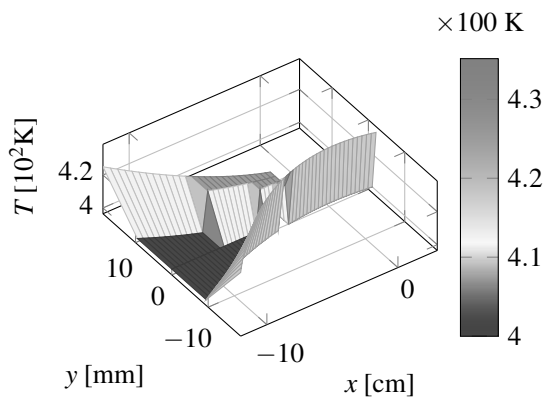


Abb. 4: Initiales Temperaturfeld in der mittleren Querschnittsfläche

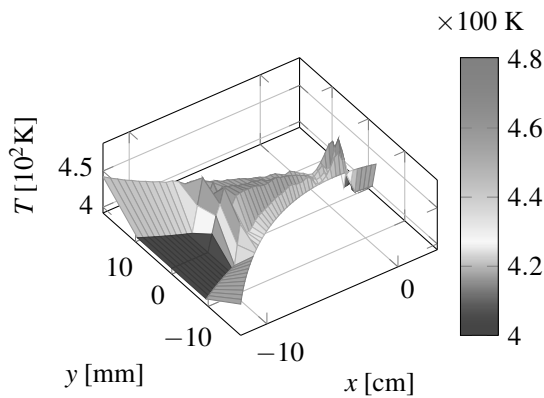


Abb. 5: Temperaturfeld in der mittleren Querschnittsfläche (vollständiges Modell)

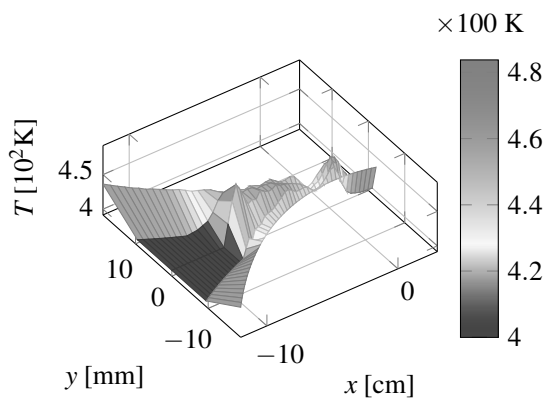


Abb. 6: Temperaturfeld in der mittleren Querschnittsfläche (reduziertes Modell)

Gemäß Abb. 4 ist die Anfangstemperatur des Fluids 425 K, während die Umgebungstemperatur 400 K und die Temperatur der Zylinder 433 K beträgt. Obwohl es Unterschiede zwischen dem mit dem Vollmodell (Abb. 5) und dem mit dem reduzierten Modell (Abb. 6) berechneten Temperaturfeld gibt, zeigt das reduzierte Modell ein qualitativ und quantitativ ähnliches Verhalten wie das Vollmodell. Vergleicht man ferner das initiale Temperaturfeld wahlweise mit Abb. 5 oder 6, lässt sich anhand der Temperaturänderung erkennen, welchen Raum das Fluid im stationären Zustand eingenommen hat.

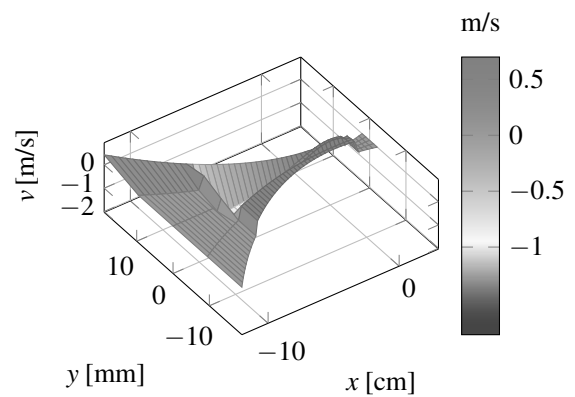


Abb. 7: Geschwindigkeitsfeld in x -Richtung in der mittleren Querschnittsfläche (vollständiges Modell)

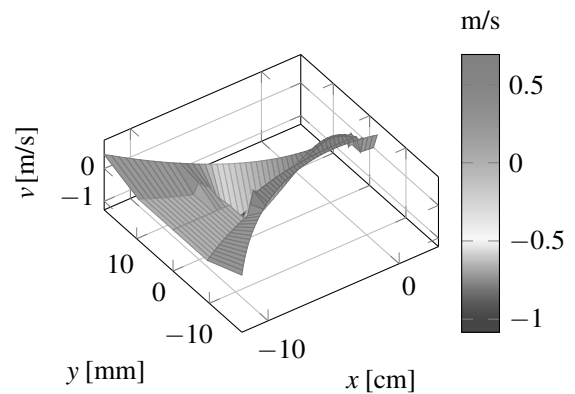


Abb. 8: Geschwindigkeitsfeld in x -Richtung in der mittleren Querschnittsfläche (reduziertes Modell)

Wie beim Temperaturfeld gibt es für die Geschwindigkeiten in der x -Richtung nur kleine Unterschiede zwischen den Ergebnissen des Vollmodells (Abb. 7)

und den mit dem reduzierten Modell (Abb. 8) berechneten Ergebnissen. Zu beachten ist, dass im Zylinderspalt negative Geschwindigkeitswerte auftreten. Die negativen Geschwindigkeiten ergeben sich aus dem hohen Druckgradienten, der beim Eintritt des Fluids in den Spalt entsteht, sodass der Druckgradient der Strömung entgegenwirkt, was zu einer Rückströmung des Fluids führt. Dies resultiert in der bereits eingangs beschriebenen Wulstbildung. In einigen Spaltabschnitten nimmt ferner der Druck kontinuierlich ab. In diesen Abschnitten wirkt der Druckgradient in positiver x -Richtung, was zu erhöhten Geschwindigkeiten führt, wie den Abb. 7 und 8 entnommen werden kann.

7 Zusammenfassung

Die Simulation von Modellen, die das Verhalten von hochviskosen, nicht-isothermen Fluiden beschreiben, ist in der Regel mit einem hohen Rechenaufwand verbunden. Daher finden derartige Modelle weder für modellbasierte Regelungs- und Steuerungsverfahren noch für die modellbasierte Diagnose Verwendung. Mit Hilfe der Störungstheorie kann jedoch gezeigt werden, dass sich die Geschwindigkeiten und die Temperatur des Fluids auf unterschiedlichen Zeitskalen entwickeln, was darauf hindeutet, dass zwei reduzierte Modelle, d.h. ein schnelles und ein langsames Teilmodell, verwendet werden können. Mit Hilfe des schnellen Teilmodells erfolgt unter Annahme einer konstanten Fluidtemperatur, welche sich auf der langsamen Zeitskala entwickelt, die Berechnung der Fluidgeschwindigkeiten. Im Gegensatz zum schnellen Teilmodell berechnet das langsame Teilmodell die Temperatur unter der Annahme von stationären Werten für die Geschwindigkeiten, die sich auf der schnelleren Zeitskala entwickeln. Ein Vergleich der mit den verschiedenen Modellen (vollständiges Modell vs. reduziertes Modell) berechneten Ergebnisse zeigt eine hohe Übereinstimmung. Obwohl der Unterschied im Rechenaufwand sehr groß ist (ca. 10 Stunden für das vollständige Modell, ca. 10 Minuten für das reduzierte Modell), ist eine weitere Modellreduktion erforderlich, um eine sinnvolle Nutzung des Modells zur Steuerung und Diagnose zu ermöglichen.

Danksagung

Die Autoren danken der EU und dem Land NRW für die finanzielle Förderung.

Literatur

- [1] Harlow, F. H., Welch J. E., *Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface*, The Physics of Fluids, vol. 8, no. 12, pp. 2182–2189, 1965.
- [2] Amsden, A. A., Harlow, F. H., *The smac method: A numerical technique for calculation incompressible fluid flows*, Los Alamos Scientific Laboratory, University of California, Tech. Rep., 1970.
- [3] Tome, M., Filho, A., Cuminato, J. A., Mangiavacchi, N., Mc-Kee, S., *Gensmac3d: a numerical method for solving unsteady three-dimensional free surface flows*, International Journal for Numerical Methods in Fluids, vol. 37, no. 7, pp. 747–796, 2001.
- [4] McKee, S., Tome, M., Ferreira, V., Cuminato, J., Castelo, A., Sousa, F., Mangiavacchi, N., *The mac method*, Computer & Fluids, vol. 37, no. 8, pp. 907–930, 2008.
- [5] Harder, D., Skeli, E., Weidemann, D., *Modelling and simulation of high-viscosity, non iso-thermal fluids with a free surface*, in Proc. of the 15th Conference on Informatics, Automation, and Robotics (ICINCO), pp. 557–563, 2018.
- [6] Lichte, B., *Verlässliche und effiziente Simulation physikalisch-technischer Systeme durch Nutzung von Fachwissen*, Shaker Verlag, 2006.
- [7] Panreck, K., *Verkopplungsorientierte Modellbildung und Simulation stationärer Extrusionsprozesse*, Dissertation, Univ. Paderborn, 1995.
- [8] Kokotovic, P. V., *Singular perturbation techniques in control theory*, in Singular Perturbations and Asymptotic Analysis in Control Systems, Kokotovic, P. V., Bensoussan, A., Blankenship, G. L. (Editoren), pp. 1–55, Springer, 1987.
- [9] Verhulst, F., *Methods and applications of singular perturbations - Boundary layers and multiple timescale dynamics*, Springer, 2005.
- [10] Hirt, C. W., Shannon, J. P., *Free-surface stress conditions for incompressible-flow calculations*, Journal of Computational Physics, vol. 2, no. 4, pp. 403–411, 1968.
- [11] Nichols, B. D., Hirt, C. W., *Improved free surface boundary conditions for numerical incompressible-flow calculations*, Journal of Computational Physics, vol. 8, no. 3, pp. 434–448, 1971.
- [12] Weickert, J., *Applications of the theory of differential-algebraic equations to partial differential equations of fluid dynamics*, Dissertation, TU Chemnitz, 1997.

Modelling and simulation of multi-physics applications in case of sudden transformations of material properties

Robert Courant^{1*}, Jürgen Maas¹

¹ Mechatronic Systems Laboratory, Technische Universität Berlin, 10623 Berlin, Germany

*robert.courant@emk.tu-berlin.de

Abstract. Within this paper, we present a practical approach to generate and simulate coupled models for switching domains in material sciences, for example in crystal structures of ferroelectric materials or magnetic shape memory alloys (MSMA). Instead of developing combined averaged FE-models, we propose an approach by augmenting existing mechanical and electromagnetic FE-code. The necessary domain variables and inequalities are described and the implementation in COMSOL Multiphysics by smoothing the discontinuities is shown for MSMA. The simulation results are compared with experimental data and remaining deviations are discussed. The individual strengths and weaknesses of our approach compared to averaged models result in different use cases.

Introduction

With growing computational power, micromechanical models gain increasing attention in material sciences. Particularly interesting problems are numerical models for switching domains, for example in crystal structures [1]. One typical application are ferroelectric materials, which include piezoelectric and electrostrictive effects. The latter are mainly caused by domain reorientation [2]. Another application of switching domain simulations are magnetic shape memory alloys (MSMA) [3].

Ferroelectric materials and MSMA are made both of a tetragonal crystal lattice (in the relevant low-temperature phase) that moves into specific orientations depending on the external load. Beside the reaction of both materials to mechanical stresses, ferroelectric materials are susceptible to electrical fields and MSMA to magnetic fields additionally [4]. The behaviour is mostly analogue and will be first described using MSMA as an example.

Magnetic shape memory alloys The lattice of MSMA's martensitic phase has a shorter c-axis having a higher magnetic permeability in that direction – the so-called easy axis, marked with an arrow in Fig. 1 – compared to the a- and b-axes – denoted as hard axes. It is advantageous from an energetic point to align the easy axis with an external magnetic field or compressive mechanical load. The energy difference creates a driving force towards a reorientation [5]. If the energy difference exceeds a certain threshold – the so-called twinning stress σ_{tw} – the concerning domain orientation switches along diagonal twin boundaries as shown in Fig. 1. Section 1.1 briefly describes a common mathematical model by Likhachev and Ullakko [6] with discrete domain variables denoting the lattice orientation and the governing switching conditions.

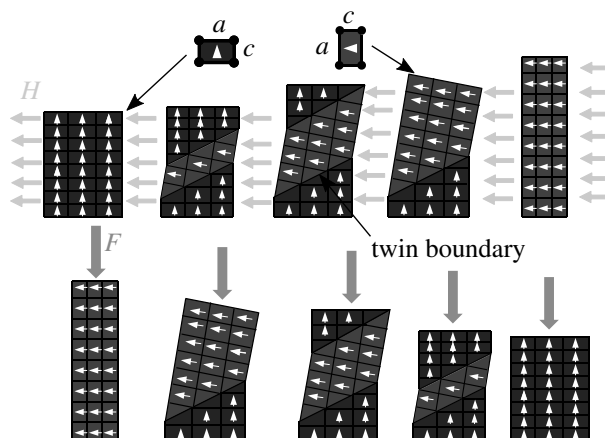


Figure 1: Domain wall movement of a MSM-element under magnetic influence and mechanical load.

Ferroelectric materials The lattice of ferroelectric materials has a longer c-axis with an ionic dipole

moment, which translates to a polarization on a continuum scale [7]. For MSMA, the direction of the c-axis is typically neglected, because the magnetic 180°-domain motion is almost unconstrained [8]. In contrast, for ferroelectric materials the polarization direction is important. It is advantageous from an energetic point to align the polarization with an external electric field and to move the c-axis out of a compressive mechanical load. An example of each case is shown in Fig. 2. Section 1.1 presents a corresponding mathematical model by Hwang and McMeeking [7] with discrete domain variables denoting the lattice orientation and the governing switching conditions.

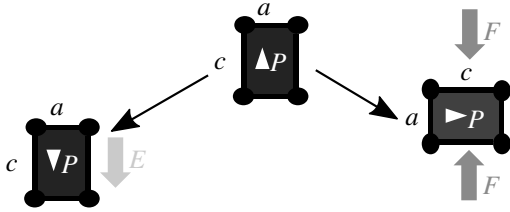


Figure 2: 180°-domain motion of ferroelectric material under electrical influence (left) and 90°-domain motion under compressive mechanical load (right).

Modelling An efficient way to model the macroscopic behaviour of switching domain materials are averaged energy-based models, using continuous phase variables denoting multiple lattice orientations coexistent in a continuum. Their simulation procedure is typically a magneto-static analysis with a special constitutive material model including the continuous phase variables.

For MSMA, one of the most popular models is given by Kiefer and Lagoudas [8]. It formulates the Gibbs free energy as the sum of the Gibbs energy of all possible martensitic variants weighted by their domain variables and a mixing term. The magnetic part contains the magnetic anisotropy energy for a crystal alignment different from the external field and the Zeeman energy for a magnetization direction outside the magnetic easy axis. The mixing term is approximated by analytical hardening functions. Section 1.2 presents some of the necessary equations.

A different modelling approach is presented in [9]. There, the mechanical energy is expressed as a piecewise quadratic function in terms of the strain instead of the stress. A Stoner-Wohlfarth hysteresis model covers

the magnetic anisotropy energy and the Zeeman energy. Additionally, all three possible crystal orientations are taken into account instead of the two-dimensional simplification. [10] gives a more comprehensive overview of the different models available.

In contrast to the averaged concepts, section 2 presents the proposed approach of using fundamental mechanical and electromagnetic models that are coupled with discrete phase variables and the inequalities from section 1.1. To improve convergence, adequate smoothing techniques are applied. In section 3, for validation our approach is compared with experimental data. Finally, the presented modelling techniques are categorised for different use cases in section 4.

1 Mathematical models

This section extends the literature work and presents the used mathematical description of MSMA and ferroelectric materials as well as different averaged models of MSMA.

1.1 Material behaviour

MSMA One way to model the magnetic (F_M) and mechanical (F_{mec}) driving forces for a typical configuration with a unidirectional magnetic field using the energy differences is given in [6] by

$$F_M(H) = G_h(H) - G_e(H) = - \int_0^H M_h(\tilde{H}) d\tilde{H} + \int_0^H M_e(\tilde{H}) d\tilde{H}, \quad (1)$$

$$F_{mec}(\boldsymbol{\sigma}) = \epsilon_0 (\sigma_{xx} - \sigma_{yy}), \quad (2)$$

with the free magnetization energy G , the magnetization curves $M(H)$ in the easy (e) and hard (h) direction, as well as the absolute value H of the magnetic field. The transformation strain $\epsilon_0 = 1 - \frac{c}{a}$ can be calculated from the lattice lengths, σ_{ij} are components of the stress tensor $\boldsymbol{\sigma}$.

The magnetic driving force can be generalized to a magnetic stress for different field directions H_i (again, only the absolute value is used) [3]:

$$\begin{aligned} \sigma_{M,i}(H_i) &= \frac{1}{\epsilon_0} (G_h(H_i) - G_e(H_i)) \\ &= \frac{1}{\epsilon_0} \left(- \int_0^{H_i} M_h(\tilde{H}) d\tilde{H} + \int_0^{H_i} M_e(\tilde{H}) d\tilde{H} \right). \end{aligned} \quad (3)$$

In typical applications, the loads act in one plane as shown in Fig. 3. Therefore, only two domain states are relevant, modelled with a variable $p \in \{0, 1\}$ with $p = 0$ for the easy axis in x -direction and $p = 1$ for y -direction. The switching conditions are then described by

$$\begin{aligned} \sigma_{M,x}(H_x) + \sigma_{xx} &> \sigma_{tw} + \sigma_{yy} + \sigma_{M,y}(H_y) \rightarrow p = 0, \\ \sigma_{M,y}(H_y) + \sigma_{yy} &> \sigma_{tw} + \sigma_{xx} + \sigma_{M,x}(H_x) \rightarrow p = 1. \end{aligned} \quad (4)$$

If none of these inequalities is true, the phase variable p remains constant.

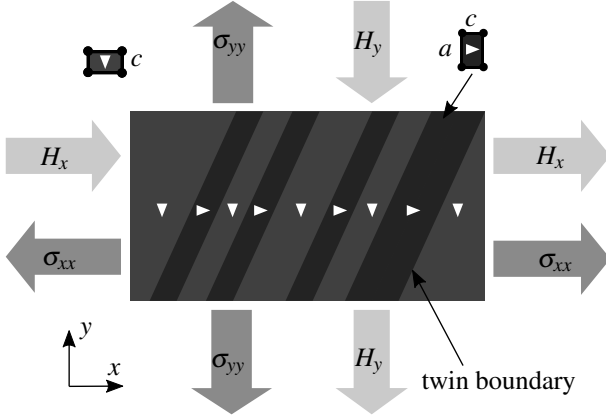


Figure 3: Directional magnetic and mechanical loads of an MSM-element.

Ferroelectric materials For ferroelectric materials, the electric field E and the polarization P have to be considered instead of magnetic field and magnetization. With three lattice axes and two polarization directions each, six states are possible in general. The critical field energy $W_E = 2P_0E_0$ for the switching can be calculated from the so-called spontaneous polarization P_0 and field E_0 , at which the polarization switches [7]. To switch to one of the six states, the electrical or mechanical work has to exceed the corresponding threshold

$$E_i \partial P_i + \sigma_{jk} \partial \epsilon_{jk} \geq 2P_0E_0. \quad (5)$$

If multiple switches are possible, the energetically most favourable is chosen.

1.2 Averaged energy-based models

As the full model of Kiefer and Lagoudas [8] is rather complex, we only show a simplified version not taking into account 180° -domain walls (which have already

been neglected in section 1.1). The Gibbs energy is in this case

$$\begin{aligned} G = & -\frac{1}{2\rho} \boldsymbol{\sigma} : \mathbf{S} \boldsymbol{\sigma} - \frac{\mu_0}{\rho} \underbrace{M_{\text{sat}} [(1-p)\mathbf{e}_x + p\mathbf{e}_y]}_{\mathbf{M}} \cdot \mathbf{H} \\ & + G_{\text{an}}(p, \boldsymbol{\theta}) + \frac{1}{\rho} f(p, \boldsymbol{\epsilon}_r) + G_0(T), \end{aligned} \quad (6)$$

with the mass density ρ , the stress $\boldsymbol{\sigma}$, the isotropic elastic compliance \mathbf{S} , the vacuum permeability μ_0 , the magnetization \mathbf{M} the magnetic field \mathbf{H} , the saturation magnetization M_{sat} , the phase p , the magnetic anisotropy energy G_{an} dependent on the angle $\boldsymbol{\theta}$ between the magnetization and the easy axes of both phases, the hardening function f dependent on the phase p as well as the reorientation strain $\boldsymbol{\epsilon}_r$ and a reference state G_0 which is constant for isothermal problems. The term $\mathbf{M} \cdot \mathbf{H}$ describes the Zeeman energy.

From this energy, the driving force $\pi_p = -\rho \frac{\partial G}{\partial p}$ can be derived, which can be solved for the closed-form solution of the phase $p(\boldsymbol{\sigma}, \mathbf{H})$. The hardening function describes the hysteretic nature with two different analytical expressions dependent on the direction of the phase change $\left| \frac{\partial p}{\partial t} \right|$. The required fitting parameters can be obtained from a single constant-stress hysteresis loop.

2 Extended fundamental models

Different techniques to directly simulate the switching as described in section 1.1 are used. For ferroelectric materials, in [7] and [2] a basic linear piezoelectric model is applied and extended with specific code for the respective switching criterion. Only one domain can change per analysing step. The switched domains are used to update some of the effected parameters in the linear stiffness matrix, other changes are only updated every loading step to speed up the convergence. The highly nonlinear model demands that only a few domains – each represented by a single element – can be simulated effectively.

For MSMA, a finer mesh inside the diagonal domains (also called slices, shown in Fig. 3) is required to accurately capture the local flux inside. Our approach is based on the work of [3]. There, (4) is evaluated by averaging for each domain. While in [3] representative points in the middle of each slice are used, we average over all elements within the domain.

The BH-curves describing the magnetization behaviour for both axes are according to [11]. For an anisotropic implementation in COMSOL Multiphysics, those have to be transferred into $\mu_r(B)$ using

$$B = \mu_0 (H + M) = \mu_0 \mu_r H. \quad (7)$$

Input and output curves for both axes depicted in Fig. 4. The resulting magnetic stress according to (3) is shown in Fig. 5. The energy difference resulting in σ_M represents the area between the two blue curves in Fig. 4. Above $5 \cdot 10^5$ A/m, the curves are identical and therefore a saturation is reached.

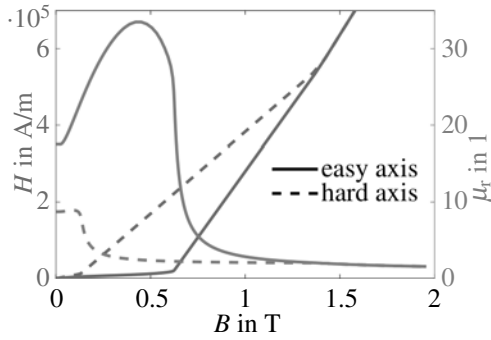


Figure 4: HB- and $\mu_r(B)$ -curves for easy and hard axis used in the simulation model.

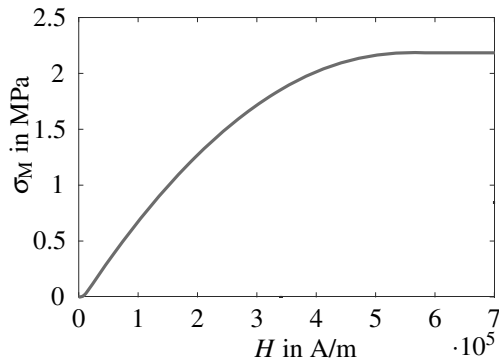


Figure 5: Magnetic stress curve $\sigma_M(H)$ used for simulation.

For each slice, the inequalities (4) to identify the phase p_i for the next iterative step based on the last p_{i-1} are implemented in the following smoothed way

$$\begin{aligned} p_i &= s_1(p_{i-1} + e_1 - e_2), \\ e_1 &= s_2(\sigma_{M,x}(H_x) + \sigma_{xx} - \sigma_{tw} - \sigma_{yy} - \sigma_{M,y}(H_y)), \quad (8) \\ e_2 &= s_2(\sigma_{M,y}(H_y) + \sigma_{yy} - \sigma_{tw} - \sigma_{xx} - \sigma_{M,x}(H_x)), \end{aligned}$$

where the inequalities are replaced by differences e_1 and e_2 which are smoothed by s_2 . The output is limited to $[0, 1]$ by the saturation function s_1 . The functions s_1 and s_2 are shown in Fig. 6. As stresses are typically above 1 hPa, the phase usually switches directly between 0 and 1. A lower gradient in s_2 improves the convergence but generates more intermediate phases, not given in reality.

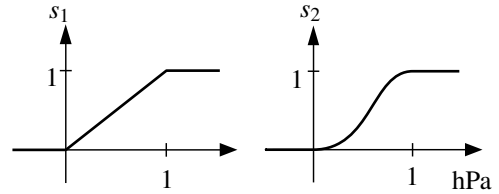


Figure 6: Smoothing functions used in (8).

Currently, it is not possible to measure the material parameters width and twinning stress for each slice individually. However, averaged measurements published in [12] allow to estimate typical distributions. While we keep the width constant at 0.1 mm to get a better mesh, the twinning stresses σ_{tw} of the slices are randomly distributed around 0.4 MPa as shown in Fig. 7.

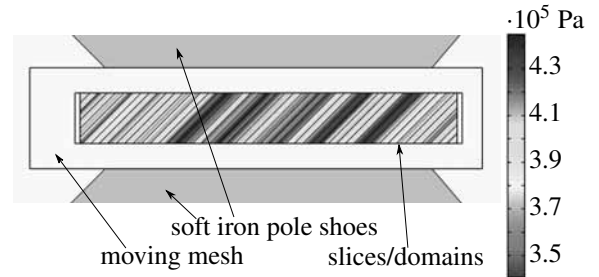


Figure 7: Random twinning stress distribution $\sigma_{tw}(x)$.

Based on a coupled magnetic and mechanical simulation, we are able to calculate the phase of each domain. The phase induces a strain in each slice of the mechanical model and determines the assignment of the corresponding magnetic permeability to the global coordinates.

As discontinuous jumps lead to severe convergence problems, we added smoothing to the load stepping by using a low-pass filter. This can be done in a time-dependent solver by introducing additional ODEs

$$-p_{s,i} + p_i - d_p \frac{\partial p_{s,i}}{\partial t} = 0, \quad (9)$$

with the index s indicating the smoothed variable and the damping d_p . The Index i cycles through all slices. The smoothing is not only beneficial from a numerical point of view but describes the behaviour also under real conditions, as the domain reorientation does not occur instantaneously in reality. It is even more efficient to use a static parametric solver and to replace the derivate with the difference quotient

$$-p_{s,i} + p_i - d_p \frac{p_{s,i} - p_{s,i-1}}{I_i - I_{i-1}} = 0, \quad (10)$$

where instead of the time t , the coil current I is used as a measure for the excitation. The proposed approach to add one variable per slice has little impact on the computational cost compared to the fine mesh needed for the magnetic simulation. An efficient way to implement the different variables and constants given in (8) for each slice is to use COMSOL's Java interface to loop over the domains.

The result of an intermediate load step can be seen in Fig. 8. Red domains have a vertical easy axis, blue domains indicate vertical hard axis. Thus, an external vertical flux is diagonally orientated in the blue domains and almost vertically in the red slices to minimize the magnetic resistance. The crooked outline of the MSM element is caused by the mechanical strain in the switched red regions, as the elongation also causes a transverse contraction. To include the geometric non-linearity in the magnetic simulation, a moving mesh domain has been set up on the surrounding air as shown in Fig. 7.

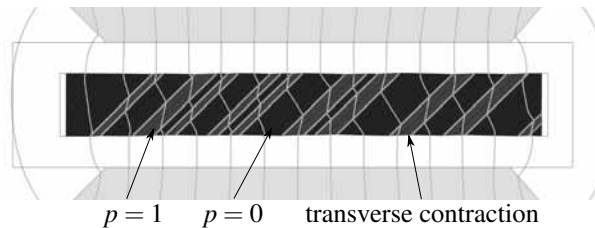


Figure 8: Distribution of the phase variable p (marked blue and red) for an intermediate magnetic field, magnetic flux lines in light grey.

3 Results

Simulated results in Fig. 9 show the expected hysteretic strain-excitation behaviour for an MSMA-element under axial compressive load. In accordance with (4),

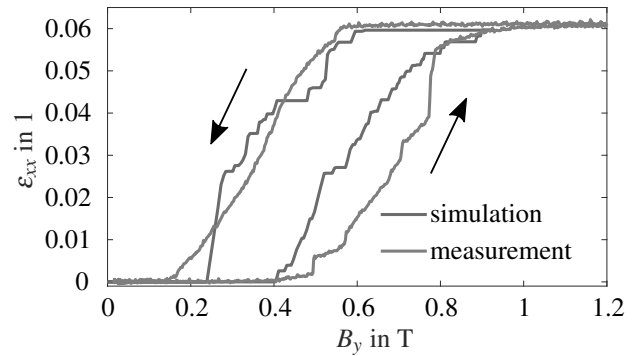


Figure 9: Simulated and measured strain of a loaded MSM-element over the driving flux density (averaged over the element).

the magnetic stress σ_M has first to overcome the compressive mechanical stress $\sigma_{xx} = -0.5$ MPa induced by the load and the twinning stress σ_{tw} until each segment switches. Because from this point a bigger part of the flux passes through that slice, the field in neighbouring domains decreases and the excitation has to be even bigger until the next segment switches. The hysteresis is mainly caused by the dissipative nature of σ_{tw} , which inhibits switching in both directions.

In non-switched domains (marked in blue), the flux is oriented diagonally almost perpendicular to the twin boundaries to minimize the magnetic reluctance. This local behaviour can only be replicated by modelling those slices instead of a continuum. For the same reason, our model offers better results than averaged models when simulating the inhomogeneous flux in the air next to the MSMA which helps to understand the spread of measurements of the magnetic flux at different positions on the MSMA surface. This becomes even more important for MSMA with a single highly mobile twin boundary instead of the discussed multiple fine twins. The local effects in this so-called Type II MSMA are discussed in [13].

For comparison, the curve of a measurement provided by ETO MAGNETIC for the equal load σ_{xx} is depicted in the same diagram. It can be seen, that the startpoints and endpoints of the elongation with increasing flux densities are quite similar. The shown measured result starts to move slower while other analysed samples have a more linear shape. The difference is mainly caused by the unique distribution of the twinning stresses σ_{tw} inside the slices and can indeed lead to different results. As the material parameters differ

between probes and can change over time, the efficient measurement of material parameters for a particular probe remains a complex challenge.

Comparing simulation and measurement, the starting point again fits quite well, but the final state is reached earlier in case of the simulation. This might be caused by the design of the pole shoes as shown in Fig. 7, which is chosen to begin the switching in the middle. In contrast, the experimental test setup has bigger pole shoes extending the outer sections.

4 Conclusion

Our proposed method directly uses the governing inequalities (4) which are comparably easy to understand. While we showed the implementation only for MSMA, the analogies with ferroelectric materials allow the application for this material class as well. More in general, our method could be applied for the implementation of new physical phenomena with switching domains – maybe even other discontinuous processes – without the need to derive a specific combined model. Due to the highly non-linear behaviour of inequalities, several smoothing techniques have to be applied. While they help to achieve convergence, the simulation is still rather inefficient, because the step size has to be lower at (smoothed) switches with high gradients in many variables. Hence, in many applications energy models with averaged phase variables are superior, once they have been developed. Other applications benefit from the proposed model accurately describing local effects, as averaged models cannot fully reflect inhomogeneous fields inside and around the sample.

Further investigations with varying mesh settings, damping and tolerances might lead to improved performance. The presented version is used for a quasi-static load case, transient analyses should be possible with the static parametric implementation of the damping as the phase is not directly dependent on the time.

Acknowledgement

We would like to thank Dr. M. Laufenberg and Dr. E. Pagounis of ETO MAGNETIC for providing samples and measurements of MSMA materials.

References

- [1] J. Huber. Micromechanical modelling of ferroelectrics. *Curr. Opin. Solid. ST. M.* 2005; 9(3): 100–106, doi: 10.1016/j.cossms.2006.05.001.
- [2] F. Li, D. Fang. Simulations of Domain Switching in Ferroelectrics by a Three-Dimensional Finite Element Model. *MECH MATER.* 2004; 36(10): 959–973. doi: 10.1016/j.mechmat.2003.01.001.
- [3] T. Schiepp. *A Simulation Method for Design and Development of Magnetic Shape Memory Actuators* [PhD Thesis]. University of Gloucestershire; 2015. oai: eprints.glos.ac.uk:2974.
- [4] H. Janocha. *Actuators – Basics and Applications.* Berlin, Heidelberg: Springer; 2004. 346 p.
- [5] R. C. O’Handley. Model for strain and magnetization in magnetic shape-memory alloys, *J. Appl. Phys.* 1998; 83(6): 3263–3270, doi: 10.1063/1.367094
- [6] A. A. Likhachev, K. Ullakko. Magnetic-field-controlled twin boundaries motion and giant magneto-mechanical effects in Ni–Mn–Ga shape memory alloy, *Phys. Lett. A.* 2000; 275(1): 142–151. doi: 10.1016/S0375-9601(00)00561-2.
- [7] S. C. Hwang, R. M. McMeeking. A finite element model of ferroelectric polycrystals, *Ferroelectrics.* 1998; 211(1): 177–194, doi: 10.1080/00150199808232342.
- [8] B. Kiefer, D. C. Lagoudas. Magnetic field-induced martensitic variant reorientation in magnetic shape memory alloys. *Philos. Mag.* 2005; 85(33): 4289–4329. doi: 10.1080/14786430500363858
- [9] B. Krevet, M. Kohl, P. Morrison, S. Seelecke. Magnetization- and strain-dependent free energy model for FEM simulation of magnetic shape memory alloys. *EPJ ST.* 2008; 158(1): 205–211. doi: 10.1140/epjst/e2008-00677-y.
- [10] J. Wang, P. Steinmann. Finite element simulation of the magnetomechanical response of a magnetic shape memory alloy sample, *Philos. Mag.* 2013; 93(20): 2630–2653, doi: 10.1080/14786435.2013.782443.
- [11] M. Schautzgy, U. Kosiedowski, T. Schiepp. 3D-FEM-Simulation of Magnetic Shape Memory Actuators, *Proc. of 2016 COMSOL Conf.*; 2016 Oct; Munich.
- [12] E. Pagounis, M. Maier, M. Laufenberg. Properties of large Ni–Mn–Ga single crystals with a predominant 5M-martensitic structure, *3rd Int. Conf. Ferromagn. Shape Mem. Alloy*; 2011; pp. 207–208
- [13] N. Gabdullin. *Modelling and design of high-speed, long-lifetime and large-force electromagnetic actuators based on magnetic shape memory alloys* [PhD Thesis]. University of London; 2016. oai: openaccess.city.ac.uk/id/eprint/16130.

Reducing response time with data farming and machine learning

Falk Stefan Pappert^{1*}, Oliver Rose¹

¹Fakultät für Informatik, Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85579 Neubiberg, Germany * falk.pappert@unibw.de

Abstract. In industry, there are numerous applications for simulation. However, simulation in our area usually takes some time even if a preexisting model just needs to be parameterized; there is still the run time, which will usually take at least a few minutes if not hours. In our current case, a planner wanted to know for a given product mix situation and for an equipment group with specific characteristics how much he can utilize the equipment without violating flow factor targets. A question, which arises several times during a typical workday as new orders are coming in and the situation on the shop floor is continuously changing. Since the user is usually asking the same question just with different parameters we are able to solve the waiting time problem while still giving good decision support. Instead of simulating every scenario at the time the user actually needs these answers, we use data farming to generate a large set of data points that are then used to train a neural network. This neural network then substitutes for the simulation and responds to the user immediately.

Introduction

A crucial task in modern industry is capacity planning. Robinson et al. [1] point out why accurate capacity planning is so important, yet so difficult to achieve in the highly sophisticated semiconductor industry. A planner faces numerous questions every day from short-term operative questions to long-term strategic ones. A necessary starting point to make any reasonable decisions is to know the available equipment capacity and how its' utilization influences the material flow.

As cycle times vary from product to product flow factors (cf. Equation 1) are good indicators to evaluate a production system.

$$\text{flow factor} = \frac{\text{actual cycle time}}{\text{raw process time}} \quad (1)$$

The trade off between utilization and flow factor can be visualized as operating curves (cf. [2]), which relate a system's flow factor against its utilization. Operating curves are an important tool in managing semiconductor

fabs (cf. [3]).

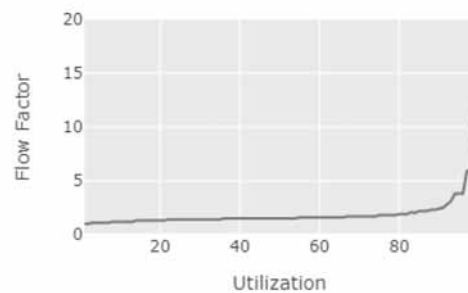


Figure 1: Operating Curve of a basic single equipment without any special features

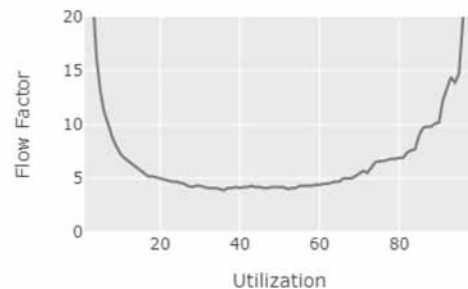


Figure 2: Operating Curve of a single batch-equipment with infrequent but long breakdowns

Examples of operating curves are shown in Figures 1 and 2. Although these curves represent the behavior of the system at all utilization levels, usually not the whole operating curve is relevant to a planner. What typically is of interest to our colleagues is whether there is enough capacity for a given product mix or load. In modern days, this question has changed to whether it is possible to maintain a given flow factor with the given product mixes and loads. Therefore, it is important to know until which point an equipment group can be utilized before it violates flow factor targets. These thresholds are basi-

cally what we are looking for. Figure 1 and Figure 2 also show that whether a system can handle a given material flow is not just based on its' utilization. Numerous factors are influencing equipment behavior; batching, breakdowns, and maintenance are just some examples. Based on these characteristics equipment groups are able to handle different utilization levels before reaching certain flow factors. As this differs from equipment to equipment this question needs to be answered often for different equipment groups.

It is the goal of our research to develop an approach that answers this question in a most timely fashion while still being sufficiently accurate to base planning and investment decisions upon. Traditionally, this is done at our industry partner with a calculation based on look up tables, which only included some factors. Although the look up is quite quick, the results were far from optimal since too few influencing factors were considered. A typical solution approach would simply be to build or generate a simulation model for a given equipment group and run some simulation experiments. But this would still take some time, with large equipment groups maybe even a few minutes. Hence, this approach would not meet the response time requirement for the given problem. Byrne [4] proposed an approach to limit the number of necessary design points to calculate an operating curve. Although this would speed up simulation, there would still be some waiting time for results.

In the first section of this paper, we will give an introduction to the general idea behind our approach. In Section two, we will discuss some software development aspects of creating such a system. In the third section, we will briefly show the features considered in our current project. In the following sections, we will furthermore discuss the simulation model, data farming, and training of our artificial neural networks. In Section seven, we will shortly show our current results and give an outlook on our future plans in Section eight.

1 System overview

As we have previously discussed, we aim to build a system which is able to quickly provide answers to the same question with changing parameters or configurations, that is repeatedly asked during a normal workday. If these questions would only be asked from time to time or a response would not be that time critical a common approach would be to build a simulation model and answer the question after analyzing a simulation

experiment.

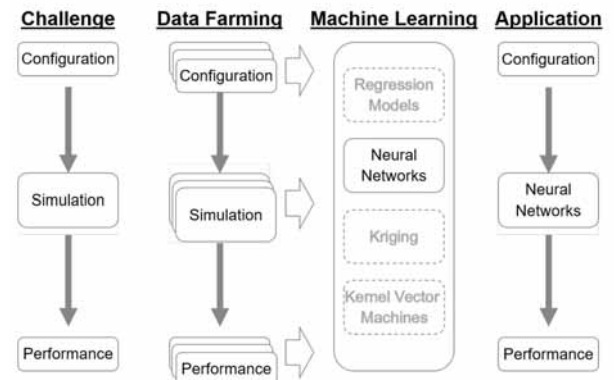


Figure 3: Generalization of discussed approach for a fast response system based on data farming and machine learning

If short response times are very important, Figure 3 shows a generalized approach on how we answer this question. The challenge column would be the normal simulation experiment approach. The user has a question about a given system configuration. A simulation model representing this system configuration is built and its' performance is evaluated. After a reasonable number of simulation runs the user gets the answer.

As the simulation runs are the time consuming part we changed the system. Instead of creating or parameterizing a simulation model each time the user needs an answer to the question we move the simulation runs to a point in time long before the user asks our system. We use data farming to create the results to a huge number of possible factor combinations. The resulting data set is then used as the supporting points for machine learning algorithms, in our case neural networks, to approximate a function that reproduces a response to a given configuration and thereby replaces the simulation in the moment the user queries the system. Instead of directly asking for the results of a time-consuming simulation experiment, the user asks the neural network that is able to respond almost immediately.

Figure 4 shows a basic overview of our system.

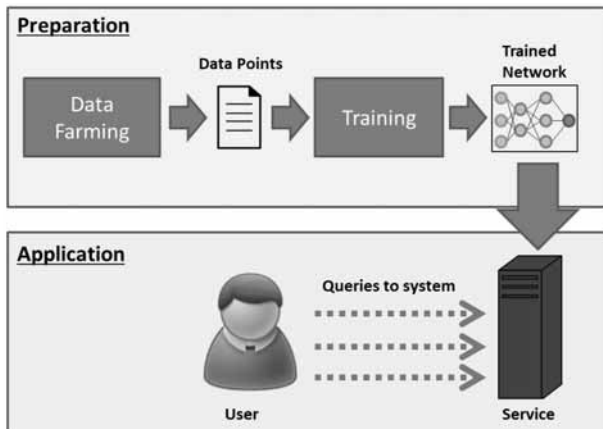


Figure 4: System overview

2 System architecture

In this section, we will discuss our system architecture from a software development point of view. We will start with the initial basic design and point out changes we have done to improve the system.

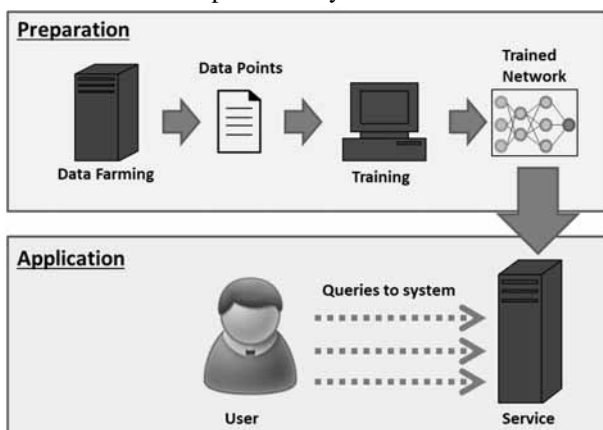


Figure 5: Basic system architecture

Basing our architecture (cf. Figure 5) on the system overview shown in Figure 4 we planned for one big simulation-based data farming component, which would generate a huge set of supporting points. These data points would be transferred as a file to an R (cf. [5]) script handling data preparation and training of the neural network. With this setup, we were able to obtain reasonable results but we found that there is still a lot of room for improvement. One of our first and surprisingly valuable changes was a switch from using R to train our neural networks to Keras (cf. [6]). With R, depending on our data set, we were sometimes observing training times of a couple of days, which we attributed to reaching some memory boundaries. We often had to abort

after some time as no further progress was visible and it was hard to predict the remaining training time. However, even training times of a few hours considerably limit the amount of network configurations one can test in the hopes of improving results. The switch to Keras with an underlying Tensorflow (cf. [7]) library immediately improved training times incredibly. Furthermore, being able to utilize GPUs (graphics cards) for training improved training speeds to a point where instead of several hours or days we were looking at seconds and minutes for training. This new dimension of training times opened up the opportunity to consider neural architecture search to further improve the results of the neural network and thereby the whole system in the future.

A second big change to our system is the move from a monolithic piece of software to a service-based architecture using RESTful Web Services (cf. [8]). In this change, we see three major benefits to our system:

1. Ease of communication between system parts,
2. Scalability and distribution on multiple machines,
3. Replaceability of components.

In the beginning of the project, we made the conscious decision to implement different parts of our system with different languages. We see benefits in developing in Java with its object-oriented concept and type system paired with available IDEs supporting numerous ways of testing and debugging that make it very suitable to larger and more complex software projects. R on the other hand offered much easier access to mathematical functionality and neural networks. Nowadays, Python basically is the de facto standard language for data analyses and machine learning with a number of libraries and frameworks available and new systems usually being accessible only or at least first with Python.

Communication across these language barriers is often not easy with “direct calls”. As most modern languages nowadays offer libraries, to easily implement web services, this is an elegant approach to handle communication between system parts written in different programming languages without much additional implementation overhead. Gone were complicated command line calls and file-based communications.

Most parts in our system can be quite computation intensive. While it is still reasonable to run small test cases on a single office PC, larger experiments benefit from good scalability and distribution. Furthermore, different parts of the system benefit differently from

available hardware. While the simulation is mostly CPU and memory intensive, training neural networks significantly benefits from the availability of modern graphics cards. Being able to assign services to machines with the best fitting hardware is therefore another aspect, which is easily taken care of with services. Besides distributing the different services on different machines, some services may in the future need additional computing power. With a service-oriented architecture, it is also easy to introduce load balancers which distribute requests of the same type on several machines offering the same service without any necessary changes to the client side. This makes setting up such a system very comfortable for different experimental environments.

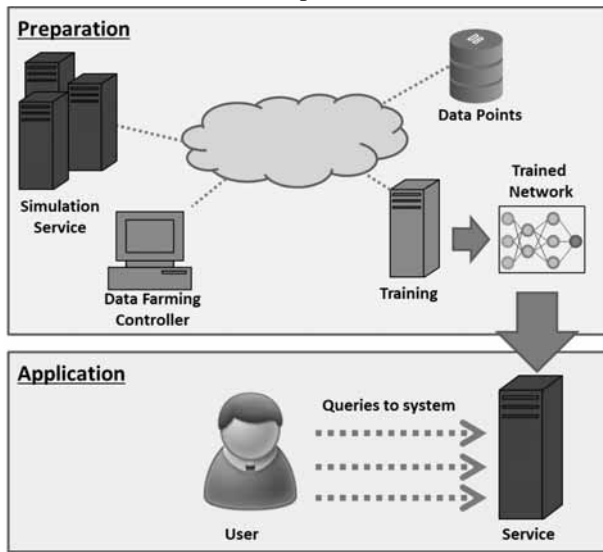


Figure 6: Service based system architecture

Replaceability without the need to touch any other system component is also a great benefit. As we try different frameworks and approaches the current architecture offers us to simply replace some services while others stay the same. Changing the simulator, the persistence approach or even the machine learning technique are all simply done by putting the new component up as a service replacing the old one. Hence, with this change we gained a lot of scalability and flexibility for future experiments.

3 Factors

Starting with Robinson et al. [1] and Hopp and Spearman [9] and a review of the previous planning

methods we defined relevant features for our equipment group model. Values for our factor levels were chosen based on a fab dataset from our industrial partner by looking for natural clusters and using representatives thereby capturing realistic workings points.

Most factors can be easily defined with single numerical values. These are shown as quantitative factors. Some factors shown as categorical in Table 1 represent more complex definitions. Product mix for example represents the number of different products as well as their percentage of the released material flow. For categorical features we selected three levels based on real equipment groups going from a low impact to a high impact setting with regard to the resulting flow factors.

Feature	Factor	#	Type
Batching	MaxBatch	5	Quant.
	MinBatchPercentage	3	Quant.
Breakdown	MeanTimeBetweenFailure	3	Quant.
	BreakdownCapaLoss%	2	Quant.
Dedication	Dedication	3	Cat.
Equipment #	ToolCount	7	Quant.
Maintenance	TimeToMaintenance	3	Quant.
	MaintCapaLos%	2	Quant.
Product Mix	ProductMix	3	Cat.
Rework	ReworkPercentage	3	Quant.
Process Time	RPT	6	Quant.
Setup	SetupDuration	3	Quant.

Table 1: Feature and factor overview

While initially considering only one factor per feature we now split some features into two factors for better scalability. This makes it easier for a future algorithm to generate new test points to validate and improve the resulting model. Additionally, the effect of some features is hard to capture with a single factor. For example, when considering two systems suffering from 25% loss of capacity due to breakdowns; one breaking down after 3 hours of productive time for about an hour while the other one runs fine for 3 weeks followed by a week of repair. We would expect the second system to perform worse with regard to cycle time and flow factor. Hence, we split breakdown into capacity loss and mean time between failures. As the increase in factors brings a significant increase in design points considering a full design, we have not yet updated our training data set to include all new points. Although the training data set does not include all these points, we are still able to address them better and test for them.

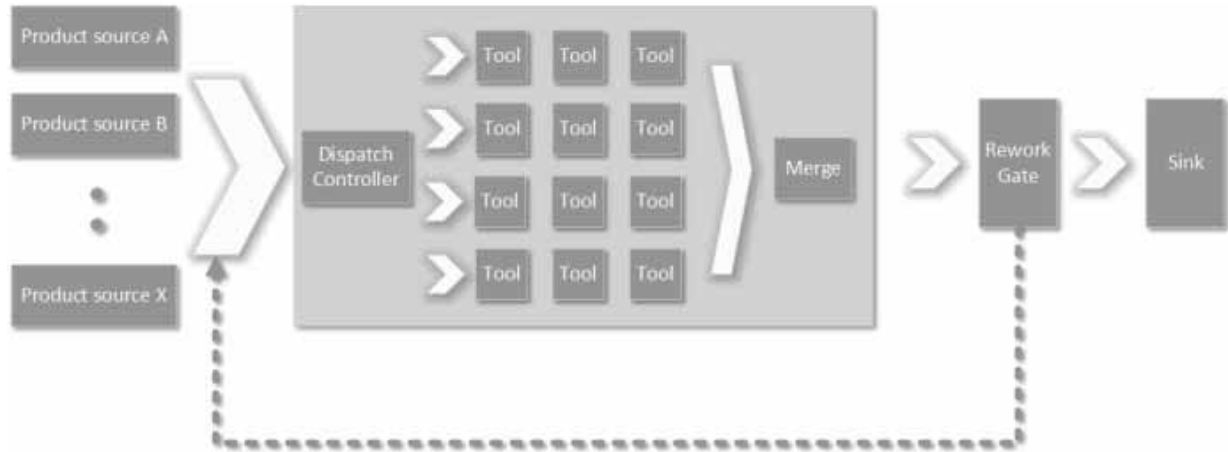


Figure 7: Simulation model structure

4 Evaluation and Simulation

We use an inhouse developed factory simulator for all simulation runs in this project. As we have mentioned before the simulator is currently running as a service and simulation runs can be started by calling the service and handing over parameter values for each factor under consideration.

The simulation service will then automatically generate a simulation model based on the given parameters. Figure 7 shows a visualization of the equipment group model. As the goal of the simulation is to determine reasonable utilization values for the lowest flow factor possible and the location of defined flow factor thresholds, the next step is a static capacity analysis. This is necessary to run the simulation only with reasonable loading scenarios without wasting calculation time for extremely low utilization settings or incredibly overloaded systems. Based on the static capacity analysis we can now simply calculate the necessary lot releases to run the model at a specific utilization point.

We use a search strategy akin to binary search to look for the location with the lowest possible flowfactor. For each utilization point under evaluation simulation runs are performed until the sample size for this point is determined to be large enough for a stable estimation. Flow factor thresholds are searched for similarly while reusing the results of previously tested plus new utilization points. Once the lowest flow factor value and all requested flow factor thresholds are determined the results are handed back.

Verification and validation are difficult when considering data farming, as it is almost impossible to evaluate every single simulation run. We deployed different

strategies to ensure our simulation results reflect real world behaviour. The basis for this were unit tests to continuously check the simulation software during development. This was done to avoid unintended effects during programming. We additionally compared sample simulation results with the results from other simulation software packages. Additionally, we had a panel of experts reviewing results generated by the simulator and compare them to real factory data of equipment groups with similar characteristics. Of course, this cannot be done for all data points but helps to validate the system.

5 Data farming

When considering the number of factors and factor levels, we are looking at a huge number of data points to evaluate. In addition to all these data points, we are also looking at several simulation runs per data point. As we have mentioned before each data point is determined by calculating several utilization points for which we run a number of simulations each. Not all utilization points take the same amount of repetitions as we determine this number on the fly during the evaluation. After simulating an initial set of replications, we calculate the confidence interval half-length and mean. Then, we compare their quotient with the relative error we aim for. If the quotient is still larger than the relative error, we run another set of replications. We repeat this until the relative error is smaller than the quotient (cf. [10]).

On average, we ran about 825 simulations for a single data point to determine the location of the lowest flow factor value and three thresholds. Considering even just one factor per feature we were looking at almost 460000 data points which total in almost 380 million simulation runs just to generate the supporting points for

our project.

Although the evaluation of single data points is feasible on a normal PC, running these almost 460000 of data points on one of our simulation servers took several weeks. With our change in architecture and therefore much better scalability, we hope any future additions to our current data set will be available much faster.

6 Training

With all these data points from simulation, we were still only looking at the supporting points for our system. As we have mentioned before we moved from using R to Keras to implement the training of our neural networks which considerably increased training speed and made it much easier to test different layer configurations. Typically, we aim to minimize the mean squared error (MSE) of our testset. When trying to evaluate the usefulness of any trained network we additionally present the predicted results graphically.

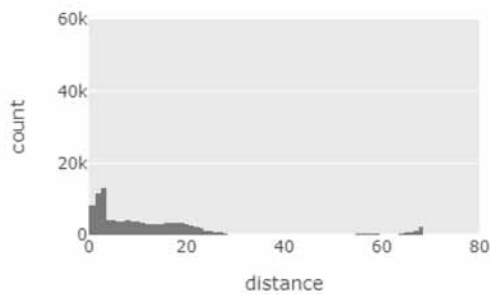


Figure 8: Visualization of the distance between predicted and simulated value; example 1

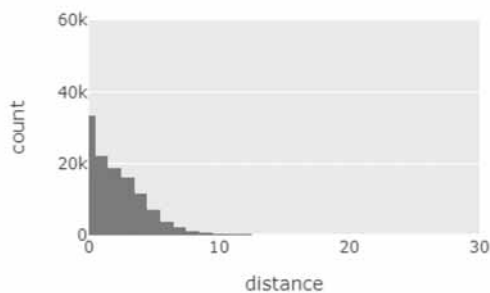


Figure 9: Visualization of the distance between predicted and simulated value; example 2

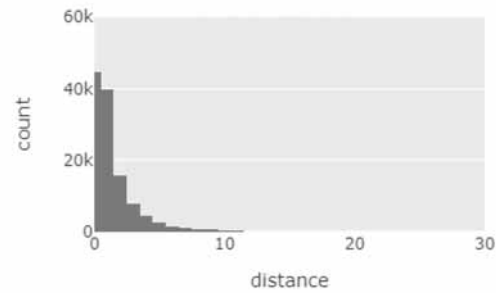


Figure 10: Visualization of the distance between predicted and simulated value; example 3

Figures 8 to 10 show the results of network configuration and training parameter sets we tested. The diagrams are histograms of how successful the predictions have been. Starting from the left results are grouped by the error in prediction compared to the simulation result. The first bar represents less than 1% distance and each following bar an additional 1%. E.g., a scenario for which the network predicted 75.5 but the simulation estimated 73 would fall into the third bin. Please be aware that the x-axis in Figure 8 is using a different range from the other shown results. We chose to do this to be able to show the set of extremely poor predictions that is not present in the other results.

Although the quality of the trained networks can differ greatly between network configurations, all of the results shown here were able to reduce the MSE continuously during training and on a first glance seemed to work quite well. Only when visualizing what the results meant with regard to the actual problem at hand, it became obvious that some of these networks are not useful at all to solve our problem.

Besides network architecture we found that training parameters like batch sizes and the number of episodes have a significant impact on result quality. In fact, Figures 8 and 9 are based on the same network configuration but used different batch parameters for training.

7 Results

We set out to achieve two objectives. First, we wanted to create a system, which is able to respond immediately to a user query. Second, we needed to have sufficiently good results to base planning and investment decisions

upon.

On the first objective, we are where we want to be. The application server running on a better office PC with a modern graphics card responds within 300ms to a user query. The majority of this time is actually spent on allocating the graphics card. Running the network for the prediction just on the CPU without GPU support this response time could actually be reduced even more as predicting is quite fast with just the CPU.

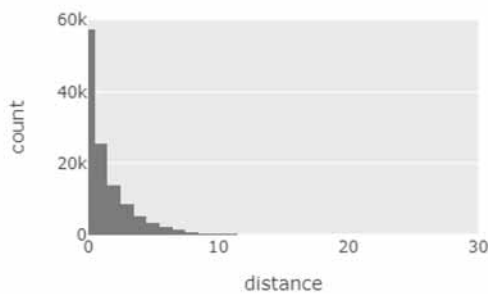


Figure 11: Visualization of the distance between predicted and simulated value; current results

Looking at the second objective, Figure 11 shows our currently best results derived just with manual testing of different network architectures and training parameters.

About 80% of our test points are predicted with an error of less than 3%. Almost all datapoints are predicted with less than 10% error from our simulation results. Furthermore, we are no longer seeing any artifacts as in Figure 8. Although these are reasonably good results and a prediction quality of 3% or better for a majority of data points would be good enough to base planning on these numbers, we are still looking at 20% of predictions being off by up to and 10%. Considering high utilization scenarios overestimating possible utilization by 10% error could have a serious impact on the performance of the material flow and ability to maintain promised delivery dates. On the other hand, underestimating utilization thresholds by 10% would mean significant loss of production capacity or triggering an investment in equipment before it is actually necessary. We therefore still see some need for improvement.

8 Summary and Outlook

In this paper, we presented our approach to create a

system with minimal response time to a user query, which we would usually answer by simulation. We discussed some aspects of software architecture to improve scalability and flexibility of the software. The presented system uses a simulation model to do data farming for supporting points, which are then presented as training data set to machine learning methods like neural networks. The resulting trained system is able to respond to the user queries within moments.

Although the system already works as a proof of concept, the accuracy is still not where it would need to be to be applicable in an industrial setting. We are working on two approaches to improve prediction quality for all points.

First, we have seen during our manual configuration of the the neural network that network architecture and training parameters tend to have a big impact on result quality. With our improved training speeds, we are planning to improve prediction quality by automating the process of finding a good network configuration. There are several promising approaches to do this. We are currently working on adding a neuroevolution (A broader explanation can be found in [11]) service to our system. As an alternative, we are also looking at Auto-Keras (cf. [12]).

The second approach targets the somewhat infrequent supporting points within our training data set. Since adding additional levels to factors drastically increase the number of total points to calculate for a full design, simply adding more levels would be a very computation intensive approach. Instead, we are seeking to improve the quality of the systems response by automatically searching for points with bad predictions and adding additional supporting points near those points to our training data set. Ideally, this would improve prediction quality in those areas and therefore for the whole system.

Acknowledgements

We would like to thank Dr Thomas Mayer for many interesting discussions on system architecture and machine learning.

References

- [1] Robinson, J.K., Fowler, J., Neacy E. *Capacity Loss Factors in Semiconductor Manufacturing*. FabTime Inc 2003

<https://www.fabtime.com/files/CapPlan.pdf>.

- [2] Aurand, S., Miller, P. The operating curve: a method to measure and benchmark manufacturing line productivity. 1997 IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop ASMC 97 Proceedings. *1997 IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop*; 1997 Sep; Cambridge, MA, USA. IEEE. 391-397. doi: 10.1109/ASMC.1997.630768.
- [3] Fayed, A, Dunnigan B. Characterizing the Operating Curve — how can semiconductor fabs grade themselves?.. *2007 International Symposium on Semiconductor Manufacturing*; 2007 Oct; Santa Clara, CA, USA. Place of Publication: publisher. 1-4. doi: 10.1109/ISSM.2007.4446827.
- [4] Byrne, N.M. *A framework for generating operational characteristic curves for semiconductor manufacturing systems using flexible and reusable discrete event simulations* [dissertation]. School of Mechanical and Manufacturing Engineering. Dublin City University; 2012.
- [5] Verzani, J. *Using R for Introductory Statistics*. 2nd Edition. New York, USA: CRC Press; 2014.518p
- [6] Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 2nd Edition. Sebastopol: O'Reilly; 2019. 600p
- [7] Abadi, M., Barham, P. et.al. TensorFlow: A System for Large-Scale Machine Learning. Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16). *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)*; 2016 Nov; Savannah, GA, USA. 265-283. ISBN: 978-1-931971-33-1.
- [8] Fielding, R.T. *Architectural Styles and the Design of Network-based Software Architectures* [dissertation]. University of California, Irvine; 2000.
- [9] Hopp W.J., Spearman, M.L.. *Factory Physics*. 3rd Edition. New York: McGraw-Hill; 2008.
- [10] Law, A.M., Kelton, W.D. *Simulation Modeling and Analysis*. 3rd Edition. New York: McGraw-Hill; 2000. 760p
- [11] Stanley, K.O., Clune, J., Lehman, J. et al.. Designing neural networks through neuroevolution. *Nat Mach Intell*. 2019; 1: 24-35. doi: 10.1038/s42256-018-0006-z.
- [12] Jin, H., Song, Q., Hu, X. Auto-Keras: An Efficient Neural Architecture Search System. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. *25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; 2019; Anchorage, AK, USA, NY, USA: Association for Computing Machinery. 1946-1956. doi: 10.1145/3292500.3330648.

Neural Network Application for Event Detection in Hybrid Dynamical Systems

Stefanie Winkler^{1*}, Andreas Körner¹, Felix Breitenecker¹

¹Department of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8–10, 1040 Vienna, Austria; *stefanie.winkler@tuwien.ac.at

Abstract. This contribution investigates a feed-forward neural network approach for event detection in hybrid dynamical models. Machine learning algorithms are commonly used in software development. In recent years these approaches have also been increasingly applied in modelling and simulation of physical systems. A significant amount of these models use artificial neural networks. However, hybrid dynamical systems describe a combination of different methods to describe a continuous process, which experiences behavioural changes at discrete events. Accordingly, the models of such systems are based on a combination of discrete and continuous methods and are often illustrated as automaton. Based on these two areas an approach, to predict the event time of the discrete processes, is presented. The different required elements are defined and a general approach is outlined. The feasibility of this concept is examined on the basis of one examples. If the given imbalanced data is resampled, training can be successful. Unfortunately, even then, the generalised classification of events often does not work sufficiently. The evaluation of the approximation results of the discrete events in hybrid systems suggests that neural networks are not suitable to classify the system states with regard to the occurrence of an event. In the outlook we suggest an alternative approach to predict the event with neural networks.

Introduction

The goal of this contribution is to investigate the applicability of artificial neural networks in the event detection for hybrid dynamical systems. On the one hand, the use of neural networks has increased, especially in fields such as pattern recognition and software development. In recent years neural networks are also applied in different engineering applications. On the

other hand, hybrid approaches present a possibility to model complex systems. A hybrid model benefits from combining the advantages of different methods. In the case of hybrid dynamical systems, a dynamical process, characterised by finite changes of the model description, is described. These changes are called discrete events. This contribution focuses on autonomous events where no external factors are involved initiating the event.

Based on the hybrid dynamical automaton, as discussed in [7, 9], a model for applying neural networks is formulated. Additionally, the basic structure of neural networks will be presented. A multi-layered neural network will be applied to substitute the event in the hybrid dynamical model. Combining continuous, discrete and machine learning approaches, a model will be defined.

For the hybrid system in the case study the bouncing ball is chosen. The state space of this examples is one-dimensional and the automaton of the simplified assumption consists of one dynamical process. The goal of the trained network will be to determine if a state vector is initiating an event or not. One of the challenges of this approach is to create a useful learning dataset. In hybrid systems, states without an event are more frequent than states where an event occurs.

1 Artificial Neural Networks

Neural networks and machine learning are a central part of modern technology. Various software packages in computer science and engineering apply machine learning methods. In system identification problems for example, neural networks are used to approximate suitable model structures. In the early 1950s the first neural network was introduced and consisted of so-called perceptrons [13]. The task of a perceptron is to test incoming signals against a predefined threshold, whereas

exceeding this value results in 1 and otherwise in 0. Despite the application purpose, the structure of a neural network always consist of three different types of layers: the input layer, the hidden layer and the output layer.

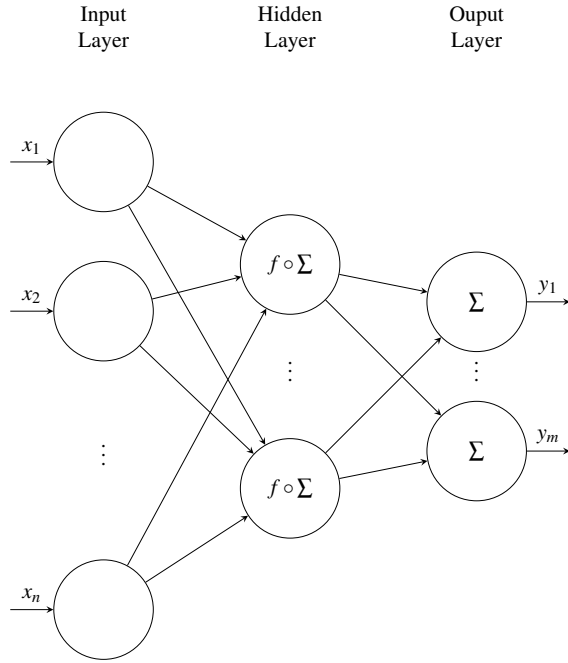


Figure 1: The structure of a basic artificial neural network, called multi-layer perceptron (MLP).

Each layer consists of a particular number of nodes, referred to as perceptrons or neurons. If every neuron of each layer is interconnected to every neuron in the following layer, the network is called *fully connected* [6]. Artificial neural networks (ANN) can be characterised by the way the signals are processed. If a neural network is executed and the signal flows from the input layer through the hidden layer and exits at the output layer, the network structure is categorised as *feed-forward*. Such networks are often called *multi-layer perceptrons* (MLP). For most of the applications, a multi-layer feed-forward network is sufficient [8]. In Figure 1 a one layered feed-forward network is depicted. It consists of one input, one hidden and one output layer. The shape of the input and output layer is defined by the given inputs $x \in \mathbb{R}^n$ and the corresponding outputs $y \in \mathbb{R}^m$. In contrast, there is no predefined structure for the hidden layers. The number of hidden layers as well as the amount of neurons in these lay-

ers are arbitrary. The connections between the neurons in the different layers are called edges. Each edge carries an individual weight w_{ij}^l , where $l \in \mathbb{N}$ defines the target layer of the connection and i, j specify the end and starting neuron, respectively. The weighted sum of the inputs together with a bias b_j^l form the input of the neuron in most applications. The neurons of the hidden layer as well as the neurons of the output layer apply an activation function to the incoming signal.

In order to obtain a good performing neural network, it is necessary to determine the biases and weights of the network structure. The iterative process determining these parameters is called training. The complexity of the model, the number of parameters as well as the accessibility, size and range of the given datasets affect the success of the training. In this study only supervised training methods are considered. The iterative optimisation process uses the given dataset to improve the results of a predefined cost function, also referred to as loss function. At the beginning of the training the parameters are initialised randomly using a certain distribution. The loss function of a neural network is similar to the cost function in optimisation problems. It evaluates the performance of the given neural network and by minimising the loss function the results are improving. For classification networks, a possible loss function for convex optimizers is the support vector machine (SVM) loss or Hinge loss and the cross entropy loss, respectively. There are different methods to determine the network parameters w and b . Apart from the Gradient Descent, common optimisation algorithms are the Newton's Method, Conjugate Gradient, Quasi Newton Method, Levenberg Marquardt and Adams Algorithm. Depending on the size of the input-output dataset, an adequate training algorithm has to be chosen. The loss function and the optimization algorithm are embedded in a learning method. The most common method is the back-propagation.

2 Hybrid Dynamical Automaton

In applied mathematics and computer science the creation of modelling standards is an important aspect. The advantage of a modelling frameworks is that they can be applied not only to one unique problem description but to an entire class of problems. Based in related work the structural and graphical framework for hybrid dynamical systems is defined. Characterisations of different event formulations and transitions are given.

The complex structure of hybrid systems is commonly illustrated using an automaton [9, 4]. Automata are often applied to depict abstract machines as well as theoretical concepts in computer science, such as combinational logic. It supports structuring mathematical tasks and illustrating finite state machines [2]. It shows current states, update rules as well as transition conditions. By expanding the description possibilities of the update rules, the automata concept can be applied to model hybrid dynamical systems, as shown in Figure 2.

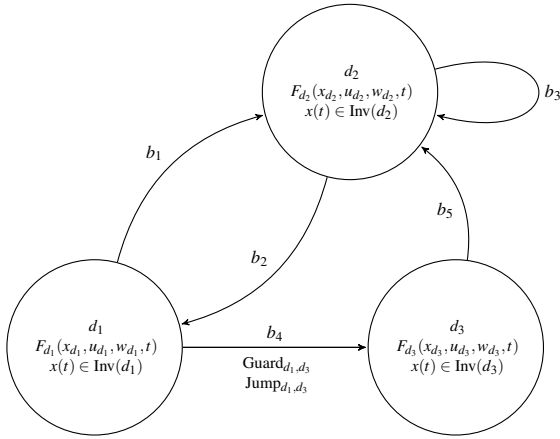


Figure 2: Conceptual structure of a hybrid automaton based on the work of [10].

In terms of layout, an automaton is an ambiguous description. It just characterises the basic structure of the model in a compact way. The nodes of the automaton describe different local dynamics whereas the connecting lines, called edges, define transition conditions. If fulfilled, the transition from one location (node) to the next is initiated.

Focusing on modelling of hybrid dynamical systems, a mathematical definition of the hybrid automaton is given [7, 4, 10].

Each node of the hybrid automata contains an individual model description, e.g. differential-algebraic equations. The state and time events, respectively, enable the transition from one node to the next. A characterisation of the different possible changes during the event, transitioning from one subsystem to the next, is formulated based on [10].

3 Artificial Hybrid Events

3.1 Idea & Concept

The illustration of the hybrid dynamical automaton in Figure 2 suggests a possible scenario to apply neural network for event detection in hybrid dynamical models. It proposes to apply neural networks to administrate the event handling.

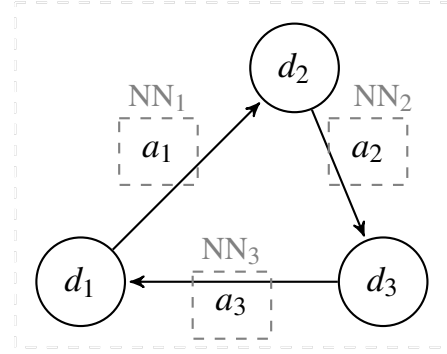


Figure 3: Three framework applications for including neural network concepts (red dashed rectangles) in modelling hybrid dynamical systems.

3.2 Network Approach

The definition is given independently from any specific feed-forward network structure and facilitates every event type. The characterisation of the event, as introduced in the hybrid dynamical automaton, requires at least three elements. The mapping Act is considered to be a-priori knowledge since the local processes and descriptions are accessible. The guard G defines the set of states initiating the events, whereas the jump map J executes the actual changes happening during the transition from one local description to the next.

Considering that the local dynamics are known, the jump relation is already implicitly defined. An investigation of the given dynamical descriptions determines a-priori if J describes a coordinate transform. A neural network approximation of that transform is unnecessary. Secondly, assuming a hybrid system with changing parameters and variables the dynamical behaviour of the system can be characterised mathematically and the jump relation is a consequence of the model description. Hence, the following framework focuses on the

approximation of the guard region.

The local descriptions and their state variables are known, whereas the guard region has to be determined. In contrast to both previously explained applications, the output data for the training set of the ANN has to be defined differently. There is no feasible output of the system, which can be directly applied for training. Instead the data has to be analysed and classified. Two different categories can be defined:

$$O_j = \begin{cases} 0, & \text{no event occurred at } x_j, \\ 1, & \text{an event occurred at } x_j. \end{cases}$$

With this classification a training set for the network A_{ac}^b can be given. The *input data* for training the neural network include

- the state vector of the system $x(t), t \in T \subset \mathbb{R}^+$,
- the given initial conditions x_0 ,
- input u and
- external variables w including any system parameters p .

Hence, the input can be given as $I(x_0, x, u, w, t)$. The *output data* for the training process is $O \in \{0, 1\}^q, q \in \mathbb{N}$, where q depends on the number of classified state values included in the dataset. After embedding the trained network into the model structure, at each time step the state vector of the dynamical process is classified by the network and an event is initiated if the state vector is labelled 1.

Both previous definitions attempt to approximate the relation between input and output values. In contrast, the approximation of the event guard represents a classification task. Hence, neural network structures such as EQL and HONN are not applicable instead MLP suited for classification tasks are required for this framework application. Therefore, datasets containing past state vectors classified with regards to the occurrence of an event can be very imbalanced [4]. As suggested in [5], resampling methods for training will be applied to level out the imbalance and its results will be compared.

4 Case Study

For the application of the approach a hybrid dynamical systems is chosen. The example is used to investigate

the applicability and the experimental results are analysed and compared with the original data.

4.1 Example Description

The bouncing ball, as defined in [11] and [1], describes a ball, bouncing off the ground. The state variable of interest is the height over time t . The acting force in the observed system is the gravity, accelerating the ball to the ground. Thus, the ODE of the dynamical behaviour of the bouncing ball can be described as

$$\ddot{h}(t) = -g, \quad (1)$$

where g is the gravitational constant. Considering an initial height h_0 and velocity v_0 , two state variables can be defined, namely the height $h(t) =: x_1(t)$ and the velocity $x_2(t) := \dot{x}_1(t)$. Hence, using the state vector $x = (x_1, x_2)^T \in \mathbb{R}^2$ equation (1) can be transformed into a state space description resulting in

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ -g \end{pmatrix}, \quad x(0) = \begin{pmatrix} h_0 \\ v_0 \end{pmatrix}. \quad (2)$$

An advantage of the chosen academic example resides in the existence of an analytical solution given as

$$x(t) = \begin{pmatrix} -\frac{g}{2}t^2 + v_0t + h_0 \\ -gt + v_0 \end{pmatrix}. \quad (3)$$

In Figure 4 (a), the height of the ball is depicted over time. Two different processes can be distinguished, the flying and falling phase of the ball, respectively and the bounce. The latter affects the behaviour of the ball and therefore defines the state event of the system.

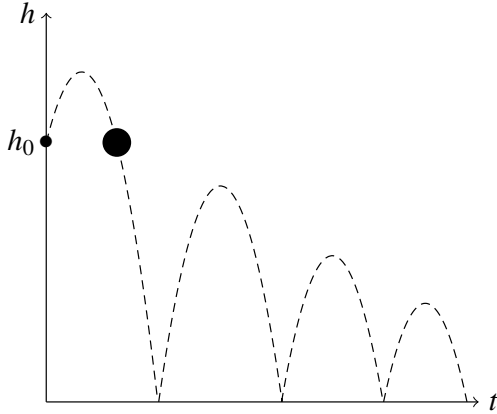
Due to the fact that the free fall phase is interrupted when the ball reaches the ground, the *event guard* $G_{d,d}$ can be defined. An event occurs if the state vector fulfils

$$x \in G_{d,d} := \{x(t) \in \mathbb{R}^2 : x_1(t) = 0 \wedge x_2(t) \leq 0\}. \quad (4)$$

This state event can be given in all three forms. The event function can be defined as $e_b(x) := x_1$ with $e_b(x) = 0$ initiates the event, whereas the threshold for the event can be given as $\Delta x_1 := 0$.

Model descriptions are often implying a certain degree of abstraction. In this application, the bounce is described by a simplified assumption without modelling the physical process in detail. When the ball hits the ground the behaviour of the ball is affected. The friction is realised as simple damping factor λ with $0 < \lambda < 1$.

(a)



(b)

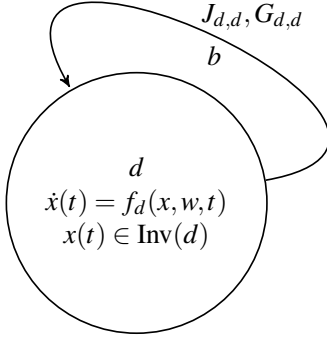


Figure 4: (a) Height of the ball over time. (b) Hybrid dynamical automaton of the bouncing ball.

The reflection of the ball results in inverting the velocity component. Hereby, the time delay due to any deformation work is neglected. Thus, the *jump map* $J_{d,d}$ at the event can be characterised by the linear transform

$$x(t^+) = J_{d,d}(x(t^-)) := \begin{pmatrix} 1 & 0 \\ 0 & -\lambda \end{pmatrix} x(t^-). \quad (5)$$

4.2 Event Classification Results

Focusing on the problem of event detection with neural networks, possible datasets consist of different state values of the system and its label if an event occurred. A MLP is chosen as network structure. The framework

application to replace the guard region in the hybrid model of the bouncing ball is given as

$$\begin{aligned} A_{\text{hae}} &= (\{d\}; \mathbb{R}^2; \{b\}; W_{\text{ae}}; E_{\text{ae}} = (d, b, G_{\text{ae}}, J_{d,d}, d); \text{Inv}; \text{Act}), \\ \text{Act}(d) &:= F_d, \\ G_{\text{ae}}(b) &:= A_{\text{ae}}^b. \end{aligned} \quad (6)$$

where W , Inv and $J_{d,d}$ remain as defined in the hybrid automaton. The network definition does not depend on the application example. Therefore one network definition can be used for events of both systems, occurring during the bounce, the free fall and the pendulum phase.

The framework application for the network is given as

$$\begin{aligned} A_{\text{ae}}^i &= (4, I, O, N_i, \{u_j^{(l)}\}_{j=1, \dots, n_l}^{l=2,3,4}, T), \quad i \in \{d, r, p\}, \\ u_j^{(l)}(z_j^{(l)}) &= \sigma(z_j^{(l)}), \quad l = 2, 3, \quad j = \{1, \dots, n_l\}, \\ u^{(4)}(z^{(4)}) &= z^{(4)}, \\ T &= (M, A, C, S, V), \\ C &= L_{\text{cross}}, \end{aligned} \quad (7)$$

where A is the Scaled Conjugate Gradient Algorithm, M is the back-propagation method and N_i depends on the application example. The input data consists of the state vector for various initial conditions and various points in time. For each data point the occurrence of an event is evaluated

$$O_i = \begin{cases} 0, & \text{no event at } I_i, \\ 1, & \text{event occurs at } I_i. \end{cases}$$

The resulting dataset $O_i \in \{0, 1\}, i \in \mathbb{N}$ forms the output data for training. The size and structure of the state vector depends on the application example. Hence, the input for the training data is given as

$$I_i = (t_i, x_1(t_i), x_2(t_i), g)$$

The event classification is especially challenging. In most hybrid systems the occurrence of an event is rather rare. A dataset for the pendulum with various initial conditions results in $\approx 0.6\%$ events. This problem is often referred to as binary classification of imbalanced datasets [16, 5]. In [3] different resampling methods are discussed, to balance the dataset and enable successful classification. One method suggests to create synthetic observations drawn from a uniform distribution within the data of the small category. Due to the fact that the

Ex	$S = (E/\text{NoE})$	k	V
BB ₁	98485/6010853	—	[0.75, 0.15, 0.15]
BB ₂	49243/49243	$\frac{1}{2}$	[0.75, 0.15, 0.15]
BB ₃	49243/49243	$\frac{1}{2}$	[0.75, 0.15, 0.15]

Ex	Corr. pos.	Corr. neg.	False. pos.	False. neg.
BB ₁	1.2%	98.4%	0%	0.4%
BB ₂	49.9%	50%	0%	0.1%
BB ₃	50%	50%	0%	0.001%

Ex	$S = (E/\text{NoE})$	k	N
BB ₁	98485/12021707	—	[4, 30, 20, 1]
BB ₂	98485/98485	1	[4, 30, 20, 1]
BB ₃	98485/147728	$\frac{3}{2}$	[4, 30, 20, 1]

Ex	Corr. pos.	Corr. neg.	False. pos.	False. neg.
BB ₁	0.6%	49.6%	49.6%	0.2%
BB ₂	49.9%	50%	0%	0.001%
BB ₃	40%	0.6%	59.4%	0.001%

Table 1: Comparision of the classification MLP for both examples.

events are defined by physics this approach is not applicable. Another possibility is to remove a certain amount of random data from the majority class to balance the dataset. A parameter k is introduced to coordinate the balance in the training set. For each data point characterising an event, k data points classified as 'no event occurred' are added to the training set.

In Table 1 the results for all three events of the two examples are given. The classification of imbalanced data with regards to hybrid events is a challenge. Even though some of the values for wrong categorisation seem sufficiently small, the reason is not a good approximation but the ratio between the two categories.

For all three event types this approach is inapplicable. Even resampling the dataset only improves the network performance in training but the testing is still not feasible. In case of the bouncing ball example, not even the events in the training set can be classify correctly by the network. In case BB₃ at least the new data is classified correctly but the 50 events which have been classified false in training, stay incorrect.

5 Conclusion & Outlook

In the case of the event detection approach, the available descriptions of the local dynamics can be used to classify the available data. If the discrepancy between model output and system data increases an event can be identified. The gathered state vectors where the discrepancies are within a certain range, can be labelled

'no event'. This procedure results in an imbalanced dataset with two classes, the majority class 'no event' and the minority 'event'. The results listed in Table 1 show the results of a trained MLP for the classification of imbalanced binary datasets. If the data is re-sampled, as suggested in [15] and [3], the training can be successful. Unfortunately, even then, the generalised classification of events often does not work sufficiently. The case study confirms the findings in [14], that the traditional feed-forward neural network has difficulties to learn from imbalanced datasets [16]. To cope with imbalanced binary datasets in classification scenarios possible alternatives to neural networks are presented in [12]. Therefore, facilitating methods such as regression and projection into the third framework application is a future objective.

The focus of the framework was, to investigate different application scenarios for neural networks. In the case of classifying state vectors in 'event' and 'no event', the neural network performance was not sufficient. The classification experiments for both application examples have shown, that neural networks are not applicable for imbalanced datasets like this. Other methods might be more effective. Therefore, adding elements to the framework to enable an inclusion of other methods in the hybrid dynamical model is a future objective. In this context, also the preprocessing of the input data in neural network applications can be included in the framework definition. A future review of networks other than feed-forward, might require adding new elements. For example a recurrent network might require more descriptive elements than defined in the framework at the moment. If the dynamical description of each node involves several implicit formulations, it might be laborious or even impossible to formulate an explicit jump relation. In such cases, the measured output and input data of the different nodes could be used to train another network J_{ae}^b to replace the jump relation J in the model description. In addition, a combined framework application, to facilitate the replacement of events and local dynamics at the same time, can be one of the next steps.

References

- [1] Berk Altın, Pegah Ojaghi, and Ricardo G. Sanfelice. A model predictive control framework for hybrid dynamical systems. *IFAC-PapersOnLine*, 51(20):128–133, 2018.

- [2] Alberto Bemporad and Manfred Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.
- [3] Evgeny Burnaev, Pavel Erofeev, and Artem Papanov. Influence of resampling on accuracy of imbalanced classification. *arXiv:1707.03905 [cs, stat]*, page 987521, 2015.
- [4] Luca P. Carloni, Roberto Passerone, Alessandro Pinto, and Alberto L. Angiovanni-Vincentelli. Languages and tools for hybrid systems design. *Foundations and Trends® in Electronic Design Automation*, 1(1):1–193, 2006.
- [5] Nitesh V. Chawla. Data mining for imbalanced datasets: An overview. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook*, pages 875–886. Springer US, 2010.
- [6] Simon Haykin. *Neural networks and learning machines*. Prentice Hall, 3rd ed edition, 2009. OCLC: ocn237325326.
- [7] Thomas A Henzinger. The theory of hybrid automata. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science*, pages 278–292, 1996.
- [8] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, January 1989.
- [9] Andreas Körner, Stefanie Winkler, and Felix Breiteneker. Possibilities in state event modelling of hybrid systems. *SNE Simulation Notes Europe*, 28(3):109–111, 2018.
- [10] Andreas Körner. *Mathematical Characterisation of State events in Hybrid Modelling*. phdthesis, Technische Universität Wien, 2015.
- [11] Andreas Körner and Felix Breiteneker. State events and structural-dynamic systems: Definition of ARGESIM benchmark c21. *SNE Simulation Notes Europe*, 26(2):117–128, 2016.
- [12] Sung-Chiang Lin, Yuan-chin I. Chang, and Wei-Ning Yang. Meta-learning for imbalanced data and classification ensemble in binary classification. *Neurocomputing*, 73(1):484–494, 2009.
- [13] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
- [14] Yi L. Murphey, Hong Guo, and Lee A. Feldkamp. Neural learning from unbalanced data. *Applied Intelligence*, 21(2):117–128, 2004.
- [15] Giang H. Nguyen, Abdesselam Bouzerdoum, and Son L. Phung. A supervised learning approach for imbalanced data sets. In *2008 19th International Conference on Pattern Recognition*, pages 1–4. IEEE, 2008.
- [16] Jinghua Wang, Jane You, Qin Li, and Yong Xu. Extract minimum positive and maximum negative features for imbalanced binary classification. *Pattern Recognition*, 45(3):1136–1145, 2012.

Reduction of Complexity in Q-Learning a Robot Control for an Assembly Cell by using Multiple Agents

Georg Kunert¹, Thorsten Pawletta¹, Sven Hartmann²

¹Research Group Computational Engineering and Automation, Wismar University of Applied Sciences: Technology Business and Design, Philipp-Müller-Straße 14, D-23966 Wismar, Germany; *georg.kunert@cea-wismar.de

²Department of Informatics, Clausthal University of Technology, Julius-Albert-Straße 4, D-38678 Clausthal-Zellerfeld

Abstract. Production systems in Industry 4.0 are characterized by a high degree of system networking and adaptability. They are often characterized by jointed-arm robots, which have a high degree of adaptation. Networking and adaptivity increase the flexibility of a system, but also the complexity of the control, which requires the use of new development methods. In this context, the Simulation-Based Control approach, a model-based design method, and the concept of Reinforcement Learning (RL) are introduced and it is shown how a task-based robot control can be learned and executed. Afterwards, the time complexity of the Q-learning method will be examined using the application example of a robot-based assembly cell with two differently flexible system configurations. It is shown that, depending on the system configuration, the time complexity of learning can be significantly reduced when using several agents. In the studied case, the complexity decreases from exponential to linear. The modified RL structure is discussed in detail.

Introduction

Production systems of Industry 4.0 have a high degree of networking and adaptivity. The latter characterizes the flexibility of a system to adapt to changing influences [1]. Systems are often characterized by jointed-arm robots which, according to [2], have a high degree of flexibility in terms of design and control. Adaptivity and networking increase the complexity of the control software, which requires the application of new development methods. In recent years, similar methods have been established under various terms, such as *Model-Based Design* (MBD) [3], *Rapid Control Prototyping* (RCP) [4] or *Virtual Commissioning* (VC) [5]. What they have in common is that they are based on a continuous model- and simulation-based development from the design to the operation phase. The *Simulation-Based-Control* (SBC) approach [6] was developed adequately for this purpose by the Computational Engineering and Automation research group at the University of Applied Sciences in Wismar. This approach was adapted in [7] and [8] specifically for

task-oriented control development for jointed-arm robots. In [9], a connection of the SBC approach with machine learning methods based on *Reinforcement Learning* (RL) according to [10] is shown. The RL is defined by a structure of at least one agent, which has a learning method and an environment. The specific learning method is Q-learning. Based on predefined tasks and transformation modules, a control strategy is learned and automatically transformed into an SBC-compliant program. Since Q-learning is a model-free algorithm, it can be applied to various problems. However, impracticable computing times result relatively quickly, even though learning can often be accelerated for problems with limited state space by means of parallel processing and binary trees [11]. A significant reduction of the state space and, thus, the computational effort is achieved with model-based RL approaches [12]. Here the agent already has process-relevant knowledge at the beginning of the learning process, but thereby loses its universality. For problems with large state space, the computing time can be reduced by combining *artificial neural networks* (ANN) as function approximators [13], [14]. However, this increases the complexity of the software architecture. Furthermore, the design and training of the ANN requires experience and time. In contrast, the simple architecture of the original Q-learning is an advantage if the computational effort can be mastered.

In this paper, we investigate how the computing time for Q-learning of a task-based robot control can be reduced by using several agents. Two differently flexible system structures of an assembly cell are considered and the time complexity of learning with one agent compared to several agents is analyzed. The learning is performed on simulated system environments. The generation of an executable robot control based on the SBC approach and basics of the RL approach using Q-learning are discussed in the following background section.

1 Background

Starting from the adapted SBC framework for articulated arm robot systems according to [7] and [8], this section deals with the principle of integration with a machine learning process and the basics of RL based on Q-learning according to [10] and [13].

1.1 The SBC Approach

As shown in Figure 1, an SBC-based robot control is layered in analogy to the concept of the Robot Operating System (ROS) [15].

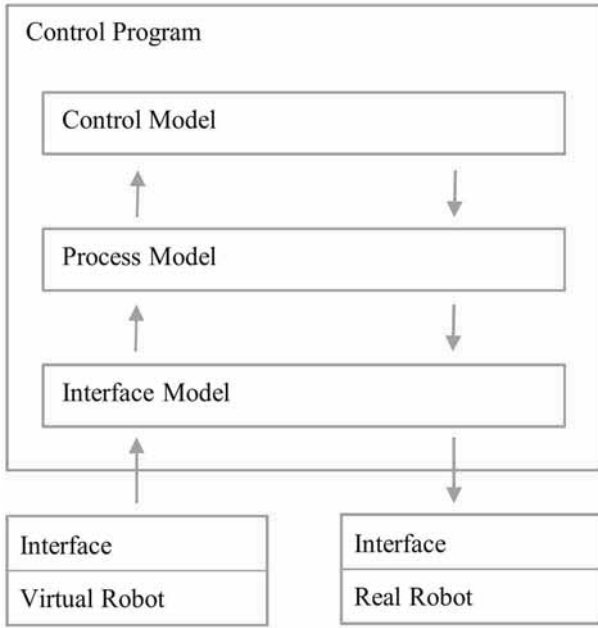


Figure 1: Structure of an SBC-based robot control.

The *Control Model* (CM) specifies the control strategy by composing predefined basic tasks. The *Process Model* (PM) implements the task transformer based on predefined task-specific modules. Additionally, the PM maps the state information. The *Interface Model* (IM) implements the interface to the hardware using a robot middleware. With the *RCV Toolbox for MATLAB* according to [16] as middleware, SBC-based robot controls can be developed and operated independently of robot manufacturers and model-based with *MATLAB/Simulink*. Virtual and real robots can additionally interact with a virtual process environment in the form of a simulation model.

With the integration of the SBC framework with a RL procedure, as introduced in [9], the task-based control

specification of the CM according to formula 2 is automatically generated from a sequence of learned state/action tuples according to formula 1:

$$[(s_0, a_0), \dots, (s_t, a_k), \dots, (s_{target}, cancel)] \quad (1)$$

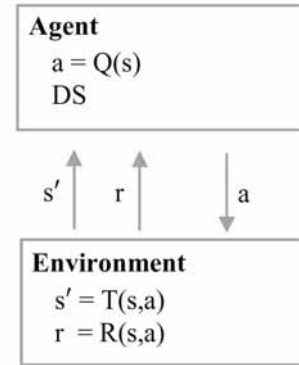
with $s_t \in S, a_k \in A, S$ state set, A action set

$$[move(\dots), pick(\dots), \dots, stop(\dots)] \quad (2)$$

Learning takes place offline using a simulated process environment. In principle, the learning algorithm could also run during the operating phase and adapt the control strategy in certain time windows if the real-time requirements are met.

1.2 Reinforcement Learning with Q-learning method

Besides supervised and unsupervised learning, the RL method forms a third class of machine learning procedures. The goal of RL is to learn a behavioral strategy $\pi: S \rightarrow A$ that assigns an action $a \in A$ to each state $s \in S$. The RL does not require explicit training data. It trains itself using a real or simulated environment according to the *trial and error principle*. The RL is based on a framework as shown in Figure 2.



T – Zustandsübergangsmodell
R – Belohnungsmodell
Q – spezifische Lernmethode
DS Datenstruktur erforschter Zustände

Figure 2: RL framework with Q-learning method.

In model-free RL, the agent only knows the allowed action set A at the start of training. The environment is defined by different states, $s \in S$. When an action $a \in A$ takes effect, the environment determines a subsequent state s' with the state transition model $T: S \times A \rightarrow S$ and

computes a reward value $r \in R$ for the current action. The subsequent state s' and the reward r are sent back to the agent. During training, the agent receives information about the possible states of the environment and the benefits of actions through iterative interaction, and gradually learns a behavioral strategy π . After completion of all training episodes, the behavioral strategy π is derived from the Q-matrix.

Learning takes place in phases, also known as episodes. These are independent of each other, always start in an initial state s_0 of the environment and end when a target state s_{target} or abort state s_{abort} is reached. At the beginning of the training, the agent selects an action $a \in A$ purely randomly (*exploration*). As the learning process progresses, the agent increasingly uses the knowledge it has acquired to select an action (*exploitation*). The ratio of exploration to exploitation is adjusted in the course of training.

Q-learning is based on a table function called Q-matrix. A matrix element $Q(s, a)$ represents the benefit Q of an action a when it is performed in the state s of the environment. From the cardinality $|A|$ follows the column dimension of Q . The row dimension of Q grows dynamically during the training with the number of states explored. The data of explored states are stored in an indexed data structure DS based on the row index of Q . This allows the agent to clearly recognize already explored states. The first episode of the training starts with an empty Q-matrix and an empty data structure. Subsequent episodes build on the already acquired knowledge in Q and DS . After each interaction with the environment, the agent checks whether the received state s' is known. If not, the Q-matrix is extended by a new row and the state data are added to the data structure DS . The successive adjustment of the Q-values results from formula 3 according to [13]:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \cdot \max_{a'} Q(s', a') - Q(s, a) \right] \quad (3)$$

The updated Q-value of the current state/action tuple (s, a) is calculated from the previous Q-value, the currently received reward r , and the maximum Q-value of all possible actions in the currently received subsequent state s' . The influence of the individual variables is determined by the hyperparameters: (i) discount factor γ and (ii) learning rate α . The discount factor controls the

influence of rewards expected in the future and the learning rate controls the influence of the current observation. In environments with deterministic behavior, a high learning rate can be applied with up to $\alpha = 1$.

2 Application Example

An automated assembly cell (AC) with an articulated arm robot for the production of different assemblies is considered as an example. By means of RL, product-compliant assembly sequences are to be learned, on the basis of which robot controls can be generated according to Section 1.1. Figure 3 shows the system layout of the AC.

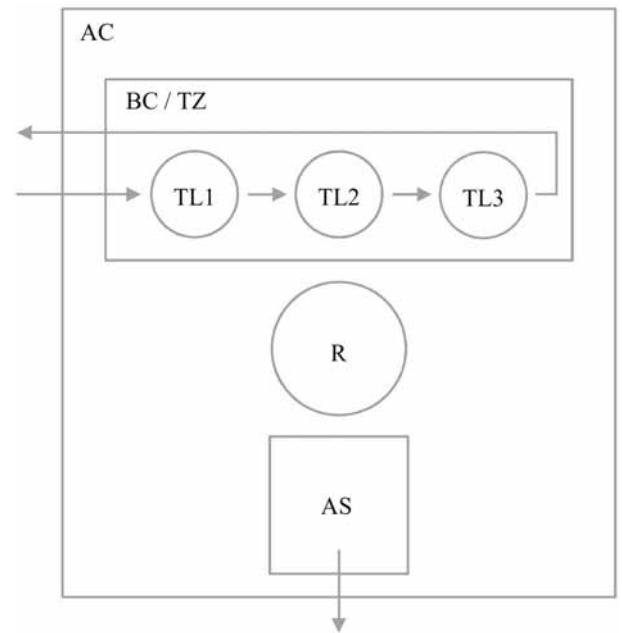


Figure 3: System layout of the assembly cell with maximum three TLs as TZ.

The AC consists of the articulated arm robot (R), an assembly station (AS) and a belt conveyor (BC). The BC feeds input parts from upstream production sections and serves as a transfer zone (TZ) to the robot. Depending on the specific system configuration, different numbers of transfer locations (TL) are possible, which influences the system flexibility. Unused input parts and assembled modules are automatically removed. The sequence in which the individual parts enter the AC is unknown and is assumed to be random.

Figure 4 shows an example of an assembled module as an exploded view, whose assembly is examined below for two system variants of the AC:

1. Minimum variant with only one TL as TZ and
2. Variant with three TLs as TZ.

For the desired complexity consideration, the assembly is reduced to a pick & place application. The assembly of the module in Figure 4 is subject to six rules.

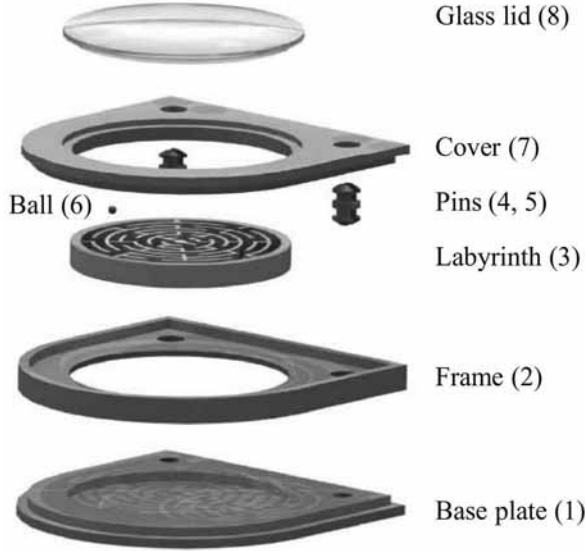


Figure 4: Exploded view of an assembled module according to [7].

Four of the rules are shown in [7]. Rules 5 and 6 were added:

1. Base plate (1) must be mounted first.
2. Pins (4, 5) may only be mounted after cover (7).
3. Comprehensive parts must be mounted after enclosed parts.
4. The glass cover (8) must be mounted last.
5. Each part is only assembled once and the pins (4, 5) are not identical.
6. Gripping attempts on empty TLs of the TZ are prohibited.

In the following two sections, the learning of an assembly strategy with first one agent for both system variants and then using several agents for the second variant is examined.

3 Learning with one Agent

According to Section 1.2, the RL framework consists of an agent and an environment. In the application case under study, the robot forms the agent and the BC and AS form the environment. In this section, the learning of an

assembly strategy based on this RL framework is examined for the two system variants of the AC.

3.1 System variant with one TL as TZ

In the case of only one TL, the robot has two possible actions $A = \{0, 1\}$. The action $a = 0$ encodes the task None and the action $a = 1$ encodes the task *Pick&Place*. The state of the environment results from the TZ allocation and the assembly state of the module on the AS. In the case of only one TL, there are nine possible states of the TZ with $S_{TZ} = \{0, 1, \dots, 8\}$. The state value 0 stands for no component and the values greater than zero for a component according to the part numbers in Figure 4. The final assembled module consists of eight individual parts. Accordingly, the assembly state on the AS can be represented by an 8-tuple. Each tuple element describes the non-assembly or assembly of a component by the values $0, 1, \dots, 8$.

A state $s \in S$ of the entire environment is, therefore, described by a 9-tuple. The first eight elements describe the assembly state on the AS and the ninth element the state of the TZ. Figure 5 shows the representation of states $s \in S$ of the environment with a state vector (SV).

Initial state $s_0 \in S$

0	0	0	0	0	0	0	0	X
---	---	---	---	---	---	---	---	---

A possible subsequent state $s' \in S$

1	0	3	0	0	0	0	0	8
---	---	---	---	---	---	---	---	---

Target state $s_{target} \in S$

1	2	3	4	5	6	7	8	X
---	---	---	---	---	---	---	---	---

Figure 5: Mapping of the state as a state vector (SV).

State s_0 represents the initial state, s' a possible later subsequent state and s_{target} the target state. The indicated state s' encodes an assembly of the components' base plate and labyrinth as well as the allocation of the TZ with a component glass lid. The target state s_{target} is reached when the first eight elements of SV are not equal to zero. The character X in the initial and target state stands for any state $s_{TZ} \in S_{TZ}$ of the TZ.

The environment reacts to an action a of the agent, as shown in Figure 2, with a reward value r and a subsequent state s' . The reward model R of the environment defines three possible rewards for an action $a = 1$ based on the six assembly rules according to Section 2:

- $r = -\infty$, if the action is not allowed,
- $r = 0$, if the action is allowed and $s' \neq s_{target}$ and
- $r = 1$, if the action is allowed and $s' = s_{target}$.

The action $a = 0$, i.e. a refusal to mount the component on the TL, always leads to the reward value $r = 0$.

The state transition model T works according to the Markov Decision Process (MDP) paradigm. Listing 1 shows the specification of T in MATLAB pseudocode.

```
%A={0,1} action set of the agent
1 num_parts = 8;
2 idx = num_parts + 1;
3 a = action(agent); %current action
%Pick&Place only
4 if a > 0
5     SV(SV(idx)) = SV(idx);
6     SV(idx) = 0;
7 end
%Pick&Place and None
8 SV(idx) = randi(num_parts + 1) - 1;
```

Listing 1: State transition model for an TZ with one TL.

With a permitted action $a = 1$ (*Pick&Place*), the last element of the SV that represents the part in the TC is restored according to its value in the SV (lines 4-7), thereby updating the assembly state. In addition, for each allowed action $a = 1$ or $a = 0$ the last element of SV is assigned a random integer value from the interval $[0, num_parts]$, which represents a new input part in the TC (line 8).

Each episode starts with the transfer of a start state s_0 of the environment to the agent. A reward is not transferred at the beginning of an episode. Learning is done as described in Section 1.2. The column dimension of the Q-matrix corresponds to the cardinality of the action set $|A| = 2$. An episode ends when the agent receives a reward, $r = 1$, from the environment. The training must include enough episodes to learn a strategy, π . It should be noted that, with the random generation of new input parts in the TZ, the state transition model T is subject to stochastic influences.

Without assembly rules, this system structure results in $2^n(n + 1) - n$ states of the environment depending on the number of parts n to be assembled. Due to assembly rules, the number of states is considerably reduced. For the assembly under consideration with eight components, 144 states result according to [7].

The complexity of an algorithm as a function of the

input data is described with the big O notation [17]. An empirical analysis showed that a linear time complexity $O(n)$ of the learning process resulted depending on the number of components, n , to be assembled. The computing time for learning an assembly strategy was less than one hour on a standard PC with an implementation in MATLAB.

3.2 System variant with three TLs as TZ

In the variant with three TLs the allocation of the TZ is analogous to a shift register. In fixed time units, the BC moves one position to the right. The robot always has access to all three TLs. This results in four possible actions for the robot, $A = \{0,1,2,3\}$. The action $a = 0$ again encodes the task *None* and the other three actions encode the task *Pick&Place* with different parameterization depending on the TL to be approached.

The state of the environment expands by two elements to an 11-tuple. The first eight elements describe the assembly state on the AS. The other three elements each describe the non-occupancy or occupancy of a TL. Due to the three TLs, the number of possible states of the environment increases. For the TZ with three TLs and eight possible input parts, as well as the case of non-occupancy of a TL, results in 729 states $s_{TZ} \in S_{TZ}$ with $S_{TZ} = \{(0,0,0), (1,0,0), \dots, (5,1,1)\}$. The values 1 to 8 encode a component according to the part numbers in Figure 4 and the value 0 encodes the non-occupancy of a TL.

The calculation of rewards $r \in R$ is analogous to the variant with only one TL. The state transition model T , which has been extended to a TZ with three TLs, shows Listing 2.

```
%A={0,1,2,3} action set of the agent
1 num_parts = 8;
2 a = action(Agent); //current action
3 idx = a + num_parts;
//Pick&Place only
4 if a > 0
5     SV(SV(idx)) = SV(idx);
6     SV(idx) = 0;
7 end
%Pick&Place and None
%Move TL allocations (TL1→TL2→TL3)
%cardinality |A| is 4
8 for k = |A| - 1 : -1 : 2
9     idx = k + num_parts;
10    SV(idx) = SV(idx - 1)
11 end
```

```

%generate new input part on TL1
12 idx = num_parts + 1;
13 SV(idx) = rand_i(num_parts +1) - 1;

```

Listing 2: State transition model for the TZ with three TLs.

The 11-tuple representing the state of the environment is again implemented as a state vector (SV). If the agent selects a permitted action $a = \{1,2,3\}$ for a Pick&Place task (line 2), the input part is taken from the corresponding TL in the TZ and mounted at the AS by the robot (line 4-7). Subsequently, for each allowed action $a \in \{1,2,3\}$ of the robot, the shift register movement is realized by the BC (lines 8-11) and a new input part is randomly generated in the form of an integer value in the interval $[0, \text{num_parts}]$ for the first TL in the TZ (lines 12-13).

Learning a behavioral strategy π is analogous to the representation in Section 3.1. However, despite the applicable assembly rules, the state space of the environment increases exponentially with $144 \cdot 9 \cdot 9 = 11644$ states. The complexity of the learning algorithm corresponds to $O(n^c)$ with n the number of components to be assembled and c the number of TLs in the TZ. In an empirical study, the computing time was about 10 hours and it was about ten times longer than the computing time for the variant with only one TL.

4 Learning with Multiple Agents

Based on the structure of the application problem, a multi-agent approach to reduce the computing time appears to be appropriate for the second system variant with three TLs in the TZ. Figure 6 shows an RL framework consisting of three agents, an environment, and a *management* component.

The three agents are identical. Their structure and behavior correspond to the individual agent in Section 3.1, and their action set $A = \{0,1\}$ represents the two tasks *None* and *Pick&Place*. The models T and R of the environment correspond to the representation in Section 3.2. Accordingly, the environment reacts to the action set $A = \{0,1,2,3\}$. In contrast to the original RL framework, the agents and the environment do not communicate directly with each other, but via the intermediate management. The management decomposes each state s' as shown in Figure 7. The first eight elements of s' , which encode the assembly state on the AS, are passed on to all three agents. From the ninth to the tenth element, which

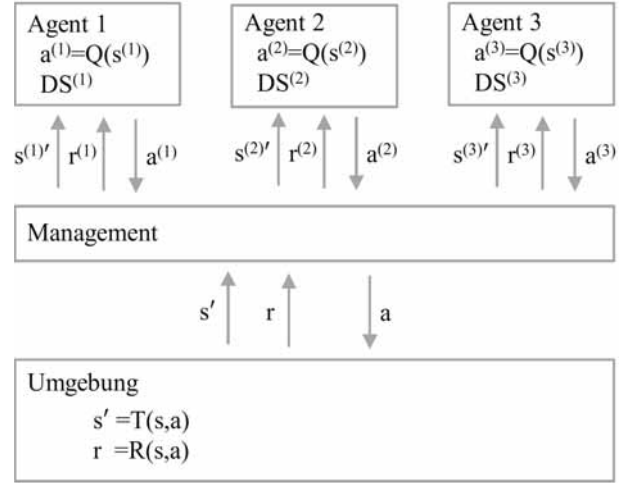


Figure 6: RL framework with multiple agents.

Next state s' of the environment

1	0	0	0	0	0	0	0	8	2	6
---	---	---	---	---	---	---	---	---	---	---

Next state $s^{(1)'} for agent 1$

1	0	0	0	0	0	0	0	8
---	---	---	---	---	---	---	---	---

Next state $s^{(2)'} für agent 2$

1	0	0	0	0	0	0	0	2
---	---	---	---	---	---	---	---	---

Next state $s^{(3)'} for agent 3$

1	0	0	0	0	0	0	0	6
---	---	---	---	---	---	---	---	---

Figure 7: Decomposition of a next state s' of the environment by the management.

code the allocation of the three TLs in the TC, each agent only receives the state of one TL at a time. This reduces the state space from the viewpoint of each individual agent to 144 states, analogous to the agent in Section 3.1.

An episode begins with the decomposition of the initial state s_0 of the environment by the management, analogous to Figure 7. Then, the management sends to each agent i the corresponding reduced state vector $s^{(i)'}$, as shown in Figure 6. Each agent reacts individually with an action $a^{(i)} \in \{0,1\}$. The management selects one of the three actions according to the following criteria:

1. An action $a = 1$ (*Pick&Place*) is preferred over an action $a = 0$ (*None*).

2. If several agents send an action $a = 1$, the agent with the highest index i is selected. The prioritization of the agents follows from the way the TZ works as a shift register. Parts on the second and first TL are still available in the next and next but one cycle. According to Figure 7, *Agent 1* has the lowest priority and *Agent 3* the highest.

The management remembers the selected agent. If an action $a = 0$ was selected, the action is forwarded to the environment unchanged. If an action $a = 1$ was selected, the action is converted into a value of the action set $A = \{1,2,3\}$ based on the index i of the agent and sent to the environment as action a . The environment reacts to the action with a reward r and a follow-up state s' . These are calculated with the reward model R as well as the state transition model T according to Section 3.2. The management sends the reward r as $r^{(i)}$ to the agent i , whose action was selected before. The other agents receive an empty reward value. An empty reward means that there is no learning for the agent in this iteration step because its action has been discarded. The subsequent state s' is decomposed by management as described and the reduced states $s^{(i)'}$ are sent to the agents. Each agent learns a behavior strategy $\pi^{(i)}$ related to the TL assigned to it. Finally, an overall assembly strategy π is derived from the individual strategies.

The number of necessary episodes is hardly different from learning with one agent. Since only one agent is active in each iteration step, the state space to be analyzed has a maximum of 144 states. The empirical investigations revealed a linear complexity $O(n)$ of learning depending on the number n of components to be assembled. The multi-agent approach is scalable regarding the selection of components in the TZ and the complexity is, therefore, independent of the number of TLs. The computing time requirement is reduced to about one hour.

5 Conclusion and Outlook

Starting from the basics of RL using the method of Q-learning and the model-based SBC approach, it was shown how, in principle, a task-based control can be learned and executed.

Subsequently, the learning of a typical pick and place strategy for two differently flexible system structures was considered using the example of a robot-based assembly

cell. The focus of the consideration was the time complexity of learning. For the simple system structure with binary selection option, it was found that the learning algorithm has a linear complexity $O(n)$ depending on the number n of assembled components. In contrast, the second system structure with c many simultaneous choices per assembly step had an exponential complexity $O(n^c)$ of learning.

As a result, an RL multi-agent framework with the Q-learning method was designed for the second system structure. It could be shown that, for learning with the multi-agent approach, a linear complexity $O(n)$ results as a function of the number n of components to be assembled and that the complexity is independent of the number of simultaneous choices due to the scalability of the approach.

The application example under study is characterized by learning an assembly sequence that conforms to the assembly. The different system structures were purposefully modelled as environments for the RL according to the MDP paradigm. In subsequent investigations, further influencing variables of a production process are to be taken into account during learning, such as the introduction of input parts as a function of buffer capacities. This requires the integration of the RL with a more complex dynamic simulation model of the production, which was not explicitly developed according to the MDP paradigm. Conceptual approaches for such simulation-based RL experiments are presented in Schmidt [18] and Adams [19].

References

- [1] Bundesministerium für Wirtschaft und Energie. *Was ist Industrie 4.0?* (Federal Ministry for Economic Affairs and Energy. *What is Industry 4.0?*) <https://www.plattform-i40.de/PI40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html> [Retrieved 27-July-2020].
- [2] Hägele M., Nilsson K., Pires J.N. Industrial Robotics. In: Siciliano B., Khatib O., editors. *Handbook of Robotics*. Berlin: Springer Pub; 2008. 963-986.
- [3] Nicolescu G., Mosterman P.J. *Model-Based Design for Embedded Systems*. Boca Raton / FL: CRC Press; 2010.766 p.
- [4] Abel D., Bollig A. *Rapid Control Prototyping – Methoden und Anwendungen (Methods and Applications)*. Berlin: Springer Pub., 2006, 400 p.
- [5] Turnbull C. *What is Virtual Commissioning?* <https://virtualcommissioning.com/what-is-virtual-commissioning/> [Retrieved 27-July-2020].

- [6] Pawletta T., Pawletta S., Maletzki G.: Integrated Modeling, Simulation and Operation of High Flexible Discrete Event Controls. In I. Troch, F. Breitenacker, editors, *Proc. Mathematical Modelling - MATHMOD 2009* Feb, Vienna. Argesim Report No. 35, 13 p., ISBN 978-3-901608-35-3
- [7] Maletzki, G. *Rapid Control Prototyping komplexer und flexibler Robotersteuerungen auf Basis des SBC-Ansatzes (Rapid control prototyping of complex and flexible robot controls based on the SBC approach)* [Dissertation]. Universität Rostock / Hochschule Wismar; 2013. In: ASIM FBS 25 doi: 10.11128/fbs.25.
- [8] Freymann, B. *Aufgabenorientierte Multi-Robotersteuerungen auf Basis des SBC-Frameworks und DEVS (Task-oriented multi-robot controls based on the SBC framework and DEVS)* [Draft Dissertation]. TU Clausthal / Hochschule Wismar; 2020 (unpublished).
- [9] Kunert G. Pawletta T. Generating of Task-Based Controls for Joint-Arm Robots with Simulation-based Reinforcement Learning. *SNE – Simulation Notes Europe*. 2018; 28(4):149-156. doi:10.11128/sne.28.4.1044
- [10] Sutton R., Barto A. *Reinforcement Learning*. 2nd Edition. Cambridge/ MA: MIT Press; 2012. 334 p.
- [11] Jammer D., Pawletta S., Kunert G., Pawletta T. Beschleunigung eines Reinforcement-Learning-Algorithmus durch Parallelverarbeitung für Robotikanwendungen (Accelerate a reinforcement learning algorithm through parallel processing for robotics applications.). In Durak U. et al., editors. *Proc. ASIM STS/GMMS Symposium*; 2019 Feb; Braunschweig. Wien: ARGESIM Verlag. 49-52. doi: 10.11128/arep.57.
- [12] The MathWorks. Reinforcement Learning With MATLAB – Part 1. Ebook. The MathWorks Inc.; 2019. 24 p.
- [13] Russel S., Norvig P. *Artificial Intelligence: A Modern Approach*. 4th Edition. Cambridge / MA: MIT Press; 2020. 1115 p.
- [14] Zai A., Brown B. *Deep Reinforcement Learning in Action*. Shelter Island / NY: Manning Pub.; 2020. 359 p.
- [15] ROS-Industrial. *Homepage*. <https://rosindustrial.org> [Retrieved 13-May-2020].
- [16] Deatcu, C., Freymann, B., Schmidt, A., Pawletta, T. MATLAB/Simulink Based Rapid Control Prototyping for Multivendor Robot Applications. *SNE – Simulation News Europe*. 2015; 25(2): 69-78. doi:10.11128/sne.25.2.1029.
- [17] Filho W.F. *Computer Science Distilled*. Las Vegas / NV: Code Energy LLC Pub.; 168 p.
- [18] Pawletta T., Durak U., Schmidt A. Modeling and Simulation of Versatile and Adaptable Systems with an Application in Engineering. In Zhang L. et al., editors, *Model Engineering for Simulation*. Elsevier Inc. Pub., 2019, Chap. 18, 29 p.
- [19] Adams S. et al. Reinforcement Learning from Simulated Environments: An Encoder Decoder Framework. In *Proc. SCS SpringSim '20*, 2020 May 19-21, Fairfax / VA, 12 p.

Visual Analytics for Data-Driven Analysis in Semiconductor Manufacturing

Patrick Boden*, Sebastian Rank, Thorsten Schmidt¹

¹Chair of Logistics Engineering, Technische Universität Dresden, Münchner Platz 3, 01187 Dresden, Deutschland;

*patrick.boden@tu-dresden.de

Abstract. This article dedicates to the process of data visualization. In complex production systems, visualizations are essential to support explorative data analysis and the communication of findings. Quite often, the results of data analyses are static KPIs or visualizations that can only be adapted by extensive preprocessing. This usually requires specific knowledge of software tools that only a few employees have, which hinders an easy adaptation by stakeholders or even a reasonable subsequent use. A new approach called Visual Analytics pretends to overcome the mentioned issues through interactive visualizations combined with automated algorithms.

1 Motivation

Analysis of system status is crucial for management, control, and optimization of production and logistics systems. For this purpose, industrial applications acquire an increasing amount of data, which must be processed by system experts to generate and communicate information. This process requires the visualization of data because meaningful visualizations support exploratory analysis and enable to share the findings with stakeholders.

Quite often, the results of data analysis are static KPIs, diagrams, or tables. Hence, the value of the analysis is limited and the results can only be used to find answers to initially specified questions. Furthermore, static figures and visualizations are usually not suitable to serve as a basis for communication to stakeholders with different needs like the form of presentation or the selection of the underlying data. Working with un-customized visualizations results in misinterpretations and loss of time due to the time-consuming adaptation of the analysis. Numerous scientific papers about the generation of meaningful visualizations confirm this.

Fortunately, latest approaches intended for exploratory data analysis allow a visual and interaction-based analysis even of a large amount of data. Therefore, sophisticated software tools usually applied in the fields of data management and data analysis are combined with interactive visualizations. They allow the creation of interactive, adaptable dashboards which should allow analysis in depth. The method is also known as Visual Analytics and is the focus of this paper.

The present work summarizes the most important lessons

learned from the implementation and application of Visual Analytics applications for visualization tasks in the semiconductor industry.

This work based on the findings gathered in several research projects in the semiconductor industry, as well as literature research in the field of Visual Analytics and own software demonstrators. As the semiconductor domain is characterized by highly complex production systems, new approaches that support data analysis are of particular interest.

2 State of Technology

The manufacturing of semiconductor products is highly demanding from a technological and intralogistics point of view. It requires the execution of hundreds of production steps in a cost-intensive production environment to process a wafer which in the end holds the semiconductor chips.

Scientific publications, industry reports, and presentations provide numerous examples for the visualization of data. See for example Ben-Salem et al. (2016), Hammel et al. (2012) and van Roijen et al. (2014).

Usually, results are prepared in form of tables and diagrams. Most common are histograms, pie charts, or box plots. The way of data selection and aggregation, as well as the presentation form rely on the preferences of the author's document which involves the risk of e.g. misinterpretation or wrong focus depending on the addressees' needs.

From our experience, for data analysis and visualization nearly without exception standard software tools are applied. These include for small data sets Excel and for more extensive data sets software tools such as R, Matlab, or Python. Common to all is a static diagram as the result of the visualization process. However, if adaptable graphs are needed, these tools require deep and specific knowledge as source code modifications become necessary.

In general, the following limitations are associated with static reporting:

- Limited access to the data analysis process and the neglect of human capabilities

- Shortening of information and a lack of adaptability of the analysis
- Specific knowledge of data analysis tools and limited possibilities for collaborative work

Concepts from the field of Visual Analytics could contribute to overcome these limitations.

3 The Concept of Visual Analytics

Visual Analytics allows dynamic analysis of data through interactive graphical data exploration combined with automated algorithmic techniques for data analysis. One of the definitions for the term Visual Analytics is provided by Keim et al. (2008): "Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of vast and complex data sets." The approach on the one hand focuses on the strength of computerized analysis, like statistics, data management, and machine learning capabilities. On the other hand, it involves human capabilities such as human cognition or human perception. That allows the analysis of complex data sets in an intuitive way. Patterns found in the analysis can be investigated in several detail levels. New metrics, as well as additional data sets, can enrich the analysis.

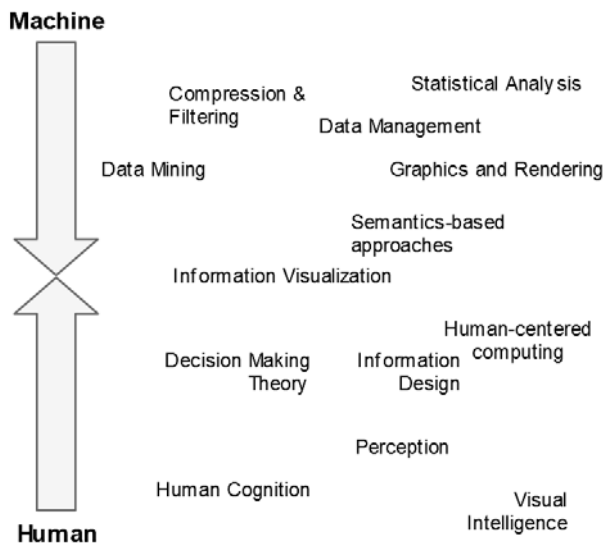


Figure 1: Disciplines integrated into the field of Visual Analytics (see Keim et al. 2008)

An extract of the related disciplines is given in figure 1. Following Keim et al. (2008) Visual Analytics enables the synthesizing of information from massive data sets,

the detection of expected and even unexpected features, and effective communication of findings. For more details about techniques and applications (see Sun et al. 2013).

Visual Analytics requires software tools that allow sophisticated interactive data analysis and visualizations even for large data sets. For this purpose, interaction mechanisms must be provided as well as the automated adaptation of the analysis. Interaction can be enabled in different ways. First, interaction with graphic elements of the visualization is possible by zooming, hovering, or selecting. In this way, parts of the visualization can be examined more closely and areas of interest can be selected for further analysis. Another form of interaction is the selection, rearrangement and expansion of the underlying data. Also, the specific application of statistical methods is an essential way of interaction. Both refers to the use of comparatively simple calculation methods such as averaging up to statistical tests to confirm hypotheses or machine learning applications. The basis for Visual Analytics applications is efficient access to data. In order to achieve the full advantage of Visual Analytics, heterogeneous data from various sources in different levels of aggregation are required and need to be processed in a fast way.

4 Conclusion

So far, data analysis in science and industry quite often has been based on static visualizations. Data was highly aggregated and an investigation of addressees' related aspects was nearly impossible.

That is a critical limitation, especially in complex production and logistics systems that depend on various experts' collaborative work. This results in unnecessary delays due to time-consuming adaptation of the analysis and severely limited access to and understanding of the data.

Software tools from the area of Visual Analytics offer intuitive access to the analysis of the data. Dashboards prepared by experts can be adapted and extended even with limited knowledge of the respective domain. By combining human strengths with automated data processing, new insights can be gained.

References

- [1] Ben-Salem, A.; Yugma, C.; Troncet, E.; Pinaton, J.: AMHS design for reticles in photolithography area of an existing wafer fab. In: 27th Annual SEMI Advanced

Semiconductor Manufacturing Conference (ASMC), 2016, pp. 110-115.

[2] Hammel, C.; Schmidt, T.; Schöps, M.: Network optimization prior to dynamic simulation of AMHS. In: Proceedings of the 2012 Winter Simulation Conference, 2012, pp. 1956-1966.

[3] Keim, D.; Andrienko, G.; Kekete, J.-D.; Görg, C.; Kohlhammer, J.; Melancon, G.: Visual Analytics: Definition, Process, and Challenges. In: Information Visualization, 2008, pp. 154-175.

[4] Sun, G.-D.; Liang, R.-H.; Liu, S.-X.: A Survey of Visual Analytics Techniques and Applications: State-of-the-Art Research and Future Challenges. In: Journal of computer science and technology, 2013, pp. 852-867.

[5] Van Roijen, R.; Joshi, P.; Bailey, D.; Conti, S.; Brennan, W.; Findeis, P.: Defect reduction by nitrogen purge of wafer carriers. In: ASMC 2013 SEMI Advanced Semiconductor Manufacturing Conference, 2013, pp. 338-341.

Note

The project iDev40 has received funding from the ECSEL Joint Undertaking under grant agreement No. 783163. The JU receives support from the European Union's Horizon 2020 re-search and innovation program. It is co-funded by the consortium members, grants from Austria, Germany, Belgium, Italy, Spain and Romania. It is coordinated by Infineon Technologies Austria AG.

The information and results set out in this publication are those of the authors and do not necessarily reflect the opinion of the ECSEL Joint Undertaking.



Simulationsgestützte Auslegung von Reglern mithilfe von Machine Learning

Dominic Brown^{1*}, Martin Strube¹

¹Institut für Kommunikationssysteme und Technologien, Ostfalia Hochschule für angewandte Wissenschaften, Salzdahlumer Str. 46/48, 38302 Wolfenbüttel; *dominic.brown@itison-ikt.de

Abstract. Ein zunehmend interessantes Einsatzgebiet der künstlichen Intelligenz (KI) stellt die Auslegung von Reglern für technische Systeme mithilfe von Machine Learning und dabei besonders mit Reinforcement Learning dar. Die Anwendung von Machine Learning ermöglicht die Auslegung von Reglern, ohne die üblicherweise mit dem Entwurfsprozess einhergehenden Aufwände für menschliche Experten. Stattdessen wird durch die Verwendung von neuronalen Netzen in Kombination mit Reinforcement Learning ein neuronaler Regler entwickelt, der die Regelung eines technischen Systems selbstständig erlernt. Dabei ist das Vorhandensein eines Simulationsmodells des zu regelnden Systems bei vielen Anwendungen eine Grundvoraussetzung für einen KI-basierten Ansatz zur Reglerauslegung. In dem Beitrag wird dargestellt, wie ein neuronaler Regler mit Reinforcement Learning die Regelung eines nichtlinearen Systems in Form eines inversen Pendels erlernt. Es werden die genutzte Lernmethode und die einzelnen Phasen beim Training des neuronalen Reglers beschrieben. Durch einen Vergleich der KI-basierten Reglerauslegung mit und ohne Simulationsmodell wird anschließend erläutert, welche Vorteile das simulationsgestützte Auslegen von Reglern mithilfe von Machine Learning bietet. Abschließend werden der neuronale Regler und ein konventioneller Regler gegenübergestellt und deren Verhalten bei der Regelung des inversen Pendels verglichen.

Einleitung

Der Einsatz von künstlicher Intelligenz (KI) innerhalb technischer Systeme gewinnt zusehends an Bedeutung und gilt als ein Innovationsmotor in der Fertigungs- und Verarbeitungsindustrie. „Durch den Einsatz von KI-Technologien soll die Effizienz und Effektivität industrieller Prozesse gesteigert werden. [...] In der Regel ist für die Bewältigung komplexer und komplizierter Prozessherausforderungen Erfahrungswissen und intelligentes Vorgehen erforderlich. Daher erscheint KI neben einfachen Wenn-dann-Routinen und klassischer Automatisierungs- und Regelungstechnik sehr gut geeignet, komplexe Situationen in industriellen Prozessen zu meistern.“ [1].

Bei der Reglerauslegung mit den Methoden der klassischen Regelungstechnik analysiert ein menschlicher Experte das technische System und legt anschließend

eine Regelung für das System aus. Dabei muss der Experte über die notwendigen Kenntnisse und Fähigkeiten verfügen, um die komplexe Struktur des technischen Systems zu analysieren und einen geeigneten konventionellen Regler auszulegen. Ein konventioneller Regler ist durch seine linearen Eigenschaften für einige Anwendungen nur bedingt geeignet. "These conventional controllers are linear in nature and have a fixed control law. However many systems where these controllers are used are very non-linear and have changing characteristics depending on the operational parameters." [2].

Als Alternative zu konventionellen Reglern haben sich neuronale Regler etabliert, wie aus Veröffentlichungen zum Thema [3], [4], [5] hervorgeht. Durch den Einsatz künstlicher neuronaler Netze in Kombination mit Machine Learning und der Lernmethode Reinforcement Learning wird ein neuronaler Regler ausgelegt. Der neuronale Regler erlernt die Regelung eines technischen Systems durch die Beziehung zwischen den Eingangs- und Ausgangsgrößen des Systems. Dadurch kann ein Regler ohne die Aufwände menschlicher Experten ausgelegt werden, die normalerweise mit dem Entwicklungsprozess eines Reglers verbunden sind. Die Vorteile des neuronalen Reglers liegen darin, dass Abweichungen von der erlernten Systemdynamik gut kompensiert werden können und durch seine nichtlinearen Eigenschaften ist der neuronale Regler prädestiniert für die Regelung nichtlinearer Systeme.

Ebenso wie bei der klassischen Regelungstechnik ist beim KI-basierten Ansatz zur Reglerauslegung das Vorhandensein eines Simulationsmodells vor Vorteil, in manchen Anwendungsfällen sogar zwingend notwendig.

1. Inverse Pendel

Das inverse Pendel ist eine klassische Aufgabenstellung der nichtlinearen Regelungstechnik. Das Prinzip des inversen Pendels findet sich auch in praktischen Anwendungen wieder. Beispielsweise beim Ausbalancieren eines Segway Personal Transporter oder bei den aufrecht

landenden Raketen von SpaceX. Hier wird die Bauform eines rotierenden inversen Pendels genutzt, das einen Aktor und zwei Freiheitsgrade besitzt. Das System des rotierenden inversen Pendels ist in Abbildung 1 dargestellt. Es besteht aus einem Pendel, das drehbar an einem Arm gelagert ist. Der Arm ist an einer Welle befestigt, die über einen Elektromotor mit einem Drehmoment beaufschlagt wird.

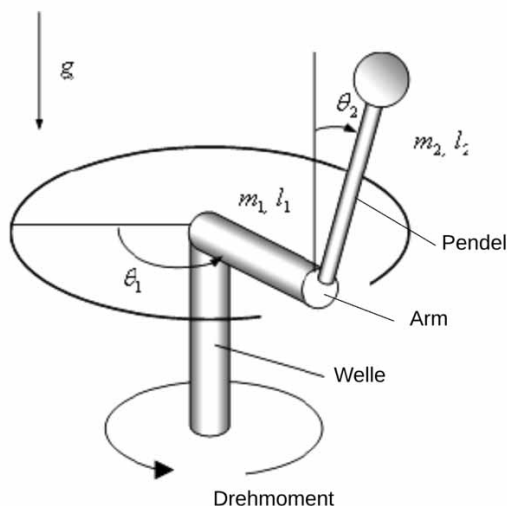


Abbildung 1: Rotierendes inverses Pendel [6]

Bei der Regelung eines rotierenden inversen Pendels wird zunächst das Pendel aus der stabilen Ruhelage in die instabile Ruhelage aufgeschwungen, also senkrecht nach oben zeigend. Dafür muss durch den Regler die benötigte Spannung an den Elektromotor anlegt werden. Dadurch wird ein Drehmoment auf die Welle aufgebracht und Bewegungen mit dem Arm ausgeführt, die das Pendel nach oben schwingen. Anschließend soll das Pendel vom Regler in der instabilen Ruhelage ausbalanciert werden. In der instabilen Ruhelage wird das Pendel den Zustand der geringsten Gesamtenergie anstreben, der erreicht wird, wenn das Pendel umfällt und die stabile Ruhelage erreicht hat. Um das Pendel in der instabilen Ruhelage auszubalancieren, muss mit dem Elektromotor ein Drehmoment aufgebracht werden, um Ausgleichsbewegungen mit dem Arm auszuführen.

Als rotierendes inverses Pendel wird der Quanser QUBE-Servo 2 verwendet. Beim Quanser QUBE-Servo 2 ist der Bewegungsbereich für den Arm durch Anschläge begrenzt. Ein optischer Drehimpulsgeber erfasst die Winkelposition des Arms und die Winkelposition des Pendels. Die Winkelposition wird zur Berechnung der Winkelgeschwindigkeit und der Winkelbeschleunigung

verwendet. Als Schnittstelle wird das QFLEX 2 USB-Panel (USB 2.0) genutzt.

2. Reinforcement Learning

Machine Learning, im Deutschen maschinelles Lernen, gilt als Schlüsseltechnologie der künstlichen Intelligenz. Die Grundidee des maschinellen Lernens besteht darin, Computer mit speziellen Algorithmen in die Lage zu versetzen, selbstständig Lösungen für ein konkretes Problem zu finden. Ähnlich wie beim Menschen spielt dabei die Weiterentwicklung durch gewonnene Erfahrung eine wichtige Rolle.

Beim maschinellen Lernen werden u.a. künstliche neuronale Netze genutzt, deren Struktur dem Aufbau des menschlichen Gehirns nachempfunden ist. Mit den dazugehörigen Lern-Algorithmen können neuronale Netze ähnlich wie Menschen durch Versuch, Wiederholung und Feedback, zu den erzielten Ergebnissen, lernen.

Dabei werden unterschiedliche Lernmethoden genutzt, wie z.B. das Reinforcement Learning. Beim Reinforcement Learning (RL), im Deutschen bestärkendes Lernen, lernt ein aus neuronalen Netzen bestehendes KI-System (Agent) nach dem Grundsatz Versuch und Irrtum durch die Interaktion mit der Umgebung eine bestimmte Aufgabe zu lösen. Dabei wird nach jeder durchgeführten Aktion mithilfe einer Belohnungsfunktion bestimmt, wie gut diese Aktion in der aktuellen Situation geeignet war, um die gestellte Aufgabenstellung zu lösen. "The essence of RL is learning through interaction. An RL agent interacts with its environment and, upon observing the consequences of its actions, can learn to alter its own behaviour in response to rewards received. This paradigm of trial-and-error learning has its roots in behaviorist psychology and is one of the main foundations of RL." [7].

Zu Beginn des Lernprozesses besitzt der Agent kein Wissen über die Umgebung und kann Aktionen nur zufällig auswählen. Durch die während des Lernens erhaltenen Belohnungen entwickelt der Agent eine Strategie, um immer besser geeignete Aktionen auszuführen, um dadurch die erhaltenen Belohnungen zu maximieren.

Beim simulationsgestützten Ansatz lernt der Agent durch Interaktion mit einem Simulationsmodell anstelle von der realen Umgebung. Dies bietet im Falle der Reglerauslegung im Wesentlichen drei Vorteile:

1. Die durch das Prinzip von Versuch und Irrtum gene-

rierten Stellgrößen können zu keiner Beschädigung des zu regelnden Systems führen.

2. Die Entkopplung von physikalischen Randbedingungen ermöglicht eine stark beschleunigte Simulation der Regeleingriffe und damit auch des Lernverfahrens.
3. Die Reglerauslegung kann bereits zu einem Zeitpunkt erfolgen, in dem das reale System noch nicht bereitsteht.

Da das bestärkende Lernen zur Steuerung des Lernprozesses die erhaltenen Belohnungen nutzt, besteht hier die besondere Herausforderung darin, eine für den konkreten Anwendungsfall geeignete Belohnungsfunktion zu definieren.

Der in dieser Untersuchung genutzte Algorithmus für das bestärkende Lernen basiert auf der Methode Actor-Critic. "Actor-critic approaches achieve a high performance because they require minimal computation for action selections even in case of very high or infinite number of possible actions. Actor-critics can also learn an explicitly stochastic policy which is very useful in continuous learning problems." [8]. In Abbildung 2 ist der Informationsfluss bei der Methode Actor-Critic innerhalb des Agenten dargestellt und die Interaktion des Agenten mit der Umgebung abgebildet.

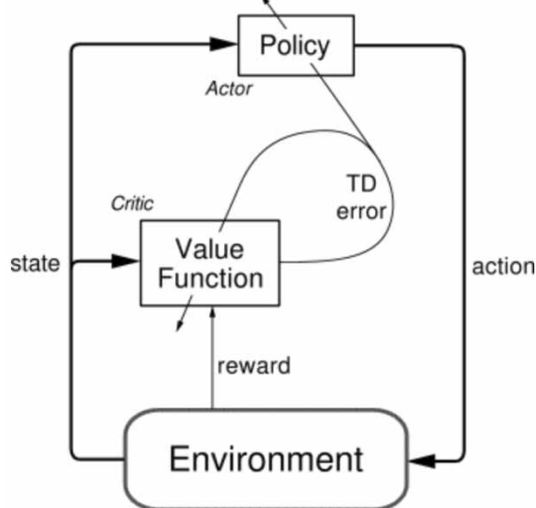


Abbildung 2: Informationsfluss bei der Methode Actor-Critic [9]

Der Actor (Actor) erhält den Zustand (state) von der Umgebung (Environment) und führt abhängig von seiner Strategie (Policy) eine Aktion (action) aus. Die Strategie bildet damit das erlernte Verhalten des Aktors ab.

„A policy defines the learning agent's way of behaving at a given time. Roughly speaking, a policy is a mapping from perceived states of the environment to actions to be taken when in those states. [...] The policy is the core of a reinforcement learning agent in the sense that it alone is sufficient to determine behavior.“ [9]

Der Kritiker (Critic) evaluiert die Strategie des Aktors mit einer Wertfunktion (Value Function). Mit der Ausgabe des Kritikers wird das Lernverfahren beim Actor und beim Kritiker durchgeführt. „Almost all reinforcement learning algorithms are based on estimating value functions--functions of states (or of state-action pairs) that estimate how good it is for the agent to be in a given state (or how good it is to perform a given action in a given state). The notion of "how good" here is defined in terms of future rewards that can be expected, or, to be precise, in terms of expected return. Of course the rewards the agent can expect to receive in the future depend on what actions it will take. Accordingly, value functions are defined with respect to particular policies.“ [9]

3. Auslegung des neuronalen Reglers

Der neuronale Regler wird mit der Programmiersprache Python und dem Machine Learning Framework Tensorflow ausgelegt. Unabhängig davon, ob das RL simulationsgestützt oder am physikalischen Modell des Pendels erfolgt, besteht der zu entwickelnde neuronale Regler aus zwei künstlichen neuronalen Netzen, eins für den Actor und eins für den Kritiker. Als physikalisches Modell wird der Quanser QUBE-Servo 2 genutzt (siehe Abbildung 3). Das Setup für das Training des neuronalen Reglers ist unabhängig von der Anwendung auf dem physikalischen Modell oder dem Simulationsmodell. Vor diesem Hintergrund werden die hierzu notwendigen Schritte im Folgenden anhand des Setups am physikalischen Modell erklärt.

3.1. Architektur der künstlichen neuronalen Netze

Der Actor und der Kritiker verwenden für die künstlichen neuronalen Netze dieselbe Architektur. Es wird jeweils ein vollvermaschtes mehrschichtiges Perzeptron (MLP) mit zwei verborgenden Schichten mit je 64 Einheiten und der tanh-Aktivierungsfunktion verwendet. Die künstlichen neuronalen Netze werden auf einem Computer mit dem Betriebssystem Ubuntu 18.10 und der CPU Intel CORE i7-7700HQ trainiert.

3.2. Proximal Policy Optimization

Für das Training der künstlichen neuronalen Netze wird der Algorithmus Proximal Policy Optimization (PPO) in der Implementierung PPO2 von der Organisation OpenAI genutzt. Bei der Implementierung des Algorithmus werden die Hyperparameter aus der Publikation von PPO [10] und von OpenAI verwendet.

Das künstliche neuronale Netz des Aktors repräsentiert die Strategie und nimmt den Zustand des inversen Pendels als Input, d.h. die Winkelposition des Arms und des Pendels, sowie jeweils die Winkelgeschwindigkeit und -beschleunigung. Als Output gibt das künstliche neuronale Netz einen Wert aus, der mittels Controller als Eingangsspannung an dem Gleichstrommotor des inversen Pendels anliegt.

Das künstliche neuronale Netz des Kritikers stellt die Wertfunktion dar und schätzt die zukünftigen Belohnungen bei Einhaltung der Strategie ab. Mit der Ausgabe der Wertfunktion und der erhaltenen Belohnung wird das neuronale Netz des Kritikers angepasst und mit dem Algorithmus PPO das neuronale Netz des Aktors.

Der Algorithmus PPO stellt dabei sicher, dass sich die neue Strategie des Aktors nicht zu stark von der alten Strategie unterscheidet, damit es nicht zu Einbrüchen beim Lernprozess kommt.

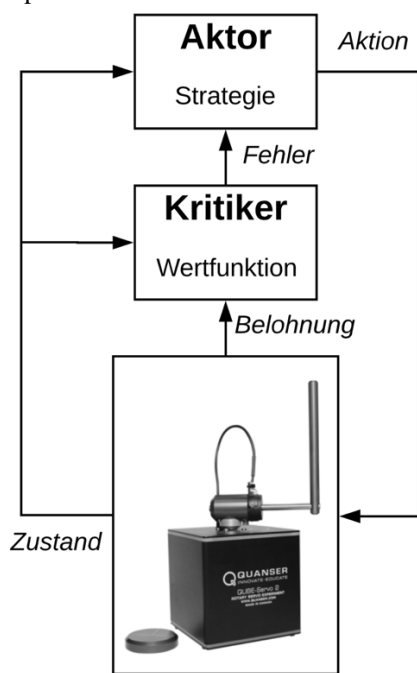


Abbildung 3: Aufbau des neuronalen Reglers und des Quanser QUBE-Servo 2

3.3. Definition der Belohnungsfunktionen

Es muss eine geeignete Funktion zur Berechnung der Belohnungen definiert werden, um den neuronalen Regler für das beabsichtigte Verhalten zu trainieren. Das Ziel ist, dass der neuronale Regler das Pendel aufschwingen und in der instabilen Ruhelage ausbalancieren kann. Der Arm soll beim Ausbalancieren in der instabilen Ruhelage mittig liegen.

Die Belohnungsfunktion wird mit dem Zustand des inversen Pendels definiert. Dabei wird die Winkelposition des Arms (THETA) und die Winkelposition des Pendels (ALPHA) genutzt. THETA kann einen Wert von bis zu 120° annehmen, da der Bewegungsbereich des Arms durch Anschläge begrenzt ist. Ein Vorzeichen gibt an, ob der Arm sich nach links oder rechts bewegt und wird bei der Berechnung der Belohnung nicht berücksichtigt. Bei 0° ist der Arm zentriert. ALPHA hat einen Wert von 0° , wenn das Pendel senkrecht nach oben zeigt, und 180° , wenn das Pendel nach unten zeigt. Ein Vorzeichen gibt an, ob das Pendel sich nach rechts oder links bewegt und wird nicht berücksichtigt. Zur einfacheren Berechnung und Definition der Belohnungsfunktion wird der Betrag von ALPHA und THETA gebildet und die Werte im Bereich von $[0, 1]$ skaliert.

Die Abläufe beim Aufschwingen und Ausbalancieren des inversen Pendels sind sehr verschieden. Dies wird auch beim Training des neuronalen Reglers berücksichtigt, indem dieser in unterschiedlichen Phasen trainiert wird. In der ersten Phase des Trainings soll der neuronale Regler lernen, das Pendel in der instabilen Ruhelage auszubalancieren. Dabei befindet sich das Pendel zu Beginn in der instabilen Ruhelage und soll dort vom neuronalen Regler ausbalanciert werden. Wenn das Pendel die instabile Ruhelage verlässt, wird das Training unterbrochen und das Pendel wird von einem konventionellen Regler in die instabile Ruhelage aufgeschwungen, wo das Training wieder beginnt. Um das beabsichtigte Verhalten zu trainieren, wird bei der Belohnungsfunktion für diese Phase der Winkel des Pendels (ALPHA) deutlich höher gewichtet als der Winkel des Arms (THETA). Dadurch kann der neuronale Regler eine höhere Belohnung bekommen, wenn das Pendel aufrecht steht als wenn der Arm zentriert ist. Dementsprechend passt der neuronale Regler sein Verhalten bei der Regelung des inversen Pendels an. Nach einer Trainingsdauer von 3,5 Stunden kann der neuronale Regler das Pendel in der instabilen Ruhelage ausbalancieren. Die Belohnungsfunktion in (1) wird verwendet.

$$\text{Belohnung} = 1 - 0,9 * |\text{ALPHA}| - 0,1 * |\text{THETA}| \quad (1)$$

In der zweiten Phase soll der neuronale Regler lernen, das Pendel in die instabile Ruhelage aufzuschwingen. Dabei ist das Pendel anfangs in der stabilen Ruhelage, d.h. nach unten gerichtet, und soll in die instabile Ruhelage aufgeschwungen werden. Wenn das Pendel für 10 Sekunden in der instabilen Ruhelage ausbalanciert wurde, wird das Training unterbrochen und das Pendel kehrt wieder in die stabile Ruhelage zurück, wo das Training wieder beginnt. Nach einer Trainingsdauer von 4,8 Stunden kann der neuronale Regler das inverse Pendel zuverlässig in die instabile Ruhelage aufschwingen. Die Belohnungsfunktion in (1) wird dabei verwendet.

Nach der zweiten Phase neigt der Arm dazu, in Richtung der Anschläge zu driftet, wenn der neuronale Regler das Pendel ausbalanciert. Daher ist eine dritte Trainingsphase erforderlich. Das Ziel in der dritten Phase ist, dass der Arm beim Ausbalancieren mittig bleibt und nicht mehr in Richtung der Anschläge driftet. Dabei ist das Pendel in der instabilen Ruhelage und wird durch den neuronalen Regler ausbalanciert. Wenn das Pendel die instabile Ruhelage verlässt, wird das Training unterbrochen und das Pendel wird durch einen konventionellen Regler in die instabile Ruhelage aufgeschwungen, wo das Training wieder beginnt. In (2) ist die Belohnungsfunktion für die dritte Phase definiert. Der Winkel des Arms wird höher als zuvor gewichtet, um das beabsichtigte Verhalten mit dem neuronalen Regler zu trainieren.

$$\text{Belohnung} = 1 - 0,7 * |\text{ALPHA}| - 0,3 * |\text{THETA}| \quad (2)$$

Nach ca. 20 Minuten Trainingszeit werden in der dritten Phase gute Ergebnisse mit dem neuronalen Regler erzielt. Insgesamt dauert das Training ca. 9 Stunden, wobei 2,7 Millionen Aktionen während des Trainings vom neuronalen Regler auf dem inversen Pendel ausgeführt wurden. Im Ergebnis kann der neuronale Regler das inverse Pendel aus der stabilen Ruhelage in die instabile Ruhelage aufschwingen und dort ausbalancieren. Somit kann der hier entworfene neuronale Regler zur Regelung des inversen Pendels genutzt werden.

4. Vergleich KI-basierte Reglerauslegung mit und ohne Simulationsmodell

Aufgrund der bereits unter Kapitel 2 aufgeführten Vorteile eines simulationsgestützten Ansatzes, konnte die

benötigte Zeitpanne für das Training eines neuronalen Reglers am Simulationsmodell im Vergleich zum Training an dem realen System sehr deutlich verkürzt werden. Dies führt vor allem dazu, dass unterschiedliche Algorithmen und Parameter für den Entwurf und das Training eines neuronalen Reglers bei der simulationsgestützten Vorgehensweise schnell und kostengünstig evaluiert werden können.

Weiterhin hat sich gezeigt, dass aggressive Lernstrategien beim Training am physikalischen Modell mit einem erheblichen Risiko von Beschädigungen des Systems oder sogar Verletzungen von Menschen einhergehen.

Ein weiterer Vorteil beim Training am Simulationsmodell ist, dass kritische Situationen häufiger simuliert werden können, als sie tatsächlich in der Realität vorkommen. Dadurch kann der neuronale Regler das korrekte Verhalten in der Simulation lernen, bevor diese kritischen Situationen an einem realen System geregelt werden müssen. Dadurch kann die Robustheit der Regelung gesteigert werden.

Ein Nachteil der simulationsgestützten Auslegung entsteht durch Abweichungen zwischen dem Simulationsmodell und dem realen zu regelnden System. Bei den durchgeführten Untersuchungen hat sich gezeigt, dass geringfügige Vereinfachungen im Simulationsmodell schnell dazu führen können, dass die daran trainierten neuronalen Regler nicht auf das zu regelnde physikalische System übertragen werden können.

Die Vorteile einer simulationsgestützten Herangehensweise müssen daher Anwendungsfall spezifisch mit den durch eine notwendige hohe Modellgenauigkeit erforderlichen Aufwänden abgeglichen werden.

5. Vergleich zwischen dem neuronalen Regler und einem konventionellen Regler

Das Verhalten des trainierten neuronalen Reglers wird mit dem Verhalten eines konventionellen Reglers bei der Regelung des inversen Pendels verglichen. Ein konventioneller Regler wird von Quanser, dem Hersteller des genutzten inversen Pendels, für Matlab Simulink zur Verfügung gestellt [11].

Der konventionelle Regler nutzt beim Aufschwingen des Pendels einen energiebasierten Regelungsansatz und zum Ausbalancieren einen PD-Regler (siehe Abbildung 4).

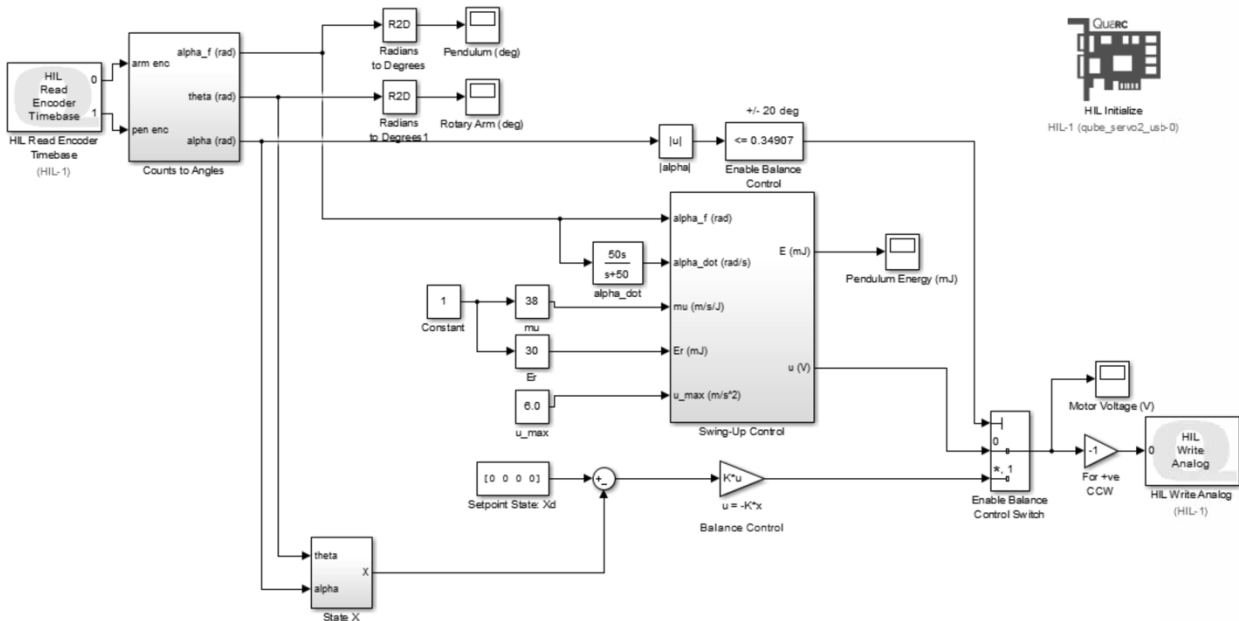


Abbildung 4: Aufbau konventioneller Regler in Matlab Simulink [11]

5.1. Verhalten des konventionellen Reglers

In Abbildung 5 ist das Verhalten des konventionellen Reglers bei der Regelung des inversen Pendels dargestellt.

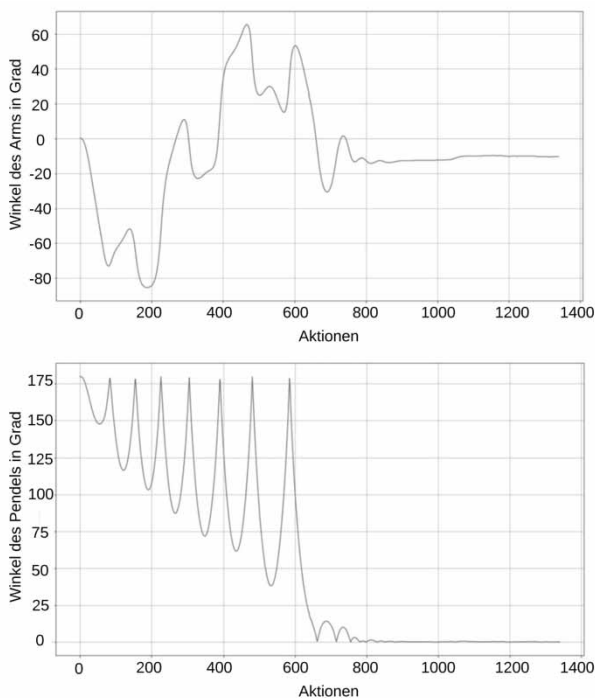


Abbildung 5: Verhalten des konventionellen Reglers

Zunächst ist der Arm mittig und hat einen Winkel von

0°. Der Regler lenkt den Arm ungleichmäßig nach links und rechts aus, um das Pendel aufzuschwingen. Das Pendel befindet sich zu Beginn in der stabilen Ruhelage mit einem Winkel von 180° und zeigt nach unten. Um den Winkel des Pendels in der Abbildung darzustellen, wird das Vorzeichen nicht berücksichtigt. Das Pendel wird vom Regler immer weiter ausgelenkt, bis das Pendel die instabile Ruhelage bei 0° erreicht, in der es ausbalanciert wird. Der Regler benötigt ca. 700 Aktionen, bis das Pendel die instabile Ruhelage erreicht hat. Beim Ausbalancieren ist der Arm um ca. 10° außerhalb der Mitte.

5.2. Verhalten des neuronalen Reglers

Das Verhalten des neuronalen Reglers bei der Regelung des inversen Pendels ist in Abbildung 6 dargestellt. Es ist zu erkennen, dass sich das Verhalten des neuronalen Reglers vom Verhalten des konventionellen Reglers deutlich unterscheidet. Der Arm wird gleichmäßig ausgelenkt, um das Pendel aus der stabilen Ruhelage in die instabile Ruhelage aufzuschwingen. Das Pendel erreicht die instabile Ruhelage nach ca. 400 Aktionen und damit deutlich schneller als mit dem konventionellen Regler. Der Arm wird anschließend mittig ausgerichtet und erreicht schließlich einen Winkel von 0°. Der neuronale Regler zeigt bei der Regelung des inversen Pendels das mit den Belohnungsfunktionen beabsichtigte Verhalten. In den Belohnungsfunktionen wird der Winkel des Pendels höher gewichtet als der Winkel des Arms. Daher ist

es für den neuronalen Regler wichtiger, das Pendel in die instabile Ruhelage aufzuschwingen und dort auszubalancieren. Der Winkel des Arms wird weniger stark gewichtet, daher wird der Arm erst zentriert, wenn sich das Pendel in der instabilen Ruhelage befindet.

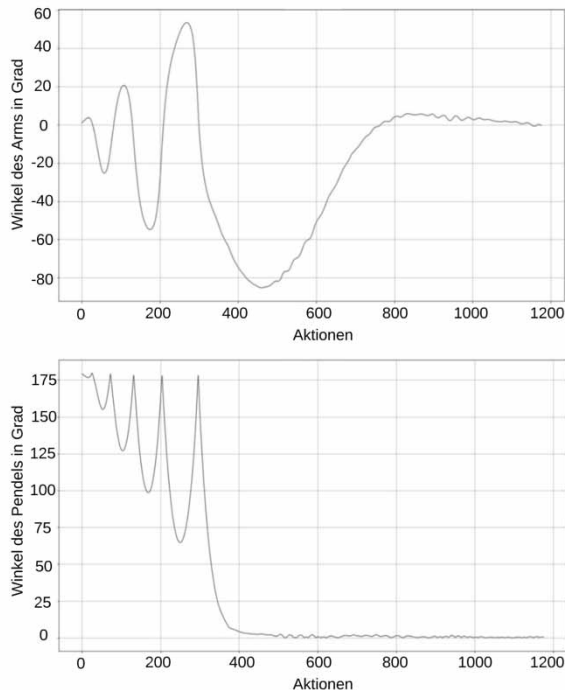


Abbildung 6: Verhalten des neuronalen Reglers

6. Fazit und Ausblick

In diesem Beitrag wurde gezeigt, wie ein neuronaler Regler für die Regelung eines rotierenden inversen Pendels mithilfe von maschinellen Lernverfahren ausgelegt werden kann. In der Gegenüberstellung des Regelverhaltens ist der durch maschinelles Lernen erzeugte Regler einem konventionellen Regler ebenbürtig. Des weiteren wurde verdeutlicht, welche Vorteile, aber auch Herausforderungen sich durch den simulationsgestützten Entwurf neuronaler Regler ergeben.

Die durchgeführten Untersuchungen lassen deutliches Potenzial simulationsgestützter Auslegungen von Regelungen mithilfe maschinellen Lernens erkennen, da sehr leistungsfähige Rechensysteme heute ein paralleles und schnelles Training verschiedenster neuronaler Regler ermöglichen.

Referenzen

1. Bundesministerium für Wirtschaft und Energie, Technologieszenario „Künstliche Intelligenz in der Industrie

4.0“, 2019

2. D. M. Charney and G. M. Josin, "Neural network servo control of a robot manipulator joint in real-time," 1991 IEEE International Joint Conference on Neural Networks, Singapore, 1991, pp. 1989-1994
3. S. K. Suman, M. K. Gautam, R. Srivastava and V. K. Giri, "Novel approach of speed control of PMSM drive using neural network controller," 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, 2016, pp. 2780- 2783
4. N. Yadaiah, A. K. Priya and G. T. Ram Das, "Design of Neural Hysteris Band PWM Current Controller," 2006 IST IEEE Conference on Industrial Electronics and Applications, Singapore, 2006, pp. 1-5
5. D. V. Samokhvalov and I. S. Nosirov, "Feed drive of the metal cutting machines with neural network controller," 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), St. Petersburg, 2017, pp. 373-375
6. Teng Fong, Tang & Jamaludin, Z. & Abdullah, Lokman. (2014). SYSTEM IDENTIFICATION AND MODELING OF ROTARY INVERTED PENDULUM. International Journal of Advances in Engineering & Technology. 6. 2342-2353.
7. K. Arulkumaran, M. P. Deisenroth, M. Brundage and A. A. Bharath, "Deep Reinforcement Learning: A Brief Survey," in IEEE Signal Processing Magazine, vol. 34, no. 6, pp. 26-38, Nov. 2017.
8. Marochko, Vladimir. (2017). Pseudorehearsal in actor-critic agents
9. R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, Cambridge, MA:MIT Press, 2018
10. John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347.
11. Quanser CUBE Servo 2, <https://www.quanser.com/products/qube-servo-2/>

Sequence to Sequence Modelle zur hochaufgelösten Prädiktion von Stromverbrauch

Benjamin Wörrlein, Steffen Straßburger

Fachgebiet Informationstechnik in Produktion und Logistik, Technische Universität Ilmenau, Max-Planck-Ring 12, 98693 Ilmenau, Deutschland; benjamin.woerrlein@tu-ilmenau.de, steffen.strassburger@tu-ilmenau.de

Abstract. Modelling power consumption for jobs on a machine tool is commonly performed by measuring the real power consumption of comparable jobs and machines. The so gathered data is then processed to represent the time-averaged sums of power consumptions of previous jobs. These values of power consumption are then used for upcoming comparable jobs. This approach allows for no high-resolution prediction of power consumption and further presumes static processing times of jobs. Here we propose a new approach to model power consumption that incorporates a Sequence-to-Sequence model, which generates time series according to dynamic data, that describes a numerical control code and environment settings such as state of tools, etc.

Einführung

Aufgabe der Simulation ist es, durch Nachbilden eines Systems Erkenntnisse über dessen Verhalten zu gewinnen. Das Verhalten eines Systems wird typischerweise anhand der dynamischen Veränderungen seines Systemzustands über ein diskret-ereignisgesteuertes oder kontinuierliches Simulationsparadigma in einem Simulationsmodell beschrieben. Kommen bei der Erstellung des Simulationsmodells mehrere verschiedene Modellierungsansätze (ggf. auch nur mehrere Weltsichten innerhalb eines der genannten Paradigmen) zum Einsatz, so spricht man von *hybrider Simulation* [8]. Die Kombination von diskret-ereignisgesteuerter und kontinuierlicher Simulation als eine Spielart der hybriden Simulation wird traditionell auch als „kombinierte Simulation“ bezeichnet [3].

Die Untersuchung von Fragestellungen der Energieeffizienz innerhalb der Simulation ist mittlerweile ein verbreiteter Untersuchungsansatz. Häufig basieren existierende Arbeiten auf der Berücksichtigung des Stromverbrauchs von Ressourcen (Maschinen, Öfen, ...) anhand messtechnisch erfasster Betriebszustände, die über einen definierten Zeitraum als konstant angesehen werden [5, 12].

Der über einen Zeitraum gemittelte Stromverbrauch von

Ressourcen wird hierbei einem Ressourcenzustand zugeordnet und kann dann statusbasiert mit ereignisdiskreten Simulationsansätzen abgebildet und analysiert werden. Kritisch ist hierbei zu hinterfragen, für welche Anwendungsfälle diese quasi-statischen Betriebszustände genügend Realitätsnähe liefern. Zur Ermittlung und Glättung von Lastspitzen vieler Ressourcen bietet ein derartiger Ansatz keine ausreichende Realitätsnähe.

Ein Lösungsansatz hierfür wird in [10] vorgestellt. Er basiert auf der Grundidee der kombinierten Simulation, der z. B. auch in [9] vorgeschlagen wird. Während der Produktions- und Logistikanteil des Modells klassisch mit ereignisdiskreter Simulation abgebildet wird, wird in [10] für den Stromverbrauch der System-Dynamics-Ansatz angewendet. Hiermit können die Zeitreihen der real gemessenen Stromverbräuche hochaufgelöst in der Simulation reproduziert werden. Dies bietet den Vorteil eines hochaufgelösten Gesamtbilds des Stromverbrauchs der Produktion.

Nachteilig ist hierbei jedoch, dass nur der Stromverbrauch real gemessener Aufträge wiedergegeben werden kann. Ein Stromverbrauch unbekannter Auftragsarten kann nicht ohne vorherige Messung am Realsystem prognostiziert werden. Weiterhin lassen sich mit dem in [10] erläuterten Ansatz keine Ursache-Wirkzusammenhänge zwischen Steuerparametern (z.B. halber Vorschub, langsamere Hochheizphase) und dem resultierenden Stromverbrauch darstellen.

Der Stromverbrauch einer Maschine aber geschieht über einen Zeitraum, in welchem kontinuierlich Energie benötigt wird, um einen vorher bestimmten Prozess abschließen zu können. Die Dauer dieses Prozesses wird von einer Vielzahl äußerer und innerer Faktoren bestimmt.

Am Beispiel einer Werkzeugmaschine wird deutlich, dass Faktoren wie der spezifische NC-Code eines Fertigungsauftrags maßgeblich darüber entscheiden, wann eine Maschine wieviel Strom verbraucht. Informationen

wie der NC-Code beschreiben einen zukünftigen Prozess anhand einer festen Abfolge von Schritten, welche zur Bearbeitung des Prozesses nötig sind. Auch wenn die Abfolge von Schritten vorgegeben und vor dem tatsächlichen Prozessstart bekannt ist, so enthalten diese noch keine Beschreibung der Zeit, die benötigt wird, um die einzelnen Schritte durchzuführen.

Wir verwenden hier ein in [13] zuerst vorgeschlagenes Sequence-to-Sequence Modell zur Abbildung des Wirkzusammenhanges zwischen der Zeitreihe des Stromverbrauchs eines Fertigungsauftrags und alternativer Beschreibungen, wie dem NC-Code, Betriebsparametern, usw.

Der Schwerpunkt der Untersuchungen liegt hierbei auf dem Gebiet des maschinellen Lernens bzw. des *Deep Learnings*.

Ziel des vorgeschlagenen Verfahrens ist es, mittels entsprechend trainierten künstlichen neuronalen Netzen (KNN) Zeitreihen für den Stromverbrauch von Fertigungsaufträgen prognostizieren zu können.

Die Grundidee hierbei ist es, einem Sequence-to-Sequence Modell NC-Codes, sowie entsprechende Betriebszustände etc., und gemessene, hochaufgelöste Zeitreihen des Stromverbrauchs vorhergehender Fertigungsaufträge in einer Trainingsphase zu übergeben. In der Trainingsphase stellt das Sequence-to-Sequence Modell den Zusammenhang zwischen den beiden Beschreibungen her. Ist das Training abgeschlossen, muss dem nun gewichteten neuronalen Netz nur ein NC-Code, und gegebenenfalls weitere betrachtete Faktoren, wie Werkzeugstandzeit etc., übergeben werden, woraufhin dieses eine Zeitreihe des Stromverbrauchs eines Fertigungsauftrags anhand eines NC-Codes generiert.

Perspektivisch kann das KNN dann zu beliebigen, ggf. auch unbekannten Aufträgen mit abweichenden NC-Codes eine Zeitreihe des erwarteten Stromverbrauchs prognostizieren. Diese ließen sich dann in hybriden Simulationen des gesamten Produktionssystems verwenden.

Der Beitrag stellt ein Lösungskonzept für das skizzierte Verfahren sowie eine prototypische Implementierung und Evaluierung vor und ist hierzu wie folgt gegliedert: Kapitel 1 führt in die benötigten theoretischen Grundlagen des Verwendeten maschinellen Lernverfahren ein. Insbesondere werden Notwendigkeit und Grundidee einer *Deep-Learning*-Methode, welche Sequenzen unterschiedlicher Länge und Taktzeiten aufeinander abbilden kann, gesondert vorgestellt. Anschließend wird die prinzipielle Funktionalität von klassischen

KNN zu zeitsensitiven, rekurrenten neuronalen Netzen (RNN) abgegrenzt. Aufbauend auf dieser Einführung in RNN wird auf *Sequence to Sequence* (Seq2Seq)-Modelle, insbesondere RNN-Encoder-Decoder-Architekturen (RNN-ED) eingegangen, welche eine Zuordnung Sequenzen unterschiedlicher Länge zueinander erlauben.

Hierauf aufbauend wird in Kapitel 2 ein Konzeptvorschlag der Gesamtarchitektur mit seinen Eingangs- und Zielsequenzen entwickelt. Das Konzept wird im Kontext eines Fertigungsauftrages (FA) an einer Werkzeugmaschine (WZM) erstellt. Als Eingangssequenz wird sich hier des NC-Codes und entsprechender Betriebszustände bedient. Anschließend wird der vektorisierte NC-Code auf aufgenommene Zeitreihen der Wirkleistung einer WZM in [kW], hier den Zielsequenzen, trainiert.

In Kapitel 3 werden die notwendigen Schritte zur Vorverarbeitung der Eingangs- und Zielsequenzen erläutert. Einerseits wird eine Methode definiert, die eine Eingangssequenz von Symbolen, wie den NC-Code, etc., in eine für ein KNN verständliche Form überführt. Dieser als Vektorisierung bezeichnete Vorgang wird anhand eines *Word2Vec-Tokenizers* durchgeführt und gibt so eine vektorisierte Form des NC-Codes aus. Andererseits werden empirische Erkenntnisse im Hinblick auf die Beschaffenheit der Zielsequenzen, hier Zeitreihen, vorgestellt und erläutert. Unsere Forschung weist darauf hin, dass die Verteilung der Häufigkeit einzelner Merkmalsausprägungen von entscheidender Bedeutung ist.

Eine prototypische Umsetzung des Konzepts erfolgt in Kapitel 4. Dieses wird mit der API *Keras* und dem Backend *Tensorflow* umgesetzt. Nach erfolgreicher Trainingsphase erfolgt eine Vorstellung der Ergebnisse. Dies geschieht anhand einer Gegenüberstellung der Zeitreihen des Trainingsdatensatzes und der Erzeugten. Hier sollen insbesondere die charakteristischen *features* der beiden Zeitreihengruppen einander gegenübergestellt werden.

Eine kritische Betrachtung der Ergebnisse und ein Ausblick über weitere Forschungsrichtungen erfolgt in Kapitel 5.

1 Sequenzmodellierung durch Sequence-to-Sequence

Die hier vorgeschlagene Methode der Sequenzmodellierung zeichnet sich dadurch aus, dass sie bekannte, asynchrone Zustands- bzw. Parameterverläufe, als eine Form von apriorischem Wissen, zur Modellierung von System-

verläufen ermöglicht. Verläufe werden als asynchron zueinander definiert, wenn sie über denselben Start- und Endzeitpunkt verfügen, sich die Taktung in ihren Einträgen aber voneinander unterscheidet. Um solche Verläufe einander zuzuordnen wird sich einer Methode des maschinellen Lernens zu Nutze gemacht. Vorteilhaft bei der Verwendung von Algorithmen des maschinellen Lernens ist es, dass diese sich, im Anschluss an eine Modellierungsphase, anhand von Realdaten selbst parametrieren.

1.1 Sequenzmodellierung als Ergänzung der ereignisdiskreten Simulation

Die grundsätzliche Limitation ereignisdiskreter Simulationsansätze besteht darin, dass Zustandsänderungen zwischen zwei Ereignissen nicht abbildbar sind. Für eine Aktivität, d. h. die Zeitspanne zwischen zwei Ereignissen, kann jedoch die Notwendigkeit bzw. der Wunsch bestehen, einen Zustandsverlauf eines zur Aktivität gehörenden Merkmals (z. B. den Verlauf des Stromverbrauchs während der Bearbeitung) zu beschreiben. Hierfür könnte z. B. aus einem internen (in der ereignisdiskreten Modellierung nicht betrachteten) Zustandsverlauf heraus zu bestimmten Taktzeiten eine Folge von Ausgaben erzeugt werden, welche das zeitliche Verhalten dieses Merkmals widerspiegeln [7].

Gerade aber der Zustandsverlauf eines technologischen Systems kann von einer Vielzahl äußerer Einflüsse bedingt werden. Dies bedeutet, dass Merkmalsbeschreibungen Y , die ab dem Start einer Aktivität ausgegeben werden sollen, in Relation zu etwaigen Einflussgrößen X verstanden werden müssen.

Es fehlt eine Methode, welche es erlaubt, Zielmerkmalsverläufe Y_T , aus asynchronen, aber bekannten Parameter- bzw. Zustandsverläufen X_T abzubilden. Gerade für die Abbildung komplexer Merkmalsverläufe aufeinander bieten sich Methoden des maschinellen Lernens an, da diese einerseits Zusammenhänge zwischen Merkmalsverläufen selbstständig erkennen und andererseits lernen, diese aufeinander abzubilden. Weiter besitzen Methoden des maschinellen Lernens erst nach erfolgreicher Lernphase einen parametrisierten Systemzustand, analog zum Systemzustand innerhalb einer Simulation, und setzen daher keinen bekannten Zustandsverlauf während der Aktivität voraus. Diese Eigenschaften machen Methoden des maschinellen Lernens perspektivisch zu einem potenten Verfahren innerhalb eines *Hybrid System Model* nach [8].

1.2 Rekurrente Netze und Encoder-Decoder Architekturen

KNN werden zur Identifikation von Zusammenhängen in komplexen Datenstrukturen verwendet. Hierfür nehmen Aufnahmeschichten einer Netzarchitektur die Daten auf und leiten diese als abstrahierte Information durch die verdeckten Schichten eines KNN. Verdeckte Schichten bestehen wiederum aus verdeckten Einheiten, den eigentlichen Neuronen. Diese Neuronen sind sich selbst parametrierende Einheiten. Je mehr verdeckte Schichten ein KNN hat, desto höher kann der Abstraktionsgrad der aufgenommenen Information sein. Besitzt ein KNN mehr als eine verdeckte Schicht, so kann es Abstraktionen, die in einer Schicht gewonnen wurden, in einer weiteren Schicht verknüpfen, und somit eine komplexere Abstraktion, mit jeder hinzugenommenen Schicht, erzeugen. Diese tiefe Staffelung von neuronalen Schichten wird als *Deep Learning* bezeichnet [4].

Ändern sich Muster über die Zeit, wird diese zeitliche Abfolge von Mustern als Sequenz verstanden. Damit ein KNN zeitliche Muster verarbeiten kann, müssen rekurrente Verbindungen in der Netztopologie vorhanden sein, welche eine Rückkopplung abstrahierten Wissens zulassen [1, 14]. Solche rückgekoppelten bzw. rekurrenten neuronalen Netze (RNN) eignen sich besonders für Daten, welche in sequentieller Form vorliegen [4].

Für die zeitsensitive Verarbeitung von Sequenzen muss weiter eine neuronale Zelle bereitgestellt werden, welche einerseits ihren eigenen Zustand behält und diesen weitergeben kann, andererseits Zugriff auf Nachfolgezustände besitzt und sich an diesen referenzieren kann. Die Anforderungen an eine solche neuronale Zelle mit Gedächtnis werden durch neuronale *Long Short Term Memory* (LSTM)-Zellen [6] und deren vereinfachte Form *Gated Recurrent Unit* (GRU) [2] erfüllt.

Handelt es sich bei den Daten eines KNN um Sequenzen, werden diese als *Sequence to Sequence* (Seq2Seq) Architekturen bezeichnet. Durch die Aufnahmeschicht eines KNN findet eine Codierung der Eingangssequenz statt. Wird die Eingangssequenz als Abstraktion, in eine spezifische neuronale Schicht codiert, so ist dies ein Encoder. Wird eine Zielsequenz aus der Abstraktion einer neuronalen Schicht heraus generiert, so wird dieser Teil einer Netzwerktopologie als Decoder bezeichnet [4].

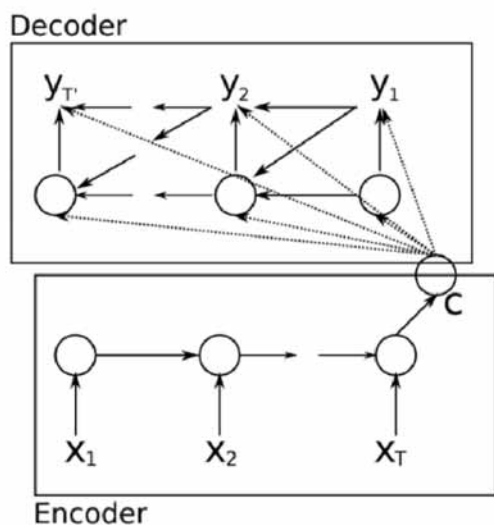


Abbildung 1: Encoder-Decoder Architektur mit den Sequenzen X_T und $Y_{T'}$ unterschiedlicher Länge $T \neq T'$ und dem Kontext C [2]

Ist es Aufgabe eines Seq2Seq-Modells, Sequenzen unterschiedlicher Länge auf einander abzubilden, werden solche Strukturen allgemein als rekurrente Encoder-Decoder-Netzwerke (RNN-ED) bezeichnet [4]. Sollen Sequenzen unterschiedlicher Länge und unterschiedlicher Attribute aufeinander abgebildet werden, so müssen diese um eine zusätzliche Beschreibungsart, einen *Kontext* (vgl. Kontextvektor C Abb. 1) erweitert werden. Der Kontext kann als Zwischenschicht zwischen den verdeckten Schichten eines Encoders und Decoders verstanden werden [2].

Der Kontext C ist ein Vektor einer Sequenz, welche die Einheiten der verdeckten Schicht des Encoders aufnimmt und diese auf den Decoder abbilden soll [4]. Der Vektor selbst lässt sich anhand einer verdeckten Schicht beschreiben [4]. So wird bei der RNN-ED-Architektur der Kontext C als ein Resultat der finalen verdeckten Schicht des Encoders mit der Eingangssequenz X_T , beschrieben. Da der Encoder in der Trainingsphase seinen finalen verdeckten Zustand weitergibt, muss die ganze Sequenz X_T durchlaufen worden sein (siehe Abb. 1).

Weiterführende Erläuterungen zum hier verwendeten Encoder-Decoder können [2, 4, 11] entnommen werden.

2 Konzept

Wie eingehend erwähnt, wird zur konzeptuellen Überprüfung vorgeschlagen, eine Menge der Eingangssequenzen \mathbb{X} und Zielsequenzen \mathbb{Y} als Sequenzpaarungen

$\{X_i, Y_i\}$ der Menge i zu verwenden, welche derselben zeitlich-räumlichen Entität angehören. Von einer zeitlich-räumlichen Entität wird hierbei von einem Prozess ausgegangen, welcher am selben Ort und zur selben Zeit stattfindet. Hierfür wurde das technologische Verfahren des Spanens eines Fertigungsauftrages (FA) auf einer Werkzeugmaschine (WZM) identifiziert (siehe Abb. 2).

Ein NC-Code beschreibt eine Abfolge notwendiger technologischer Prozesse bis zur Beendigung eines FA und kann somit als eine konkrete Beschreibung einer dem Prozess zugrundeliegenden Zustandsfolge verstanden werden. Der NC-Code bestimmt also maßgeblich das Verhalten innerhalb des Spanraums einer WZM. Weiter gilt ein FA erst als abgeschlossen, wenn der NC-Code einmal komplett durchlaufen wurde.

Weiter wird der Beschreibung des FA entsprechende Betriebsmodi, wie Schruppen und Schlichten, angehängt. Der NC-Code, gemeinsam mit dem Betriebsmodus stellen hier die Eingangssequenz X_i eines RNN-ED dar.

Basis der Ausgangszeitreihen Y_i quasi-kontinuierlicher Ausgabewerte ist der Stromverbrauch [kW] desselben FA bei Durchlauf des NC-Codes im jeweiligen Betriebsmodus. Der zeitliche Stromverbrauch soll konkret Aufschluss darüber geben, wann mit wieviel Verbrauch gerechnet werden muss, sobald über die Einsteuerung eines FA entschieden werden muss. Die Zeitreihen wurden unter Feldbedingungen aufgenommen und verfügen über dieselbe Taktung von 500 ms.

In der Trainingsphase wird ein ungewichtetes KNN, bestehend aus einem RNN-ED, anhand der Ein- und Zielsequenzen $\{X_i, Y_i\}$ parametrisiert (siehe Abb. 2).

Aufgabe der Inferenzphase ist es, ein aussagekräftiges Stromverbrauchsprofil \hat{Y}_i explizit quasi-kontinuierlich über die Zeit auszugeben.

Die so postulierte Methode der Erzeugung quasi-kontinuierlicher Zeitreihen stellt, wenn sie in Kombination mit den Modellierungsmöglichkeiten diskret-ereignisgesteuerter Simulationssysteme verwendet wird, eine Methode der hybriden Simulation dar.

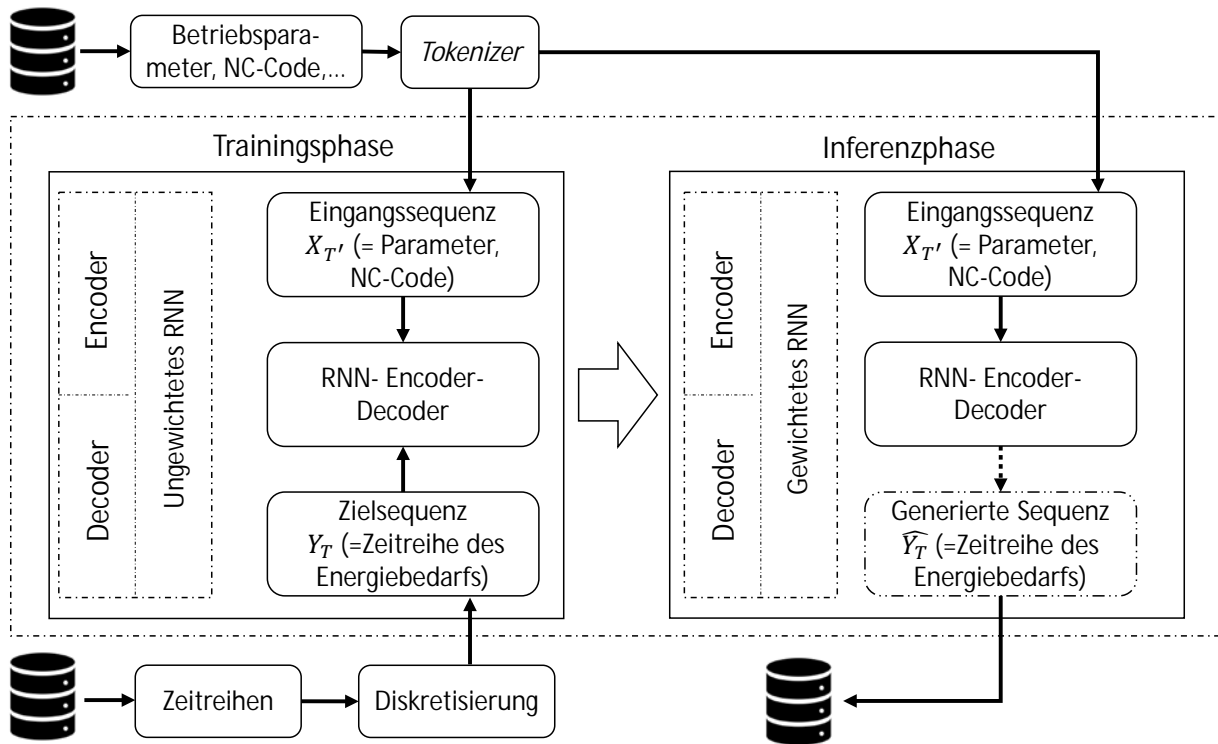


Abbildung 2: Bestandteile einer RNN-Encoder-Decoder Topologie für Sequenzen unterschiedlicher Länge und Symbolik während der Trainings- und Inferenzphase

3 Versuchsvorbereitung

Dem Sequence-to-Sequence Modell werden in der Trainingsphase einerseits Eingangssequenzen, bestehend aus den jeweiligen NC-Codes und Betriebsmodi, andererseits Zielsequenzen, bestehend aus Zeitreihendaten, eines FA gegenübergestellt. Es wurde eine Menge ($i = 51$) Sequenzpaarungen aufgenommen. Diese müssen entsprechend der hier vorgestellten empirischen Ergebnisse vorverarbeitet werden, damit das Modell einen aussagefähigen Zusammenhang zwischen den beiden Mengen erlernen kann.

3.1 Vorbereitung der Eingangssequenzen \mathbb{X}

Die Eingangssequenzen \mathbb{X} werden um verschiedene Betriebsmodi $\{x_{11}, x_{12}\}$, in denen die Werkzeugmaschine betrieben werden kann, erweitert. Diese Betriebsmodi spiegeln eine gängige Arbeitsroutine bei der Bearbeitung eines FA wider. Der NC-Code läuft zum ersten Mal $\{\text{Schruppen} = x_{11}\}$, um eine größere Menge an überschüssigem Material abzutragen und dem Material seine

Form zu geben. Danach wird der gleiche NC-Code noch mehrere Male ausgeführt $\{\text{Schlichten} = x_{12}\}$, um die Oberfläche des nun in Form gebrachten Materials zu glätten. Diese beiden Modi resultieren in Zeitreihen der Leistungsaufnahme, die zwar in ihrer Länge vergleichbar sind, aber in ihren Merkmalsverläufen unterschiedliche Eigenschaften aufweisen. Die Eingangssequenz X_i wird entsprechend beschrieben als:

$$X_i = \{\{x_{11}, x_{12}\}, x_2\}$$

wobei x_2 der NC-Code ist.

Die Beschreibungen der Eingangssequenz müssen hierfür erst in eine Abfolge numerischer Werte übersetzt werden, welche die Struktur der Eingangsfolge beibehält. Dies wird durch einen sog. *Tokenizer* realisiert. Ein Tokenizer weist jedem Symbol bzw. jeder Menge von Symbolen, welche im NC-Code usw. vorhanden sind, einen numerischen Wert bspw. anhand der Häufigkeit des betreffenden Symbols zu.

$$[\dots G \ 00, X0 \ Y0 \ Z0, \dots] \xrightarrow{\text{Tokenizer}} [\dots 1 \ 2 \ 3 \ 4 \ 5 \ \dots]$$

Weiter entfernt der Tokenizer Symbole bzw. Symbolbeschreibungen, welchen ein geringer Informationsgehalt, wie zum Beispiel Kommata und Groß-/Kleinschreibung, unterstellt wird. Werden alle Einträge eines (langen) NC-Codes übernommen, kann dies in einem Vektorraum resultieren, der zwar einen Prozess detailliert beschreibt, aber aufgrund seiner Größe nicht mehr rechentechnisch verarbeitet werden kann. Eine Möglichkeit, die Dimensionen des Vektorraums zu begrenzen, ist es, dem *Tokenizer* eine Anzahl maximal abbildbarer Symbolmengen, d. h. Wörter, anzuzeigen. Dies war jedoch im hier verwendeten Anwendungsfall, aufgrund der relativen Kürze des NC-Codes nicht nötig.

Im Anwendungsfall wurde ein *Word2Vec*-Tokenizer verwendet, welcher alle Symbole und Symbolmengen in den Vektor übernimmt. Die Symbole der Sequenzen von \mathbb{X} werden abschließend in eine Sequenz von ganzzahligen Werten tokenisiert, wobei jedes eindeutige Wort durch genau eine ganze Zahl repräsentiert wird. Dies ermöglicht es, wiederkehrende Muster innerhalb eines NC-Codes zu modellieren.

3.2 Vorbereitung der Zeitreihen \mathbb{Y}

Die Grundlage der Werte für die quasi-kontinuierliche Zielzeitreihe \mathbb{Y} bildet der reale Stromverbrauch eines FA beim Durchlauf eines NC-Codes. Eine äquidistante Messreihe gibt konkret Auskunft darüber, wann wie viel Verbrauch entstanden ist, sobald eine Entscheidung über die Bearbeitung eines Auftrags getroffen werden muss. Die Zeitreihendaten wurden unter Feldbedingungen aufgezeichnet und haben die gleiche Taktung $\Delta t=500$ ms, sowie eine mediane Länge von 2295 Zeitschritten für den Schruppprozess x_{11} und 2256 Zeitschritten für den Schlichtprozess x_{12} .

Der Energieverbrauch der Werkzeugmaschine und damit die Zeit, die für die Bearbeitung eines spezifischen Auftrags benötigt wird, wird zunächst bei jeder Bearbeitung eines Auftrags überwacht und als Zeitreihendatensatz gespeichert.

Die Menge von Zeitreihen \mathbb{Y} muss weiter diskretisiert werden. Diskretisierung ist der Prozess der Portionierung kontinuierlicher Werte in diskrete Gruppen von Werten oder *bins*, die den ursprünglichen Werten der Daten ähneln. Dies stellte sich als notwendig heraus, da die hier vorgestellten empirischen Ergebnisse zu der Schlussfolgerung führten, dass eine Verteilung von Merkmalshäufigkeiten P^f , welche einer diskreten Gleichverteilung $P^{d.u.d.}$ oder einer *long tail*-Verteilung $P^{long\ tail}$, bei

welcher der *tail* zu einer diskreten Gleichverteilung tendiert, das Seq2Seq-Modell darin behindern, eine sinnvolle gemeinsame Verteilung von $\{X_i, Y_i\}$ zu erlernen.

Um den richtigen Diskretisierungsparameter zu finden, d. h. den Grad der Diskretisierung zu bestimmen, wurden mehrere Trainingsläufe mit alternativen Diskretisierungsparametern durchgeführt. Die verschiedenen Diskretisierungsparameter wurden auf den gesamten Zeitreihendatensatz angewendet und anschließend entsprechend der Merkmalsfrequenz f der diskretisierten Werte klassifiziert (siehe Abb. 3). Dazu wurde die Verteilung der Merkmalshäufigkeiten P^f mit einem gaussischen Kerndichteschätzer (KDE) analysiert.

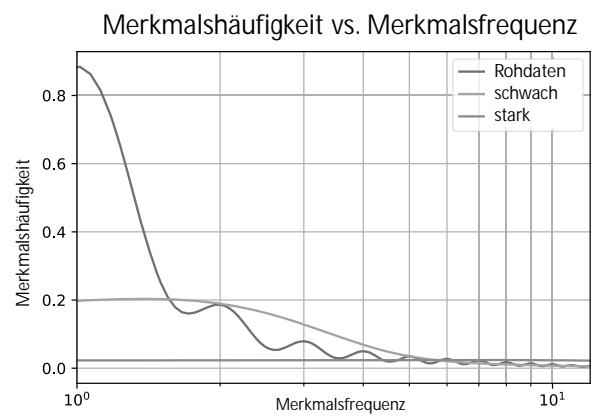


Abbildung 3: Der KDE-Plot zeigt, dass die Rohdaten meist einmalige Werte enthielten, während eine *starke* Diskretisierung zu einer gleichmäßigen Verteilung führt, bei der die Wahrscheinlichkeit, dass ein Wert zu einer beliebigen Frequenz gehört, die gleiche ist wie bei jeder anderen Frequenz. Eine *schwache* Diskretisierung führt zu einer *heavy-tail* Verteilung der Frequenzen.

Das vorgeschlagene Konzept wurde für alle drei Häufigkeitsverteilungen erprobt und führt nur bei der schwachen Diskretisierung zu zufriedenstellenden Ergebnissen.

4 Versuchsdurchführung

Als Metriken zum Vergleich der erzeugten Zeitreihen \hat{Y}_i und den Trainingszeitreihen Y_i werden die mittlere Länge $len(\hat{Y}_i)$ und die durchschnittliche Summe $sum(\hat{Y}_i)$ der Zeitreihen, wie sie im Trainingsset gefunden werden, herangezogen. Außerdem wurden die Zeitreihen visualisiert und Merkmale charakteristischer Muster (*features*) zu diesen Visualisierungen hinzugefügt

(siehe Abb. 5). Das Hinzufügen von *features* hilft, die Zeitreihen \hat{Y}_i und Y_i intuitiver auf visueller Ebene zu vergleichen.

Die auf Basis der Rohdaten, welche nicht diskretisiert wurden, erzeugte Zeitreihe stimmt mit der geringen Aussagekraft, wie in [13] beschrieben, überein. Die erzeugten Sequenzen zeigten keinen sinnvollen Werteverlauf und versäumten es weiterhin, ein EOS-Token zu erzeugen, d. h. die Methode erzeugt infinit neue Zeiteiheneinträge bis ein generisches Abbruchkriterium gefunden wurde.

Die Ergebnisse für die starke Diskretisierung, wie in Abbildung 4 dargestellt, stellen eine Verbesserung dar. Es wurde ein EOS-Token erstellt, wie auch die meisten anderen Zeitreihenmerkmale, doch die generierte Zeitreihe kann deutlich von denen der Trainingsdaten unterschieden werden (vgl. Stichprobenbeispiele in Abbildung 5) und führen schlussendlich zu niedrigen Werten in den Metriken.

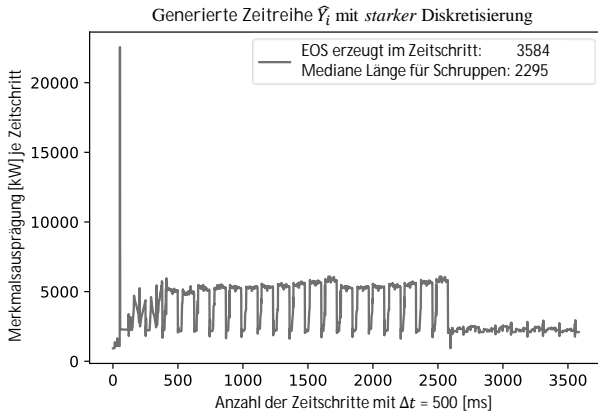


Abbildung 4: Ergebnis für eine *starke* Diskretisierung und Parametereinstellung $\{x_{11}, x_2\}$. Ein EOS-Token wurde, wie die meisten anderen Merkmale, erstellt, doch die generierte Serie kann klar von den Trainingsdaten unterschieden werden (siehe Abbildung 5).

Die *schwache* diskretisierte Zeitreihe hingegen zeigt einerseits hohe Werte für $len(\hat{Y}_i)$ und $sum(\hat{Y}_i)$:

$$\{x_{11}, x_2\}: \frac{len(\hat{y}=2258)}{len(\bar{y}_i=2295)} = 98.4\%; \frac{sum(\hat{y}=6847.7)}{sum(\bar{y}_i=6927.9)} = 98.8\%$$

$$\{x_{12}, x_2\}: \frac{len(\hat{y}=2204)}{len(\bar{y}_i=2256)} = 97.7\%; \frac{sum(\hat{y}=4843.3)}{sum(\bar{y}_i=4871.8)} = 99.4\%$$

Andererseits ist bei näherer Betrachtung der Zeitreihen \hat{Y}_i und Y_i für $\{x_{11}, x_2\}$, wie in Abbildung 5 dargestellt, eine auffällige Ähnlichkeit zu erkennen. Die generierte Zeitreihe schafft es, den Verlauf der *features*, wie in den

Stichproben gezeigt, mit einer bemerkenswerten Präzision nachzuahmen. Sie vermag nicht nur, einen EOS-Token, der der Lage der im Trainingsset gefundenen Zeitreihe entspricht (grünes *feature*), eine ausgeprägte Lastspitze (rotes *feature*), eine Folge von Untersequenzen (*features* wechselnder Blautöne) zu reproduzieren, sondern auch, diese in der richtigen Reihenfolge und Dimensionalität zu erzeugen.

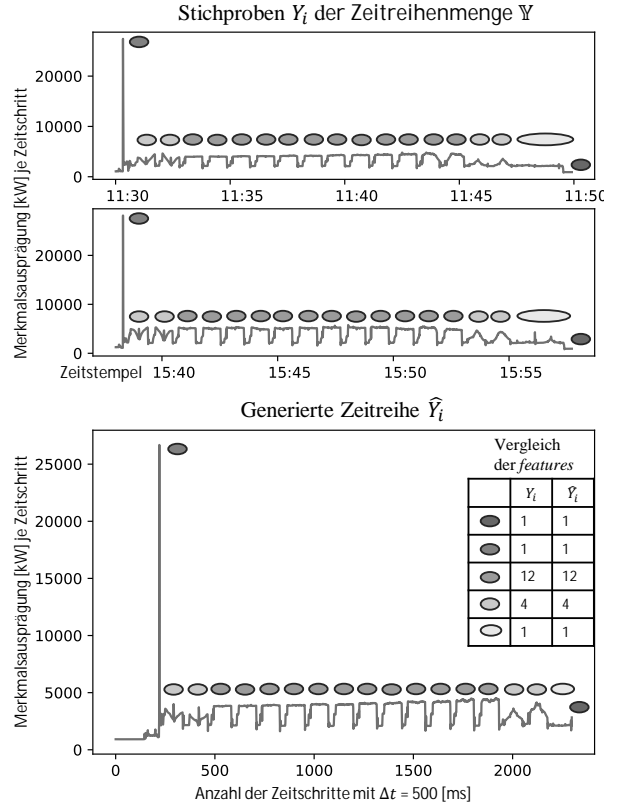


Abbildung 5: Vergleich der Proben Y_i aus dem Trainingsset \mathbb{Y} und der generierten Zeitreihe \hat{Y}_i für den *schwachen* Diskretisierungsparameter und die Sequenzkombination $\{x_{11}, x_2\}$. Die angezeigten Sequenzen zeigen deutlich die gleichen Muster in ihrem Verlauf. Um den visuellen Vergleich zwischen verschiedenen Sequenzen zu erleichtern, wurden den lokalen Mustern, *Points of Interest-features* hinzugefügt. Die Tabelle auf der rechten unteren Seite vergleicht die Anzahl der *features* miteinander.

Die in Abbildung 6 dargestellten Zeitreihen \hat{Y}_i und Y_i für $\{x_{12}, x_2\}$ zeigen ebenfalls deutlich, dass es dem Seq2Seq-Modell gelungen ist, den Verlauf der Labels innerhalb des Trainingssets zu erfassen, obwohl die Trainingszeitreihe nur wenige Merkmale enthielt, die überhaupt erlernt werden konnten.

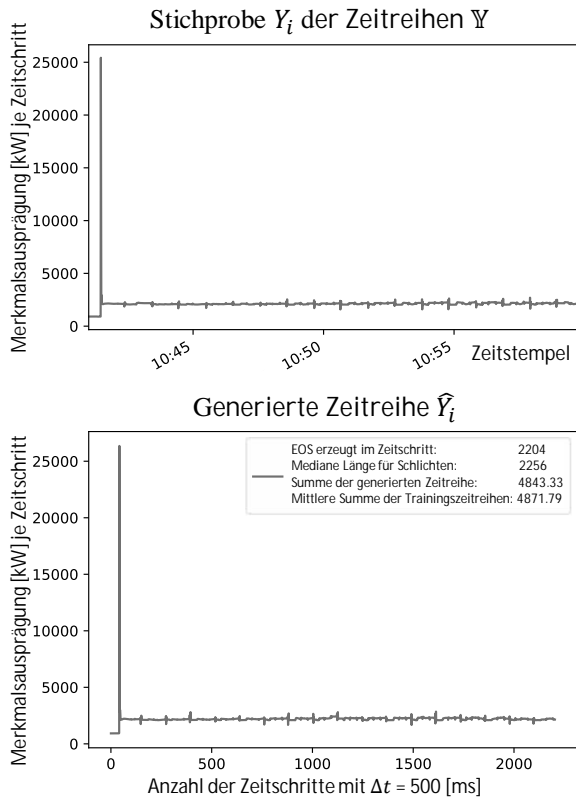


Abbildung 6: Vergleich der Proben Y_i aus dem Trainingsset Y und der generierten Zeitreihe \hat{Y}_i für die schwache Diskretisierung und die Sequenzkombination $\{x_{12}, x_2\}$. Es wurden keine Merkmale hinzugefügt, da die Zeitreihe nur wenige Merkmale aufweist.

5 Kritische Betrachtung

Die grundlegende Funktionalität der beschriebenen Methode wurde im Testszenario bestätigt. Die generierten Zeitreihen müssen jedoch in weiteren Forschungsarbeiten noch kritisch hinterfragt und validiert werden. Zum einen fehlen Evaluationsmethoden für generative Modelle des maschinellen Lernens, um die generierten Zeiteiheneinträge auf die Aussagekraft ihrer Einträge zu überprüfen. Dies geschieht derzeit durch Analyse und den Vergleich der generierten Zeitreihen durch einen Experten des Anwendungsfalles mittels optischer Inspektion [4], wie in Abbildung 5 dargestellt.

Für eine abschließende Bewertung der verwendeten Methoden ist es ratsam, die qualitative und quantitative Datenbasis des Seq2Seq-Modells zu erweitern. Der hier verwendete Datensatz ist von geringer Größe. Dennoch ist die Größe des Datensatzes beispielhaft für reale Situ-

ationen, die sich schnell und in kurzer Zeit ändern können. Algorithmen des maschinellen Lernens hingegen konvergieren in der Regel aber umso besser, je mehr Daten vorhanden sind, aus denen gelernt werden kann.

Eine Methode, in der der Trainingsdatensatz um synthetische Zeitreihen erweitert wird, die so beschaffen sind, dass sie einen *gemittelten* Verlauf der Zeitreihen des Trainingsdatensatzes darstellen, könnte dieses Problem lösen. *Dynamic Time Warping* könnte verwendet werden, um solche *Ground Truth*-Zeitreihen zu erzeugen, die dann iterativ dem Trainingsset hinzugefügt werden könnten, bis ein vorteilhaftes Lernverhalten erreicht wird.

Zusätzliche *End-of-Sequence-Token* könnten zur Beschreibung von Ereignissen wie Maschinenausfall verwendet werden. Die hier verwendeten EOS-Token markierte lediglich das Ende eines abgeschlossenen FA. Jedoch sind einige Aufträge aufgrund von Systemveränderungen wie z.B. Verschleiß des Werkzeugs anfällig für Werkzeugbruch, etc. Das Hinzufügen eines alternativen EOS-Tokens zum Trainingsdatensatz, der einen Maschinenausfall markiert, zusammen mit zum Zustandsdaten der Werkzeuge usw., könnte auch die Frage beantworten, ob ein Auftrag im Hinblick auf die bekannten Systemgrößen erfolgreich ausgeführt werden kann.

Das hier vorgestellte Seq2Seq-Modell erlaubt es außerdem, Zeitreihen auf Basis faktorieller Kombinationen zu generieren, welche in den Trainingsdaten nicht vorhanden sind. Da der Decoder nicht direkt auf die in der Eingangssequenz gefundenen Beschreibungen parametrisiert wird, sondern auf eine Zusammenfassung ebendieser in Form eines *Kontextvektors*, können faktorielle Kombinationen von Eingangsparametern verwendet werden, die im ursprünglichen Trainingsdatensatz nicht repräsentiert sind. Sobald die jeweiligen Eingabeparameter und ihre spezifische Wirkung auf die Zeitreihe modelliert worden sind, können beliebige Kombinationen von Eingangsparametern verwendet werden. Dies würde zu Faktorkombinationen führen, die für einen Simulationsexperten von hohem Interesse sind und die in einer generischen Simulationsstudie nur mit hohem Aufwand modelliert werden könnten.

Darüber hinaus muss dem vorgeschlagenen Modell eine geeignete Bewertungsmethode hinzugefügt werden, da die Validierung der generierten Zeitreihen sich nicht an einer (nicht vorhandenen) idealen Zeitreihe orientieren kann.

Weiter könnte bei erfolgreicher Etablierung und Validierung der Methode eine Lösung entwickelt werden, die auf Basis nicht in den Trainingsdaten vorhandener

NC-Codes, plausible Stromverbrauchsprognosen für neue FA's generiert. Dies hätte ein hohes praktisches Potenzial und wäre auch aus wissenschaftlicher Sicht ein Durchbruch.

Die Weiterentwicklung der oben beschriebenen Methode des maschinellen Lernens und ihre Anwendung für hybride Simulationsmodelle ist derzeit Gegenstand laufender Forschung.

Die Übertragung der Grundidee auf andere Formen von Eingangssequenzen und Zeitreihen anderer Messwerte ist ebenfalls denkbar und ein möglicher Gegenstand weiterer Untersuchungen.

Literaturverzeichnis

- [1] Rüdiger W. Brause. 1995. *Neuronale Netze. Eine Einführung in die Neuroinformatik* (2., überarbeitete und erweiterte Auflage). Leitfäden der Informatik. Vieweg+Teubner Verlag, Wiesbaden.
- [2] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar.
- [3] Tillal Eldabi, Mariusz Balaban, Sally Brailsford, Navonil Mustafee, Richard E. Nance, Bhakti S. Onggo, and Robert G. Sargent. 2016. Hybrid Simulation. Historical lessons, present challenges and futures. In *Proceedings of the 2016 Winter Simulation Conference (WSC). Simulating complex service systems*. IEEE, Piscataway, NJ, 1388–1403. DOI: <https://doi.org/10.1109/WSC.2016.7822192>.
- [4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press.
- [5] Holger Haag. 2013. *Eine Methodik zur modellbasierten Planung und Bewertung der Energieeffizienz in der Produktion*. Zugl.: Stuttgart, Univ., Diss., 2013. Stuttgarter Beiträge zur Produktionsforschung, 11. Fraunhofer Verlag, Stuttgart.
- [6] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8, 1735–1780.
- [7] Jan Lunze. 2017. *Ereignisdiskrete Systeme*. De Gruyter, Berlin, Boston.
- [8] Navonil Mustafee, Sally Brailsford, Anatoli Djanatliev, Tillal Eldabi, Martin Kunc, and Andreas Tolk. 2017. Purpose and benefits of hybrid simulation: Contributing to the convergence of its definition. In *Proceedings of the 2017 Winter Simulation Conference (WSC)*. IEEE Press, Piscataway, NJ, 1631–1645. DOI: <https://doi.org/10.1109/WSC.2017.8247903>.
- [9] T. Peter and S. Wenzel. 2015. Simulationsgestützte Planung und Bewertung der Energieeffizienz für Produktionssysteme in der Automobilindustrie. In *Simulation in production and logistics 2015. 16. ASIM Fachtagung Simulation in Produktion und Logistik, Dortmund, 23. - 25. September 2015 ; Tagungsband*, Markus Rabe and Uwe Clausen, Eds. ASIM-Mitteilung, 157. Fraunhofer Verl., Stuttgart, 535–544.
- [10] Anna C. Römer, Martina Rückbrod, and Steffen Straßburger. 2018. Eignung kombinierter Simulation zur Darstellung energetischer Aspekte in der Produktionssimulation. In *ASIM 2018 : 24. Symposium Simulationstechnik, 4. bis 5. Oktober 2018, HafenCity Universität Hamburg : Tagungsband*. ARGESIM/ASIM, Wien, 73–80.
- [11] Ilya Sutskever, Oriol Vinyals, and Quoc Le V. 2014. Sequence to Sequence Learning with Neural Networks. In *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. MIT Press, Cambridge, MA, USA.
- [12] Sebastian Thiede. 2012. *Energy efficiency in manufacturing systems*. Zugl.: Braunschweig, Techn. Univ., Diss., 2011. Sustainable production, life cycle engineering and management. Springer, Berlin.
- [13] Benjamin Wörrlein, Sören Bergmann, Niclas Feldkamp, and Steffen Straßburger. 2019. Deep-Learning-basierte Prognose von Stromverbrauch für die hybride Simulation. In *Simulation in Produktion und Logistik 2019*. Verlag Wissenschaftliche Scripten, Auerbach, 121–131.
- [14] Andreas Zell. 2003. *Simulation neuronaler Netze* (4., unveränd. Nachdr.). Oldenbourg, München.

Vorhersage und Regelung der Methanproduktion durch maschinelles Lernen

David Wagner^{1*}, Wolfgang Schlüter¹

¹Fakultät Technik, Biomasse-Institut Hochschule Ansbach, Residenzstraße 8, 91522 Ansbach, Germany; *david.wagner@hs-ansbach.de

Abstract. Das Anaerobic Digestion Model 1 (ADM1, [1]) ist das umfassendste Modell zur Biogasproduktion. Es enthält 32 dynamische Zustandsgrößen aus reinen Differentialgleichungen (DE) und 26 Zustandsgrößen, sowie 8 algebraischen Variablen, die mittels differential-algebraischen Gleichungen (DAE) beschrieben werden. Es ist daher äußerst flexibel einsetzbar. Das Modell selbst basiert auf Kinetiken erster Ordnung bzw. Monod-Kinetiken. Die Komplexität der Gleichungen führt in Simulationen und Optimierungen häufig zu numerischen Problemen. Um die Biogasproduktion prädiktiv zu steuern, wurden vielfach Modellreduktionen oder -abwandlungen vorgenommen. Darunter leidet die Genauigkeit der Vorhersagen und die Flexibilität. Im folgenden Beitrag wird statt einer Reduktion der Modellstruktur des ADM1 das Modell im vollen Umfang genutzt, um randomisierte Datensätze zu erzeugen. Mithilfe dieser Daten werden maschinelle Lernverfahren trainiert und im Anschluss ein tiefes neuronales Netz (DNN) aufgebaut. Es zeigt eine über 99%-ige Übereinstimmung zum ADM1 Modell, wenn lediglich Gleichgewichtszustände vorhergesagt werden, und eine 96.7%-ige Übereinstimmung bei Vorhersage zeitlicher Verläufe des Methanstroms (\dot{m}_{CH_4}). Zudem wird gezeigt, dass das so erhaltene DNN zur Prozesssteuerung verwendet werden kann. Es ist damit in der Lage ein Auswaschen des Reaktors rechtzeitig zu verhindern sowie einen zuvor eingestellten Bedarfsverlauf von \dot{m}_{CH_4} exakt nachzubilden. Damit bietet sich die Möglichkeit zum Einsatz an flexibilisierten Anlagen, die ihre Biogasproduktion am Preis der Strombörse ausrichten.

Einleitung

Biogas wird durch den anaeroben Abbau von landwirtschaftlichen Substraten, wie Gülle oder Pflanzenresten, hergestellt. Der Prozess gliedert sich in fünf Stufen:

Desintegration (A), Hydrolyse (B), Acidogenese (C), Acetogenese (D) und der Methanogenese (E), den Prozessablauf zeigt Abbildung 1. Mit Ausnahme der Desintegration, katalysieren verschiedene Bakterienkonsortien die jeweiligen Reaktionen.

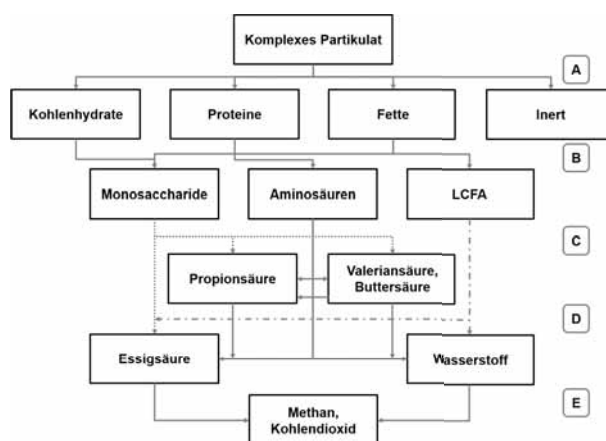


Abbildung 1: Prozess der Biogasproduktion (leicht abgewandelt nach [1]).

Insgesamt werden 105 Parameter für die Beschreibung benötigt. Entsprechende Berechnungen sind langwierig und unterliegen numerischen Instabilitäten. Genaue Parameterkalibrierungen sind mit erheblichem Aufwand verbunden [2, 3]. Für valide Vorhersagen über Gasströme sind diese jedoch erforderlich. Je nach Anlage können die verwendeten Parameter stark vom realen Zustand abweichen. Die eingeschränkte Messbarkeit der Zustandsgrößen limitiert zusätzlich die Identifizierbarkeit der Parameter [4]. Große Konfidenzintervalle und Unsicherheiten im Prozess sind die Folge. Ergebnisse aus Optimierungen unterliegen damit starken Schwankungen. Eine modellprädiktive Regelung (MPC - model predictive control) von Anlagen ist daher mit dem ADM1 nur schwer umsetzbar. Alternativ haben Autoren diverser Publikationen eingeschränkte Modelle oder andere Regler verwendet. So bspw. das Aci-

dogenese / Methanogenese Modell (AM2) [5] oder ein Modell der Totalalkalinität [6], sowie Regler mit Fuzzy-Logic [7]. Zwar sind einige Arbeiten der modellbasierten Regelung mithilfe des ADM1 veröffentlicht (Löffler 2012, Cimatoribus 2009, Gaida 2012), allerdings gelten hierbei ebenfalls die vorgenannten Herausforderungen bei der Kalibrierung des ADM1. Weiterhin ist den Autoren keine industriell genutzte Strategie auf Basis des ADM1 bekannt.

Eine Alternative stellen die hier untersuchten Black-Box (statistische) Modelle dar, die den Prozess als solchen außer Acht lassen. Für viele Anwendungen spielt der eigentliche Prozess nur eine untergeordnete Rolle. Für die Prozessregelung sind lediglich die Führungs-, Stell-, Stör- und Regelgrößen relevant. In generischen PID-Reglern wird die Regelgröße in jedem Zeitschritt neu mit der voreingestellten Führungsgröße verglichen und demnach die Stellgröße beeinflusst. Der MPC ist zusätzlich in der Lage, Vorhersagen über den Prozess zu machen. Damit kann die Stellgröße in jedem Zeitschritt bezüglich des gewünschten zukünftigen Regelgrößenverlaufs optimiert werden. Der Vorteil der MPC gegenüber einer PID-Regelung besteht in der prädiktiven Komponente, die kommende Ereignisse, wie bspw. ein Prozessversagen, antizipieren kann. In Abbildung 2 ist diese Form der Regelung schematisch dargestellt. Die Stellgrößen enthalten die manipulierten technischen Parameter im Prozess (Dilution), die Regelgrößen stellen die durch Messwerte belegbaren Prozessausgangsgrößen (\dot{m}_{CH_4}) dar. Prozessstörungen, durch Messungen oder (unerwünschte) Nebenprozesse, müssen in der Regelung ausgeglichen werden. Der Prozessausgang wird rückgeführt und mit der Führungsgröße (Referenzwert / Idealverlauf) verglichen woraus eine Regelabweichung bestimmt wird. Der Regler selbst kann dabei auf physikalischen oder statistischen Modellen basieren.

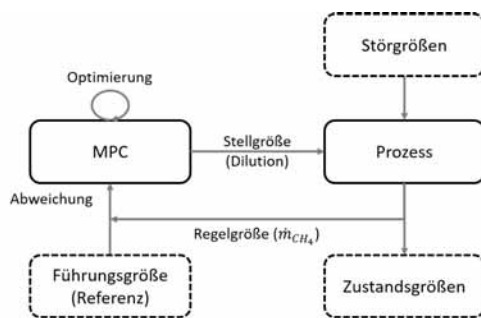


Abbildung 2: Vereinfachtes Schema der MPC-Regelung.

Der MPC-Regler macht Vorhersagen zum Regelgrößenverlauf über einen gleitenden Zeithorizont. Die Op-

timierung orientiert sich an einem vorgegebenen Idealverlauf. Um möglich zeitgerecht auf Prozessstörungen zu reagieren, muss die Stellgröße schnell optimiert werden. Die benannten physikalischen Modelle haben den Nachteil, lediglich Subprozesse abzubilden oder durch ihre Komplexität und damit verbundene Trägheit drohendes Prozessversagen nicht zeitgerecht antizipieren zu können. Im Folgenden werden daher diverse maschinelle Lernmethoden und neuronale Netze mit dem Ziel verglichen, das ADM1 akkurat nachzubilden. Die maschinellen Lernmethoden werden anhand verschiedener, durch das ADM1 generierter, Datensätze trainiert. Um zu prüfen welche Prozessgrößen (Stellgrößen) den größten Einfluss auf den Methanmassenstrom (als Regelgröße) haben, wird eine explorative Datenanalyse (EDA) vorgeschaltet. Nachfolgend werden die relevanten Prozessgrößen zufällig variiert und die Simulationsergebnisse als Trainingsdatensätze genutzt. In einem eingeschränkten Datensatz wird die Dilutionsrate (Fütterungsrate) limitiert, so dass es zu keinem Prozessversagen kommen kann. In einem zweiten komplexen Datensatz wird diese Einschränkung gelockert, so dass es zu einem Auswaschen der Biomasse kommt. In einer ersten Analyse werden nur die Gleichgewichtspunkte vorhergesagt. Im Anschluss werden auch Zeitverläufe nachgebildet, die für einen Einsatz im MPC essentiell sind. Dazu wird das maschinelle Lernverfahren ausgewählt, welches die geringsten Abweichungen zeigt (Minimum residualen quadratischen Fehler - $\min(RSS)$). In der dargestellten Anwendung dient das Verfahren sowohl zur Regelung des Gleichgewichtszustandes (Endpunktregler), als auch zur kompletten Verlaufsregelung des Methanstroms. Das Vorgehen ist in Abbildung 3 gezeigt.

1 Material & Methoden

1.1 Datenvorbereitung & -analyse

Die Güte maschineller Lernverfahren hängt in erster Linie von der Datenbasis ab. Die Simulation des ADM1-Modells erfolgt in Matlab 2019a (auf Basis von [8]). Es werden dafür alle Eingangsparameter im ADM1 variiert. Die Simulationen fließen anschließend in die EDA ein, in der die Korrelation (Corr) zwischen den Eingangsparametern untereinander und dem Gasstrom analysiert wird. Es gilt:

$$Corr(x,y) = \frac{Cov(x,y)}{\sqrt{Var(x)Var(y)}} \quad (1)$$

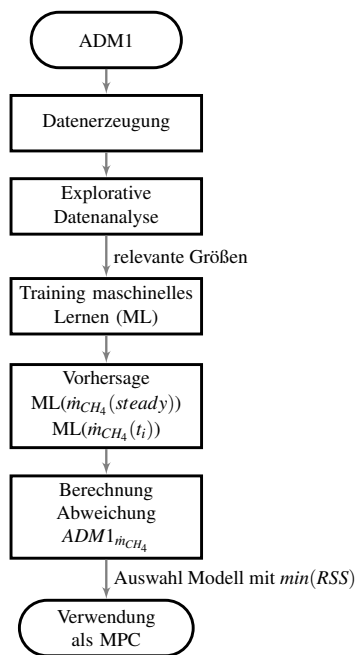


Abbildung 3: Vorgehensschema zur Generierung des MPC-Reglers.

Die Grenzen der Simulation bilden dabei die theoretischen Parametergrenzen. In Tabelle 1 sind die Wertebereiche der relevanten Eingangsgrößen gezeigt.

Parameter	<i>min</i>	<i>max</i>
Dilution (<i>dil</i>)	$1e-2 \text{ h}^{-1}$	$5e-2 / 2e-1 \text{ h}^{-1}$
Kohlenhydratanteil (<i>ch</i>)	0 %	100 %
Proteinanteil (<i>pr</i>)	0 %	100 %
Lipidanteil (<i>li</i>)	0 %	100 %

Tabelle 1: Randbedingungen für die Simulationsdaten.

Im eingeschränkten Datensatz kommt es unter keinen Umständen zu einem Prozessversagen, da keine Bakterien ausgewaschen werden. Die obere Grenze der Dilution wird auf $5e-2$ limitiert. Für den komplexen Datensatz ergibt sich unter Umständen ein Auswaschen der bakteriellen Masse und damit ein Einbruch der Gasproduktion. Für beide Datensätze werden insgesamt 12.000 Punkte zufällig innerhalb der Grenzen mittels ADM1 simuliert und der Methanmassenstrom (\dot{m}_{CH_4}) zeitlich aufgezeichnet. Es wird für die Datenerzeugung das Standardsubstrat aus Batstone et al. verwendet, die Substratanteile werden jedoch (proteins - *pr*, carbohydrates - *ch*, lipids - *li*) ebenso wie die Dilutionsrate (*dil*) variiert.

Die diversen Wertebereiche der Einflussgrößen machen eine Skalierung vor dem Training notwendig.

1.2 Werkzeuge & Bibliotheken

Alle im Folgenden gezeigten Berechnungen finden unter Windows10 Enterprise (Version 1903) auf einem 64-Bit-System mit Core i7-6600 Kern und 2.6 GHz statt. Die Größe des Arbeitsspeichers beträgt 16 GB.

Für die verwendeten Programme und Werkzeuge siehe Tabelle 2. Matlab dient lediglich der Simulation der Daten mit dem ADM1. Andere Operationen finden in Python innerhalb der Anaconda-Umgebung statt. Bei den Regressionen handelt es sich um lineare Regression (LR), polynomische Regression (PR), Kernel-Ridge Regression (KRR) und Random Forest (RF). Bei den polynomischen Regressionen wird ein Polynom 6. Grades verwendet. Die verwendeten neuronalen Netze werden mit NumPy (Ein-Schichten Netz) bzw. Pytorch (Multi-Schichten Netz) aufgebaut. Für die Validierung der einzelnen Modelle werden jeweils 10 % der Daten als Testdatensatz zurückgehalten. Trainings- und Testdaten werden dabei zufällig ausgewählt. Alle Optimierungen werden unter denselben Bedingungen in Python durchgeführt. Die Optimierung des ADM1 wird über eine Schnittstelle (*Matlab-Kernel*) zwischen Python und Matlab realisiert. Die lineare Optimierung wird mit 20 verschiedenen Startwerten aufgerufen und die erhaltenen Ergebnisse miteinander verglichen.

Programm	Werkzeug	Funktion
Matlab (9.6.0)	<i>ode15s-solver</i>	Simulation
Python (3.7.1)	scikit-learn (0.22.2)	Regression
-	NumPy (1.17.0)	Arrayoperation
-	PyTorch (1.4.0)	Neuronale Netze
-	<i>BFGS / shgo-sobol</i>	Optimierung lokal / global

Tabelle 2: Verwendete Werkzeuge bzw. Bibliotheken in Matlab und Python.

1.3 Neuronale Netze

Die verwendeten neuronalen Netze lassen sich in zwei grundlegende Topologien einordnen. Im einfachsten Fall ist lediglich eine Schicht enthalten (Verbindung Input- und Output layer - *NN*). Es werden nur lineare Beziehungen zwischen Ein- und Ausgabe ermittelt. Um

die Vorhersagegenauigkeit zu steigern werden schrittweise Hidden Layer zugefügt (H_n^m in Abbildung 4). Diese erhöhen den Grad der Nichtlinearität. Die Stärke (bspw. $w_{1,1}$) der Verbindungen zwischen den Knoten determiniert die Aktivierung des Neurons. Die einzelnen Werte fließen in die Gewichtsmatrix W ein. Eine Aktivierung findet statt, wenn die Werte einen Schwellenwert, den *bias* (b), überschreiten.

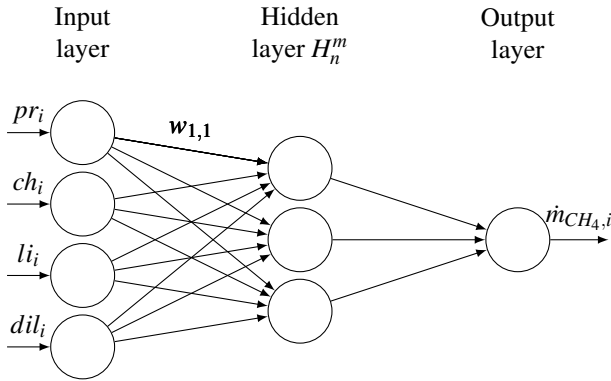


Abbildung 4: Tiefes neuronales Netz zur \dot{m}_{CH_4} -Vorhersage.

Die Netze werden durch die Optimierer Adam und SGD (stochastic gradient descent) trainiert. Die Verlustfunktion ist in beiden Fällen der RSS (Gleichung 2). Sie wird ebenso, mit Ausnahme der KRR, auch für die Regressionen bei scikit-learn verwendet.

$$RSS = \sum_{i=1}^n (\hat{m}_{CH_4,i} - \dot{m}_{CH_4,i})^2 \quad (2)$$

Der Index i repräsentiert den jeweiligen Datenpunkt ($n = 12.000$). Während $\dot{m}_{CH_4,i}$ den wahren (durch ADM1 simulierten) Prozessausgang darstellt, handelt es sich bei $\hat{m}_{CH_4,i}$ um den modellseitig ermittelten Methanmassenstrom. Für die KRR wird ein Regularisierungsparameter α zur Gleichung 2 hinzugefügt, dieser bestraft die Anzahl der Koeffizienten c_p (die Komplexität).

$$RSS_{kr} = \sum_{i=1}^n ((\hat{m}_{CH_4,i} - \dot{m}_{CH_4,i})^2 + \alpha(c_p)^2) \quad (3)$$

Die in der Optimierung berücksichtigten Parameter sind einerseits die Einträge in der Gewichtsmatrix W und andererseits die bias-Terme b . Es ergibt sich damit für den Fall eines Input- und eines Output-layers fol-

gende Gleichung:

$$\hat{m}_{CH_4,i} = w_{1,1}li_i + w_{1,2}ch_i + w_{1,3}pr_i + w_{1,4}dil_i + b \quad (4)$$

Das im Folgenden verwendete tiefe neuronale Netz (DNN) enthält acht Hidden Layer (als H_n^m in Abbildung 4 dargestellt).

2 Ergebnisse & Diskussion

In dem hier dargestellten Szenario wird ein MPC-Regler auf Basis eines Black-Box Modells betrieben. Das ADM1 (im Realfall der Biogasreaktor) entspricht der Regelstrecke, die technischen Parameter (wie z.B. dil) den Stellgrößen und \dot{m}_{CH_4} der Regelgröße. Ein gravierender Nachteil neuronaler Netze ist die mangelnde Extrapolierbarkeit, da ihre Gültigkeit im Allgemeinen auf den Raum der Datenaufzeichnung beschränkt ist. Die mangelnde Extrapolierbarkeit gilt jedoch hierbei ebenfalls für das ADM1. Die Kalibrierung von Parametern muss für jede Anlage neu erfolgen, mit den eingangs beschriebenen Problemen der Nicht-Identifizierbarkeit einzelner Parameter und den daraus folgenden Unsicherheiten bei der Optimierung. Trotzdem gelten die hier dargestellten Ergebnisse lediglich als *Proof of Principle*. Sie zeigen, dass sowohl maschinelle Lernverfahren als auch neuronale Netze die Genauigkeit der Vorhersagen des ADM1 erreichen können. Zudem wird aufgezeigt, dass die Rechenzeiten für die Optimierung deutlich geringer sind. Im Anschluss wird das tiefe neuronale Netz (DNN) genutzt, um mit dessen Hilfe die Dilution entsprechend voreingestellter Idealverläufe des Methanstroms zu optimieren. Wie abschließend gezeigt, lässt sich damit ein Auswaschen der Anlage präventiv verhindern.

2.1 Datenvorbereitung & -analyse

Die Verteilung der skalierten Eingangswerte, die eine gleichmäßige Abdeckung des Raums deutlich macht, ist in Abbildung 5 dargestellt.

Die Skalierung erhöht die Wahrscheinlichkeit der Konvergenz im Trainingsprozess. Es ist allerdings darauf zu achten, dass für eine sinnvolle Aussage eine Reskalierung durchgeführt werden muss. Das Ergebnis der explorativen Datenanalyse ist als Korrelationsmatrix in Abbildung 6 gezeigt.

Es ist deutlich zu erkennen, dass die Dilutionsrate den größten Einfluss auf den Gasfluss (CH_4) hat. Dies

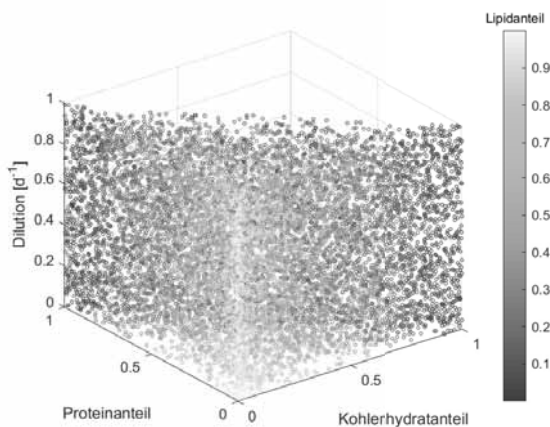


Abbildung 5: Verteilung der Datenpunkte.

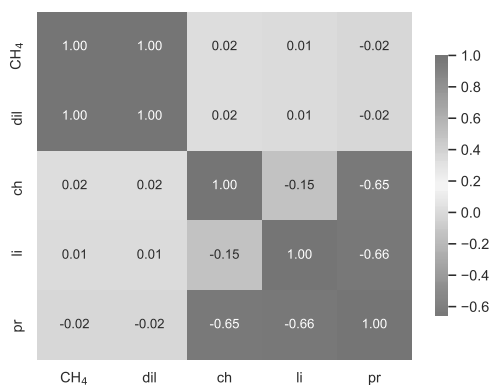


Abbildung 6: Korrelationsmatrix relevanter Größen.

ist zu erwarten, da die Dilution mit der Fütterungsrate gleichzusetzen ist. Ein hoher Substrateinstrom sorgt für eine gute Versorgung der Bakterien und damit hohen Produktivitätsraten. Andererseits ist die Fütterung weitgehend steril. Daher erfolgt bei Dilutionsraten, jenseits der Bakterienwachstumsraten, ein Auswaschen der Biomasse. Die Korrelationen zwischen Substranteilen und Gasfluss sind verschwindend gering. Dadurch kann man vermuten, dass bereits lineare Modelle hohe Vorhersagegenauigkeiten erzielen können. Eine Dimensionsreduktion ist jedoch nicht sinnvoll, da die Anteile der Substrate auch direkt mit der einstellbaren Dilution zusammenhängen.

2.2 Modelldiskriminierung & -anpassung

2.2.1 Simple Regressionsmodelle

Die verwendeten Regressionsmodelle sind:

- Random Forest (RF) - nichtlinear, multiple Entscheidungsbäume
- Polynomial Regression (PR) - nichtlinear
- Kernel Ridge Regression (KRR) - nichtlinear mit Komplexitätsbestrafung
- Linear Regression (LR) - linear

Im ersten Versuch werden die Modelle dazu genutzt den Gasstrom im Gleichgewicht vorherzusagen. Dabei kommt der vereinfachte Datensatz ohne mögliches Auswaschen zum Einsatz.

	RF	PR	KRR	LR
Acc_{Train} in %	99.99	99.99	99.99	99.75
RSS_{Test}	3.0e-4	1.2e-6	1.4e-2	2.3e-1

Tabelle 3: Genauigkeit der getesteten Regressionsmodelle auf dem vereinfachten Datensatz.

Tabelle 3 zeigt sowohl die Genauigkeit der Daten auf dem Trainingsset als auch die Abstraktionsfähigkeit anhand der Abweichung zwischen Testdaten und vorhergesagten Gasströmen. Die polynomiale Regression zeigt die geringste Abweichung. RF-Regressionen neigen generell zum Overfitting, weshalb die Abweichung auf den Testdaten geringfügig höher ist, als bei der PR. Die Abweichung bei KRR ist nochmals höher, was vor allem an den zusätzlichen Parametern liegt (vornehmlich von α). Auch die LR zeigt eine sehr gute Übereinstimmung mit den Testdaten. Aufgrund der vorrangigen Abhängigkeit des Gasstroms von der Dilution ist dieses Ergebnis nicht überraschend. Alle Regressionmethoden sind für den definierten Bereich ohne Auswaschen geeignet. Bei Übertragung auf den größeren Dilutionsbereich im komplexen Datensatz (siehe Tabelle 4) versagen viele jedoch.

	RF	PR	KRR	LR
Acc_{Train} in %	99.94	77.67	55.15	41.77
RSS_{Test}	2.7e-1	4.7e+1	9.3e+1	1.2e+2

Tabelle 4: Genauigkeit der getesteten Regressionsmodelle auf dem komplexen Datensatz.

Lediglich der RF-Regressor ist in der Lage Genauigkeiten über 99% sowohl auf den Trainings- als auch Testdaten zu erzeugen.

2.2.2 Einfaches neuronales Netz (NN)

Aufgrund der Ergebnisse bei den Regressionsmethoden, kann davon ausgegangen werden, dass auch ein DNN in der Lage sein wird, die Daten genau vorherzusagen. Im ersten Schritt wird der eingeschränkte Datensatz auf ein neuronales Netz ohne Hidden Layer angewandt. Dieses Konstrukt entspricht einer LR und sollte daher ähnliche Genauigkeiten liefern. In diesem Fall besteht die Gewichtsmatrix lediglich aus einem Vektor mit vier Komponenten, hinzu kommt ein bias-Term. Das Training des Netzes liefert für die Gleichung (4) die folgenden Werte:

$$w = [-0.62 \quad -0.62 \quad -0.60 \quad 1.02], b = [0.68]$$

Da bereits die Korrelationsmatrix den Einfluss der Substratzusammensetzungen innerhalb der Nährstoffe annähernd gleich bewertet, ergeben sich (bis auf die Dilution) auch ähnliche Gewichtungsfaktoren.

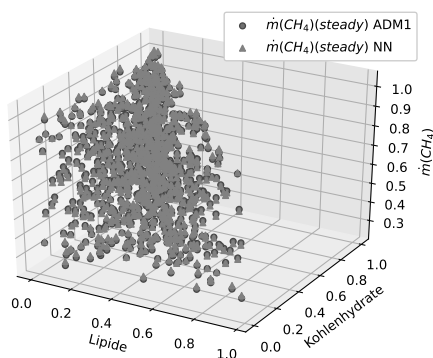


Abbildung 7: Testdaten vereinfachter Datensatz (blau) und Vorhersage durch NN (rot).

Die Genauigkeit mit diesem simplen neuronalen Netz ist dabei sogar der einfachen linearen Regression überlegen. Ein Vergleich zwischen den Testdaten und dem vorhergesagten Gasstrom ist in Abbildung 7 zu sehen. Wird dasselbe Netz auf den komplexen Datensatz angewandt, ergibt sich ein anderes Bild. Einerseits resultiert das Auswaschen in einem nichtlinearen Verhalten des Gasstroms, andererseits sind bei bestimmten Zusammensetzungen aus Substrat und Dilution die Gasströme gleich oder nahe Null. Dieses Verhalten lässt sich mit einem Ein-Schichten Netz nicht wiedergeben (siehe Abbildung 8).

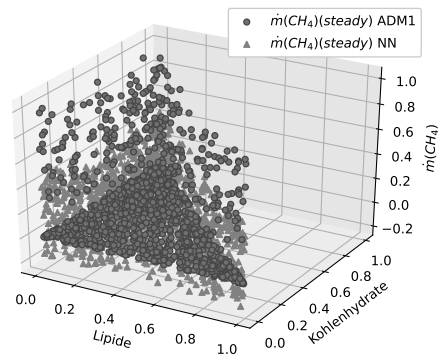


Abbildung 8: Testdaten komplexer Datensatz (blau) und Vorhersage durch NN (rot).

2.2.3 Tiefes neuronales Netz (DNN)

Der RF-Regressor kann auch Daten unter Auswaschkonditionen abbilden. Der Grund dafür ist, dass RF-Regressoren aus einer Vielzahl Entscheidungsbäumen bestehen. Sie erkennen somit auch komplexere Sachverhalte, wie bspw. welche Kombination aus Dilution und Substratzusammensetzung zu einem Auswaschen führt. Der Nachteil ist die hohe Ressourcennutzung der erhaltenen Modelle und die angesprochene Tendenz zum Overfitten. Aus diesem Grund wurde zusätzlich ein DNN aufgebaut. Das erstellte Netz enthält acht Hidden Layer. Es ergeben sich insgesamt 510.528 Parameter (Gewichtungsmatrix und bias-Terme), die innerhalb des Trainings optimiert werden.

Die Genauigkeit bei der Bestimmung des Methanmassenstroms sowohl für den vereinfachten Datensatz als auch den komplexen Datensatz liegt weit über 99%.

2.2.4 Verlaufskurvenvorhersage

Um das DNN im MPC-Kontext nutzen zu können bedarf es zusätzlich noch Informationen über den Zeitverlauf der Methanproduktion. Der Zeitverlauf wird mit 16 Zeit-Massenstrompunkten aufgelöst. Diese gehen in das Training des DNN ein.

Auf den Testdaten beträgt die Genauigkeit 96.70%. Das DNN ist in der Lage, aus der Dilution und der Substratzusammensetzung, vorherzusagen, ob es zu einem Auswaschen während des Prozesses kommt. Beispielsweise sind in der Abbildung 9 zeitliche Verläufe und deren Übereinstimmung mit den simulierten Daten ohne und mit Auswaschen gezeigt. Diese Werte wurden ohne

die Optimierung von Hyperparametern, wie bspw. Dropout, Lernrate oder Lernratenabschwächung, ermittelt.

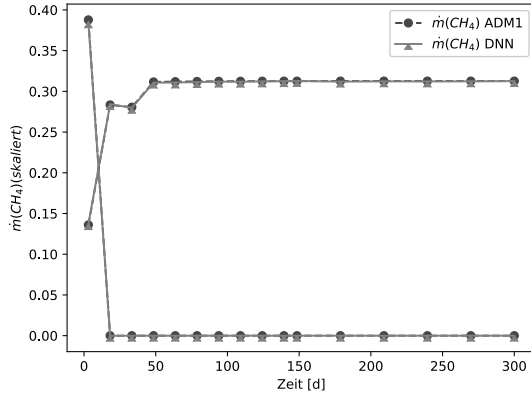


Abbildung 9: Beispielverlauf ohne (obere Kurve) /mit (untere Kurve) Auswaschen. Testdaten blau, Daten von DNN vorhergesagt rot.

2.2.5 MPC-Regelung

Nachdem gezeigt werden konnte, dass alle Datensätze adäquat mit dem DNN dargestellt werden können, wird das DNN in einen MPC integriert. Die Funktionsweise ist schematisch in Abbildung 10 gezeigt.

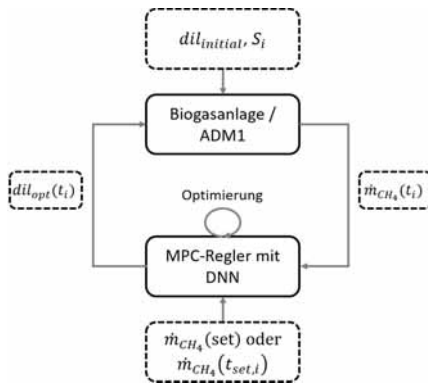


Abbildung 10: Ablauf der MPC-Regelung mittels DNN.

Das ADM1 stellt die Regelstrecke dar und wird mit einer zufällig gewählten Kombination aus Substrat (S_i) und zugehöriger Dilution ($dil_{initial}$) initialisiert, um eine beliebige Biogasanlage zu simulieren. In jedem Zeitschritt wird der aktuelle Methanstrom ($\dot{m}_{CH_4}(t_i)$), als Regelgröße, an den Regler (das DNN) übermittelt. Daraufhin vergleicht der Regler den derzeitigen Stand mit dem zuvor eingestellten Referenzwert für den Methanmassenstrom. Es ergibt sich eine Regelab-

weichung $\varepsilon = \dot{m}_{CH_4}(t_i) - \dot{m}_{CH_4}(t_{set,i})$. Diese wird mittels Einstellung der durch das DNN optimierten Dilutionsrate $dil_{opt}(t_i)$ (Stellgröße) minimiert. Statt zu jedem Zeitschritt, kann die Optimierung auf einen gewünschten Endpunkt ($\dot{m}_{CH_4}(set)$) erfolgen (hier 1500 m^3), wobei es sich dann um einen Endpunktreger handelt und die Dilution lediglich einmal eingestellt bzw. optimiert wird (siehe dazu Abbildung 11).

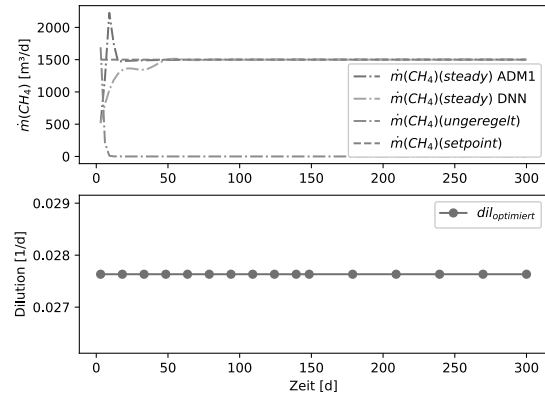


Abbildung 11: Endpunktreger des Methanstroms. Daten des ADM1 blau, durch das DNN vorhergesagter Verlauf gelb, Referenzwert in rot und unregelter Verlauf grün. Unten: Verlauf der Stellgröße.

Um Schwankungen in der Methanproduktion zu verhindern und möglichst auch Überproduktion zu vermeiden, wie in Abbildung 11 sichtbar, kann der MPC die Dilution auch zu jedem Zeitschritt optimieren. Durch die Anpassung der Dilutionsrate kann die Methanproduktion an einen zuvor bestimmten Idealverlauf angeglichen werden (siehe dazu Abbildung 12).

Aus Abbildung 12 lässt sich deutlich entnehmen, dass eine zuvor eingestellte Trajektorie des Methanmassenstroms durch den Regler verfolgt werden kann. Das typische Überspringen, wie aus der Anwendung von PID-Reglern bekannt, bleibt aus. Außerdem wird ein Prozessversagen ohne die Regelung (siehe grüner Verlauf Abbildung 11) vermieden. Die Rechenzeit für die Ermittlung der optimalen Dilutionsrate (globales Minimum) beträgt bei Verwendung des DNN durchschnittlich lediglich 2 s , bei der lokalen Optimierung 0.5 s pro Optimierung. Zum Vergleich werden dieselben Optimierungen mit dem ADM1 aus Python angestoßen, wobei allein die lokale Optimierung durchschnittlich 633 s reine Rechenzeit benötigt.

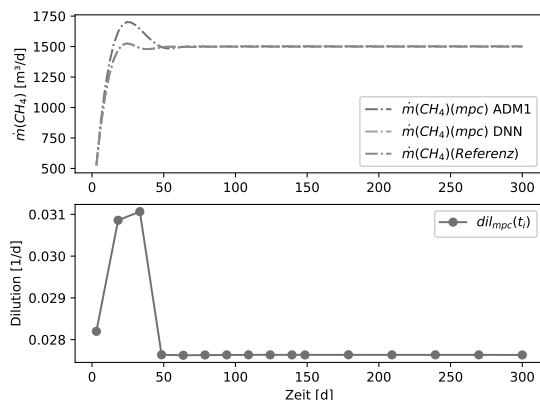


Abbildung 12: Kontinuierliche Regelung des Methanstroms. Daten des ADM1 blau, durch das DNN vorhergesagter Verlauf gelb, Referenzverlauf rot. Unten: Verlauf der Stellgröße.

3 Schlussfolgerung & Ausblick

Diese Veröffentlichung zeigt, maschinelle Lernmethoden können komplexe physikalische Modelle ersetzen. Selbst Verlaufskurven von physikalischen Größen lassen sich mit einfachsten so nachbilden. Die Verfahren können in eine Regelung integriert werden, die ohne Kalibrierungsmaßnahmen auskommt. Für Biogasanlagen stehen die benötigten Trainingsdaten zumeist durch die kontinuierliche Überwachung der Gasmengenzusammensetzung und der zugeführten Substrate zur Verfügung, wohingegen eine Kalibrierung der Modellparameter für das ADM1 zeit- und messintensiv ist. Ein neuronales Netz ist, einmal trainiert, in der Lage, Optimierungen zeitnah durchzuführen. Dies ermöglicht die Anwendung in einem MPC-Regler, um Methanstromtrajektorien nachzubilden und ein Prozessversagen zu verhindern. Durch einen solchen MPC-Regler lassen sich Fahrpläne, die den ökonomischen Vorgaben der Strombörse folgen, realisieren. Allein aus Zeitgründen sind die dafür erforderlichen Optimierungen mit dem ADM1 nicht realistisch umsetzbar.

Literatur

[1] D.J. Batstone, J Keller, I Angelidaki, S.V. Kalyuzhnyi, S.G. Pavlostathis, A Rozzi, W.T.M. Sanders, H Siegrist, and V.A. Vavilin. The IWA Anaerobic Digestion Model No 1 (ADM1). *Water Science and Technology*, 45(10):65–73, 2002.

[2] D. Poggio, M. Walker, W. Nimmo, L. Ma, and M. Pourkashanian. Modelling the anaerobic digestion of solid organic waste - Substrate characterisation method for ADM1 using a combined biochemical and kinetic parameter estimation approach. *Waste Management*, 53:40–54, 2016.

[3] S. Astals, M. Esteban-Gutiérrez, T. Fernández-Arévalo, E. Aymerich, J.L. García-Heras, and J. Mata-Alvarez. Anaerobic digestion of seven different sewage sludges : A biodegradability and modelling study. *Water Research*, 47(16):6033 – 6043, 2013.

[4] Lei Xue, Dewei Li, and Yugeng Xi. Nonlinear model predictive control of anaerobic digestion process based on reduced ADM1. *2015 10th Asian Control Conference: Emerging Control Techniques for a Sustainable World, ASCC 2015*, 2015.

[5] Olivier Bernard, Zakaria Hadj-Sadok, Denis Dochain, Antoine Genovesi, and Jean Philippe Steyer. Dynamical model development and parameter identification for an anaerobic wastewater treatment process. *Biotechnology and Bioengineering*, 75(4):424–438, 2001.

[6] H. O. Méndez-Acosta, B. Palacios-Ruiz, V. Alcaraz-González, V. González-Álvarez, and J. P. García-Sandoval. A robust control scheme to improve the stability of anaerobic digestion processes. *Journal of Process Control*, 20(4):375–383, 2010.

[7] A. Puñal, L. Palazzotto, J. C. Bouvier, T. Conte, and J. P. Steyer. Automatic control of volatile fatty acids in anaerobic digestion using a fuzzy logic based approach. *Water Science and Technology*, 48(6):103–110, 2003.

[8] C Rosen, D Vrecko, K V Gernaey, M N Pons, and U Jeppsson. Implementing ADM1 for plant-wide benchmark simulations in Matlab / Simulink. *Water Science and Technology*, 54(4):11–19, 2006.

Prediction of PM emissions during transient operation of marine diesel engines using artificial neural networks

Michèle Schaub^{1,2*}, Michael Baldauf¹, Egon Hassel²

¹ Institute for Ship Simulation and Maritime Systems (ISSIMS), Wismar University of Applied Sciences, Richard-Wagner-Str. 30, 18119 Rostock-Warnemünde, Germany; *michele.schaub@hs-wismar.de

² Chair of Technical Thermodynamics, University of Rostock, Albert-Einstein-Str. 2, 18059 Rostock, Germany; *michele.schaub@uni-rostock.de

Abstract. Many internal combustion engine emission limits are already prescribed for land transport. Stricter regulations are also expected for international shipping in the future. The International Maritime Organisation (IMO), a subdivision of the UN, has been negotiating for years on direct regulation of particulate emissions from ships. Therefore, in addition to exhaust aftertreatment systems, internal engine and operational measures are also of interest. This article focuses on an operational measure. A new type of assistance software is presented, which shows the nautical ship officer the environmentally relevant consequences of his actions already during manoeuvre planning and later during its execution. The well-known "black flag" on the funnel of a ship is usually the result of transient engine operation. A corresponding assistance software is dependent on a model of transient engine operation including the resulting particle emissions. Over decades, the reaction kinetics for the formation and oxidation of soot have been investigated. Such physical models are to be preferred, provided they meet the quality criteria and the calculation time requirements. At present, however, no model exists which describes the emissions of particulate matters (PM) in transient operation and can make predictions for several minutes within a few milliseconds. An alternative to physical modelling is data-based modelling. The shorter computing time is a major advantage here. On the other hand, there is a great need for training (measurement) data. Two different approaches of Artificial Neural Networks (ANN) were investigated for their applicability to the special case of PM emission prediction under transient engine operating conditions. The advantages and disadvantages of both approaches are discussed in the paper. The results were examined for their applicability in the Maritime Simulation Centre in Warnemünde (MSCW).

Introduction

Emissions of particulate matters (PM) from the combustion process of a marine diesel engine consist of organic and inorganic components. While their generation is relatively low in stationary ship operation, the PM load of the exhaust gas mass flow in transient operation assumes

comparably high values (Figure 1).

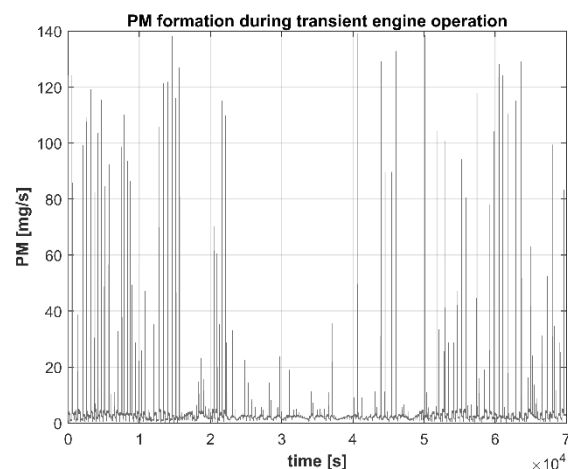


Figure 1: Measurement data from a marine diesel engine with various load changes: Peaks of PM emission are relatively high compared to the stationary PM level.

Even the stationary characteristic map, in which the particle load is shown as a function of engine speed and torque, shows strong non-linearities depending on the engine (Figure 2). The transient operation, however, contains even more non-linearities, which shall be represented by a model.

Over decades, the reaction kinetics for the formation and oxidation of soot have been investigated. Such physical models are to be preferred, provided they meet the quality criteria and the calculation time requirements. At present, however, no model exists which describes the emissions of PM in transient operation and can make predictions for several minutes within a few milliseconds of time. An alternative to physical modelling is data-based modelling. Two different approaches have been selected, to study the specific problem of simulation and prediction of PM. Both use Artificial Neural Networks (ANN).

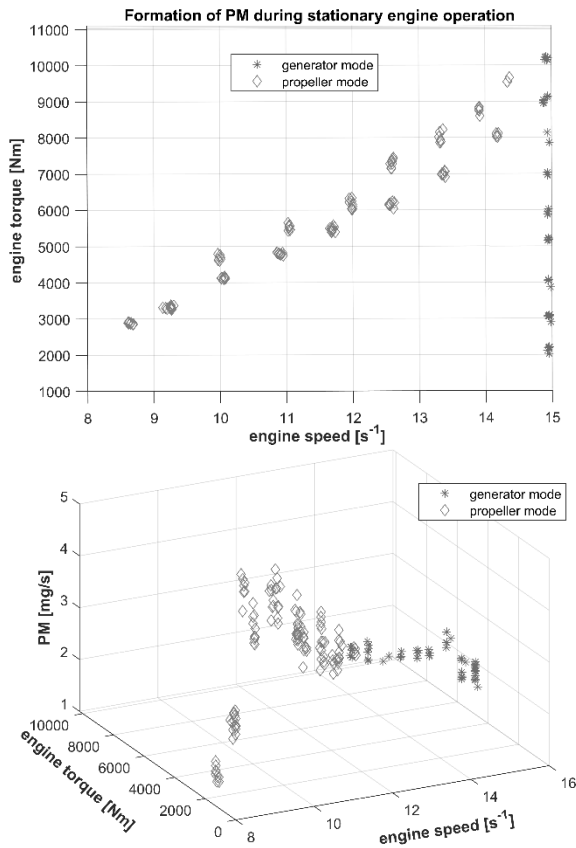


Figure 2: Measurement data from stationary load points from a marine diesel engine. Upper figure: top view of the engine map with generator mode (blue stars) and two times propeller mode with 100% resp. 85% nominal power (orange diamonds). Lower figure: Map display of PM over engine speed and torque.

1 ANN for prediction of dynamic, non-linear processes

There are many possibilities to generate data-based models. The main criteria for the present case study is that such a model must be able to deal with multi-dimensional input. This requirement is based on the assumption that for dynamic modelling, several past values are needed in order to predict the future behaviour of the system.

The current level of PM emissions depends on various factors, e.g. on the current engine speed as well as on the fuel injection. There are also other influencing aspects as e.g. the operation time of the engine and the system's internal temperatures. However, for the use in an assistance software, the two factors mentioned above should be sufficient for this study, in particular because the other operating parameters are by default not accessible at all or only to a limited extent on board sea-going

ships.

The selected architecture of ANN is a feed-forward multi-layer perceptron (MLP) network which means that the links between input, hidden and output layers are unidirectional without feedback between those multiple layers. Hidden neurons as well as the output neuron consist of two types of parameters to be adjusted during the model training: the synaptic weights which are multiplied by the neuron's input and the so called threshold of each neuron's activation function. For the second approach described here below, additional filter coefficients are to be adapted.

1.1 ANN with external dynamics

The expression external dynamics was coined by [1] and describes that the historical values, describing the dynamics, act as input values on the ANN. How many historical values are needed, can be taken from the dynamics of the output value in a first step, in a second step the best fitting number can be determined by a k-fold cross-validation. The following two aspects are important:

1. The curves of the input values should clearly indicate a single output value. If the uniqueness is not guaranteed, further past values have to be added.
2. Not more historical values than necessary should be included, because otherwise the number of hidden neurons has to be increased as well and thus also the number of parameters to be estimated.

1.2 ANN with internal dynamics

Black box systems, like the ANN with external dynamics, are not or only very difficult to interpret. With a large number of input variables, interpretability becomes even more difficult and the number of parameters to be estimated becomes even larger. To counteract these disadvantages, an ANN with internal dynamics is presented and discussed in [2], [3] and [4]. The internal dynamics uses an ARMA filter [5], which creates a memory effect. Such a filter is integrated in every neuron of the hidden layer as well as in the output neuron. Thus, the input space shall be limited to the current input values and at the same time reduce the number of necessary neurons. In contrast to this, the 5 coefficients of the ARMA filter have to be applied additionally for each neuron.

2 Measurement data

2.1 Test bed engine

As already mentioned in the introduction, no data from a theoretical model could be used for the investigations, which is why measurement data from the engine test bench (Table 1) was used directly.

MAN B&W 6L23/30	
Type	medium speed 4-stroke marine diesel engine
Bore	225 mm
Stroke	300 mm
Rated output	960 kW
Rated speed	900 min ⁻¹
Compression ratio	13.5:1
Fuel injection system	unit injector system
Load	water vortex brake

Table 1: Specification of the used engine test bench MAN B&W 6L23/30

The engine allowed test runs on the generator curve as well as on the propeller curve. In generator mode, the controller is set so that the engine speed should remain constant despite changing load. In propeller mode, load and speed change, which corresponds to the operation of a fixed pitch propeller ship.

2.2 Data series

Over a period of six days various load changes were systematically carried out as shown in Figure 1. The first approx. 25000 seconds concern the transient operation on the propeller curve with a nominal power of 100%. The following measurements refer to the generator mode, while from second 43000 on the propeller mode was repeated, but with a nominal power of 85% (see Figure 2). For the training of the two different ANNs, the individual load cycles were strung together in such a way that no discontinuities appear in the data. A smoothing of the data was deliberately avoided, as this would have resulted in the loss of important information.

3 Practical Implementation

3.1 Experiments with a small selection of data

For reasons of better comparison of the two approaches only a selection of the measurement data will be taken in a first approach. Figure 3 shows the selected data which are extracted from the entire measurements. They originate from the measurement data around time 25000 s in Figure 1. This involves the following load changes: from 90% to 100% power, then down to 20% and up to 60%. During these three load changes the system waits for the transient settling to a steady state, but the last load change increases from 60 to 90 and shortly afterwards already to 100% power.

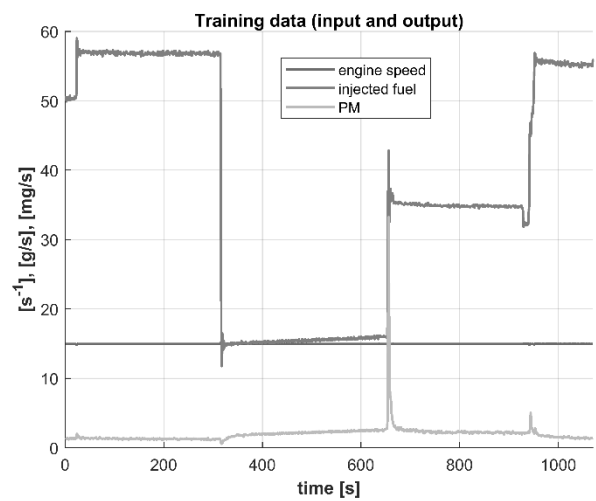


Figure 3: Four load changes of different character in generator mode to compare the two approaches.

These four resp. five load changes refer to the engine mode called generator mode which corresponds to the operation of a ship with controllable pitch propeller. Therefore, only the changes in fuel consumption influence the formation of PM, unless there is a temporary reduction of engine speed during a load change.

3.2 General remarks

The synaptic weights as well as the thresholds start with a randomly determined start value. According to experience, one can adjust the start values within a more suitable and limited range. Due to the uncertainty and by using the random function repeatability is missing. Each new try for model training can lead to a totally different output. By using a k-fold cross-validation statistics upon the selection of the adjustable parameters can be provided. Besides the model parameters (synaptic weights

and thresholds) these are e.g. the number of hidden neurons, training epochs, the learning rate as well as the number of delays when working with external dynamics resp. the filter coefficients when applying the approach with internal dynamics.

ANN are known to need a lot of training data. This example does not satisfy this need for data. It is only intended to show that it is possible to represent the relationship between input and output variables by means of a suitably selected network architecture.

For these reasons, the findings shown here are only some of many, possibly even better results.

3.3 Example: ANN with external dynamics

Settings.

- Start values: randomize(1)
- Number of epochs: 2000
- Learning rate: 0.002
- Number of hidden neurons: 30
- Number of delays: 5

Result. The number of parameters to be adjusted amounts to 151. Figure 4 shows one possible result which looks promising.

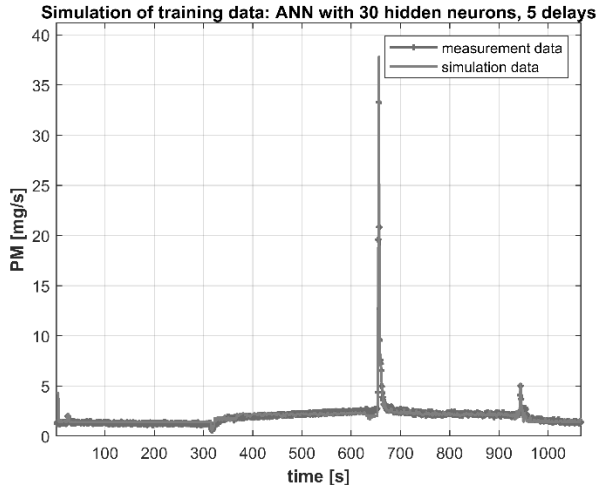


Figure 4: Simulation of training data after training the ANN

3.4 Example: ANN with internal dynamics

Settings.

- Start values of weights: randomize(0.01)
- Start values of filter coefficients: set by experience
- Number of epochs: 80
- Learning rate: 0.002

- Number of hidden neurons: 5
- Number of filter coefficients: 5

Result. The number of parameters to be adjusted amounts to 51. Figure 4 shows one possible result which looks rather worse than the former one. Training with more than 80 epochs led to worse results.

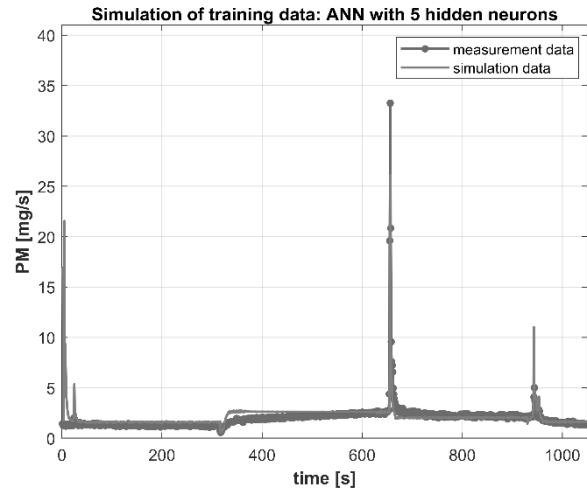


Figure 5: Simulation of training data after training the ANN

3.5 Discussion

Both approaches can reflect the trend in the relationship between input and output variables. Another common feature is the adaptation process at the beginning of the prediction: ANN with external dynamics first needs the specified number of historical values to make a prediction, whereas ANN with internal dynamics needs a few seconds to settle down, because the recursive values of the ARMA filter, the internal memory of the neurons, are not known at the beginning. If one considers the simulation of the four resp. five peaks, the ANN with external dynamics performs obviously better. On the other hand, the last load increase from 60 % to 90 and immediately to 100 % power is qualitatively better represented by the internal dynamics, in which two differentiated peaks are visible according to the measured data (see Figure 6).

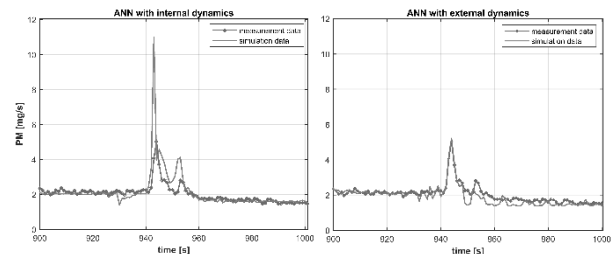


Figure 6: Load increases from 60% to 90% followed by 100%. Left side: simulation of ANN with internal dynamics. Right side: simulation of ANN with external dynamics.

Furthermore, there is also the possibility that an ANN with external dynamics provides relatively imprecise results if the optimization process gets stuck in a local optimum. Then, also negative PM values may be predicted – a result which has not yet been observed with the internal dynamics approach.

4 Results and Validation

The following section shows the application of a MLP ANN with external dynamics for larger measurement data sets. The measurements shown in Figure 1 have been carried out systematically. After the completion of these measurements, random commands were selected over a period of about 5000 s, covering the entire engine map. 75% of the systematically measured time series (up to second 52400) were used for training, the remaining systematic measurement points as well as the supplementary measurements were used for validation.

4.1 Neural Network training

Settings. The same setting as above (3.3) have been chosen for this example.

Training results. Figure 7 shows the measurement data on which the training is based (blue circles). The trained network with the 151 adapted parameters simulates the PM emissions (red line) based on the current input values of engine speed and consumption as well as their historical values.

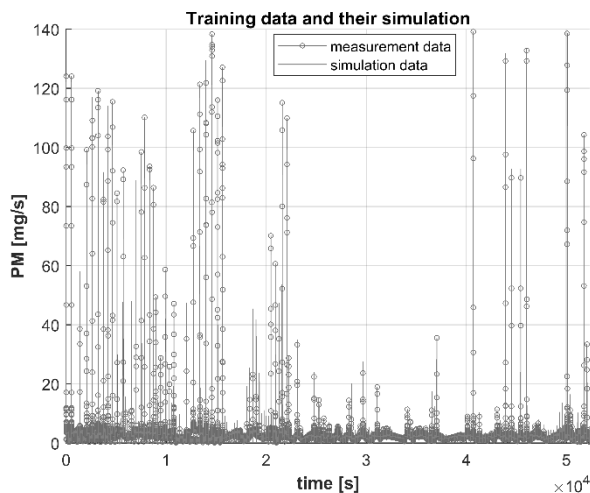


Figure 7: Simulation of training data with a MLP ANN with external dynamics

Zooming out the section selected in (3.3), Figure 8

shows that this ANN also represents the course of the measurements relatively well. Due to the fact that the 30 hidden neurons also have to cover completely different, additional non-linearities, inaccuracies occur in some places.

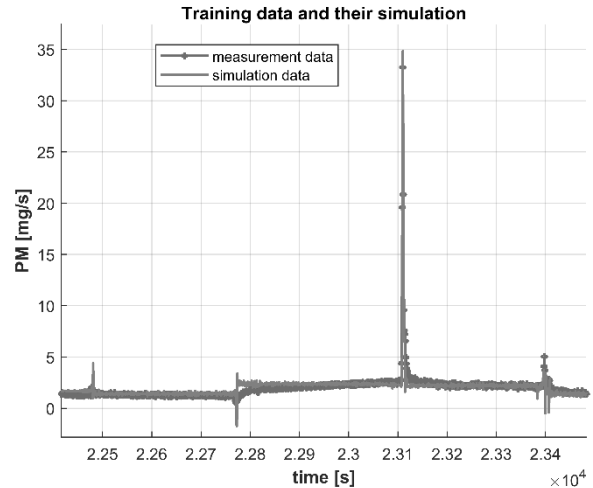


Figure 8: Zoom of above (Figure 4 and Figure 5) selected data, now simulated with the ANN trained with 75% of all measurement data.

4.2 Validation result

The validation result is presented in Figure 9. The PM peaks during transient engine operation tend to be mapped nearly correctly.

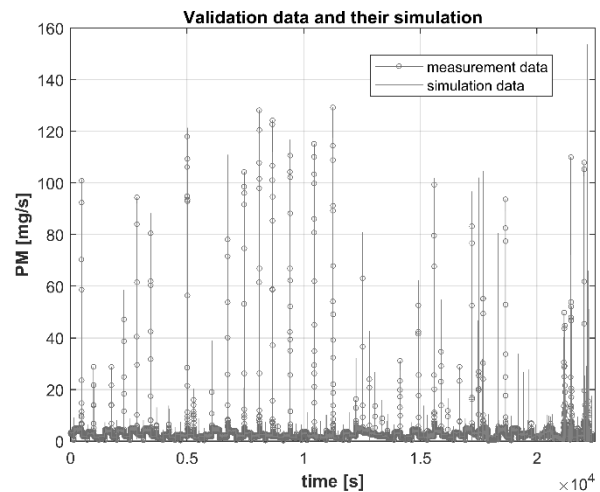


Figure 9: Simulation of validation data with a MLP ANN with external dynamics

The zoomed out section, shown in Figure 10, represents the stepwise ramp-up from 20% load on the propeller curve (85%) to 100% load in 10% steps. The stationary intermediate points are met relatively well.

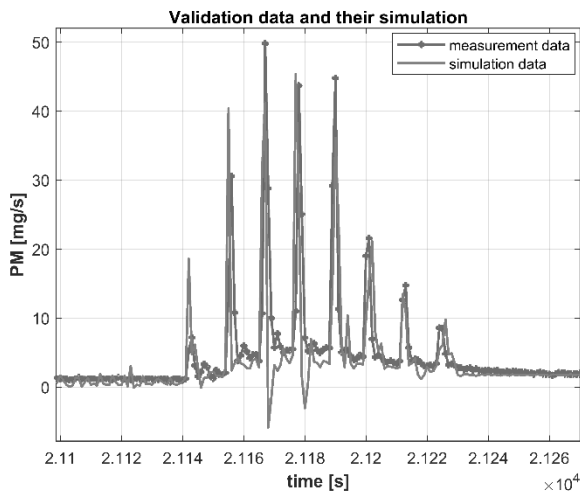


Figure 10: Zoom of validation data and simulation of a stepwise load increase in propeller mode (85%) from 20% to 100% power.

5 Application

5.1 Assistance Software

The aim of the present investigations is to find a suitable method which can subsequently be integrated into an existing manoeuvre assistance system. This system called SAMMON (Simulation Augmented Maneuvring Design and Monitoring) [6] currently supports the navigational officer in planning manoeuvres in advance as well as in their execution. The basis for the prediction is a fast calculating mathematical ship model. Currently, ship movements can be predicted up to 24 minutes ahead. An extension of the engine module enables the prediction of fuel consumption and thus also of emissions. While in online operation the ship's motion and the prediction of its future path is an essential support, the extensions offer valuable possibilities to include environmental concerns in education and training as well as in the planning of manoeuvres on board.

5.2 Practical studies in the simulation centre

The Maritime Simulation Centre Warnemünde (MSCW) offers the possibility to conduct studies for the usefulness of the above mentioned assistance software. Not only the presence of a realistic 360° ship bridge, but also the closeness to nautical students and the direct contact to graduates who are working at sea facilitate the implementation of practical and meaningful tests. Such tests had been carried out as part of the MEmBran project funded by the German government (funding code 03SX423B). [7]

The results clearly showed that, thanks to the support software, the energy input into the water can be reduced: large rudder angles, which considerably increase the ship's resistance, became less, while the number of smaller rudder angles increased only slightly. Power at the propeller was measurably reduced thanks to the assistance software (Figure 11). These energetic improvements have not noticeably influenced the maneuvering time.

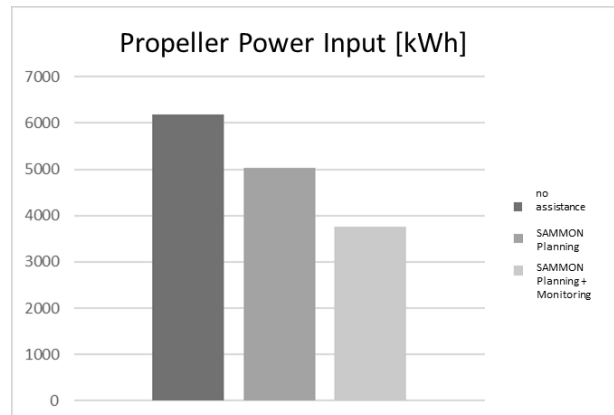


Figure 11: Comparison of the power input of the ship's propeller into the water as a mean value of all executed test scenarios: The blue bar shows the power induced without any assistance, the grey bar is the result of manoeuvres executed with a pre-planning and the yellow bar is the result of a pre-planned manoeuvre and an online monitoring during its execution.

Corresponding comparative studies for the software extended with emission data are still pending.

5.3 Comparison of different strategies

Figure 10 shows a stepwise load change from 20 to 100% in generator mode. The question arises whether this strategy is better in terms of time, energy and environment than a directly given command. With the help of the SAMMON assistance, this question would be answered in a very short time, provided that all necessary models of the ship and its sub-modules are available.

The following two figures illustrate a corresponding experiment. These are measured values from the test bench engine. The load is increased from 20 to 60% in propeller mode (100%). In Figure 12 the load is increased stepwise in 5 % steps, in Figure 13 the increase is initiated with a single command.

For the two strategies, PM emissions and time consumption are to be compared at the time when the same amount of energy has been delivered by the engine. In practice, this would mean that both vessels would have achieved the same manoeuvring target, but with different PM

emissions and in different times.

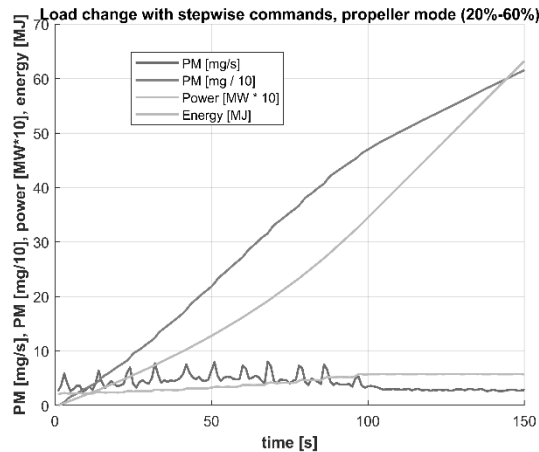


Figure 12: Load change from 20 to 60% in 5% steps. Note the blue line is representing the formation of PM over time and the reddish line is the summation of PM. The power (yellow line) integration, the delivered energy, is illustrated by the green line.

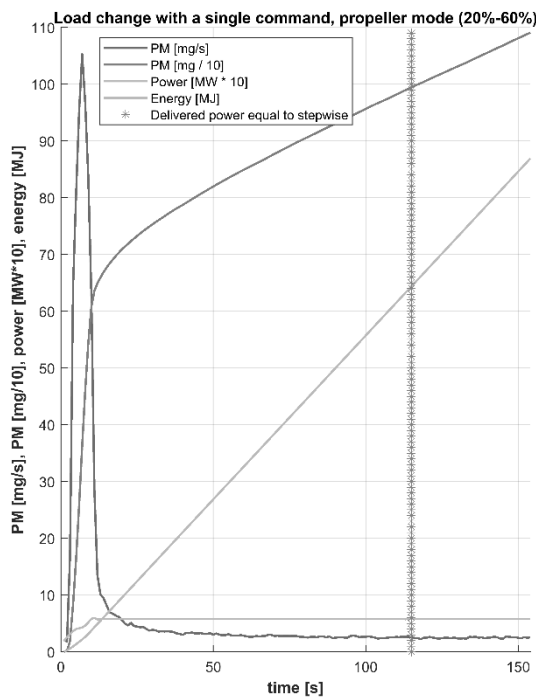


Figure 13: Load change from 20 to 60% with a single command. Note the blue line is representing the formation of PM over time and the reddish line is the summation of PM. The power (yellow line) integration, the delivered energy, is illustrated by the green line. The red vertical line shows the stage at which the same manoeuvre objective was achieved as with the step-by-step strategy.

The step-by-step strategy (Figure 12) achieves stationary operation after 150 s. The energy delivered at that

time is 64.4 MJ (green line). The 64.4 MJ is taken as the reference value. It is observed when the same value is reached in the direct command strategy. After 114 s this is the case (red vertical line in Figure 13). At that time, the same manoeuvre objective was achieved with both strategies. The direct strategy took only 76% of the time (114 s instead of 150 s), but has a 60% higher PM output than the stepwise strategy. (995 mg instead of 621 mg).

6 Potentials for Improving Training and Decision Making

Models to describe the emissions of PM in transient operation for purposes of predictions is a compelling need in today's shipping. While just 20, 30 years ago this need was rather underestimated and models allowing for detailed consideration of optimal engine operation to save fuel had priority. The impact of emissions on climate changed has changed the focus.

Development and implementation of assistance systems need to comprehensively address emissions as a consequence of navigational manoeuvres. Different manoeuvre strategies to safely arrive and berth a ship can be applied according to the prevailing environmental conditions and the actual ship status. The availability of models like those developed and researched here have the potential to improve training of navigational officers and to integrate the concepts of green manoeuvring [8] into maritime education and training of cadets but also into professional development courses of experienced navigators.

Existing tools and manoeuvring assistance systems not only predicting the ship's path according to ordered and intended orders of engine(s), rudder(s), thrusters etc. are important for safety of navigation and the avoidance of groundings or allisions with pillars, jetties or berth constructions etc. However, such tools are increasingly required to also support bridge team's decision making in respect to minimize emissions. Especially when manoeuvring a ship, engine operates in transient modes where emissions are particularly high. Due to the absences of suitable models training and simulation exercises could not completely meet those requirements in the past [9]. Same is valid for assistance systems not taking into account such issues appropriately. Sophisticated models as investigated and tested here have great potential to contribute to substantial improvement in maritime education and training as well as in environmentally-friendly

ship operation by ships' crews as well.

IMO's concept of e-Navigation aims at enhancing "berth to berth navigation and related services for safety and security at sea and protection of the marine environment." It is the authors' hope, that the studied models may contribute especially to the second mentioned aspect of the e-Navigation concept.

7 Summary and Outlook

This paper shows us how existing, data-based methods can be applied to the concrete problem of online predictions of PM emissions. Two model approaches, both of which use ANN, were programmed and trained on established optimisation procedures in order to use them for predictions. A successful validation depends very much on the amount and quality of the training data but also on the settings for start values and other framework conditions.

In general, both methods are able to represent the validation data, if not exactly then at least with the right trend, giving a valuable impression to the trainee about the consequences of his/her actions.

A third method still to be investigated, using a static ANN for stationary and an ARMA-filter for the transient part of PM exhaust gas, is one outcome of the presented studies and will be further pursued.

References

- [1] Isermann, R. *Mechatronische Systeme. Grundlagen*. Berlin: Springer Verlag; 2008. 249 p. doi: 10.1007/978-3-540-32512-3
- [2] Ayoubi, M. *Nonlinear System Identification Based on Neural Networks with Locally Distributed Dynamics and Application to Technical Processes* [dissertation]. [Inst. of Automatic Control, D] TU Darmstadt; 1996.
- [3] Ayoubi, M. Fault Diagnosis with dynamic neural structure and application to a turbocharger. *IFAC Proceedings Volumes*, 27; 1994; Finland. p. 597-602. doi: 10.1016/S1474-6670(17)48090-6
- [4] Nelles, O. *Nonlinear system identification. From classical approaches to neural networks and fuzzy models*. Engineering online library, Berlin: Springer; 2001. 563 p.
- [5] Bohn Ch., Unbehauen H. *Identifikation dynamischer Systeme. Methoden zur experimentellen Modellbildung aus Messdaten*. Wiesbaden: Springer Vieweg; 2016. p 290-total pages of chapter. doi: 10.1007/978-3-8348-2197-3
- [6] ISSIMS GmbH. *SAMMON*. <https://www.issims-gmbh.com/yoomla/sammon>, abgerufen am: 09.09.2020
- [7] Projektträger Jülich. *Statustagung Maritime Technologien. Tagungsband der Statustagung 2019*. Jülich: Forschungszentrum Jülich GmbH, Zentralbibliothek, Verlag; 2019.
- [8] Baldauf, M., Benedict, K., Kirchhoff, M., Schaub, M., Gluch, M., & Fischer, S. (2018). Energy-Efficient Ship Operation: The Concept of Green Manoeuvring. DOI:10.1007/978-3-319-69143-5_11
- [9] Baldauf, M., Baumler, R., Ölçer, A.I., Nakazawa, T., Benedict, K., Fischer, S., & Schaub, M. (2013). Energy-efficient Ship Operation – Training Requirements and Challenges. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, 7, 283-290. DOI:10.12716/1001.07.02.16

Erweiterung der Entwicklungsplattform LoRra um eine Schnittstelle zum Internet der Dinge

Sven Jacobitz^{1*}, Xiaobo Liu-Henke¹

¹Institut für Mechatronik, Ostfalia Hochschule für angewandte Wissenschaften, Salzdahlumer Str. 46/48, 38302 Wolfenbüttel; *sve.jacobitz@ostfalia.de

Kurzfassung. Der vorliegende Beitrag stellt eine Erweiterung der kostengünstigen RCP-Entwicklungsplattform LoRra um eine Schnittstelle zum Internet der Dinge vor. Anhand einer Literaturrecherche werden Anforderungen an die Erweiterung erarbeitet. Mit MQTT als Kommunikationsprotokoll erfolgt die Konzeption und Implementierung der Erweiterung. Durch Grundblöcke des Real-Time Interface in Xcos wird die Konfiguration der MQTT-Verbindung, das Veröffentlichen sowie das Abbonieren von Themen ermöglicht. Anhand eines Anwendungsbeispiels wird die Erweiterung unter Echtzeitbedingungen verifiziert und optimiert.

Einleitung

Das Internet der Dinge (engl. Internet of Things, IoT) ist ein aufstrebendes Paradigma, durch welches eine Vielzahl digitaler smarter Geräte mit dem Internet verbunden werden. Der Begriff IoT ist dabei bereits 1999 von Kevin Ashton das erste mal erwähnt [1]. Hinter diesem Begriff verbirgt sich keine einzelne Technologie, sondern eine Sammlung von Infrastruktur, Diensten, Anwendungen und Steuerungsinstrumenten [2]. Durch die Vielzahl der vernetzten Geräte werden neue Funktionen, wie z.B. die Vorhersage von Wartezeiten in Verkehrsstaus oder ein optimales Energiemanagement im Smart Home, ermöglicht oder die Effizienz vorhandener Dienste gesteigert. Marktforscher gehen davon aus, dass die Anzahl der vernetzten IoT-Geräte bis zum Jahr 2030 fast 30 Mrd. beträgt [3].

Für kleine und mittelständische Unternehmen (KMU) stellt dieser Trend eine große Herausforderung dar. Um konkurrenzfähig zu bleiben, müssen sie immer mehr intelligente Hard- und Software in ihre Produkte integrieren. Schnell entstehen aufgrund des rasant steigenden Funktionsumfangs und Vernetzungsgrades komplexe Softwarekomponenten, welche in starker Wechselwirkung miteinander stehen [4]. Um

die Forderung nach einer schnellen Marktreife zu erfüllen, ist die Entwicklung und Absicherung der entstehenden eingebetteten mechatronischen Systeme unter Anwendung einer effizienten Entwicklungsmethodik unabdingbar [5]. Das durchgängig modellbasierte Rapid Control Prototyping (RCP) ist eine solche Methodik. Essenziell für RCP ist die durchgängige Unterstützung durch eine CAE-Werkzeugkette, um einen hohen Automatisierungsgrad zu erreichen. Etablierte CAE-Werkzeugketten sind sehr kostenintensiv, was insbesondere für KMU eine große Hemmschwelle zur Einführung des RCP-Prozesses darstellt [6].

Im Rahmen des durch die EU geförderten Forschungsprojektes *Low-Cost Rapid Control Prototyping-System mit Open-Source-Plattform für die Funktionsentwicklung von eingebetteten mechatronischen Systemen (LoCoRCP)* wurde an der Ostfalia die kostengünstige Entwicklungsplattform LoRra für die Funktionsentwicklung eingebetteter mechatronischer Systeme entwickelt [7]. In folgendem Beitrag wird diese Entwicklungsplattform für die Forschung im Niedersächsischen Zukunftslabor Mobilität um eine Schnittstelle zum IoT erweitert.

1 LoRra-Entwicklungsplattform

Der durchgängige, modellbasierte RCP-Prozess zur Entwicklung und Absicherung vernetzter mechatronischer Systeme besteht aus den Prozessschritten Modellbildung, Analyse / Synthese, automatisierte Generierung von Quelltext, automatisierte Implementierung auf einer Echtzeithardware und Onlineexperiment. Durch diese Prozessschritte lassen sich Model-in-the-Loop- (MiL-), Software-in-the-Loop- (SiL-) und Hardware-in-the-Loop- (HiL-) Simulationen realisieren.

Ein wesentlicher Bestandteil der Methode ist die automatisierte Transformation von Blockschaltbild-Modellen Programme. Das Blockschaltbild der Funktion wird dabei ohne Eingriffe des Benutzers in äqui-

valenten, hoch performanten Programm Quelltext transformiert. Hierdurch werden sowohl zufällige Fehler durch manuelle Programmierung vermieden, als auch Entwicklungszeit eingespart [8]. Die modular aufgebaute Entwicklungsplattform LoRra unterstützt diesen Prozess durchgängig. Abbildung 1 illustriert den RCP-Entwicklungsprozess sowie die durchgängige Unterstützung mittels LoRra [9].

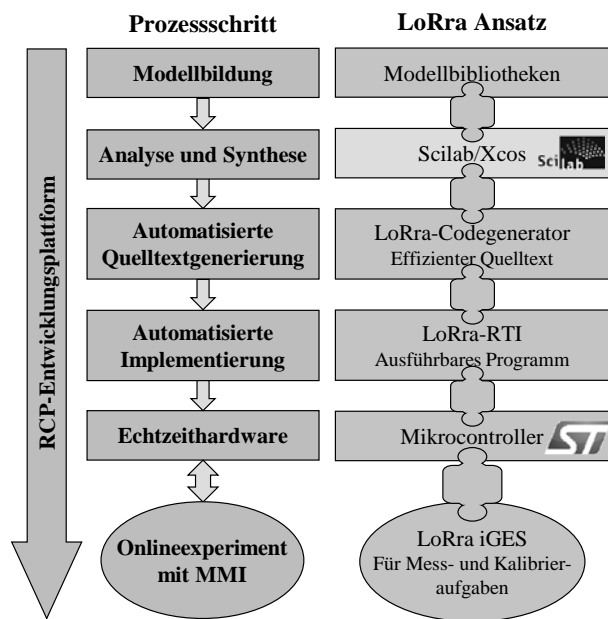


Abbildung 1: Konzept der Entwicklungsplattform LoRra [7].

Für den Prozess der Modellbildung sind domänenübergreifende Modellbibliotheken vorhanden. Mittels Versions- und Konfigurationsmanagement können Modellvarianten übersichtlich direkt in der Simulationsumgebung von Scilab / Xcos zusammengestellt und verwaltet werden. Das Open-Source-CAE-Werkzeug Scilab wird zur Analyse und Synthese der Funktionen verwendet. Es bietet dabei einen ähnlichen Funktionsumfang wie das kommerziell häufig verwendete Matlab / Simulink. Das entstehende Funktionsmodell kann direkt in die Modellbibliothek integriert werden. Durch die offenen Schnittstellen des LoRra-Code-Interface (LCI) lassen sich zudem mit geringem Aufwand vorhandene Programme und Schnittstellentreiber einbinden. Mittels MiL-Simulationen können bereits in frühen Entwicklungsstadien Optimierungen und Tests der entwickelten Funktionen durchgeführt werden.

Durch den LoRra-Codegenerator wird mittels Model-zu-Text-Transformation automatisiert effizien-

ter, modularer C-Quelltext aus dem Funktionsmodell generiert. Durch offene funktionale Beschreibungen von Grundelementen des Modells, sogenannte Grundblöcke, ist der LoRra-Codegenerator flexibel erweiterbar. Der generierte Quelltext entspricht den Spezifikationen des LCI, sodass dieser, z.B. zur Optimierung und Test mittels SiL-Simulationen, ohne manuelle Arbeiten wieder in das Xcos-Modell eingebunden werden kann.

Die Verbindung zu Modellen der Regelstrecke oder weiteren Funktionen werden bei hinreichendem Funktionsstand durch Schnittstellenblöcke des LoRra Real-Time Interface (RTI) ersetzt. Hierdurch erfolgt ohne manuelle Programmierung die Ansteuerung der Echtzeithardware. In Kombination mit einer hardware-spezifischen RTI-Basissoftware, welche unter anderem ein Echtzeitbetriebssystem und standardisierte Schnittstellentreiber beinhaltet, wird somit eine automatisierte Implementierung auf der Echtzeithardware durch das RTI möglich. Als Echtzeithardware werden kostengünstige Mikrocontroller z.B. der Serie STM32H7 eingesetzt. Mittels HiL-Simulationen kann die entwickelte Funktion somit auch unter Echtzeitbedingungen optimiert und getestet werden. Als Mensch-Maschine-Interface (MMI) steht dabei die integrierte Graphikunterstützte Experimentiersoftware (iGES) zur Verfügung. Mit dieser lassen sich Onlineexperimente intuitiv steuern und überwachen sowie Messdaten aufzeichnen.

2 Stand des Wissens

IoT ist ein Interdisziplinäres Paradigma, in welchem viele Geräte mit dem Internet verbunden sind um neue Funktionen anzubieten oder die Effizienz von Funktionen zu steigern [10]. Dabei wird eine globale Infrastruktur von Heterogenen vernetzten eingebetteten Geräten und Objekten genutzt [11]. Die Zahl der durch das IoT vernetzten Geräte steigt dabei exponentiell an. Insbesondere durch neue Technologien im Bereich der Kommunikation (z. B. 5G) werden immer mehr mobile Funktionen ermöglicht [12].

2.1 Architektur von IoT-Anwendungen

Werden Geräte miteinander vernetzt, muss dieses über eine sowohl hardware- als auch softwaretechnisch einheitliche Schnittstelle erfolgen. Insbesondere wenn die Integration neuer Geräte unterschiedlicher Hersteller ohne aufwändige Konfiguration möglich sein soll, muss hierzu eine geeignete Architektur vorhanden sein [12].

[13] gibt hierzu eine Übersicht verschiedener Lösungsansätze.

Zur Vereinheitlichung können IoT-Anwendungen in die Prozesse Datenerfassung, Datenübertragung, Datenverarbeitung und Datenspeicherung gegliedert werden [10]. Diese können je nach Anwendung auf Endgeräten oder durch im Netzwerk verbundene Geräte wie Fog- oder Cloud-Systeme durchgeführt werden [14]. Entsprechend dieser Prozesse lässt sich die Architektur einer IoT-Anwendung in vier Schichten einteilen [15]. Von unten nach oben sind dies:

1. *Perzeptionsschicht*: ist die unterste Schicht und dient vornehmlich der Datenerfassung. Neben verschiedensten Sensoren, welche oft zu einem Sensor-Hub integriert sind, können hier auch Aktoren sowie deren lokale Regelung eingeordnet werden.
2. *Netzwerkschicht*: dient der Datenübertragung zu anderen IoT-Geräten sowie der übergeordneten Datenverarbeitungsschicht. Hier werden häufig drahtlose Technologien wie WLAN, Bluetooth oder LTE eingesetzt.
3. *Datenverarbeitungsschicht*: verarbeitet und analysiert die Daten aus der Perzeptionsschicht z.B. mittels Machine-Learning-Verfahren. Dies wird häufig durch Fog- oder Cloud-basierte Architekturen bereitgestellt. Neben der Verarbeitung erfolgt hier je nach Anwendung auch die Datenspeicherung sowie die Kommunikation der Ergebnisse an weitere IoT-Geräte.
4. *Anwendungsschicht*: präsentiert und speichert die Ergebnisse der Datenverarbeitungsschicht. Sie stellt die Schnittstelle zum Benutzer dar und erfüllt in diesem Zusammenhang diverse Aufgaben.

2.2 Kommunikation zwischen IoT-Geräten

Grundlage des IoT ist die Maschine-zu-Maschine-(M2M-) Kommunikation. Die Entwicklung neuer Technologien verläuft dabei in verschiedenste Richtungen. Meist werden IoT-Geräte physikalisch über Funkt mit dem Internet verbunden. Verschiedene Funktechnologien wie 5G, Bluetooth Low Energy, Low Power wide Area Networks und viele mehr konkurrieren dabei darum der Standard zu werden [16]. Die Anforderungen an Reichweite, Energieverbrauch, Verbindungssicherheit, etc. sind dabei so stark gestreut, dass die Erfüllung

durch eine Funktechnologie für alle Anwendungen derzeit nicht möglich ist [10].

Auf höheren Ebenen des *Open Systems Interconnection model* (auch OSI-Schichtenmodell) nach ISO/IEC J7498 werden meist IP-basierte Kommunikationsprotokolle mit dem *Transmission Control Protocol* (TCP) oder dem *User Datagram Protocol* (UDP) als Transportschicht eingesetzt. Diese werden von einem IoT-Nachrichtenprotokoll überlagert [17]. Umfangreiche Forschungen wurden bereits zur Analyse, Bewertung und Vergleich verschiedener Protokolle durchgeführt. So bieten [18] und [19] eine allgemeine Übersicht über bekannte IoT-Nachrichtenprotokolle wie das *Constrained Application Protocol* (CoAP), das *Named Data Networking* (NDN) und das *Message Queuing Telemetry Transport Protocol* (MQTT). Hierbei sind CoAP und MQTT aufgrund der Flexibilität, Stabilität und Effizienz weit verbreitet [20].

CoAP wurde im Juni 2014 von der *Internet Engineering Task Force* (IETF) als Standard veröffentlicht und zählt damit zu den neueren IoT-Protokollen [21]. Es basiert auf einem Client / Server Modell und wurde für ressourcenarme Geräte sowie einer hohen Kompatibilität zum *Hypertext Transfer Protocol* (HTTP) entwickelt. Es nutzt im Gegensatz zu HTTP UDP als Transportschicht, wodurch es geringeren Overhead verursacht und mit geringerem Implementierungsaufwand verbunden ist [22]. CoAP behandelt Daten grundsätzlich als Zeichenkette. Zur Strukturierung werden Austauschformate wie XML-basierte Sprachen oder JSON genutzt [23].

MQTT ist ein einfach zu verwendendes, offenes Protokoll zur Kommunikation im IoT. Es wurde ursprünglich von IBM im Jahr 1999 entwickelt und ist seit 2013 durch die *Organization for the Advancement of Structured Information Standards* (OASIS) standardisiert [24]. Im Frühjahr 2018 wurde die Version 5.0 mit Anpassungen an neue Technologien und Herausforderungen veröffentlicht. MQTT basiert auf einem Publish / Subscribe System, bei dem ein Server (Broker) die gesamten Daten der Kommunikationspartner (Clients) verwaltet. Durch dieses Prinzip ist es möglich leistungsschwache Geräte zu vernetzen und komplexere Berechnungen auf performante Systeme auszulagern. MQTT nutzt als Transportprotokoll TCP/IP und unterscheidet bei der Nachrichtenzustellung zwischen drei Quality of Service (QoS). Hierdurch kann der Entwickler wählen ob Nachrichten ohne Rückmeldung (QoS 0, kann zu Datenverlust führen), Mindestens einmal (QoS

1, die Nachricht wird so lange gesendet, bis eine Empfangsbestätigung eingeht) oder genau einmal (QoS 2) zugestellt werden. Die Organisation der Inhalte erfolgt über eine Baumstruktur. Durch die Zweige (Topics) lassen sich Themengruppen und hierarchische Strukturen abbilden [25].

Die Effizienz, Stabilität und Sicherheit von CoAP, MQTT sowie weiteren IoT-Kommunikationsprotokollen ist bereits gut untersucht. [20] untersucht für Narrowband-IoT-Technologie anhand von stationären Versuchsaufbauten die durch das Protokoll hervorgehende Netzwerkbelastung. Aufgrund des geringeren Overheads von UDP sowie der Möglichkeit mehrere Themen in einer Nachricht zu übertragen, belastet CoAP insbesondere bei großen Datenmengen das Netzwerk weniger als MQTT. [26] kommt für stationäre Netzwerkaufbauten zu dem selben Schluss. Eine detailliertere Untersuchung anhand größerer Anwendungen sowie eine Übersicht weiterer durchgeführter Leistungsanalysen bietet [19].

3 Konzeption der Erweiterung

3.1 Anforderungen an die Erweiterung

Durch die Erweiterung des LoRra-RTI soll in erster Linie die Entwicklung von Perzeptionsschicht-Anwendungen der in Abschnitt 2.1 vorgestellten Architektur ermöglicht werden. Sie bezieht sich hierbei auf die informationstechnische Erweiterung. Zur physikalischen Datenübertragung muss eine entsprechend geeignete Echtzeithardware gewählt werden. Um die funktionalen Anforderungen zu spezifizieren erfolgt zunächst eine Vorstellung der Akteure und Objekte, mit denen die Erweiterung interagiert:

- *Regulär-Signal (Signal)*: quasi zeitkontinuierliches rationales Signal. Kann sowohl ein Skalar als auch eine Matrix sein.
- *Ereignis-Signal (Ereignis)*: zeitdiskrete Ereignisse wie Zeitperioden oder steigende Flanken, welche Blöcke aktivieren.
- *Andere IoT-Geräte*: über das IoT verbundene Geräte beliebiger Anwendungsebene, die Signale von der Erweiterung empfangen oder Signale zur Erweiterung senden.
- *Benutzer*: Funktionsentwickler oder anderer Nutzer, der die Erweiterung im Xcos-Model nutzt.

- *Zielhardware*: gewähltes Echtzeithardwaresystem, auf welchem das generierte Programm abläuft.

Für die Erweiterung ergeben sich übergeordnet vielfältige funktionale und nichtfunktionale Anforderungen, die anschließend unter anderem zur Auswahl eines Kommunikationsprotokolls genutzt werden. Die relevantesten sind:

- R1 *Versenden von Signaldaten*: Signale sollen eindeutig identifizierbar wahlweise bei jedem Berechnungsschritt oder ausgelöst durch ein Ereignis gesendet werden.
- R2 *Empfangen von Signaldaten*: Signale einer zur Übersetzungszeit festgelegten Identifikation sollen laufend empfangen werden. Nach Wahl des Benutzers soll bei Empfang einer neuen Nachricht ein Ereignis ausgelöst werden.
- R3 *Hohe Ausfallsicherheit*: es ist sicherzustellen, dass Nachrichten auch bei schlechten Verbindungsbedingungen zugestellt werden und nicht verloren gehen.
- R4 *Geringer Latenz*: Signale sollen mit geringer Zeitverzögerung an andere IoT-Geräte gesendet werden.
- R5 *Hohe Kompatibilität zu anderen IoT-Geräten*: bei dem genutzten Kommunikationsprotokoll soll es sich um ein gängiges Protokoll oder ein Protokoll mit hoher Kompatibilität zu anderen Protokollen handeln.
- R6 *Einfache Bedienung*: der Benutzer soll mit geringem Konfigurationsaufwand Signale senden und empfangen können.
- R7 *Ressourcenschonende Implementierung auf der Zielhardware*: die IoT-Erweiterung soll auf der Zielhardware ressourcenschonend (insb. Speicher und Rechenaufwand) implementiert werden.

3.2 Kommunikationsprotokoll

Aufgrund der in Abschnitt 3.1 dargelegten übergeordneten Anforderungen sowie der in Abschnitt 2.2 dargestellten Forschungen wird MQTT 5.0 als Kommunikationsprotokoll für die IoT-Erweiterung genutzt. Ausschlaggebend hierfür ist die zentral durch den Broker verwaltete Baumstruktur, durch die Signale eindeutig

identifizierbar sind (R1, R2) sowie die einfache Bedienung durch Angabe von Topics (R6) mittels Publish / Subscribe Prinzip. Zudem benötigt MQTT keinen komplexen Parser für ein- und ausgehende Daten, was die Implementierung in eingebetteten Systemen ressourcenschonender macht (R7).

Abbildung 2 illustriert ausgewählte relevante Anwendungsfälle (engl. Use Case, UC) zur Nutzung von MQTT. Als Akteure sind das User Programm, also die aus dem Xcos-Modell hervorgehende Anwendung, sowie der MQTT-Broker dargestellt. Zwei Relevante UC aus Sicht der Zielhardware sind das Senden und Empfangen von MQTT-Nachrichten. Alle weiteren UC beinhalten das Senden oder Empfangen von Nachrichten. Die restlichen dargestellten UC realisieren unter anderem die wichtigsten MQTT-Nachrichtentypen CONNECT, PUBLISH und SUBSCRIBE.

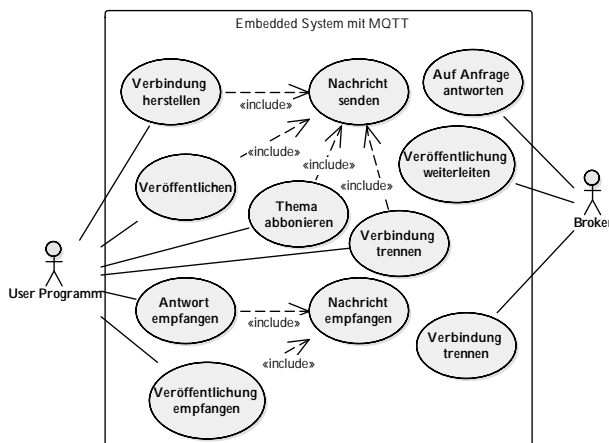


Abbildung 2: MQTT Anwendungsfälle.

Die Kommunikation der Nachrichten erfolgt in einer durch die MQTT-Spezifikation [25] vorgeschriebene Reihenfolge. Diese ist für den QoS 1 in Abbildung 3 dargestellt. Jedes MQTT-fähige IoT-Gerät kann sich als Client am Broker durch Senden einer CONNECT-Nachricht anmelden. Ist die Anmeldung erfolgreich, reagiert der Broker dies mit einer Bestätigung (engl. acknowledge, ack). Über SUBSCRIBE-Nachrichten kann ein Client spezifische Themen oder ganze Äste des Datenbaums abonnieren. Veröffentlicht ein anderer Client in einem abonnierten Thema eine neue Nachricht, wird dies durch den Broker mitgeteilt. Über das Senden einer PUBLISH-Nachricht lassen sich Daten in festgelegten Themen veröffentlichen. MQTT 5 arbeitet mit unterschiedlichen Datentypen. Die IoT-

Erweiterung nutzt aus Kompatibilitätsgründen in Zeichenketten gewandelte Fließkommazahlen, bzw. deren Darstellung im Scilab Matrizenform, zur Übertragung.

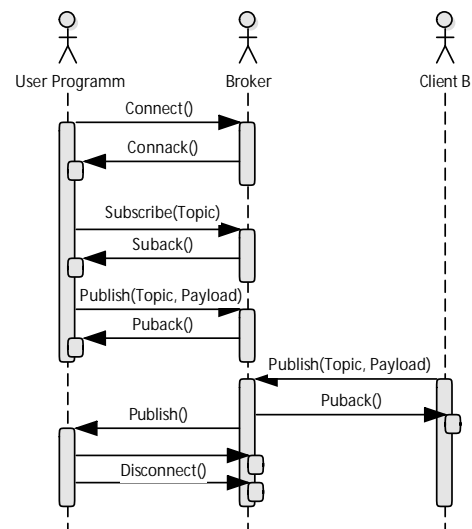


Abbildung 3: Ablauf der MQTT-Kommunikation für QoS 1.

3.3 Struktur der Erweiterung

Aus den in Abschnitt 3.1 ausgearbeiteten Anforderungen sowie den in Abschnitt 3.2 vorgestellten UC lässt sich die Struktur der IoT-Erweiterung ableiten. Diese wird wie das RTI in Modell- und Echtzeitebene gegliedert. Hieraus entstehen zwei Bestandteile, die RTI-Schnittstellenblöcke zur Bedienung der Erweiterung und eine MQTT C-Bibliothek für die Integration in der RTI-Basissoftware.

Für die Bedienung der Erweiterung werden drei Schnittstellenblöcke konzipiert:

- `lorra_rti_mqtt_config` dient der Konfiguration. Hier werden vom Benutzer Daten wie die IP-Adresse des Brokers und eine eindeutige Client ID angegeben. Dieser Konfigurationsblock muss genau einmal im Modell vorhanden sein, wenn die IoT-Erweiterung genutzt wird.
- `lorra_rti_mqtt_publish` veröffentlicht Signale in einem vom Benutzer vorgegebenen Topic. Wahlweise kann die Veröffentlichung bei jedem Berechnungsschritt oder ausgelöst durch ein Ereignis erfolgen.
- `lorra_rti_mqtt_subscribe` empfängt Signale eines vom Benutzer vorgegebenen Topics.

Wahlweise wird bei Empfang einer neuen Veröffentlichung ein Ereignis ausgelöst.

Die Erweiterung der RTI-Basissoftware erfolgt durch das Modul `lorra_rti_mqtt`, dessen Struktur durch Abbildung 4 illustriert wird. Dieses beinhaltet neben Funktionen zum Initialisieren, Senden und Empfangen auch die notwendigen Datenstrukturen.

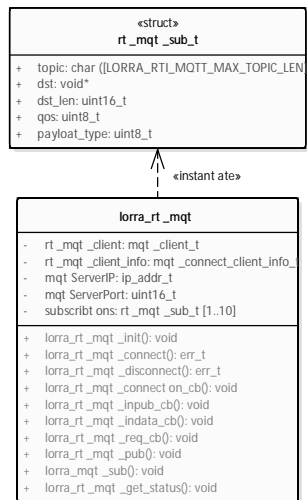


Abbildung 4: Erweiterungsmodul der RTI-Basissoftware.

4 Realisierung der Erweiterung

Zur Realisierung der IoT-Erweiterung wurden die in Abschnitt 3.3 beschriebenen benötigten Schnittstellenblöcke in Scilab implementiert. Abbildung 5 zeigt exemplarisch die Benutzerschnittstelle des Blocks `lorra_rti_mqtt_config`. Neben der Benutzerschnittstelle gehören zu jedem Schnittstellenblock Transformationsregeln für die automatisierte Quelltextgenerierung.

Die Realisierung des Basissoftware-Moduls erfolgt unter Verwendung des Open-Source-TCP/IP-Stacks LwIP [27]. Dieser wird bereits von andere Module der Basissoftware genutzt und bietet eine verbreitete Implementierung der TCP/IP Transportschicht. LwIP beinhaltet unter anderem einen bereits implementierten MQTT-Clienten, auf welchen zurückgegriffen werden kann. Das Modul `lorra_rti_mqtt` verknüpft das automatisch generierte Nutzerprogramm mit dem LwIP-MQTT-Client.

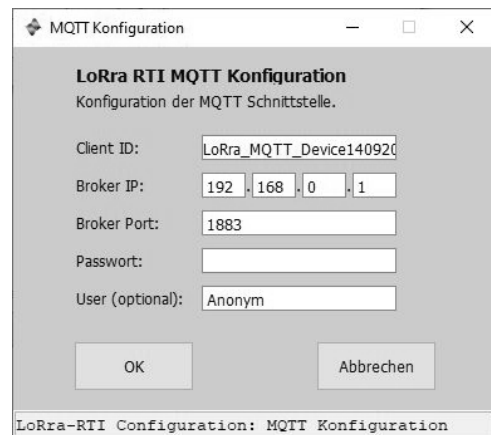


Abbildung 5: MQTT Konfigurationsfenster des LoRa-RTI.

5 Test der Erweiterung

Zur Optimierung und Test der IoT-Erweiterung wurden diverse Testfälle erfolgreich durchgeführt. Hierbei erfolgten sowohl offline automatisierte Verifikationen der Schnittstellenblöcke mittels Scilab-eigener Testumgebung als auch die Durchführung von HiL-Simulationen zur Verifikation der gesamten Erweiterung unter Echtzeitbedingungen. In diesem Abschnitt wird exemplarisch die HiL-Simulation der Drehzahlregelung eines Gleichstrommotors (GSM) mit einer Sollwertvorgabe über MQTT vorgestellt.

5.1 Versuchsaufbau

Abbildung 6 illustriert den Versuchsaufbau zur Drehzahlregelung des GSM mit Getriebe und Lastträgheitsmoment. Auf der Zielhardware (Client A), einem Mikrocontroller vom Typ STM32H7 dessen Programmierung automatisiert mittels LoRa erfolgt, wird in Echtzeit die Drehzahlregelung ausgeführt. Die Sollwerte der Regelung werden von Client B über MQTT gesendet und durch die IoT-Erweiterung verarbeitet. Mittels Inkremental Encoder wird die aktuelle Drehzahl gemessen, verarbeitet und über MQTT veröffentlicht. Mittels PWM-Ansteuerung eines Vierquadrantenstellers wird die berechnete Stellgröße umgesetzt. Über ein lokales Netzwerk sind die Clienten A und B einem Broker, realisiert durch die Open-Source-Software Mosquitto [28], verbunden.

Das Xcos-Modell der Drehzahlregelung ist in Abbildung 7 dargestellt. Hierbei wird die Drehzahl der Lastmasse, welche über ein Getriebe mit einer Über-

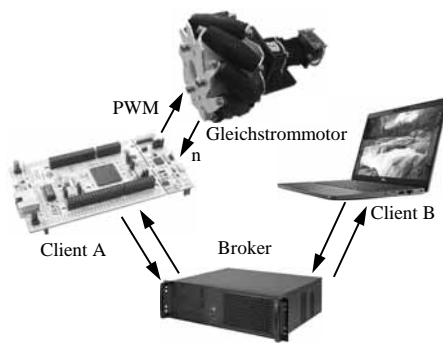


Abbildung 6: Versuchsaufbau zum Test der IoT-Erweiterung.

setzung von $i = 64$ durch den GSM angesteuert wird, geregelt. Über das MQTT-Topic `/motor_control/n_des` wird die Soll-Drehzahl übermittelt. Die gemessene Ist-Drehzahl wird zudem synchron zum Berechnungsschritt in dem Topic `/motor_control/n_act` veröffentlicht. Der modellbasiert ausgelegte PI-Regler ermittelt mit einer Abtastzeit von 10ms aus der Regeldifferenz die Stellgröße (Spannung an den Motorklemmen), welche in ein PWM-Signal transformiert wird.

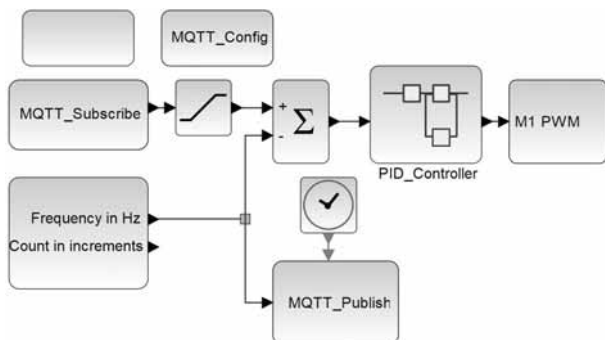


Abbildung 7: Xcos-Modell der Drehzahlregelung.

5.2 Versuchsdurchführung und Auswertung

Zur Verifikation der Funktion wird eine Vielzahl von Versuchen durchgeführt. Hier werden exemplarisch die Messergebnisse eines Sollwert-Sprungs von $0 \frac{1}{s}$ auf $2 \frac{1}{s}$ und nach 3s zurück auf $0 \frac{1}{s}$ untersucht. Die Soll-Drehzahl wird über MQTT an den Mikrocontroller gesendet und dort verarbeitet. Die Ist-Drehzahl wird per MQTT wieder zurück an den Broker gesendet. Parallel erfolgt die Aufzeichnung der Messdaten über LoRa-iGES.

Abbildung 8 illustriert sowohl die über iGES als auch über den MQTT-Broker aufgenommenen Messergebnisse der Drehzahlregelung. Die über MQTT empfangenen Messwerte stimmen gut mit den durch iGES aufgezeichneten Werten überein. Eine geringe zeitliche Verzögerung von $0,04\text{s}$ kann anhand der durchgeführten Sprungantworten abgelesen werden. Bei ca. 4s verbleibt die Ist-Drehzahl auf $0,25 \frac{1}{s}$. Dies ist auf die Messung durch den Inkremental Encoder zurückzuführen, da dieser für sehr kleine Drehzahlen bzw. den Stillstand keine Signale generiert.

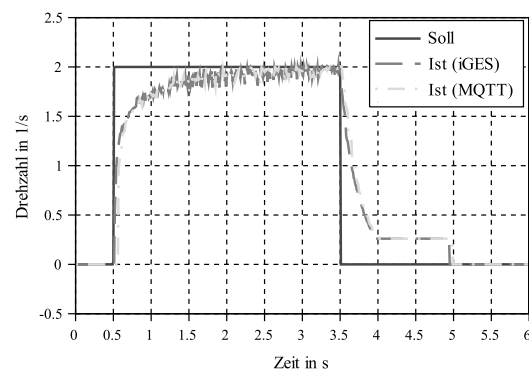


Abbildung 8: Sprungantwort der Drehzahlregelung.

Die durchgeführten Versuche bestätigen die Funktionsfähigkeit der IoT-Erweiterung. Durch das Versenden und Empfangen von Signalen über MQTT lassen sich mit LoRa entwickelte Anwendungen somit in IoT-Netzwerke einbinden. Eine detaillierte Untersuchung zu Latenzen und Netzwerkbelastung steht noch aus.

6 Zusammenfassung und Ausblick

Diese Veröffentlichung befasst sich mit einer Erweiterung der kostengünstigen RCP-Entwicklungsplattform LoRa um eine Schnittstelle zum IoT. Durch die neue RTI-Schnittstellenblöcke können Funktionen und Geräte der Perzeptionsschicht entwickelt werden. Anhand einer Literaturrecherche wurden Anforderungen an die Erweiterung formuliert sowie ein Konzept erarbeitet. Als Kommunikationsprotokoll wird das TCP/IP-basierte MQTT genutzt. Hieraus ergeben sich drei notwendige Schnittstellenblöcke: Konfiguration, Subscribe und Publish. Die RTI-Basissoftware wurde um ein

MQTT-Modul erweitert, welches das generierte Benutzerprogramm mit dem MQTT-Clienten des LwIP TCP/IP Stacks verknüpft. Anhand einer Beispielanwendung wurde die Erweiterung erfolgreich unter Echtzeitbedingungen getestet.

Weitere Arbeiten befassen sich mit der Leistungsfähigkeit der Erweiterung. Hierzu soll zum einen untersucht werden, welchen Einfluss die Kommunikation über MQTT auf die Systemdynamik (z.B. Totzeit aufgrund von Latenzen) hat und zum anderen, welche Netzwerkauslastung durch die Erweiterung hervorgerufen werden. Neben diesen Untersuchungen wird die LoRra-Plattform im cyber-physischen Industrie-4.0 Labortestfeld eingesetzt, in dem Sollwerte per MQTT kommuniziert werden.

Danksagung

Gefördert vom Niedersächsischen Ministerium für Wissenschaft und Kultur unter Fördernummer ZN3495 im Niedersächsischen Vorab der VolkswagenStiftung und betreut vom Zentrum für digitale Innovationen (ZDIN).



Literatur

- [1] van Kranenburg R, Bassi A. IoT Challenges. *Communications in Mobile Computing*. 2012;1(9):1–5.
- [2] Vermesan O, Bacquet J. *Cognitive Hyperconnected Digital Transformation: Internet of Things Intelligence Evolution*. River Publishers Series in Communications. Aalborg: River Publishers. 2017.
- [3] Morelli B. IoT Market Overview. 2018. IHS Markt Customer Care.
- [4] Quantmeyer F, Liu-Henke X. Hardware in the Loop Test Rig for Development of Control Algorithms for Electric Vehicles. *Solid State Phenomena*. 2013; 198:507–512.
- [5] Liu-Henke X, Duym S. Modellgestützte Funktionsabsicherung des vernetzten mechatronischen Kraftfahrzeugs. In: *Mechatronik 2005*, VDI-Berichte. Wiesloch: VDI-Verl. 2005 Jun; pp. 1073–1090.
- [6] Liu-Henke X, Feind R, Roch M, Quantmeyer F. Investigation of low-cost open-source platforms for developing of mechatronic functions with rapid control prototyping. In: *10th International Conference on Mechatronic Systems and Materials*. Opole, Polen. 2014 Jul; pp. 1–9.
- [7] Jacobitz S, Liu-Henke X. The Seamless Low-cost Development Platform LoRra for Model based Systems Engineering. In: *Proceedings of the 8th International Conference on Model-Driven Engineering and Software Development*. Valletta, Malta: SCITEPRESS - Science and Technology Publications. 2020 Feb; pp. 57–64.
- [8] Hanselmann H. Vom Modell zum Seriencode. *Elektronik automotive*. 2003;3:1–5.
- [9] Jacobitz S, Liu-Henke X. LoRra – Eine Low-Cost RCP-Entwicklungsplattform. In: *Tagungsband Embedded Software Engineering Kongress 2019*. Sindelfingen: Vogel and MicroConsult. 2019 Dez; pp. 63–70.
- [10] Samie F, Bauer L, Henkel J. IoT technologies for embedded computing. In: *2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*. Pittsburgh, USA. 2016 Oct; pp. 1–10.
- [11] Gluhak A, Krco S, Nati M, Pfisterer D, Mitton N, Razafindralambo T. A survey on facilities for experimental internet of things research. *IEEE Communications Magazine*. 2011;49(11):58–67.
- [12] Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things (IoT): A vision, architectural elements, and future directions: Future Generation Computer Systems, 29(7), 1645–1660. *Future Generation Computer Systems*. 2013;29(7):1645–1660.
- [13] Washizaki H, Yoshioka N, Hazeyama A, Kato T, Kaiya H, Ogata S, Okubo T, Fernandez EB. Landscape of IoT Patterns. In: *2019 IEEE/ACM 1st International Workshop on Software Engineering Research & Practices for the Internet of Things*. Montreal, QC, Canada. 2019 May; pp. 57–60.
- [14] Guth J, Breitenbucher U, Falkenthal M, Leymann F, Reinfurt L. Comparison of IoT platform architectures: A field study based on a reference architecture. In: *2016 Cloudification of the Internet of Things (CIoT)*. Paris, France. 2016 Nov; pp. 1–6.
- [15] Sikder AK, Oetracca G, Aksu H, Jaeger T, Uluagac S. A Survey on Sensor-based Threats to Internet-of-Things (IoT) Devices and Applications. *Cryptography and Security (csCR)*. 2018;.
- [16] Routh K, Pal T. A survey on technological, business and societal aspects of Internet of Things by Q3, 2017. In:

- 2018 3rd International Conference on Internet of Things: Smart Innovation and Usages. Bhimtal, India. 2018 Feb; pp. 1–4.
- [17] Farris I, Taleb T, Khettab Y, Song J. A Survey on Emerging SDN and NFV Security Mechanisms for IoT Systems. *IEEE Communications Surveys & Tutorials*. 2019;21(1):812–837.
 - [18] Jaikar SP, Iyer KR. A Survey of Messaging Protocols for IoT Systems. *International Journal of Advanced in Management, Technology and Engineering Sciences (ijamtes)*. 2018;8(2):510–514.
 - [19] Gündoğran C, Kietzmann P, Lenders M, Petersen H, Schmidt TC, Wählich M. NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT. In: *Proceedings of the 5th ACM Conference on Information-Centric Networking*. Boston, USA: ACM. 2018 Sep; pp. 159–171.
 - [20] Larmo A, Ratilainen A, Saarinen J. Impact of CoAP and MQTT on NB-IoT System Performance. *Sensors*. 2018;19(7).
 - [21] Shelby Z, Hartke K, Bormann C. The Constrained Application Protocol (CoAP). Online. 2014. Doi: 10.17487/RFC7252.
 - [22] Ruta M, Scioscia F, Pinto A, Gramegna F, Ieva S, Loseto G, Di Sciascio E. CoAP-based collaborative sensor networks in the Semantic Web of Things. *Journal of Ambient Intelligence and Humanized Computing*. 2019;10(7):2545–2562.
 - [23] Bormann C, Castellani AP, Shelby Z. CoAP: An Application Protocol for Billions of Tiny Internet Nodes. *IEEE Internet Computing*. 2012;16(2):62–67.
 - [24] Kodali RK, Soratkal S. MQTT based home automation system using ESP8266. In: *IEEE Region 10 Humanitarian Technology Conference 2016*. Agra, India. 2016 Dez; pp. 1–5.
 - [25] Coppen R, Banks A, Briggs E, Borgendale K, Gupta R. MQTT Version 5.0: OASIS Standard. Online. 2019. URL <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf>
 - [26] Moraes T, Nogueira B, Lira V, Tavares E. Performance Comparison of IoT Communication Protocols. In: *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Bari, Italy. 2019 Oct; pp. 3249–3254.
 - [27] Dunkels A. Design and implementation of the lwIP TCP/IP stack. 2001. Swedish Institute of Computer Science.
 - [28] A Light R. Mosquitto: server and client implementation of the MQTT protocol. *The Journal of Open Source Software (JOSS)*. 2017;2(13):265.

Echtzeitfähige Motorprozessmodelle für Schiffsmaschinen-Simulatoren

Georg Finger^{1*}, Karsten Wehner¹, Egon Hassel², Steffen Loest¹,
Michael Baldauf¹

¹Department of Maritime Studies ISSIMS Institute, University of Applied Sciences Wismar, Richard-Wagner-Str. 31, 18119 Rostock-Warnemünde, Germany; *georg.finger@hs-wismar.de

² Chair of Technical Thermodynamics, Rostock University, Albert-Einstein-Straße 2, 18059 Rostock, Germany

Abstract. Ein ökologisch sinnvoller und ökonomisch effizienter Schiffsbetrieb ist eine zwingende Grundanforderung der (nationalen und internationalen) Schifffahrt und kann durch vielfältige Maßnahmen erreicht werden. Viele dieser Maßnahmen sind technologische Ansätze, wie z.B. die Veränderung von Motorparametern oder Steuerungssystemen, Nutzung von alternativen Kraftstoffen, Anwendung von Abgasreinigungssystemen oder innermotorische Maßnahmen um Motorprozesse effizienter zu gestalten. Bei der Umsetzung dieser Maßnahmen wird oft außer Acht gelassen, dass immer noch Menschen an Bord sind, die für Betrieb und Wartung des Schiffes verantwortlich sind. Dieses Personal muss seine Aufgaben im Spannungsfeld von Umweltschutz und Effizienz unter allen auftretenden äußeren Bedingungen erfüllen. Die Ausbildung von technischen und nautischen Offizieren an Bord von seegehenden Schiffen beinhaltet neben der theoretischen Ausbildung an Land und der praktischen Seefahrtzeit auch Trainingseinheiten in Full-Mission Simulatoren. Diese Simulatoren, müssen auf die sich ändernden Ausbildungsanforderungen durch gezielte Verbesserung der integrierten Prozessmodelle weiterentwickelt werden.

In diesem Beitrag wird ein Ansatz zur echtzeitfähigen Integration von „Zwei-Zonen-Modellen“ zur Bestimmung von Stickoxiden im Abgas vorgestellt. Durch die zusätzliche Integration von Rußmodellen wird darüber hinaus auch die Darstellung von Rußemissionen z.B. im Hafen ermöglicht. Es wird demonstriert, wie durch den Einsatz von Assistenzsystemen der Verbrauch von Kraftstoff reduziert und eine Emissionsminderung im Hafen realisiert werden kann. Die Validierung der Ergebnisse erfolgt durch vergleichende Betrachtungen von realen Messdaten eines Prüfstandsmotor MAN 6L/2330 mit den neu entwickelten und testweise integrierten Simulationsmodulen im Schiffsmaschinensimulator. Mit den FuE-Arbeiten wird ein Beitrag zum besseren Systemverständnis des nautischen und technischen Schiffsführungs-personals erbracht und die aktive Beeinflussung von Emissionen durch ökologisch sinnvolle und ökonomisch effiziente Steuereingriffe gefördert.

Einleitung

Technische und nautische Schiffsoffiziere unterliegen in

ihren Ausbildungsanforderungen unterschiedlichen Standards. Die Mindeststandards werden in der International Convention on Standards of Training, Certification and Watchkeeping for Seafarers (nachfolgend STCW-Abkommen) geregelt. Dieses Abkommen legt fest, welchen Kenntnisstand Mannschaftsdienstgrade und Offiziere in Ihren jeweiligen Positionen vorweisen müssen. Neben rein theoretischem Wissen wird dabei auch praktische Erfahrung bei Anwendung bestimmter Verfahren und Methoden eingefordert. Eine Ausbildung findet dazu im Regelfall nicht nur an Bord von Schiffen statt, sondern in speziellen Simulatoren wie z.B. den am Bereich Seefahrt, Anlagentechnik und Logistik installierten Simulatoren für Schiffsführung (Ship Handling Simulator – SHS) und Schiffsmaschinenbetrieb (Ship Engine Simulator – SES). Beide Simulatoren sind vom DNV-GL als sogenannte Full-Mission Simulatoren zertifiziert und stellen bereits in Teilen eine detaillierte realistische Schiffs Umgebung nach. So wird z.B. im SHS mit Hilfe von realen Konsolen in Verbindung mit einer 360° - Sichtumgebung ein komplexes Seegebiet dargestellt und Trainees mit dem Verhalten eines Schiffes bei verschiedenen Umweltbedingungen vertraut gemacht [16]. Dabei wurde bisher der Fokus auf native Bewegungsmodelle gelegt. Eine Berücksichtigung des realen Verhaltens von Großmotoren findet in der nautischen Ausbildung bisher nur sehr oberflächlich statt. Bessere Möglichkeiten bietet ein Schiffsmaschinensimulator der alle an Bord von Schiffen vorhandenen Versorgungs- und Antriebssysteme nachbilden kann. Allerdings liegt der Schwerpunkt auf dem reinen Betrieb von Großmotoren. Die dabei entstehenden Emissionen ließen sich bisher nicht zufriedenstellend darstellen.

Dieser Artikel befasst sich mit der Entwicklung von Prozessmodellen zur Emissionsbildung und von Kraftstoffverbräuchen sowie deren Anwendung in Simulatoren und Assistenzsystemen.

1 Einfluss von Assistenzsystemen auf den Kraftstoffverbrauch

Im Rahmen des vom Bundesministerium für Wirtschaft und Energie geförderten Verbundprojektes MEmBRan zur Modellierung von Emissionen und Brennstoffverbräuchen wurden experimentelle und Feldstudien durchgeführt, welche eine Betrachtung des reinen Energiebedarfs am Propeller beim Manövrieren mit unterschiedlichen Assistenzsystemen verfolgte. Zunächst wurden dazu Hafenanläufe von realen Schiffen analysiert um eine Basis für den unterschiedlichen Umgang von nautischen Offizieren mit Ihren Schiffen darzustellen. Figure 1 zeigt dabei die Bandbreite der sich aus unterschiedlichen Manöverstrategien ergebenden Bahnverläufe beim Anlegen. Dabei stellt die rote Linie die Wegpunktliste der von der Schiffsführung geplanten Route dar, während die schwarzen Linien die tatsächlich zurückgelegten Strecken sind.

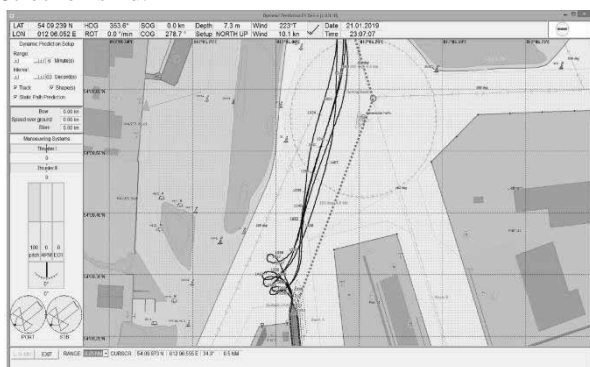


Figure 1: Hafenanlaufstrategien verschiedener Offiziere

Hierbei variierte der Leistungsbedarf an den Propellern im Manöverbetrieb um bis zu 20%[1]. Da in der realen Umgebung verschiedene Umwelteinflüsse eine Rolle spielen und eine Vergleichbarkeit dabei nur schwer erzielt werden kann, wurden weitere Versuche im Schiffsführungssimulator unter kontrollierbaren Bedingungen durchgeführt. Zur Vergleichbarkeit wurden drei verschiedene Szenarien entwickelt und in einem Full-Mission-Simulator, in denen die Probanden, erfahrene Nautiker und Kapitäne, mit dem Schiff anlegen sollten, eine Durchfahrt durch eine fiktive Insellandschaft vollführen oder bis zu einem Ankerplatz manövrieren mussten. Dabei hat jede Testperson jedes Szenario nur einmal durchgeführt um etwaige Lernprozesse auszuschließen. Ein Szenario wurde dabei „klassisch“ – das heißt ohne die Zuhilfenahme von Assistenzsystemen durchgeführt.

Ein Szenario erfolgte unter Zuhilfenahme eines Manöverplanungswerkzeuges. Bei diesem Werkzeug wird im Vorfeld der Übung festgelegt an welchem geographischen Punkt, welcher Steuereingriff (Manöver) erfolgt. Im Gegensatz zu einer reinen Wegpunktliste, bei denen jeweils nur die anzufahrenden geographischen Positionen geplant sind, enthält der Manöverplan, daher exakte Anweisung zum Einsatz der einzelnen Kontrollorgane, wie beispielsweise Ruderlage oder zu ordernde Propellerdrehzahl. Für dieses, am Institut für Innovative Schiffssimulation des Bereiches Seefahrt, Anlagentechnik und Logistik entwickelte, Werkzeug ist daher eine exakte Modellierung des Bewegungsverhaltens erforderlich. Figure 2 zeigt einen Auszug aus einem Manöverplan mit verschiedenen Manöverpunkten. Somit wird aus einem rein mentalen Modell des Nautikers ein vorhersagbares und reproduzierbares Vorgehen geschaffen.

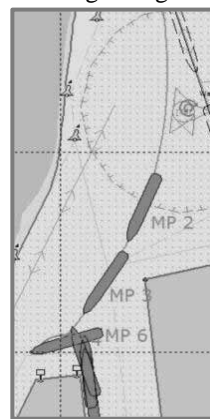


Figure 2: Auszug aus Manöverplan

Als weiteres Assistenzsystem wurde den Probanden eine Fast Time Simulation zur Verfügung gestellt, die den Probanden die Schiffsbewegung für einen Vorab definierten Zeitraum während der Übung grafisch darstellt. Somit hatten die Testpersonen während der Durchführung die Möglichkeit, ihre Handlungen zu validieren. Figure 3 stellt die durchschnittliche Leistungsaufnahme des Propellers dar. In den Versuchen zeigte sich, dass bei den mittels Assistenzsystemen durchgeführten Anlege-manövern die benötigte Propellerleistung signifikant geringer war. Mit Steigerung des Assistenzgrades sinkt der Leistungsbedarf um knapp 30%. Dieser Effekt wird besonders in küstennahen Gebieten und beim Manövrieren im Hafen deutlich. Bei langen Überfahrten verschwindet der Effekt. Bei einem Schiff, das sehr viele Manöver in kurzer Zeit ausführen muss, wie beispielsweise Fähren im Kurzstreckenverkehr, ist der Effekt sehr deutlich messbar.

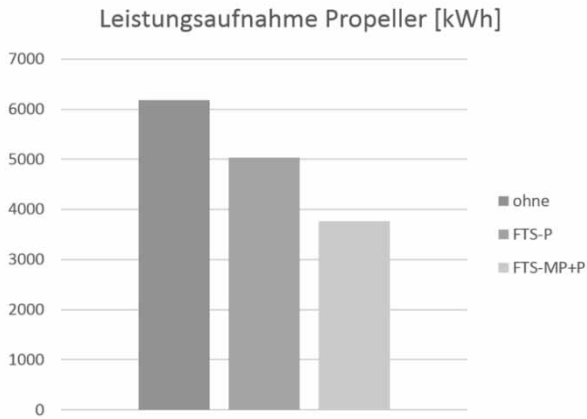


Figure 3: Durchschnittsergebnis aus einem Szenario

Zurückzuführen lässt sich diese Ersparnis auf zwei Parameter. Zunächst werden die Anzahl und Intensität der Steuereingriffe (Manöver) minimiert. Beispielsweise kann eine übermäßige Beschleunigung, die im weiteren Verlauf wieder abgefangen werden muss, vorausschauend vermieden oder minimiert werden. Des Weiteren sinkt der Einfluss von Ruderkommandos. Jedes Ruderkommando hat einen Einfluss auf die Anströmung von Propeller und Schiffskörper und hat somit eine Widerstandsänderung zur Folge. Übermäßige Quergeschwindigkeiten, führen somit immer zu einem zusätzlichen Leistungsbedarf des Propellers. Figure 4 zeigt die Gegenüberstellung von Ruderwinkeln eines Szenarios ohne Assistenzsystem (oben) und mit Prädiktion und Planung (unten). Man kann erkennen, dass die Anzahl an Korrekturen stark abnimmt, somit große Ruderlagen größtenteils bis zum Erreichen des nächsten Manöverpunktes konstant bleiben.

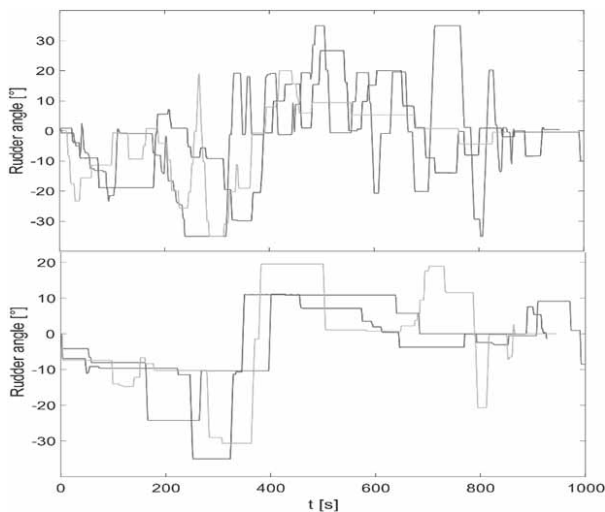


Figure 4: Ruderwinkel aus einem Szenario

Unzureichend an dieser Darstellung ist bisher, dass lediglich die am Propeller abgeforderte Leistung auf Basis des Bewegungsmodells quantifiziert wird. Technisch findet sowohl in der Prädiktion, als auch in der Simulation des Schiffsführungssimulators keine reale Simulation des Motorprozesses statt. Das führt dazu, dass somit keine Aussagen über den realen Kraftstoffverbrauch und Emissionen wie beispielsweise Stickoxide oder Ruß stattfinden kann, da der Betriebspunkt des Motors nicht bekannt ist. Somit wurde eine Kopplung des Schiffsführungssimulators mit dem Schiffsmaschinensimulator inklusive Integration von Brennraum- und Emissionsmodellen angestrebt.

2 Motormodell

Das Motormodell besteht aus verschiedenen Komponenten die nachfolgend näher betrachtet werden. Die Komponenten lassen sich in Regler, Turboladermodell und Zylinderinnenprozess aufteilen. Im Innenprozess findet dabei der Energieumsatz statt auf dessen Basis die Emissionsmodelle aufbauen.

2.1 Grundlagen des Zylinderinnenprozesses

Der Zylinderinnenprozess basiert auf dem ersten Hauptsatz der Thermodynamik für ein offenes System (Fehler! Verweisquelle konnte nicht gefunden werden.).

$$\frac{dU}{dt} = -p \frac{dV}{dt} + \dot{Q}_B + \dot{Q}_W + \dot{H}_{BB} + \dot{H}_{in} - \dot{H}_{out} \quad (1)$$

Darin beschreibt der Term $-p \frac{dV}{dt}$ die Volumenänderungsarbeit des Gases die letztlich an den Kolben abgegeben wird und letztendlich über die Welle die Arbeit unter Abzug von Reibungsverlusten am Propeller verrichtet. \dot{Q}_B ist die im Brennstoff chemisch gebundene Energie die nach Einspritzung und Verdampfung umgesetzt wird. Der Wandwärmestrom \dot{Q}_W beinhaltet die Wärme die vom Brennraum an die Wände von Laufbuchse, Kolben und Zylinderdeckel abgegeben wird und über das Kühlwasser des Motors abgeführt wird. Die Enthalpieströme \dot{H}_{in} & \dot{H}_{out} enthält die Energie die mit dem Gas über Ein- und Auslassventile aus dem Brennraum transportiert wird. Der Enthalpiestrom \dot{H}_{BB} berücksichtigt dabei die

Verluste über Undichtigkeiten der Reibpaarung von Kolbenring und Laufbuchse. Die Berechnung der spezifischen inneren Energie u erfolgt dabei nach dem Ansatz von Justi 2

$$\begin{aligned}
 u(T, \lambda) = & 0,1445 \left[1356, + \left(489,6 + \frac{46,4}{\lambda^{0,93}} \right) * \right. \\
 & * (T - T_{Bez}) 10^{-2} + \\
 & + \left(7,768 + \frac{3,36}{\lambda^{0,8}} \right) (T - T_{Bez})^2 10^{-4} - \\
 & \left. - \left(0,0975 + \frac{0,0485}{\lambda^{0,75}} \right) (T - T_{Bez})^3 10^{-6} \right]
 \end{aligned} \quad (2)$$

Basierend auf den partiellen Ableitungen aus (1) und (2) unter Berücksichtigung des Verbrennungsluftverhältnisses λ lässt sich zu jedem Zeitpunkt bzw. zu jedem Grad Kurbelwinkel φ die thermodynamische Durchschnittstemperatur T bestimmen. Aus der Temperatur lässt sich über die allgemeine Gasgleichung der Druck im Brennraum bestimmen und über die Brennraumgeometrie die Kraft auf den Kolben, welche letztendlich in das Drehmoment der Welle übergeht. Da zur Bestimmung des Energieumsatzes das Verbrennungsluftverhältnis benötigt wird, ist die genaue Kenntnis des Ladeluftdrucks nach Turboladerkompressor und Ladeluftkühler sowie des Abgasgegendruckes vor der Turboladerturbine unerlässlich. Sind die Drücke vor und nach Zylinder bekannt, so lässt sich der Massenstrom über Ein- und Auslassventil durch die dynamischen geometrischen Größen durch die Modellierung einer Drosselstelle (3) wiedergeben.

$$\dot{m} = A_1 \sqrt{\rho_1 * p_1} * \sqrt{\frac{2\kappa}{\kappa - 1} \left[\left(\frac{p_2}{p_1} \right)^{\frac{2}{\kappa}} - \left(\frac{p_2}{p_1} \right)^{\frac{\kappa+1}{\kappa}} \right]} \quad (3)$$

Über den Massenstrom und die innere Energie lassen sich somit die Enthalpieströme darstellen. Unter Berücksichtigung der Kraftstoffzuführung aus Kapitel 2.2 lässt sich somit der Brennraumdruck über ein Arbeitsspiel numerisch berechnen. In Figure 5 sind die Berechnungsergebnisse dargestellt die aus diesen Ansätzen statisch hervorgehen. Die schwarze Linie ist dabei eine Kurve die zwischen 2.5 und 3 bar verläuft und den Ladungswechsel darstellt.

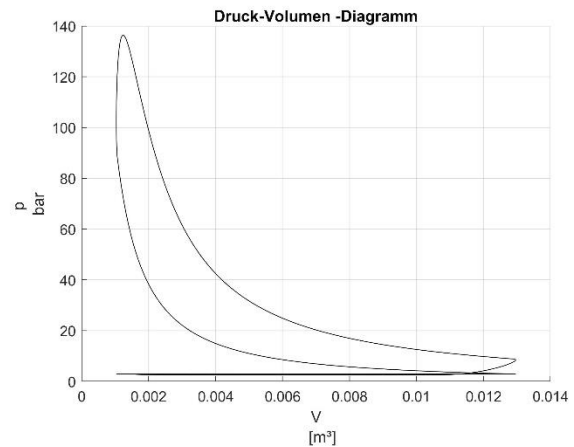


Figure 5: Druck-Volumen-Diagramm

2.2 Kraftstoffzufuhr und Umsetzung

Die Kraftstoffzufuhr zum Motor eines Schiffes wird äußerlich über zwei Komponenten geregelt. Im Vordergrund steht zunächst die Drehzahlanforderung der nautischen Schiffsführung über den Maschinentelegraphen. Dieser Sollwert wird in der Regel über Hochfahrprogramme gefiltert um eine Überlast des Motors zu minimieren. Für jedes Schiff werden diese Programme neu abgestimmt, aufgrund der Tatsache, dass sich ein Schiffeigner für eine beliebige - technisch mögliche - Kombination von Rumpf, Propeller und Maschine entscheiden kann. Der Regler am Motor ist ein konventioneller PID-Regler, der die benötigte Kraftstoffmenge auf den derzeitigen Sollwert einregelt. Programntechnisch lassen sich beide Varianten zusammenfassen. Ein Motor mit klassischer Einspritzung über Pumpe-Düse Verfahren ist dabei im Kraftstoffmassenstrom über die Drehzahl begrenzt. Figure 6 zeigt exemplarisch die Beschleunigung des Versuchsmotors auf Nennleistung.

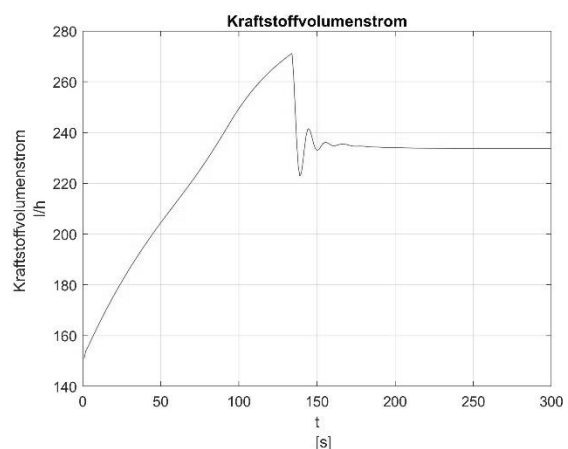


Figure 6: Kraftstoffvolumenstrom bei der Beschleunigung

Leicht ersichtlich ist in diesem Beispiel, dass der Kraftstoffverbrauch nach dem Überspringen noch absinkt. Dies ist auf den verbesserten Umsatz in Folge höherer Ladeluftdrucke zurückzuführen. Das hier dargestellte Simulationsergebnis basiert auf einer „stand-alone“ Variante die etwaige Beschleunigungsmomente aus den Bewegungsgleichungen aus Abschnitt 1 noch nicht ausreichend berücksichtigt. Da eine reine Simulation der Kraftstoffzufuhr über das Füllungsgestänge den Eintrag und die Umsetzung im Zylinderinnenprozess nicht widerspiegelt, wurde der Ansatz nach Vibe [3] verwendet, um den Kraftstoffmassenstrom in den Energieumsatz \dot{Q}_B aus (1) zu übertragen. Dabei sind Anpassungen nach Sitkei[4] und Woschni et al[5] dynamisch berücksichtigt, sodass sich der Brennverlauf $\dot{Q}_B(\varphi)$ dynamisch an äußere Umstände anpassen lässt.

2.3 Turboladermodell

Um die Drücke vor und nach Zylinder zu berechnen sind exakte Kenntnisse über das Betriebsverhalten des Turboladers und der jeweiligen Sammelrohre notwendig. Aufgrund der Tatsache, dass der Turbolader seine Antriebsenergie zur Verdichtung von Ladeluft aus der Abgasenergie nach dem Zylinder bezieht, ist es ein rückgekoppeltes System. Das System kann in der numerischen Berechnung, insbesondere bei sehr großen Schrittweiten, schnell instabil werden, da eine stationäre Simulation immer auf dem Gleichgewicht von Antriebsmoment der Turbine M_T und Arbeitsmoment des Kompressors M_C . Dies hat in der Simulation eine ungewollte Wechselwirkung zwischen Ladeluftdruck und Drehzahl zur Folge. Die Berechnung beruht dabei auf den technisch validierten Modellen der Turboladerhauptgleichung [6]. Dieses Modell ist skalierbar und ebenfalls auf unterschiedliche Turbolader/Motorkonstellationen anpassbar. Die Ergebnisse der Parametrierung des Modells an den Prüfstandsmotor des Bereichs Seefahrt, Anlagentechnik und Logistik MAN 6L23/30 sind in Figure 7 ersichtlich. Hier ist ein dynamischer Lastsprung zwischen Messung („Eng“ – in Abhängigkeit konventioneller Schiffsautomatisierungstechnik in 0.1 bar Schritten) und Simulation („Sim“) dargestellt. Der zeitliche Verlauf hat einen hohen Einfluss auf die Emissionen, da die Luftmasse im Zylinder den Verbrennungsvorgang maßgeblich bestimmt.

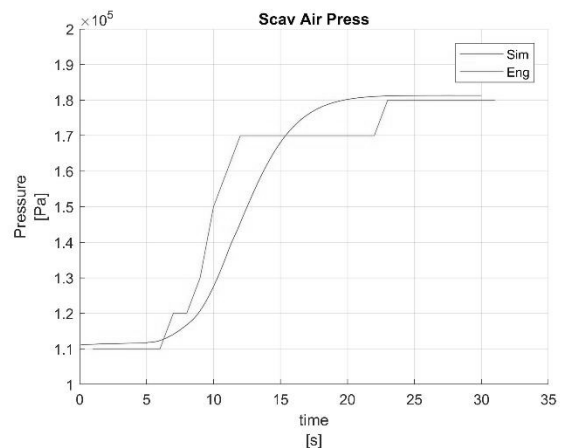


Figure 7: Ladeluftdruck Simulation und Messung

2.4 Emissionsmodell

Die Berechnung der Emissionen sind abhängig von den vorausgegangen Berechnungen und basieren jeweils auf Druck- und Temperaturwerten im Brennraum. Daher sind die zuvor beschriebenen Berechnungen unabdingbare Basis für die Modellierung von Emissionen.

Stickoxid-Modell

Die Modellierung von Stickoxiden basiert auf den in Kapitel 2.1 beschriebenen Berechnungen. Zur Berechnung der Volumenänderungsarbeit ist hier die thermodynamische Durchschnittstemperatur ausreichend. Thermische Stickoxide entstehen aus dem in der Verbrennungsluft enthaltenen Stickstoff. Hohe Temperaturen in der Flammenfront sind daher erforderlich, um die Dreifachbindung des Luftstickstoffs aufzubrechen. Daher wird nach Heider[7] die thermodynamische Durchschnittstemperatur in zwei Zonen aufgeteilt. Zone eins beinhaltet das unverbrannte Material und Zone zwei ist die bereits verbrannte Zone. Die Aufteilung erfolgt dabei nach der Funktion $B(\varphi)$ (4) unter Berücksichtigung des motorspezifischen Faktors A^*

$$T_1(\varphi) - T_2(\varphi) = B(\varphi)A^* \quad (4)$$

Die Berechnung erfolgt dabei basierend auf dem Schleppdruck p_0 eines Motors ohne Verbrennung und dem Integral des Druckverlaufs $\int_{\varphi_{BC}}^{\varphi_{ExVo}} p d\varphi$ von Brennbeginn (BC) und Öffnung des Auslassventils (ExVo) mit der Funktion (5)

$$B(\varphi) = 1 - \frac{\int_{\varphi_{BC}}^{\varphi} [p(\varphi) - p_0(\varphi)] m_1 d\varphi}{\int_{\varphi_{BC}}^{\varphi_{ExVo}} [p(\varphi) - p_0(\varphi)] m_1 d\varphi} \quad (5)$$

Durch die Temperatur in der Flammenfront lässt sich mit Hilfe des Zeldovich-Mechanismus[8](6) die Konzentrationsänderung von Stickstoffmonoxid bestimmen.

$$\frac{d[NO]}{dt} = k_{1,r}[O][N_2] + k_{2,r}[N][O_2] + k_{3,r}[N][OH] - k_{1,l}[NO][N] - k_{2,l}[NO][O] - k_{3,l}[NO][H] \quad (6)$$

Der Geschwindigkeitsbestimmende Faktor ist in Folge der hohen Aktivierungsenergie $k_{1,r}$ und kann der Literatur entnommen werden [9][10][11][12]. Eine Bestimmung der exakten Temperaturen nach diesen Faktoren ist dabei unerlässlich. Eine Abweichung um vier Kelvin im Bereich von 1600K führt bereits zu einer um bis zu 10% fehlerhaften Berechnung der Stickstoffmonoxidbildung.

Ruß-Modell

Die Modellierung von Ruß in echtzeitfähigen Verfahren ist derzeit nur beschränkt möglich. Genaue Verfahren basieren auf CFD-Rechnungen und sind weit verbreitet, aber auch rechenintensiv. Als weit verbreitetes Modell dient das Modell von Nishida und Hiroyasu [13]. In diesem Modell werden Rußbildung (7) und Oxidation (8) durch empirische Gleichung beschrieben und die Bestimmung der Gesamtrußentstehung erfolgt durch Differenzbildung der beiden Größen.

$$\frac{dm_{P,B}}{dt} = A_B * m_{B,g} p^{0,5} e^{-\frac{6313}{T}} \quad (7)$$

$$\frac{dm_{P,Ox}}{dt} = A_{Ox} * m_p x_{O_2} * p^{1,8} e^{-\frac{7070}{T}} \quad (8)$$

Dieser Ansatz basiert auf der gasförmigen Brennstoffmasse $m_{B,g}$ und dem Sauerstoffstoffmengenanteil x_{O_2} sowie den Druck- und Temperaturwerten im Zylinder. Die Berechnung erfolgt als Paketmodell im Einspritzstrahl. Ebenfalls erfolgt eine Anpassung durch die motorspezifischen Faktoren A_B und A_{Ox} . Dieses Modell wurde nun adaptiert und in einem Einzonenmodell getestet und in den SES statisch implementiert. Das Rechenresultat der Partikelbeladung in Milligramm pro Kubikmeter ist für die Trainees allerdings sehr unhandlich. Daher erfolgt eine iterative Umrechnung in die Filterschwärzungszahl nach AVL [14] (9).

$$PB = \frac{1}{0.405} * 4,95 * FSN e^{0,38FSN} \quad (9)$$

Diese kann wiederum in eine Opazität umgerechnet werden. Somit lässt sich im Simulator eine schwarze Rußfahne bei ungünstigen Lastzuständen darstellen, um eine visuelle Rückkopplung an den Trainee zu geben. Figure 8 stellt dabei die Rußbildung an verschiedenen stationären Lastpunkten gegenüber.

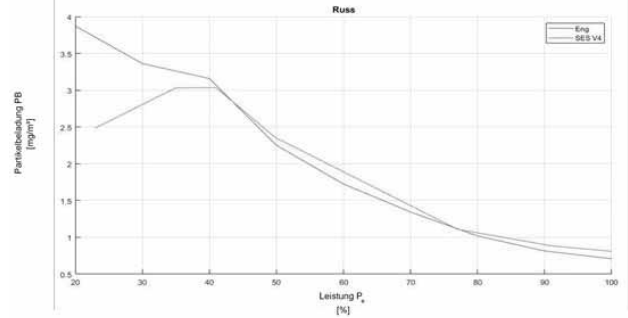


Figure 8: Rußbildung stationärer Lastpunkte

In Figure 9 ist das Ergebnis eines Lastsprunges auf einer Propellerkurve dargestellt. Die Abweichungen zwischen Simulation und Prüfstand sind auf die Tatsache zurückzuführen, dass das Ergebnis der Rußberechnung nach Zylinder erfolgt, während die Messung am Prüfstand nach dem Turbolader erfolgt und dementsprechend Mischungsverhältnisse und Trägheiten nicht vollständig berücksichtigt sind.

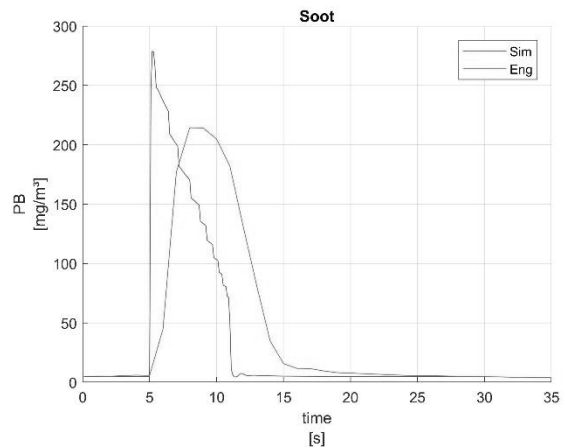


Figure 9: Rußbildung während eines Lastsprunges

2.5 Zusammenfassung und Ausblick

Die in diesem Beitrag dargestellten Methoden zur Modellbildung von Motorprozessen ermöglichen eine

komplexe und umfassendere Schiffssimulation. Die Anwendung auf vorhergehende Untersuchung zeigt, dass der Kraftstoffbedarf quantifizierbar wird und somit auch in die nautische Manöverplanung einfließen kann. Bisher erfolgte die Umsetzung der Motorprozesse in 3 Phasen. Sowohl der C-Code des Schiffsmaschinensimulators, als auch die C++ Umgebung zur Entwicklung des Bewegungsverhaltens des Schiffes sind echtzeitfähig. Problematisch wird die Berechnung bei sehr hohen Drehzahlen über 4000 rpm. Da dort nicht mehr sämtliche Umdrehungen gerechnet werden können. Ebenfalls erweist sich eine Einbindung in die Fast-Time-Simulation zur Prädiktion im operativen Betrieb als schwierig. Da dort Vorausberechnung bis 24min erfolgen sollen, ist der Rechenaufwand durch das Motormodell immens. Abhilfe schafft an dieser Stelle die Einbindung künstlicher neuronaler Netze. Diese können mit den Ergebnissen des Simulationstools angelernet werden und beschleunigen die Berechnung um ein Vielfaches. Erste Vorarbeiten sind von Schaub et al.[15] bereits durchgeführt worden. Des Weiteren müssen weitere Modelle eingepflegt bzw. angepasst werden, um nicht nur der Vielfalt an Motoren gerecht zu werden, sondern auch, um nicht-konventionelle Motoren wie Common-Rail oder Dual-Fuel Motoren zukünftig darzustellen.

Danksagung

Der in diesem Beitrag beschriebene Ansatz wurde im Rahmen des vom Bundesministeriums für Wirtschaft und Energie geförderten FuE-Projektes MEMBRAN entwickelt und getestet. An diesem Projekt waren beteiligt: MARSIG GmbH Rostock, Hochschule Wismar (Bereich Seefahrt, Anlagentechnik und Logistik), Universität Rostock (Lehrstuhl für Technische Thermodynamik), Rheinmetall Electronics Bremen (Abteilung Maritime- und Prozesssimulation) und assoziierte Partner.

Nomenklatur

SHS	- Schiffsführungssimulator
SES	- Schiffsmaschinensimulator
U	- innere Energie [J]
H	- Enthalpie [J]
V	- Volumen [m ³]
p	- Druck [Pa]
\dot{Q}_B	- Energiefreisetzung aus Brennstoff [kJ/s]
\dot{Q}_W	-Wandwärmestrom [kW]
λ	-Verbrennungsluftverhältnis [-]
φ	- Kurbelwinkel [°]
T	- Temperatur im Brennraum [K]

SES	- Schiffsmaschinensimulator
U	- innere Energie [J]
H	- Enthalpie [J]
V	- Volumen [m ³]
κ	- Isentropenexponent
A ₁	- Querschnitt Drosselfläche [m ²]
p ₁	- Druck vor Drosselstelle [Pa]
p ₂	- Druck nach Drosselstelle [Pa]
ρ	- Dichte vor Drosselstelle [kg/m ³]
M _T	- Drehmoment Turbine[Nm]
M _C	- Drehmoment Verdichter[Nm]
k	- Reaktionsgeschwindigkeitskonstante
O	- elementarer Sauerstoff
N	- elementarer Stickstoff
T ₂	- Temperatur unverbrannte Zone[K]
T ₁	- Temperatur verbrannte Zone [K]
m _{B,g}	- gasförmiger Brennstoff
m _p	- Rußpartikelmasse
x _{O₂}	- Sauerstoffmengenanteil
φ	- Kurbelwinkel [°]
φ_{BC}	- Kurbelwinkel Brennbeginn [°]
φ_{ExVo}	- Kurbelwinkel bei Auslass öffnet [°]
FSN	- Filterschwärzungszahl
PB	- Partikelbeladung [mg/m ³]

References

- [1] Finger, G., Schaub, M., Dahms, F., Hassel, E., Riebe, T., Milbradt, G. u. Wehner, K.: On-board Support System for the eco-friendly ship operation in coastal and port areas. In: OCEANS 2019 (Hrsg.): Proceedings OCEANS 2019. Marseille 2019
- [2] Justi, E.: Spezifische Wärme, Enthalpie, Entropie und Dissoziation technischer Gase, Springer Verlag (1938)
- [3] Vibe, I.: Brennverlauf und Kreisprozess von Verbrennungsmotoren, VEB Verlag Technik, Berlin (1970)
- [4] Sitkei, G.: Über den dieselmotorischen Zündverzug. Motortechnische Zeitschrift MTZ 26 (1963)
- [5] Woschni, G., Anisits, F.: Eine Methode zur Vorausberechnung der Änderung des Brennverlaufs mittelschnelllaufender Dieselmotoren bei geänderten Betriebsbedingungen. Motortechnische Zeitschrift MTZ 34, 106 ff., Franckh-Kosmos Verlags- GmbH, Stuttgart (1973)
- [6] Merker, P.: Title Grundlagen Verbrennungsmotoren. p 228
- [7] Heider, G.: Rechenmodell zur Vorausberechnung der NO-Emissionen von Dieselmotoren, Dissertation, TU München (1996)
- [8] Zeldovich, Y.B.: The Oxidation of Nitrogen in Combustion and Explosion. Acta Physicimica, USSR 21, pp. 577-628 (1946)
- [9] GRI-MECH 3.0: www.me.berkeley.edu/gri_mech (2000)
- [10] Baulch, Evaluated kinetic data for combustion modeling. Supplement I, Journal of Physical and Chemical

Reference Data 23, pp. 847 (1994)

- [11] Heywood, J.B.: Internal Combustion Engine Fundamentals. McGraw-Hill (1988)
- [12] Pattas, K.: Stickoxidbildung in der ottomotorischen Verbrennung, MTZ 34, pp. 397-404 (1973)
- [13] Nishida, K., Hiroyasu, H. : Simplified Three-Dimensional Modeling of Mixture Formation and Combustion in a DI Diesel Engine, SAE Paper, 890269 (1989):
- [14] AVL, Smoke value measurement with the filter-paper-method (2005)
- [15] Schaub M, Data-Based Prediction Of Soot Emissions For Transient Engine Operation. IAPGOS 9(4):10-13. doi:10.35784/IAPGOS.29 (2019)
- [16] Baldauf, M, Schröder-Hinrichs, JU, Kataria, A, Benedict K, Tuschling, G (2016) Multidimensional simulation in team training for safety and security in maritime transportation, Journal of Transportation Safety & Security, 8:3, 197-213, DOI: 10.1080/19439962.2014.996932

ROCS: A Realtime Optimization and Control Simulator

Andreas Britzelmeier¹, Matthias Gerdts¹, Omid Moslehi Rad¹, Sonali Rani¹, Thomas Rottmann¹

¹Institute of Applied Mathematics and Scientific Computing, Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany;
{andreas.britzelmeier,matthias.gerdts,omid.moslehi,sonali.rani,thomas.rottmann}@unibw.de

Abstract. The Realtime Optimization and Control Simulator (ROCS) is a software package written with Qt. It is conceived to be a versatile tool to develop, investigate, and visualize control and trajectory optimization tasks for automated vehicles, aircrafts, and robots in multi-modal scenarios. It is also conceived as a platform which allows to combine real driving data with virtual simulation using a vehicle in the loop.

1 Introduction

The task of simulating and modeling physical systems has always been an important instrument of scientific and industrial research and development. It allows to study and to evaluate the behavior of proposed models and to compare the outcome with the performance of the actual system under consideration. Therein, we can distinguish two major types of simulation tasks, that is, with and without real time requirements and visualization. Often computations are undertaken and afterwards visualized through graphs or non-immersive types of representation. However, with advancements in hardware technology and the increase in digitization in many systems, like cars, planes, and mobile robots, the requirement for virtual environmental simulation is drastically increasing. Another aspect which justifies immersive simulation tools comes along with automation of systems and the desire to create a digital twin. Developers are obliged to prove the reliability and safety of newly developed systems, as well as that the expected increase in utility is guaranteed. However, building prototypes often is very expensive as well as intensive testing to generate reproducible results. Therefore, the need of alternative methods to analyze the behavior and interaction of systems is imminent. Prominent examples of powerful simulation tools in the automotive industry are Virtual Testdrive (VTD)

[17, 15], and SILAB, [11]. Both are widely used in the automobile industry, compare [1]. Apart from the automobile industry, also in aerospace engineering and flight training simulative tools have a long history and are widely used. The Flight Simulator from Microsoft, see [12], as well as X-Plane, see [18], both of which are certified by the Federal Aviation Administration, are used in pilot training and research. Hence, immersive simulation tools are not just mere tools of real-time visualization but necessary tools to develop the technology of the future. An indispensable advantage of such tools comes to play whenever new technologies which require human cooperation or interaction is necessary. Then these simulative tools provide a safe environment to test acceptance and reliability, compare [8, 13]. Despite the different types of simulation tools already available, most of them are limited to a specific use case, may it be cars or planes.

In addition, there are very popular game engines, e.g. Unity [16], which are applicable to both, gaming and simulation applications. Such game engines have numerous advantages, e.g. fast and agile development, huge asset stores, optimized graphics, physics and audio engines. However, these platforms come with some specific limitations that can hinder scientific simulation purposes. Few of those complications are licensing and costs for activation of desired features, difficulty in organizing its complex directory hierarchy, non-public source code, making it difficult to track or debug issues, increase in consumption of hardware resources due to complex environment, and finally it is convenient to use only with C Sharp as the primary scripting language. Moreover, downward compatibility issues owing to new versions may arise and can make it difficult to maintain a long term project. Finally, the addition of one's own particular models, controllers, or optimizers can be cumbersome or even impossible.

These potential drawbacks motivated us to build a

research and development tool called Real-time Optimization and Control Simulator (ROCS). The idea was to build a versatile research tool which allows for in time visualization and testing of our online optimization algorithms and feedback controllers for automated agents in multi-model scenarios. A further goal was to include control interfaces to real systems. ROCS is build as a modular tool which allows for simple extension by further optimizers and controllers, and the integration of sensor data. Further it is able to visualize scenarios and conduct experimental validation with a variety of vehicles like cars, industrial or mobile robots, and flying platforms like drones, planes or quadcopters. Owing to the modularity of the tool it is comparatively simple to add models for every required type of vehicle. Different modes of simulation are implemented. One can either provide precomputed data, use feedback controllers in combination with model simulation or employ an optimizer to perform online path planning tasks. ROCS already provides a set of vehicular controllers as well as different models for cars and integrators.

The outline of the paper is as follows. In Section 2 we discuss the overall conceptual design of ROCS. Simulation aspects and two selected vehicle models in ROCS are discussed in Section 3. Section 4 addresses the controller design, while Section 5 presents the 3D simulation environment. Some simulation results are presented in Section 6. Finally, a summary and an outlook with future developments conclude the paper.

2 Design

Realtime Optimization and Control Simulator (ROCS) is designed in a modular way. We decided to implement it in C++ with Qt as it is a programming language widely used in industry and academia and facilitates integration of algorithms and modules. In addition it provides convenient 3D visualization capabilities and the slots and signal mechanism is very well suited for the realtime control purposes.

The main components of ROCS are depicted in Figure 1. The core class objects are a vehicle class, a control class, an input/output class, and a visualization class. The vehicle class contains all vehicle relevant parameters, numerical integrators for motion prediction and simulation, interfaces to controllers and graphical objects describing the shape. The control class contains a collection of tracking controllers and optimization-based path planning tools as detailed in Section 4. The visualization class serves to display the simulation and control outputs in a 3D view or in chart plots. It is also

possible to store the simulation results or measurements in a file. Likewise it is possible to use ROCS in an offline mode in order to visualize external data from a data file. The central control unit is the User Control Interface (UCI) described in Section 2.1. An automatic world generator class is part of the concept, but not fully realized up to now.

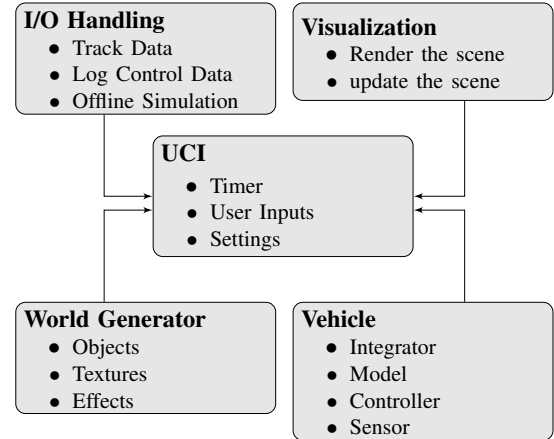


Figure 1: Information flow between simulation models and controllers.

2.1 The User Control Interface (UCI)

The central control panel of ROCS is the User Control Interface (UCI) in Figure 2. This panel allows to load reference paths, environments, and models. Moreover it provides an overview on the number and type of agents within the simulation. The properties of the agents can be edited through additional dialogs, compare Figures 4, 5. The UCI furthermore allows to select a camera perspective, to switch on or off a data logging mode, and it permits to adjust the scene timer for 3D visualization. Finally it offers options for saving and loading in order to conveniently store or re-store complex scenarios and settings.

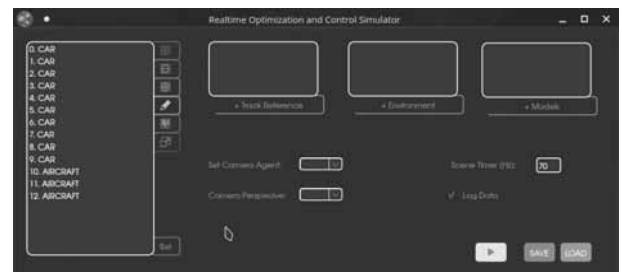


Figure 2: Central user control interface.

2.2 Handling of Multiple Agents

Due to the object oriented programming style the vehicle class can be sub-classed to differentiate among vehicle types. Furthermore, multiple objects of one vehicle can be created inheriting the same properties and functions, controlling their visualization. On top, each object is stored in a list such that each vehicle appearing in the scene can also be customized. Customization includes changing the objects model, linking to different controllers or integrators or changing the pipeline of data acquisition.

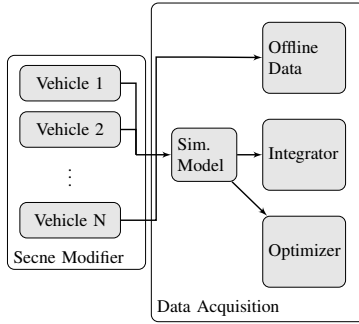


Figure 3: Customizable linking of different data structures for Simulation.

Data can be acquired three ways. The first is to load offline created data from text files which provide data required for simulation. The second method consists in online computation of simulative data through integration and feedback controllers. The computed data is then pipelined by a signal to a slot in the scene modifier class. Thereby, each individual vehicle object and the corresponding controller run in a separate thread and do not interfere with the update of the scene or other operations of the tool. Threads are managed in a synchronous and thread-safe way. The last option includes the data generation by an optimization-based path planner, which repeatedly solves optimal control problems within a model-predictive control loop. The output data can also be directed to the scene modifier by addressing the same slot from the optimizer class. Hence, we have a uniform connection through signal and slots which can be used to adjoin further modules as well.

In summary, ROCS is centered around the feedback control loops for the agents. These control processes run at a specified frequency in their own threads and are decoupled from the visualization, which is able to run at its own frequency and merely accesses simulation data generated by the control loops. Both frequencies can be synchronized in which case visualization and control work in realtime, if the hardware permits it.

3 Simulation of Multi-Agent Systems

ROCS allows to investigate heterogeneous multi-agent systems consisting of, e.g. cars, robots, or aircrafts. These agents or vehicles, respectively, can be derived from a basic vehicle class, which inherits core functionalities for any type of agent. The derived objects allow to set particular features of the individual agents. The individual agents can be added to the simulation through a dialog window, compare Figure 4.



Figure 4: Dialog for adding agents.

The individual properties, models, and parameters of the agents can be adjusted and selected in an editing dialog, compare Figure 5.



Figure 5: Dialog for editing agents.

Once all agents have been configured (including dynamics, initial states, controller types) and added to the scenario, it remains to simulate the whole multi-agent system. To this end let $N \in \mathbb{N}$ agents be given. We assume that each agent can be controlled and for each agent $i \in \{1, \dots, N\}$ we denote the control input at time t by $u_i(t)$ and the state at time t by $x_i(t)$. The motion of the i -th agent is modelled mathematically by an initial value problem of type

$$\dot{x}_i'(t) = f_i(t, x_i(t), u_i(t)), \quad x_i(t_{i,0}) = x_{i,0}, \quad (1)$$

with initial time $t_{i,0}$ for $i = 1, \dots, N$. The agents can be controlled either in open-loop, i.e., by providing the control input $u_i = u_i(t)$ as a given function of time as in (1), or in closed-loop by providing a feedback law $u_i = \mu_i(t, x)$, where $x = (x_1, \dots, x_N)^\top$ is the combined state of all agents. This leads to the closed-loop system

$$\dot{x}_i(t) = f(t, x_i(t), \mu_i(t, x(t))), \quad x_i(t_{i,0}) = x_{i,0}, \quad (2)$$

for $i = 1, \dots, N$. In both cases the overall dynamic system will be solved numerically by a Runge-Kutta method. ROCS uses standard solvers with fixed step-sizes (Euler method, Heun's method, classic 4-th order Runge-Kutta method) and variable step-sizes (DOPRI5(4)), see [10].

The outcome of the simulation can be stored in a data logging file or in a chart window, compare Figure 6.

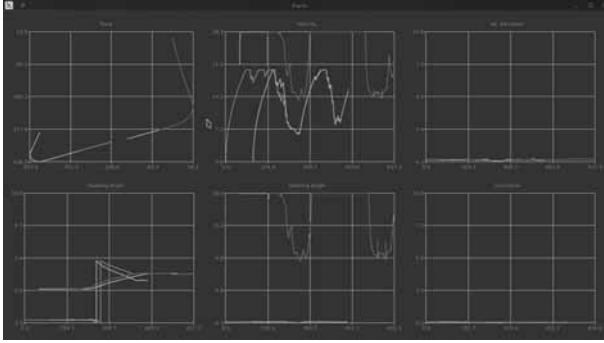


Figure 6: Chart window for detailed view of sensors, states, and controls.

The design of the feedback laws μ_i for the agents $i = 1, \dots, N$, will be outlined in Section 4. Currently, we only have individual controllers implemented, which do not take into account the behavior of the other agents. In the future we will add controllers and optimization strategies for interacting systems as outlined in [4]. This will require to set up an agent-to-agent or agent-to-cloud communication procedure, in which, e.g., position data or driving intentions are exchanged.

3.1 Vehicle Models

At the current state of development, due to the focus on autonomous driving, two vehicle models, a single track model and a kinematic model of a two wheel driven mobile robot have been implemented. The equations of motion of the single track model read as,

$$\dot{x}' = v \cos(\psi - \beta), \quad (3)$$

$$\dot{y}' = v \sin(\psi - \beta), \quad (4)$$

$$\dot{v}' = \frac{1}{m} [(F_{uh} - F_{Lx}) \cos \beta + F_{uv} \cos(\delta + \beta) - (F_{sh} - F_{Ly}) \sin \beta - F_{sv} \sin(\delta + \beta)], \quad (5)$$

$$\dot{\beta}' = w_z - \frac{1}{mv} [(F_{uh} - F_{Lx}) \sin \beta + F_{uv} \sin(\delta + \beta) - (F_{sh} - F_{Ly}) \cos \beta - F_{sv} \cos(\delta + \beta)], \quad (6)$$

$$\dot{\psi}' = w_z, \quad (7)$$

$$\dot{w}_z' = \frac{1}{I_{zz}} [F_{sh} \ell_v \cos \delta - F_{sh} \ell_h - F_{Ly} e_{sp} + F_{uv} \ell_v \sin \delta], \quad (8)$$

$$\dot{\delta}' = \frac{\delta_c - \delta}{T_c}. \quad (9)$$

Herein, x and y are the spacial coordinates and v denotes the velocity. The side slip angle is given by β , the yaw angle is ψ and the steering angle δ . The single track model is already a quite detailed model of a car, which is frequently used in the automotive industry for the investigation of the lateral motion of cars. The model includes various forces acting on the vehicle body. That is, the lateral tyre forces F_{sh}, F_{sv} , longitudinal forces F_{uv}, F_{uh} as well as air resistance in longitudinal F_{Lx} and lateral F_{Ly} direction. Further we have the vehicle mass m and the distance from the centre of gravity to the drag mount point e_{sp} . The distance from the centre of gravity to the front and rear wheel are described by ℓ_v and ℓ_h respectively. The control input to the model are the commanded steering angle δ_c and a combined acceleration and deceleration force, which enters the above force terms. Details of the model can be found in, e.g., [6, 7]. The constant $T_c > 0$ is used to model a delay in the adjustment of the steering angle towards the commanded steering angle.

Another model in ROCS describes a mobile robot with two driven wheels on the left and the right, respectively. Its equations of motion read as follows:

$$\dot{x}' = \frac{v_L + v_R}{2} \cos \psi, \quad (10)$$

$$\dot{y}' = \frac{v_L + v_R}{2} \sin \psi, \quad (11)$$

$$\dot{\psi}' = \frac{v_R - v_L}{B}, \quad (12)$$

$$\dot{v}_L' = \frac{v_L^c - v_L}{T_c}, \quad (13)$$

$$\dot{v}_R' = \frac{v_R^c - v_R}{T_c}. \quad (14)$$

Herein, x and y denote the center of gravity of the robot, ψ the yaw angle, v_R and v_L the velocity of the right and

left wheels, respectively, and B is the width of the robot. The robot is controlled by the commanded velocities v_R^c and v_L^c of the right and left wheels. The constant $T_c > 0$ is used to model a delay in the adjustment of the velocities towards the commanded velocities.

These two models are included in order to illustrate that heterogeneous agents can be considered. Further vehicle models and models for mobile robots can be found in [5]. We like to point out that further models can be integrated into ROCS in a straightforward way. This is an important feature for our research purposes.

4 Control and Path Planning

The realtime feature is implemented through timers for the control loop of each vehicle, i.e., the control loop runs at a user-defined rate and triggers the import of sensor data and the update of controls. The computed controls are then applied to the vehicle, either for simulation purposes or to control a real vehicle. As for the control we distinguish between path tracking control and path planning control. The former aims to track a predefined (spline) path while the latter generates a path and a trajectory using mathematical vehicle models and on-line optimization methods in combination with model-predictive control, see, e.g., [7]. In both, path tracking and path planning, the aim is to realize the feedback law μ_i in (2). Currently, a dynamic inversion controller and a linear model-predictive controller are used for path tracking, see [5, 3] for details. These controllers can be applied to both models in Section 3.1. The model-predictive control concept is applicable to path planning as well, compare [7]. Since model-predictive control is a powerful and versatile control paradigm, especially for multi-agent systems, we outline in brief the working principle. Further details can be found in the monographs [14, 9].

To this end we consider dynamics in discrete time $t_n = t_0 + nh$, $n \in \mathbb{N}$, where $h > 0$ is the stepsize given by the control timer in ROCS. For notational convenience we restrict the discussion to $N = 1$ agent with state x , control u , and dynamics (1). Discretization of the latter using a suitable Runge-Kutta method leads to a discrete time system. A typical path tracking task requires to solve a linear-quadratic optimization problem of the following type at each t_n with measured state x_n at t_n :

Minimize the tracking error

$$\frac{1}{2} \sum_{k=n}^{n+M-1} \|x(t_k) - x_{ref}(t_k)\|^2 + \|u(t_k) - u_{ref}(t_k)\|^2$$

subject to the constraints

$$\begin{aligned} x(t_{k+1}) &= A_k x(t_k) + B_k u(t_k) & (k = n, \dots, n+M-1) \\ x(t_k) &\in X & (k = n, \dots, n+M) \\ u(t_k) &\in U & (k = n, \dots, n+M-1) \\ x(t_n) &= x_n \end{aligned}$$

Herein, the linear dynamics are obtained by linearization at the reference path (x_{ref}, u_{ref}) . The number $M \in \mathbb{N}$ denotes the preview horizon, which has to be chosen appropriately. The sets X and U define state and control constraints. Likewise a typical path planning task consists in solving a nonlinear optimization problem of the following type at t_n with measured state x_n at t_n :

Minimize the objective

$$\varphi(x(t_{n+M})) + \sum_{k=n}^{n+M-1} \ell(x(t_k), u(t_k))$$

subject to the constraints

$$\begin{aligned} x(t_{k+1}) &= F(x(t_k), u(t_k)) & (k = n, \dots, n+M-1) \\ x(t_k) &\in X & (k = n, \dots, n+M) \\ u(t_k) &\in U & (k = n, \dots, n+M-1) \\ x(t_n) &= x_n \end{aligned}$$

Herein, φ and ℓ are suitable functions modelling the control objective, e.g. driving fastly or economically. Now, the standard model-predictive control (MPC) concept requires to solve one of the above optimization problems repeatedly on a shifted time horizon. Figure 7 shows the outcome of a path planning task using the single track model in Section 3.1 for a track on the campus of the Universität der Bundeswehr München. Obstacles can be avoided as well, see [3].

4.1 Sample Vehicle Controller

We outline path tracking controllers for the vehicle models in Section 3.1. For the implementation of the controllers we use slightly modified models in terms of a curvilinear coordinate system:

$$s'(t) = \frac{v(t) \cos(\psi(t) - \psi_m(t))}{1 - r(t) \kappa_m(s(t))}, \quad (15)$$

$$r'(t) = v(t) \sin(\psi(t) - \psi_m(t)), \quad (16)$$

$$\psi'(t) = v(t) \kappa(t), \quad (17)$$

$$\kappa'(t) = u(t), \quad (18)$$

$$\psi_m'(t) = v(t) \kappa_m(s(t)). \quad (19)$$

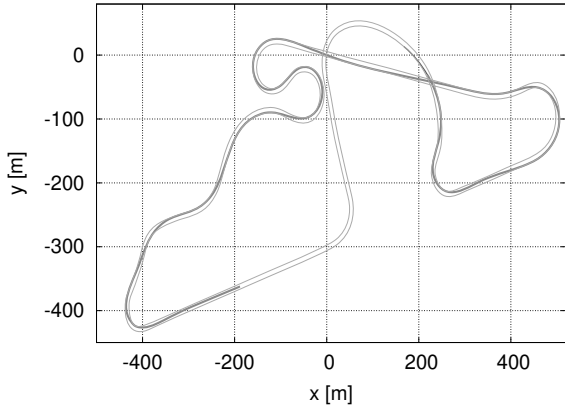


Figure 7: Nonlinear MPC result of a path planning task.

where s is the arc length along a given reference spline curve and r is the lateral offset from the reference spline. The actual heading is given by $\psi(t)$ and the corresponding reference heading is given by $\psi_m(t)$. The curvature of the driven path is denoted by κ and the curvature of the reference path is κ_m .

Both controllers are based on a simple kinematic model, Eqs. (15) to (19) and are designed to control the curvature deviation to track a given reference path. Herein, the controller class provides a control input to the vehicle models through a signal and slot connection. This allows for an easy extension with additional controllers, since the user only needs to provide an output signal. Then, the output is transformed for the respective model and eventually can be integrated employing one of the integrators, provided by the integration class, see Fig. 8.

The aforementioned transformations are model dependent. The single track model could be controlled through the steering angle δ , the commanded steering angle δ_c or the steering angle rate $\delta' = \omega_\delta$ respectively. Hence, we require a relation between the output of the controller, i.e., κ , and the control variables. For the commanded steering angle and the steering angle velocity, respectively, these relations are given by

$$\delta_c = \arctan(\ell\kappa), \omega_\delta = \ell\kappa' \cdot \cos^2(\delta). \quad (20)$$

Herein, $\delta' = \omega_\delta = \frac{\delta_c - \delta}{T_c}$ with constant $T_c > 0$. The two wheeled robot is steered through the velocities of the left and right wheel. Exploiting physical relations yields,

$$v_L^c = v_d - \frac{1}{2}B \cdot v \cdot \kappa, \quad v_R^c = v_d + \frac{1}{2}B \cdot v \cdot \kappa, \quad (21)$$

with B the width of the robot, v_d the desired longitudinal velocity, and $v = (v_L + v_R)/2$ the current velocity. Both controllers are discussed in detail in [4] and [3].

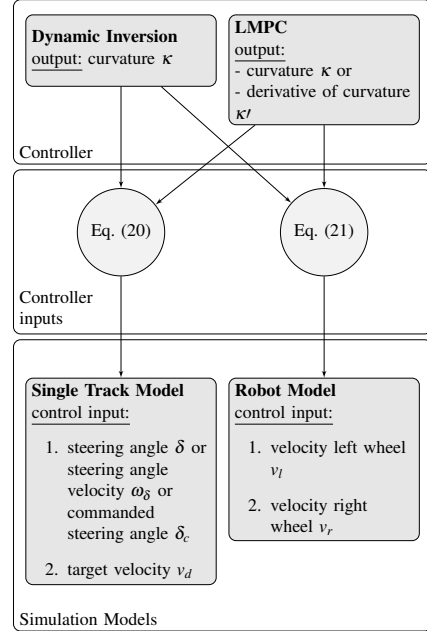


Figure 8: Information flow between simulation models and controllers.

5 Visualization

For the visualization of the control and simulation results we utilize the Qt Framework, which provides an OpenGL high level interface and allows for performant rendering in C++ applications. To visualize certain objects the data structure is based on a scene graph defined by a system of entities, where the scene graph is a tree structure made of these entities and other components. The entities to be rendered can be assigned through object files containing 3D models of, e.g., a car, an air plane, a robot, or buildings. Therefore we implemented an overloaded class of QSceneLoader addressing our requirements and managing the entities, as well as interfacing the rendering canvas. Herein, the rendering is solely a data driven process. Prebuild camera entities are provided by Qt providing viewpoints through which the scene is rendered. Multiple cameras are implemented in ROCS to capture different perspectives, e.g., the ego person's view in 9, the third person view, see Figure 10, where the camera follows in a fixed distance behind the object and a birds view in Figure 11.



Figure 9: Ego person's view of a scene.



Figure 10: Third person's view of a scene.

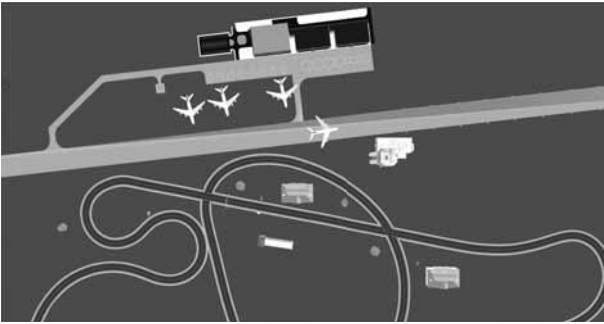


Figure 11: Bird's view of a scene.

6 Evaluation and Results

Figure 12 shows selected car data stored by the data logger function of ROCS. The virtual RAM used by ROCS for this simulation amounts to 2.37 GB and the RAM to 3.67 GB for a total of 12 simultaneously controlled cars. The GPU load amounts to 35.2 %. The computations were performed on a system with 16 GB of memory, Intel i7-8700 processor with 3.2 GHz (6 cores, 12 threads) and a Nvidia Geforce RTX 1060GB (6 GB RAM) graphic card.

The results show the output of the single track model with a linear model-predictive path tracking controller for the track depicted in Figure 7. This controller is able

to track a given geometric reference path even for comparatively high velocities with a maximum deviation of 0.15 m (see r in Figure 12).

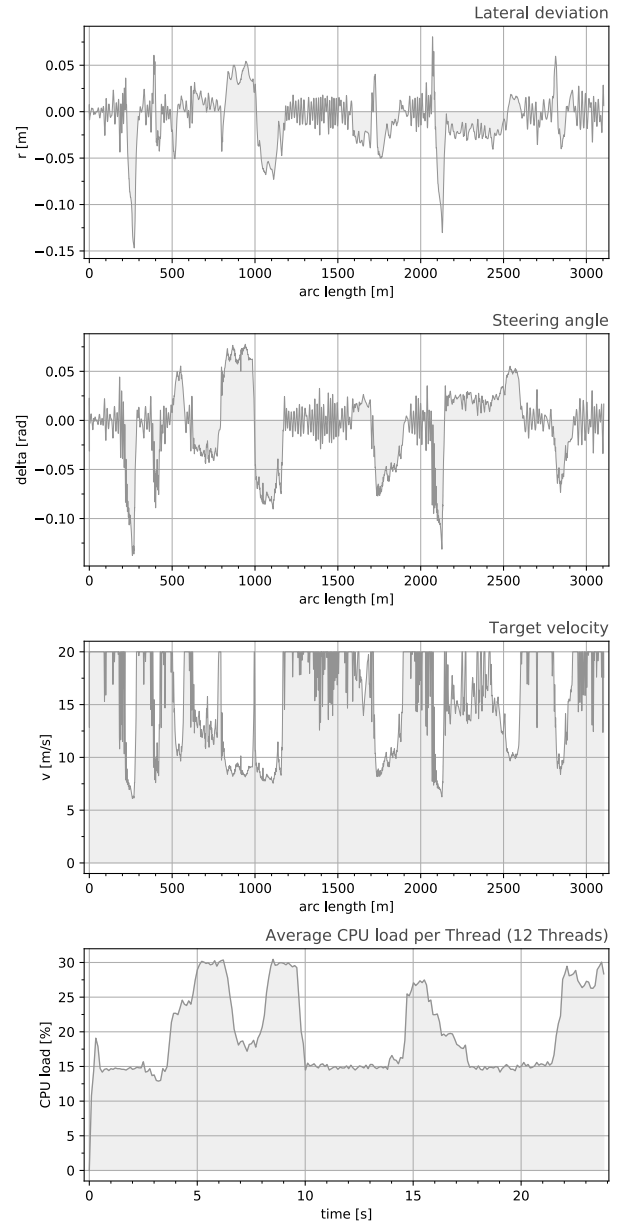


Figure 12: Data logger output (from top to bottom): lateral deviation r , steering angle δ , target velocity, and average CPU load per thread (12 threads, 12 vehicles).

7 Current Developments and Future Extensions

The development of ROCS is on-going and vehicle models from different disciplines (mobile robots, flight systems, space systems) of different complexity with appropriate controllers and path planning tools will be added step-by-step. The interfaces of ROCS will allow to directly import real sensor measurements of vehicles and to generate data to control a vehicle. This option allows to run simulation and real vehicle motion in parallel in order to overlap the two motions with the aim to design accurate digital twins. At the same time it allows to simulate a virtual world for a research vehicle at the Universität der Bundeswehr called Vehicle-in-the-loop [1, 2]. This research platform is based on a real car (Audi A6 Avant) and uses virtual environments to couple real driving experience and virtual scenarios. This concept is ideal for testing potentially dangerous scenarios in a safe way and we aim to integrate ROCS into the vehicle in the loop (VIL) for visualization, but also as an automatic control tool.

Acknowledgement

Copyright and ownership of ROCS and its derivatives solely resides with its founders Andreas Britzelmeier and Matthias Gerdt.

References

- [1] Berg, G., Karl, I., Färber, B. Vehicle in the loop - validierung der virtuellen welt. In Nichtred. Ms.-dr., editor, *Der Fahrer im 21. Jahrhundert: Fahrer, Fahrerunterstützung und Bedienbarkeit*, volume 6. Verein Deutscher Ingenieure, VDI-Verl., November 2011.
- [2] Berg, G., Nitsch, V., Färber, B. *Vehicle in the loop*. In: Winner H., Hakuli S., Lotz F., Singer C. (eds), *Handbook of Driver Assistance Systems*, Springer, 2015; pp. 199–210.
- [3] Britzelmeier, A., Gerdt, M. *A Nonsmooth Newton Method for Linear Model-Predictive Control in Tracking Tasks for a Mobile Robot With Obstacle Avoidance*. in *IEEE Control Systems Letters*, 2020; Vol. 4(4), pp. 886–891, doi: 10.1109/LCSYS.2020.2996959.
- [4] Britzelmeier, A., Gerdt, M., Rottmann, T. *Control of interacting vehicles using model-predictive control, generalized Nash equilibrium problems, and dynamic inversion*. 2020 IFAC World Congress, 2020.
- [5] Burger, M., Gerdt, M. *DAE aspects in vehicle dynamics and mobile robotics*. Applications of differential-algebraic equations: examples and benchmarks, Differential-Algebraic Equations Forum, Springer, 2019; pp. 37–80.
- [6] Gerdt, M. *Solving mixed-integer optimal control problems by branch&bound: a case study from automobile test-driving with gear shift*, *Optimal Control Applications and Methods*, Vol. 26, pp. 1–18, 2005.
- [7] Gerdt, M., Karrenberg, S., Müller-Beßler, B., Stock, G. *Generating locally optimal trajectories for an automatically driven car*. *Optimization and Engineering*, 2009; Vol. 10, pp. 439–461.
- [8] Graichen, M., Graichen, L., Rottmann, T., Nitsch, V. Using the projection-based vehicle in the loop for the investigation of in-vehicle information systems: First insights. In *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems - Volume 1: VEHITS*, pages 231–237. INSTICC, SciTePress, 2018.
- [9] L. Grüne, J. Pannek. *Nonlinear Model Predictive Control – Theory and Algorithms*. 2nd Edition, Springer, 2017
- [10] Hairer, E., Norsett, S. P., Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Series in Computational Mathematics, 2nd Ed., Vol. 8, Berlin-Heidelberg-New York, 1993.
- [11] Krueger, H.P., Grein, M., Kaussner, A., Mark, C. SILAB – A Task Oriented Driving Simulation. *North America*, page 9, 2005.
- [12] Microsoft Corporation. Microsoft flight simulator. www.flightsimulator.com, 04 2020.
- [13] Nitsch, V., Färber, B., Rüger, F. Automatic evasion seen from the opposing traffic - an investigation with the vehicle in the loop. In *IEEE 18th International Conference on Intelligent Transportation Systems*, 2015.
- [14] J. B. Rawlings, D. Q. Mayne, M. Diehl. *Model Predictive Control: Theory, Computation, and Design*. 2nd Edition, Nob Hill Publishing, Madison, 2018.
- [15] Roth, E., Dirndorfer, T., Knoll, A., von Neumann-Cosel, K., Ganslmeier, T., Kern, A., Fischer, M.-O.. *Analysis and validation of perception sensor models in an integrated vehicle and environment simulation*. *Proceedings of the 22nd Enhanced Safety of Vehicles Conference*, 2011.
- [16] Unity 3D. Unity website. unity.com, 04/2020.
- [17] VIRES Simulationstechnologie GmbH. Virtual test drive. vires.com/vtd-vires-virtual-test-drive, 04/2020.
- [18] Laminar Research. X-plane 11. www.x-plane.com, 04/2020.

Additive process chain: From virtual design to real implementation

Martin Rambke¹, Sven Lippardt¹, Tobias Mussehl¹

¹Ostfalia University of Applied Sciences, faculty of mechanical engineering, Salzdahlumer Straße 46/48, 38302 Wolfenbüttel, Germany

Abstract. In the course of the digital transformation, many companies are now striving to link virtual development and production processes with each other. The authors of this article want to discuss the following findings in the area of "additive manufacturing":

For the production of new components (in this research a node connection for a bicycle frame is used), it is often necessary to pay attention to the special features of additive manufacturing already during the design phase. Topology optimization software (MSC Apex Generative Design, former: AMendate) is used for a load-compliant design of the components in order to save material and thus reduce the production time. The designs must also be checked and compensated for distortions caused by production-related residual stresses (Simufact Additive). A validation of the distortion compensation is carried out after the production by an optical measurement of the component (Aicon). The strength and stiffness of the topology-optimized structure is verified by a test setup.

Introduction

At the Ostfalia University of Applied Sciences there are currently about 12,400 students studying in 12 faculties at four locations (Salzgitter, Suderburg, Wolfenbüttel and Wolfsburg). At the Institute for Production Technology (IPT) of the Faculty of Mechanical Engineering, seven professors and 16 research assistants are currently working in teaching and research. Four years ago, the Fabrication Laboratory (www.FabLab38.de) was initiated, which in 2017 - with the participation of other faculties - led to the foundation of the Center for Additive Manufacturing (ZaF). On meanwhile 25 "3D printers", student projects and research work in various processes (FDM, SLA, SLS, Polyjet etc.) are realized. Also in 2017, a Renishaw AM400 was applied for and procured as part of an EFRE infrastructure measure, which enables production with metal laser sintering (SLM) [1].

1 State of the art

In additive manufacturing, new components are currently being tested according to the trial and error principle.

The components are manufactured based on their original design and then a target-performance comparison is carried out. If too large deviations occur, a redesign of the component is necessary to compensate the resulting distortions. This process is repeated until the deviations of the component are within the specified tolerance. This procedure wastes a lot of time in testing before the final design of the component is determined. Figure 1 shows the testing procedure so far. In addition to the wasted development time, both manufacturing capacity and material are lost due to the production of scrap parts.

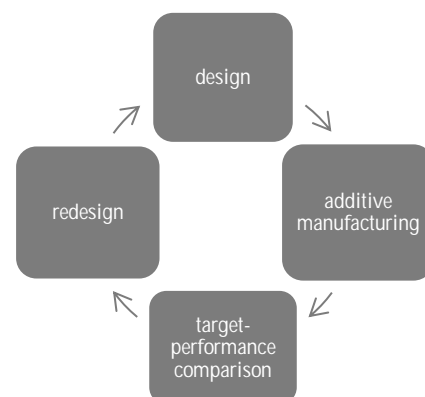


Figure 1: Procedure so far

2 Aim of the research projekt

In order to improve additive manufacturing, it is necessary to consider the entire process chain. For the implementation of the real process chain a virtual design is necessary to prevent errors in the real manufacturing process, to increase productivity and to minimize costs. The virtual design of the additive manufacturing process takes place in three steps. First, the existing design is adapted to the real load case with the help of topology optimization (MSC Apex Generative Design) in order to reduce the component weight and decrease the manufacturing time (Chapter 2). In the second step an optimization of the component alignment and the required

support structures is carried out (Chapter 3). Finally, a simulation of the additive manufacturing process is carried out (Simufact Additive) in order to examine the component for residual stresses and resulting distortion and to compensate them if necessary (Chapter 3). For the validation of the process the compensated components are manufactured and subsequently measured (Aicon). The measurement results are compared with the target component and the simulation results (Chapter 4). To check the component strength, the topology-optimized component is loaded on a test bench with the real load cases (Chapter 5). The aim is to establish a continuous process chain.

3 Topology optimization of the component

For a meaningful topology optimization of the component used, the design must first be checked. This check is necessary because the design geometry represents the design space for topology optimization. Inside the design space it is possible to remove excess material, but not to add outside. In addition to the design space, "non-design spaces" must be specified for the preparation of the topology optimization. Figure 2 shows the procedure for this.



Figure 2: Definition of „Non-Designspaces“

The "Non-Designspaces" are needed for the definition of bearing and connection areas, as this material is absolutely necessary and is left out of the design space. After the "Non-Designspaces" are defined, the occurring loads are defined. Figure 3 shows an example of the definition of a force on the component to be optimized.

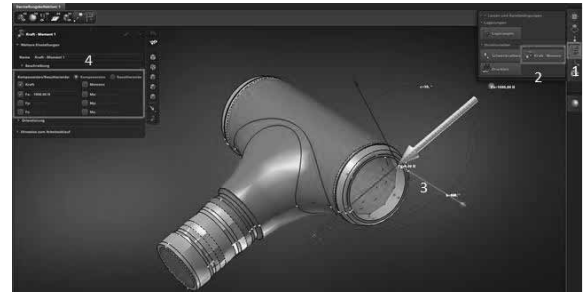


Figure 3: Definition of loads

Based on the defined forces, different load cases acting on the component can be set. After that, the topology optimization of the component is performed, whereby the "non-design spaces" remain filled with material and are not affected by the optimization.

In several iteration steps the design space is optimized considering the "non-design spaces", the defined loads and fixings. In doing so, unneeded material is removed, which has no effect on the force flow in the component. Figure 4 shows an example of the first, the thirtieth and the last iteration step. Finally, the optimized geometry is exported as an STL file and can be used for further processing in the preparation of the manufacturing job.



Figure 4: Iteration step 1 (left), iteration step 30 (middle) and iteration step 64 (right)

4 Preparation of the manufacturing job and simulation of additive manufacturing

The optimized geometry is designed for the real load cases, but machining post-processing of functional surfaces must also be considered. In the manufacturing job preparation, the corresponding component areas are provided with an offset (2 mm) so that the functionality of the component is not changed by machining post processing. Figure 5 shows the functional areas of the component with (red) and without (green) offset.

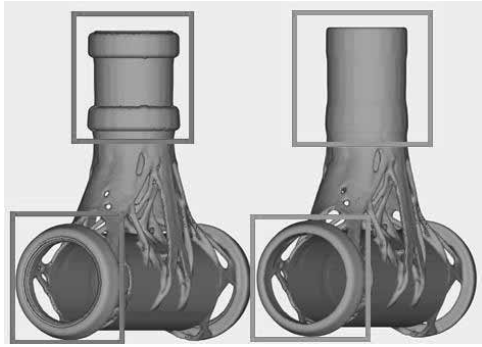


Figure 5: Part with offset for post-processing (left) and without offset (right)

A further step in the preparation of the manufacturing job is the creation of support structures to simulate the additive manufacturing process. For the creation of the support structure, the orientation of the component in the build space is highly relevant. The orientation influences the areas in which supports are required, the energy applied for each layer and the required manufacturing time. The layers should have a homogeneous cross-section (without large differences in cross-sections) in order to ensure a uniform energy input. Figure 6 shows different orientations of the component. With a suitable selection, the best possible compromise between the production time, the required support and the applied layer energy should be made.

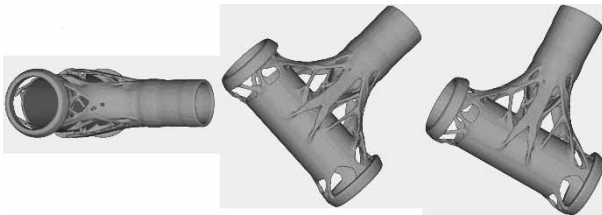


Figure 6: Orientation for minimum manufacturing time (left), minimum support volume (middle) and minimum cross-sectional differences (right)

A compromise of all three parameters is shown in Figure 7 and is used in the following step for the simulation of the additive manufacturing process.



Figure 7: Orientation of the component with supports

For the manufacturing simulation the mechanical calculation approach of "Simufact Additive" is used. It was used for this component because it is a predominantly closed geometry and the residual stresses that arise lead to a negligible deformation of the component [Cf. 2,3,4,5]. The results of the simulation are shown in Figure 8.

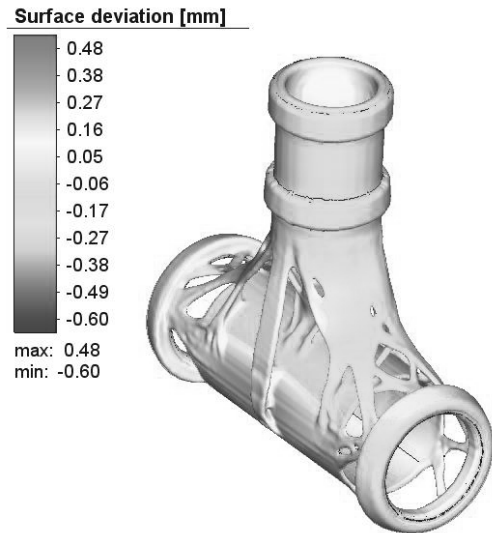


Figure 8: Simulation result of the additive manufacturing process

The calculated deviations occur in uncritical areas and that deformations in the area of the functional surfaces can be reworked by the machining post processing. For this reason, it was decided that this component is not subject to compensation.

5 Measurement of the manufactured components

The component was optically measured with a white light scanner system and a comparison was made between the real component and the simulation result. Figure 9 shows that the deviations from the calculated deformation are within ± 0.2 mm. Thus, the simulation result can be assumed to be validated, since the existing deviations can be attributed to the machine inaccuracy (melt pool is larger than the laser spot diameter, therefore the component is thickened).

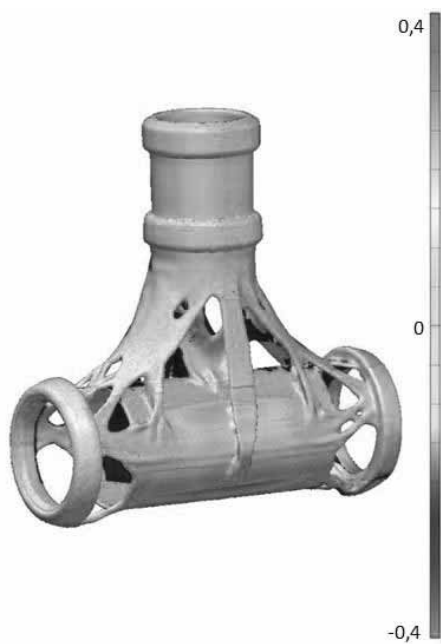


Figure 9: Comparison between simulation result and manufactured component



Figure 11: Comparison between CAD model and manufactured component

The measurement of the component was carried out before a machining post processing. As this is required to generate the accuracy, larger deviations can occur in the area of the connection points, as no distortion compensation was used for this component. Decisive for the accuracy of the production result are the areas of the component not to be machined, which are highlighted in Figure 10.



Figure 10: Area relevant for the accuracy assessment

In a second measurement, a target/actual comparison between the real component and the CAD model was carried out. Based on the result in Figure 11, it can be seen that the deviations in the surfaces to be machined are very significant (± 0.4 mm), but the rest of the component shows mainly deviations into the positive range. This is due to the larger melt pool than the laser spot diameter and results in a high volume of the component. Therefore a higher stiffness of the component can be expected and no post processing in these areas is necessary.

Based on the measurement result of the target/actual comparison, a sufficient accuracy of the component is available; in addition, the simulation results can be regarded as validated. Subsequently, the functional surfaces were reworked as preparation for the strength tests.

	Simulation/Manufactured Part	CAD-Modell/Manufactured Part
deviation in re-worked areas (before post processing)	$\pm 0,1$	$\pm 0,4$
deviation in load critical areas (no post processing)	0 - 0,2	0 - 0,4
	simulation validated	sufficient accuracy

Table 1: comparison of the deviations in different component areas

6 Load test of the topology optimized structure

For the strength tests, the reworked component was mounted on an existing tensile testing machine with a testing device (Figure 12). For the investigation of the stiffness the clamping of the component has to be taken into account, because within the topology optimization an asymmetrical load distribution was involved. The clamping of the component must correspond to the set loads from the topology optimization in order to obtain usable results.



Figure 12: Setting of the component on the test bench

The test was carried out until the component failed, with the focus being on the position of failure. The calculated stress curves from the topology optimization should be able to be checked in this way in real tests. The defined load was at a bending moment of 100 Nm. In the test setup, a lever arm of 325 mm was used to transfer the bending moment to the test specimen. Figure 13 shows that a force of 850 N was applied to the lever arm before the component failed. This corresponds to a failure bending moment of 276 Nm, which clearly fulfills the specification. Further optimization of the component is possible in order to further reduce the weight and still withstand the specified loads.

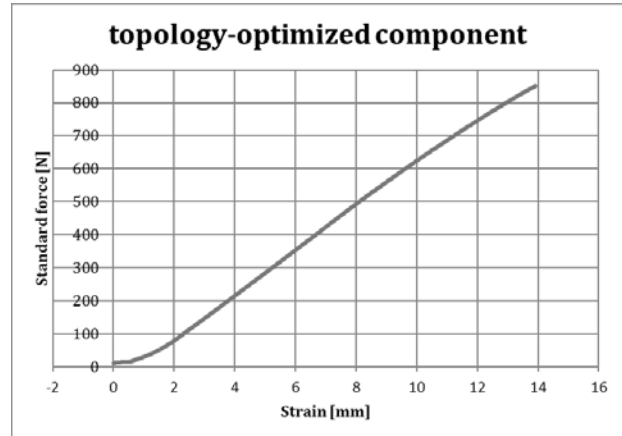


Figure 13: Force-strain diagram of the tested system (optimized component with connecting rods)

With regard to the position of the fracture point, the results from the test and the calculation are in agreement. Figure 14 shows the maximum calculated stress (marked area, left), the real component failure occurs at the predicted location (marked area, right).

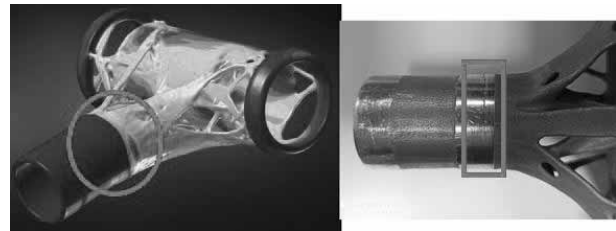


Figure 14: Comparison between calculated stresses (MSC Apex Generative Design, left) and real component failure (right)

Since only the component within the topology optimization was considered and not the connecting rods, a possible explanation is that the notch tension in the transition area has not been considered. In order to avoid this error, a topology optimization of the entire system will be investigated in a future project.

7 From the virtual to the real process chain

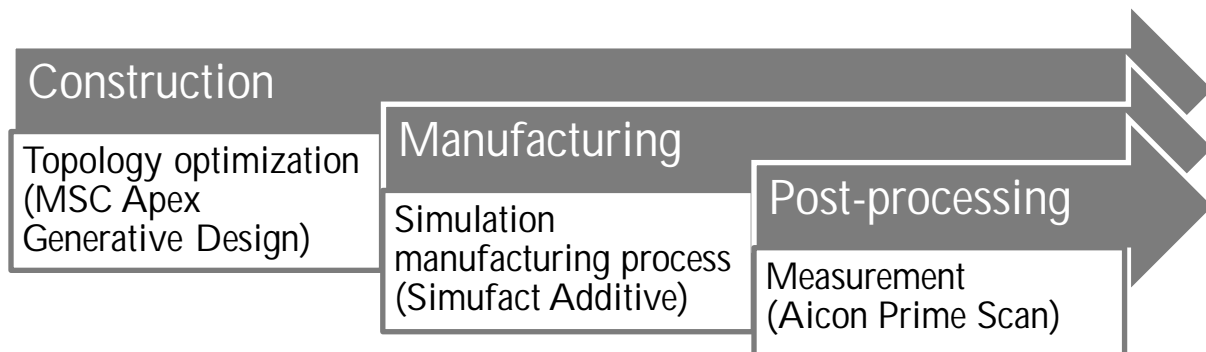


Figure 15: Support of the real process chain through the virtual process chain

Figure 15 illustrates how the virtual process chain has a supporting effect on the real process chain and thus optimizes the manufacturing process. The topology optimization provides a further improvement of the real design and opens up new potential for components. The manufacturing simulation helps to improve real production. By predicting the residual stresses that occur and the resulting distortions, it is possible to print components with the smallest possible deviation. Thus, malfunctioning components can be avoided and production resources can be saved. The measurement of the manufactured components serves to validate the manufacturing simulation results. In addition, the measurement allows the identification of component areas where an adjustment of the design or a machining reworking is necessary to maintain the tolerances.

8 Future research

Based on the results of these investigations, further approaches can be derived. On the one hand, a topology optimization of the entire system must be carried out to avoid stresses that have not been considered. Furthermore, a revision of the load-optimized structure (smoothing and closing of small holes to reduce notch effects) is recommended. To improve the load test, an adhesive connection should be made between the component and the connecting tube. Finally, a feedback of the topology-optimized component into a structural analysis for validation of the topology optimization results is planned.

References

- [1] Zentrum für Additive Fertigung; www.ostfalia.de/cms/de/zaf/detail/news/9d18688e-43a4-11e8-a117-d96edd3be9f9; Access date: 25.06.2020
- [2] Mussehl, T.; Rambke, M.: Simufact Additive: Mechanischer oder thermomechanischer Berechnungsansatz?, 20. Round Table, Simulation Manufacturing, Tagungsband, Marburg, Mai 22-23, (2019).
- [3] Mussehl, T.; Rambke, M.: Rapid-Tooling Ansätze mit dem Metall-Lasersintern - erste Ergebnisse, 19. Round Table, Simulation Manufacturing, Tagungsband, Marburg, Mai 16-17, (2018).
- [4] Mussehl, T.; Rambke, R.: Virtualisierung additive Fertigungsprozesse für das Rapid Tooling, 25. Interdisziplinäre Wissenschaftliche Konferenz Mittweida, Tagungsband, Mittweida, Oktober 24-25, (2018).
- [5] Mussehl, T.; Rambke, M.: Simulation und Kompensation der Eigenspannungen in der additiven Fertigung, ASIM – Workshop Simulation technischer Systeme/ Grundlagen und Methoden in Modellbildung und Simulation, Tagungsband, Braunschweig, Februar 21 – 22, (2019).

Systemsimulation als Teil des Systems Engineering für Scheinwerfer- und Pedalsysteme auf Basis der Modellbeschreibungssprache *Modelica*

Heinz-Theo Mammen¹, Phillip Limbach¹, Thorsten Maschkio¹

¹HELLA GmbH & Co. KGaA
Rixbecker Straße 75, 59552 Lippstadt

Kurzfassung.

Im vorliegenden Beitrag wird anhand von zwei Anwendungsbeispielen gezeigt, welche Synergien mit Hilfe der Systemsimulation erzielt werden können. Beim ersten Beispiel mit Fokus auf die Temperaturentwicklung in einem DC-Steller, konnte der bisherige messtechnische Aufwand zur Identifizierung kritischer Betriebsfälle durch die Simulation um den Faktor 4 reduziert werden. Das zweite Beispiel, ein Bremspedalsystem, zeigt eine Möglichkeit der Kontaktmodellierung zwischen starren Körpern, um Bewegungsabläufe zwischen diesen Körpern beschreiben zu können. Durch dieses neue Verfahren können Körperkontaktflächen auf einfache Weise in Modelica [1] beschrieben werden, wodurch der ursprüngliche Modellierungsaufwand für Kontaktflächen um den Faktor 10 reduziert werden konnte.

Mit Hilfe der Simulationsmodelle konnte das jeweilige Designkonzept optimiert sowie das Reib- und Temperaturverhalten einzelner Komponenten entsprechend den Spezifikationen angepasst werden.

Einleitung

Auf Grund der wachsenden Komplexität von Systemen und Subsystemen im Automotiv-Bereich gewinnt das Thema Systemsimulation als integrativer Bestandteil des Systems Engineering immer mehr an Bedeutung, um die stetige Absicherung des interdisziplinären Systemverhaltens über alle Phasen des Produktentstehungsprozesses hinweg von der Problemstellung bis zur Produktion zu gewährleisten. Im Rahmen der Systemsimulation bei Hella wurden in den vergangenen Jahren Systemmodelle für unterschiedliche Komponenten aus dem Automobilsektor entwickelt und validiert. Hierzu zählt unter anderem ein Modell eines DC-Stellers eines mechatronischen Scheinwerfermoduls mit den Komponenten Ansteuerung, elektrische Verstelleinheit und Getriebe unter Berücksichtigung des Umwelteinflusses Temperatur sowie ein Pedalsystem mit den Komponenten Gelenksystem, Positionserkennung sowie Rückstellmechanismus. Beim Pedalsystem wurde die Modellgrundstruktur automatisch aus dem Geometriedesign mit integrierter Kine-

matik heraus generiert und um Komponenten zur Beschreibung des Reibverhaltens, der Fußkraft, der Rückstellfedern sowie der Kontaktflächen erweitert.

Die Entwicklung der Modellstrukturen inklusive der Modellierung von Kontaktflächen erfolgte auf Basis der Modellierungssprache *Modelica* und dem Simulator *Dymola* [2]. Im Fokus dieses Beitrags liegen die spezifischen Eigenschaften der Modelle inklusive der Kontaktmodellierung sowie die automatische Modellgenerierung via direkter Kopplung zwischen Design- und Systemsimulationswerkzeug.

1 Modell eines DC-Stellers

Der Trend zu immer mehr mechatronischen Komponenten im Fahrzeug ist beispielhaft am Scheinwerfer erkennbar. Wo früher Leuchtmittel mit Reflektor in einem einfachen Gehäuse für eine Ausleuchtung der Straße gesorgt haben, treten heute komplexe mechatronische Scheinwerfersysteme, die ein dynamisches Kurvenlicht beinhalten, die über eine automatische Leuchtweitenregelung verfügen und die je nach Fahrstrecke (innerorts, Landstraße, Autobahn) für eine angepasste Ausleuchtung sorgen.

Da diese mechatronischen Systeme immer komplexer werden, ist eine ganzheitliche Entwicklung sinnvoll und notwendig. Unterstützt werden kann diese Entwicklung durch eine entwurfsbegleitende Modellierung von der Konzeptphase bis zum Serienbaustein.

Der DC-Steller ist eine Teilkomponente des mechatronischen Systems Scheinwerfer. Diese Teilkomponente stellt für sich auch bereits ein mechatronisches System bestehend aus einer Ansteuerung, einer elektrischen Verstellkomponente und einem Getriebe inklusive Verstellarm dar. Bild 1 zeigt einen DC-Steller eines mechatronischen Scheinwerfermoduls.

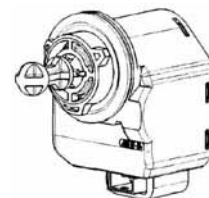
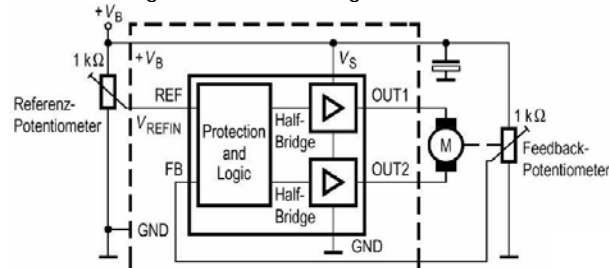
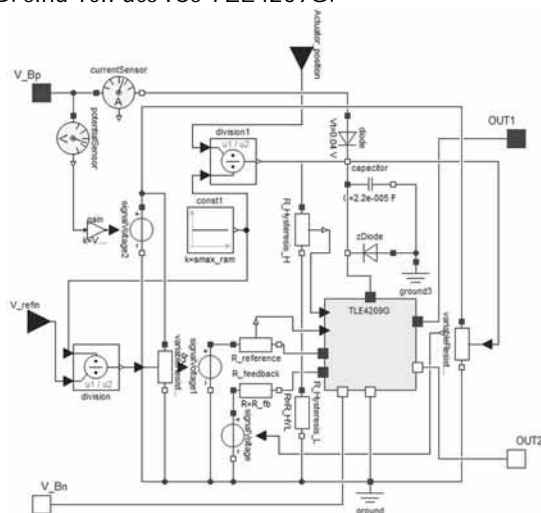


Bild 1: DC-Steller eines mechatronischen Scheinwerfermoduls

Die Ansteuerung des DC-Stellers ist in Bild 2 dargestellt. Sie besteht im Wesentlichen aus zwei Halbbrücken und einer Logik zur Ansteuerung dieser Halbbrücken.



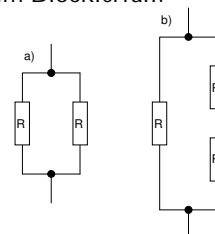
Das zur Ansteuerung eines DC-Stellers entwickelte Modell ist in Bild 3 dargestellt. Es wurde hierarchisch modelliert, d.h. die in Bild 2 dargestellten Halbbrücken z.B. sind Teil des ICs TLE4209G.



Um das thermische Verhalten des DC-Motors richtig modellieren zu können, muss das Kommutierungsverhalten zunächst beschrieben werden. Der Motor besteht aus drei Wicklungen, die über einen Kommutator bestromt werden. Im Blockierfall (Störfall) muss der Motor

[illegible]

Die Simulation soll zeigen, wie sich das Temperaturverhalten verschiedener Motoren im Blockierfall verhält. Dabei wird zwischen den zwei Blockierfällen, der 1/3- und der 2/3-Blockierung, unterschieden. Beim 1/3-Fall stehen beide Bürsten jeweils auf einer Lamelle des Kommutators, im 2/3-Fall steht eine der beiden Bürsten zwischen zwei Lamellen, wodurch diese überbrückt werden. Damit kommt es zu den folgenden zwei Ankerwiderstandszuständen im Blockierfall:



Im 2/3-Fall beträgt der Gesamtankerwiderstand $R_{ges}=1/2 R$ und im 1/3-Fall $R_{ges}=2/3 R$. Diese beiden Fälle wurden im Modell gesondert einstellbar modelliert, um das Temperaturverhalten im Blockierzustand für diese Fälle explizit ermitteln zu können.

Die Getriebekomponente beschreibt unter anderem das Verhalten eines Schneckengetriebes sowie die Anschläge für den Hin- und Rücklauf des Stellers. Zudem werden mit diesem Element Reibparameter, wie z.B. die Coulomb- und die Stribeck-Kraft modelliert. Somit ge-

hören zu den Parametern, die mit dieser Modellkomponente modelliert werden, das Übersetzungsverhältnis, die beiden Anschläge und die Reibung. In Bild 6 ist das Modell der Getriebekomponente dargestellt.

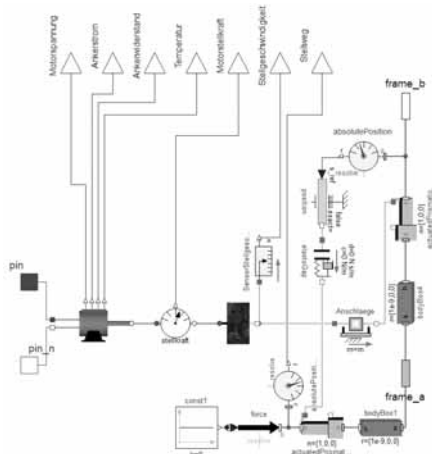


Bild 6: Getriebekomponente des DC-Stellers (inklusive Motor)

1.4 Modellvalidierung

Neben der Modellierung spielt die Modellvalidierung eine wichtige Rolle, damit am Ende des Prozesses ein Modell zur Verfügung steht, das für Untersuchungen während des gesamten Entwicklungspfad von der Konzeptidee bis zum fertigen Produkt „uneingeschränkt“ genutzt werden kann.

Die Parametrisierung eines Modells kann auf Basis von Datenblattangaben oder anhand von Messungen erfolgen. Speziell bei Motoren sind die Datenblattangaben häufig nicht ausreichend, um alle Parameter definieren zu können. Daher sind Messungen notwendig, die oft sehr aufwendig und kompliziert sein können und es wird dafür spezielles Messequipment benötigt.

Im vorliegenden Fall konnten die wesentlichen Motorparameter über Datenblattangaben berechnet werden, die Temperaturparameter wurden mit Hilfe von Messungen unter anderem im Klimaschrank ermittelt. Dabei wurden Untersuchungen zu den drei Betriebsfällen Drehbewegung, 1/3- und 2/3-Blockierfall durchgeführt.

Auf Basis der Untersuchungsergebnisse wurden die Modellparameter berechnet und ins Modell eingetragen. Nachdem alle Parameter bestimmt sind, ist es notwendig, die Funktion des Modells anhand ausgewählter Testuntersuchungen zu überprüfen. Dazu werden entsprechende Testanordnungen erstellt und die mit diesen Anordnungen erzielten Simulationsergebnisse mit Messungen verglichen. In Bild 7 ist eine entsprechende Testanordnung dargestellt.

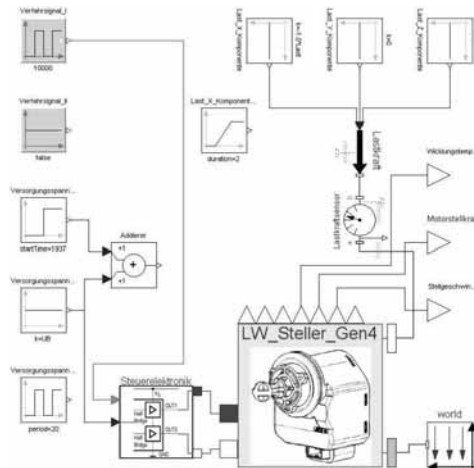


Bild 7: Testanordnung zur Validierung des Modellverhaltens

In Bild 8 ist beispielhaft ein Vergleich zwischen Messung und Simulation dargestellt. Es wurde der Ankerstrom des DC-Motors ermittelt.

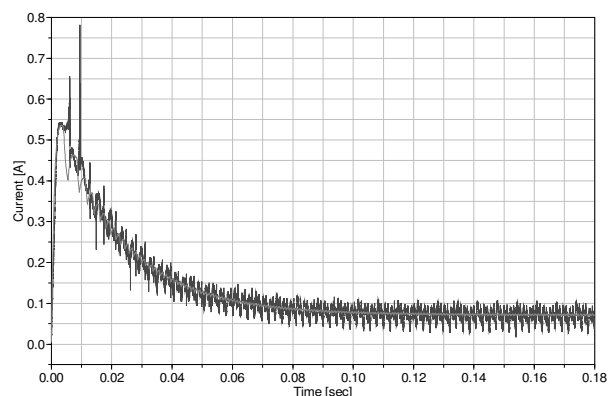


Bild 8: Ankerstrom des DC-Motormodells – Vergleich Messung (blau)/Simulation (rot)

Der Vergleich zwischen Messung und Simulation zeigt eine gute Übereinstimmung (weitere Simulationen haben dieses Ergebnis bestätigt). Damit kann dieses Modell für Untersuchungen im Entwicklungsbereich eingesetzt werden.

Die Analyse des Aufwandverhältnisses zwischen der Prototyp-Hardware-Entwicklung und der Modellierung/Simulation ergibt, dass das Verhältnis für das obige Beispiel bei 4:1 liegt, d.h. der Aufwand für die Hardware-Entwicklung ist 4mal so hoch. Mit der Modellierung/Simulation kann der Entwicklungsaufwand für einzelne Konzeptphasen also signifikant reduziert werden.

2 Modell eines Pedalsystems

Im Fahrzeug werden drei verschiedene Arten von Pedalsystemen unterschieden: das Bremspedal, das Kupplungspedal sowie das Gaspedal. Da diese Pedalsysteme nicht mehr wie vor Jahren direkt mit den zu betätigenden Systemen verbunden sind (z.B. das Gaspedal direkt über ein Seilsystem mit dem Motor), werden bei den heutigen Pedalsystemen andere Anforderungen ans Pedal gestellt. Daher sind die heutigen Pedalsysteme neben dem Betätigungsarm mit Reibelementen, Sensoren und Zusatzfedern ausgestattet, um das ursprüngliche haptische Verhalten so gut wie möglich nachzubilden.

Ein Beispiel eines Pedalsystems ist in Bild 9 dargestellt.

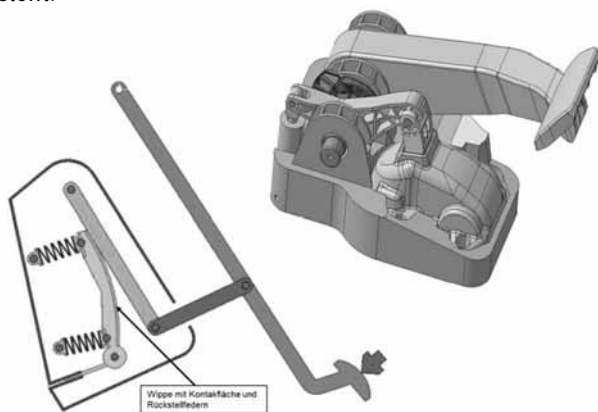


Bild 9: Designbeispiel eines Bremspedals

Das in Bild 9 dargestellte Bremspedal besteht im Wesentlichen aus einem Viergelenk, einer Wippe inklusive Rückstellfedern, einer Kontaktfläche und einem Reibelement. Die Kontaktfläche bildet die Schnittstelle zwischen dem Viergelenk und der Wippe. Mit Hilfe der geometrischen Form der Kontaktfläche läßt sich das Kraft-Weg Verhalten des Pedals entsprechend der Kundenspezifikation (siehe Bild 13) verändern.

Die wesentlichen Elemente des Pedals lassen sich mit Hilfe von Modelica-Standardelementen modellieren. Die Modellierung einer Kontaktfläche stellt eine besondere Herausforderung dar, daher wird im Folgenden darauf näher eingegangen. Eine Möglichkeit, um Kontaktvorgänge in Dymola zu simulieren, bietet die Idealized Contact Library [3].

2.1 Beschreibung der Oberfläche einer Kontaktfläche

In der Systemsimulation werden Körper meist als ideal starr angenommen. Diese Annahme ist möglich, da die Betrachtung der Kinematik im Vordergrund steht. Hierfür ist es ausreichend, die Körper durch die Lage des

Schwerpunkts sowie durch die Masse und die Trägheitsmomente im Schwerpunkt zu beschreiben. Die Ausdehnung der Körper wird dabei nicht weiter betrachtet. Für Kontaktphänomene ist jedoch sowohl die Modellierung von elastischen Körpern als auch die Definition der Oberfläche zwingend notwendig.

In der *Idealized Contact Library* steht für die Beschreibung von einfachen Geometrien, wie einem Rechteck, einem Zylinder oder einer Kugel, jeweils ein „surface“-Block zur Verfügung, der die Dimensionen der Oberfläche sowie deren Ausrichtung im angeordneten Koordinatensystem beschreibt. In der neuesten Release-Version sind zusätzlich Blöcke für die Beschreibung von Ellipsoiden sowie anderer konvexer Körper implementiert. Der „surface“-Block stellt eine dünne, masselose Fläche dar, die an jeden Starrkörper über eine „frame“-Schnittstelle angebunden werden kann.

Die eigentliche Berechnung der Kontaktkraft findet im „contact“-Block statt. Die notwendigen Informationen über die Kontaktoberfläche werden dabei über eine „contact“-Schnittstelle übermittelt. Neben der Definition des körperfesten Koordinatensystems der Oberfläche, werden auch die geometrischen Informationen der Kontaktfläche sowie in der neuesten Release-Version die Information über den Oberflächentyp transferiert.

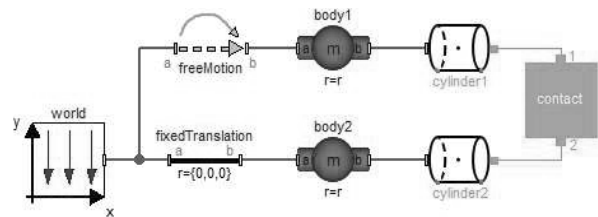


Bild 10: Aufbau eines einfachen Kontaktmodells mit zwei Zylindern

In Bild 10 ist ein Beispielmmodell für zwei in Kontakt tretende Zylinder dargestellt. Die als „cylinder1“ und „cylinder2“ bezeichneten Blöcke beschreiben die Oberfläche, die durch einen „contact“-Block (orange) verbunden sind. Komplexere Geometrien können durch Parallelschaltung einzelner Kontaktoberflächen zusammengefügt werden, wobei jedes Kontaktpaar über einen „contact“-Block verbunden sein muss [3].

2.2 Oberflächenmodellierung durch analytische Funktionen

Der Ansatz, die Kulissenoberfläche durch analytische Funktionen zu beschreiben, entspringt aus der Notwendigkeit einer krümmungsvariablen Oberfläche. Diese soll über den betrachteten Bereich tangential- und krümmungstetig sein.

Polynomfunktionen dritten Grades

In diesem Ansatz wird die Kulissengeometrie durch eine oder mehrere Polynomfunktionen beschrieben, die im Schnittpunkt die erforderlichen geometrischen Bedingungen erfüllen. Definiert man eine Polynomfunktion dritten Grades mit

$$f(z) = a_3(z - z_0)^3 + a_2(z - z_0)^2 + a_1(z - z_0) + a_0 \quad (1)$$

so lässt sich die Funktion, nach Bestimmung der Koeffizienten a_i sowie des Parameters z_0 , eindeutig beschreiben. Sind die Koordinaten von zwei Punkten auf der Funktion sowie die erste und zweite Ableitung in einem Punkt bekannt, lassen sich die unbekannten Variablen durch

$$a_0 = f(z_1) - a_1(z - z_0) - a_2(z - z_0)^2 - a_3(z - z_0)^3 \quad (2)$$

$$a_1 = \frac{f(z_2) - a_0 - a_2(z - z_0)^2 - a_3(z - z_0)^3}{(z - z_0)} \quad (3)$$

$$a_2 = \frac{f'(z_1) - a_1 - 3a_3(z - z_0)^2}{2(z - z_0)} \quad (4)$$

$$a_3 = \frac{f''(z_1) - 2a_2}{6(z - z_0)} \quad (5)$$

berechnen. Der Parameter z_0 wird gleich Null gewählt.

Soll die Polynomfunktion eine weitere Funktion in einem Punkt tangential- und krümmungsstetig schneiden, werden die erste und zweite Ableitung im Schnittpunkt durch die bereits definierte Funktion vorgegeben. Hierbei ist lediglich die Position der beiden Punkte zu definieren. Das Modell zur Beschreibung einer Polynomfunktion wurde als Modelica Modell abgebildet. Ein Teil des Kontaktflächenmodells, die Modellkomponente zur Berechnung des Kontaktpunktes auf der Ellipsenoberfläche ist in Bild 11 dargestellt.

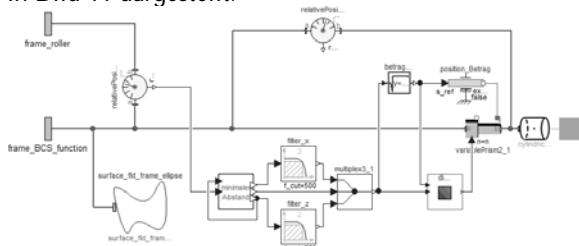


Bild 11: Modell zur Berechnung des Kontaktpunktes auf der Ellipsenoberfläche

Für die Berechnung des Kontaktpunktes auf der Kulissengeometrie wurde das Newton-Verfahren implementiert.

Ist die Polynomfunktion für einen bestimmten Bereich definiert, ist die Berechnung des minimalen Abstandes nur innerhalb dieses Bereiches notwendig und teilweise nur hierin möglich. Aus diesem Grund wird durch vorab definierte Grenzwerte die Berechnung des

minimalen Abstandes auf den Definitionsbereich eingeschränkt. Programmtechnisch wird dies durch Abfrage des berechneten Kontaktpunktes gelöst. Liegt dieser außerhalb des Bereiches, wird er gleich dem definierten Grenzwert gesetzt.

Die Kulissengeometrie lässt sich nicht durch eine einzelne Polynomfunktion beschreiben, die das gewünschte Pedalverhalten über den gesamten Pedalweg generiert. Deshalb wird das Modell sukzessiv durch mehrere Funktionen aufgebaut. Dies geschieht im ständigen Abgleich mit dem gewünschten Pedalverhalten.

Realisierung der Kulissengeometrie

Im ersten Schritt wird die Kulissenform, die zunächst durch eine einzelne Polynomfunktion beschrieben wird, entsprechend der vorgegebenen Vorspannkraft und der implementierten Wippenfederparameter derart angepasst, dass die Pedalkraft zu Beginn der Auslenkung dem Wunschverhalten entspricht. Weicht die Kraft bei größeren Pedalwinkeln von den Referenzwerten ab, wird die Position des Kontaktpunktes bei dem entsprechenden Pedalwinkel auf der Funktionsoberfläche bestimmt und als Schnittpunkt zur weiteren Funktion definiert. Im nächsten Schritt wird der zweite Punkt der „anschließenden“ Funktion derart gewählt, dass das gewünschte Pedalverhalten für einen weiteren Bereich erreicht wird. Dies wird sukzessiv durchgeführt bis die Kulissengeometrie vollständig modelliert ist.

Auf Basis der auf diese Weise definierten Kulissengeometrie kann im nächsten Schritt die ermittelte Oberflächenform direkt in CATIA [4] beim Design der Geometrie berücksichtigt werden. Hierdurch können Konstruktionsschritte und Prototypentwicklungen inklusive Messungen reduziert werden.

2.3 Validierung des Gesamtmodells

Das in Bild 12 dargestellte Gesamtmodell setzt sich im Wesentlichen aus drei Hauptkomponenten zusammen.

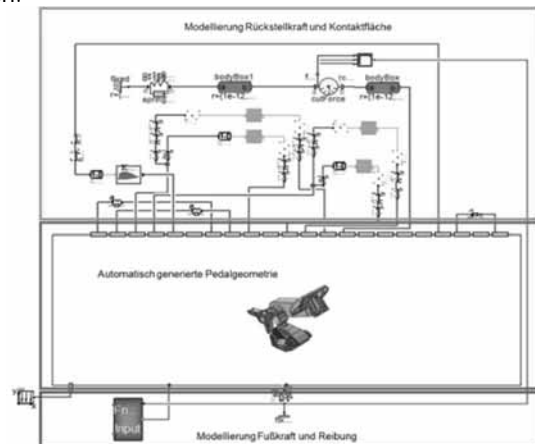


Bild 12: Gesamtmodell des Pedalsystems

Hierbei handelt es sich um die direkt aus dem CATIA Design generierte Pedalgeometrie, einer Komponente zur Modellierung der Rückstellkraft und der Kontaktfläche zwischen dem Pedalviereck und der Wippe sowie einer Komponente zur Beschreibung der Fußkraft und der Reibung, die im Pedal auftritt.

Das mit diesem Modell erzeugte Simulationsergebnis ist in Bild 13 dargestellt. Es zeigt den Verlauf der Pedalkraft (rot) über den gesamten Pedalweg im Vergleich zur Wunschkennlinie.

Beim Rücklauf weicht das Simulationsergebnis unter den gewählten Parameterbedingungen vom Wunschverhalten ab. Dennoch ist der tendenzielle Verlauf ähnlich dem gewünschten Verhalten. Durch Erhöhung des gewählten Reibwertes, der im Modell angenommen und nicht durch reale Messungen verifiziert ist, wird die Spreizung der Hysteresekennlinie vergrößert. Unter Anpassung der Kulissegeometrie, lässt sich somit das gewünschte Rückstellverhalten nahezu abbilden, wobei die Rückstellkraft bei einem Pedalwinkel von 32° unterhalb des geforderten Kraftminimums liegt. Hingehend zu einem geringeren Pedalwinkel, wird das Wunschverhalten in der Simulation annähernd erreicht.

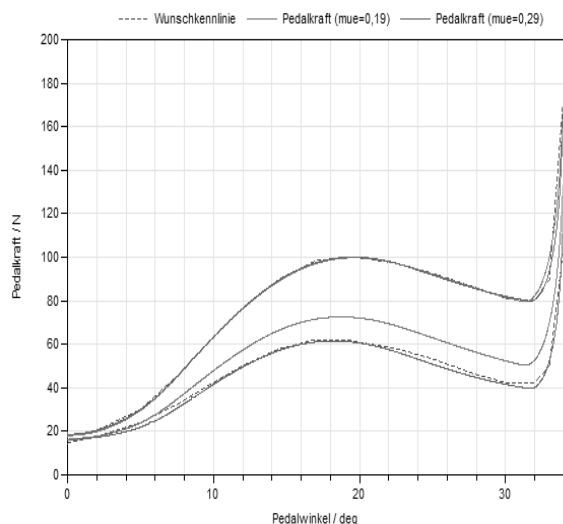


Bild 13: Einfluss der Pedalkraft bei Veränderung des Reibwertes

3 Zusammenfassung

Die Systemsimulation birgt viele Vorteile, da über alle Produktentstehungsphasen hinweg von der Problemdefinition bis zur Produktion eine zusätzliche Absicherung erfolgt, Kosten eingespart (z.B. weniger Prototypen), Zeitaufwände reduziert und Aussagen zu Systemeigenschaften getroffen werden können. Letztere wären mit Hardware-Aufbauten nicht oder nur mit erheblichem Aufwand möglich (z.B. Kraftmessungen in einem gekapselten System).

Die Modellbeispiele haben gezeigt, dass mit der Simulation Kundenanforderungen schnell überprüft, kritische Betriebsfälle simulativ abgebildet sowie Parameter- und Aufbaustudien vereinfacht durchgeführt werden können. Die dabei gewonnenen Erkenntnisse tragen zu einem umfassenderen Systemverständnis bei. Entwicklungsfehler in frühen Entwicklungsphasen, die fatale Folgen haben können, lassen sich auf diese Weise meist vermeiden.

Ein weiterer Vorteil der Systemsimulation ist in der Reproduzierbarkeit der Versuchsbedingungen zu sehen. Kraft- und Bewegungsszenarien können z.B. durch Messungen nicht exakt reproduziert werden. Insbesondere für die spätere Testphase ist dies ein wichtiger Punkt.

Ein nicht zu unterschätzender Aspekt ist auch die Möglichkeit zur Visualisierung von Bewegungsabläufen mittels Animation. Durch Animation lässt sich das Systemverhalten in seiner Gesamtheit „erleben“ und bewerten. Dies gewährleistet eine schnellere Ergebnisinterpretation und erleichtert überdies die Kommunikation zwischen den beteiligten Entwicklern. Voraussetzung hierfür ist der Aufbau eines Mehrkörpersystemmodells, mit dem unter anderem der mechanische Aufbau nachgebildet werden kann.

Bezogen auf den Entwurfsablauf lassen sich Systemmodelle auch als ausführbare Lastenhefte auffassen. Auf diese Weise dokumentiert sich der Entwicklungsprozess teilweise von selbst. Alle Experimente lassen sich auch im Nachhinein reproduzierbar nachvollziehen. Dies erleichtert den Nachweis von Fehlerursachen und vereinfacht überdies die nachträgliche Durchführung von Änderungen.

In der Konzeptphase geht es unter anderem darum, die in Frage kommenden Konstruktionsvarianten untereinander zu bewerten, um das bestgeeignete Lösungskonzept zu ermitteln.

Literaturverzeichnis

- [1] Fritzson, P.: Object-Oriented Modeling and Simulation with Modelica 2.1. John Wiley & Sons, New York 2004.
- [2] DYMOLA: Multi-Engineering Modeling and Simulation. www.dynasim.se, www.dymola.com.
- [3] Oestersötebier, F.; Wang, P.; Trächtler, A.: A Modelica Contact Library for Idealized Simulation of Independently Defined Contact Surfaces. Paderborn.
- [4] Dassault Systemes: 3DEXPERIENCE Platform <https://www.3ds.com/de/ueber-3ds/3dexperience-plattform/>

Anforderungsmanagement für die modellbasierte Entwicklung mechatronischer Systeme im digitalisierten und vernetzten Umfeld

Or Aviv Yarom^{1*}, Jie Zhang¹, Christian Raulf², Xiaobo Liu-Henke¹, Thomas Vietor²

¹Institut für Mechatronik, Ostfalia Hochschule für angewandte Wissenschaften, Salzdahlumer Str. 46/48, 38302 Wolfenbüttel, Deutschland; *o.yarom@ostfalia.de

²Institut für Konstruktionstechnik, Technische Universität Braunschweig, Hermann-Blenk-Str. 42 38108 Braunschweig, Deutschland

Abstract. Der folgende Beitrag beschreibt eine Entwurfsmethodik für die modellbasierte Entwicklung mechatronischer Systeme im digitalisierten und vernetzten Umfeld, unter Anwendung eines modellbasierten Anforderungsmanagements. Das Ziel ist eine ganzheitliche Betrachtung komplexer Systeme und derer Schnittstellen zwischen dynamischen Anforderungen und Systemstruktur. Im weiteren Verlauf wird das Vorgehen anhand eines Beispiels aus der Fertigungstechnik aufgezeigt und validiert.

Einleitung

Gesellschaft und Wirtschaft befinden sich aufgrund der zunehmenden Digitalisierung und Vernetzung in einem tiefgreifenden Wandel. Ein Beispiel hierfür ist die Automobilindustrie, die unter Einsatz des autonomen Fahrens und der Herausforderung eines flexiblen, anwendungsspezifischen Fahrzeugeinsatzes mit rasant steigenden Anforderungen an Fahrzeugentwicklung und Software zu kämpfen hat. Ein weiteres Beispiel ist die Fertigungsindustrie, die zur Erhaltung ihrer Wettbewerbsfähigkeit bei steigender Anzahl an Produktvariationen auf Industrie 4.0 (I4.0)-Lösungen zur Produktivitäts- und Flexibilitätssteigerung angewiesen ist. Dies sind nur zwei ausgewählte Domänen in denen sich der Bedarf an leistungsfähigen Komponenten aus den Disziplinen Mechanik, Elektrik/Elektronik und Informationstechnik bei der Entwicklung innovativer Produkte bzw. Systeme kontinuierlich erhöht. Das Einbringen moderner Kommunikationstechnologien ermöglicht die Vernetzung von Systemen untereinander, mit deren (System-)Umgebung oder auch mit dem Internet of Things (IoT) und schafft zusätzliche Mehrwerte für deren Nutzung und Anwendung. Darüber

hinaus erhöhen komplexe Algorithmen, z.B. aus dem Gebiet der künstlichen Intelligenz (KI), den notwendigen Aufwand für den Entwurf und die Absicherung der resultierenden cyberphysischen Systeme (CPS) massiv. Daher erfordert die Entwicklung und Validierung komplexer und intelligenter Systeme in einer zunehmend schnelllebigem, digitalisierten und vernetzten Umgebung die Umgestaltung bewährter Methoden und Prozesse.

Die vom Europäischen Fonds für regionale Entwicklung (EFRE) geförderten Innovationsverbunde *auto-MoVe (Dynamisch konfigurierbare Fahrzeugkonzepte für den nutzungsspezifischen autonomen Fahrbetrieb)* und *Synus (Methoden und Werkzeuge für die synergetische Konzipierung und Bewertung von Industrie 4.0-Lösungen)* befassen sich intensiv mit etwaigen Forschungsfragen in den Domänen automatisiertes Fahren und I4.0.

Dieser Beitrag entsteht im Rahmen dieser beiden Forschungsprojekte und befasst sich mit einer ganzheitlichen Methodik für den Entwurf intelligenter mechatronischer Systeme, vom frühen Stadium der Anforderungserhebung, bis hin zur Systemabsicherung. Die Methodik wird an einem Anwendungsbeispiel zur konfliktfreien Trajektorienplanung demonstriert und validiert.

1 Stand des Wissens

In diesem Kapitel werden zunächst bewährte sowie moderne Entwurfsmethoden aus der Konstruktionssystematik und Mechatronikforschung vorgestellt.

1.1 Modellbasiertes Anforderungsmanagement

Digitale, vernetzte mechatronische Produkte zeichnen sich dadurch aus, dass die gewünschten Fähigkeiten oder

Features nicht einem einzelnen Bauteil oder einer Baugruppe zugeordnet werden können, sondern ein komplexes Zusammenspiel vieler Komponenten aus verschiedenen Domänen erfordern. Dabei führt der Einsatz von immer mehr Softwarekomponenten zu einem Anstieg der Komplexität der technischen Systeme. Ein mechatronisches System beinhaltet dabei Elemente aus den Bereichen Mechanik, Elektronik und Informatik, beschreibt deren Zusammenwirken und grenzt sie gegenüber ihrer Umgebung ab. [1] Das zu entwickelnde System, das sogenannte System of Interest (SOI) lässt sich, in Teil- oder Subsysteme untergliedern und kann wiederum selbst in ein übergeordnetes System (komplexe Systemumgebung) eingebettet werden. [2]

Das des Systems Engineering beschäftigt sich mit der interdisziplinären Entwicklung und Umsetzung technischer Systeme. Hierbei gilt die Annahme, dass das System nicht bloß die Summe seiner Elemente ist, sondern sich über deren Zusammenhänge definiert. [3] Existiert nun ein globales abstraktes Metamodell, welches die Zusammenhänge und Wechselwirkungen der einzelnen Elemente beschreibt, spricht man vom Model Based Systems Engineering (MBSE). Dieses Metamodell soll im Folgenden als Systemmodell bezeichnet werden.

In der Konstruktionstechnik dient das Model-based Systems Engineering unter anderem dazu, verschiedene Entwicklungsaktivitäten in einem zentralen interdisziplinären Systemmodell zu dokumentieren und zu verwalten. Dieses kann z.B. genutzt werden, um kontextspezifische Sichten für die verschiedenen Entwicklungsdomänen zu generieren oder Systemabhängigkeiten und -zusammenhänge zu erkennen und zu visualisieren [2]. Um die komplexen Systemzusammenhänge zu modellieren, bedient man sich der grafischen Modellierungssprache SysML. Die wesentlichen Säulen der Systemmodellierung mit SysML werden durch die Darstellung in den drei Bereichen Anforderungen, Systemverhalten und der Systemstruktur gebildet. [4]

Zu Beginn eines jeden Entwicklungsprozesses gilt es die Anforderungen an das zu entwickelnde System zu erheben und zu dokumentieren. Die Anforderungen selbst stammen in erster Linie aus den Bedürfnissen der Stakeholder. Ergänzend fließen geltende Normen, Richtlinien und gesetzliche Rahmenbedingungen ein sowie Erfahrungswerte aus bereits vorhanden Systemen. [5] Um die Anforderungen auch in komplexen Systemen umfassend zu erfassen und zu verwalten, wurde bisher eine Gliederung in die Betrachtungsebenen "Geschäftsebene", "Systemebene" und "Komponentenebene" vorgenommen.

Zwischen den Anforderungen und der Systemarchitektur bestehen Wechselwirkungen: Die Architektur gründet sich auf den entsprechenden Anforderungen, gleichzeitig wirken sich Entwurfsentscheidungen auf die Anforderungen aus. [6] Anforderungen sind somit nicht bloß statisch, zu Beginn der Entwicklung definiert, sondern können dynamisch verändert und angepasst werden. [5] Insbesondere vor dem Gesichtspunkt interdisziplinärer Zusammenarbeit erschwert dieser Umstand den Entwicklungsprozess. Das Anforderungs- oder Requirements-Management mittels SysML wird als modellbasiertes Anforderungsmanagement (MBRE) bezeichnet. MBRE bietet die Möglichkeit, dynamische Anforderungen über den gesamten Produktentstehungsprozess bzw. -lebenszyklus im Systemmodell zu verwalten und Änderungen eindeutig zu kommunizieren und nachzuverfolgen.

1.2 Mechatronische Entwurfsmethodik

Die in der Mechatronikforschung bewährte Entwurfsmethodik basiert auf einer Modularisierung und Hierarchisierung des komplexen Gesamtsystems. Dabei wird das SOI in einem Top-Down Verfahren zunächst in intelligente, gekapselte Module aus mechatronischen Teilsystemen mit definierten Schnittstellen zur Kommunikation mit ihrer Umgebung zerlegt. Anschließend werden die Module in einer hierarchischen Struktur in Beziehung zueinander gesetzt. [7] Abbildung 1 zeigt exemplarisch die hierarchische Struktur des Forschungsfahrzeugs FREDY (Funktionsträger für regenerative Elektromobilität und Fahrdynamik) auf sechs Hierarchieebenen.

Die mechatronischen Funktionsmodule (MFM) bilden die unterste und zugleich vitalste Hierarchieebene. Sie beinhalten nicht weiter teilbare Module, bestehend aus Grundaufbau, Sensorik, Aktorik und einer grundlegenden Informationsverarbeitung. Sie besitzen physikalische und informatorische Schnittstellen zu den überlagerten mechatronischen Funktionsgruppen (MFG). Diese besitzen keine eigene Aktorik, sondern greifen auf die MFM zu und realisieren in Kombination mit einer eigenen Informationsverarbeitung höherwertige Funktionen. Mehrere MFM und MFG, bilden in ihrer Gesamtheit ein autonomes mechatronisches System (AMS). Ein AMS ist unabhängig von seiner Umgebung und verfügt über eine eigene Informationsverarbeitung und Informationsschnittstellen zu anderen Systemen. Wenn mehrere AMS Informationen durch digitale Vernetzung austauschen und so bspw. kooperativ operieren spricht man von auto-

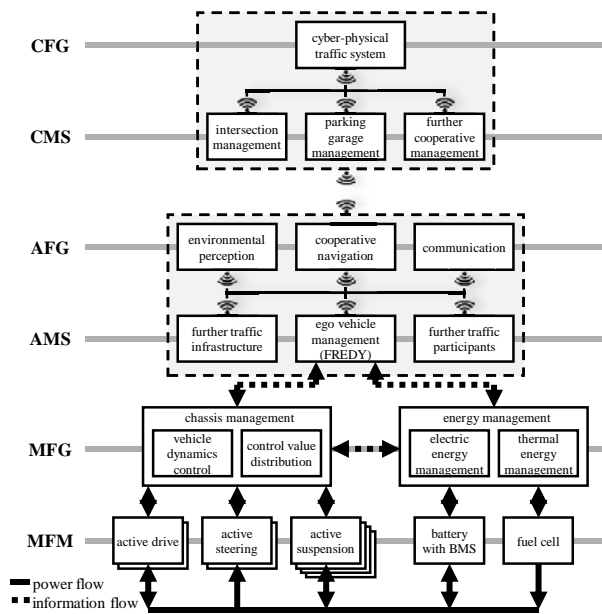


Abbildung 1: Mechatronische Strukturierung des FREDY

nome Funktionsgruppen (AFG). Agieren die AFG in einem speziellen Bereich, wie einem Kreuzungsmanagement, werden Entscheidungen in einem vernetzten mechatronischen System (CMS) getroffen und koordiniert. Das CMS reguliert den Informationsfluss und leitet entsprechende Befehle an unterlagerte AFG und AMS weiter. Mehrere CMS lassen sich zu vernetzten Funktionsgruppen (CFG) zusammenfassen, sodass Daten in strukturierten Clustern ausgetauscht werden können.

Die anschließende Auslegung jedes Funktionsmoduls mit der zugehörigen Informationsverarbeitung erfolgt Bottom-Up unter Anwendung des mechatronischen Entwicklungskreises [8]. In Kombination mit der anschließenden Absicherung und Integration wird dieses Vorgehen als mechatronische Komposition bezeichnet. Basierend auf den Anforderungen und der Spezifikation werden die physikalische und mathematische Modellierung sowie die Analyse und Parameteridentifikation durchgeführt, so dass ein validiertes Modell des Systemverhaltens die Grundlage für die modellbasierte Entwicklung von intelligenten Informationsverarbeitungen bildet. Diese Methodik zeichnet sich durch eine frühzeitige Validierung und Verifikation aus, so dass Entwicklungszeit und -kosten reduziert werden können [8]. Die Reglerentwicklung erfolgt nach dem Rapid Control Prototyping (RCP) in einem vollständig verifikationsorientierten Prozess mit MiL-, SiL- und HiL-Simulationen.

2 Kopplung von Anforderungsmanagement und mechatronischer Entwurfsmethodik

2.1 Problemstellung und Motivation

Die in Kapitel 1 vorgestellten Entwurfsmethoden befassen sich mit der strukturierten Entwicklung intelligenter mechatronischer Systeme, die wegen ihrer zahlreichen miteinander agierenden Komponenten und der digitalen Vernetzung mit ihrer Umwelt hochkomplex sind. Aufgrund ihres Ursprungs aus den Domänen der Konstruktionstechnik und der Mechatronikforschung bedienen die Entwurfsmethoden jeweils unterschiedliche Sichtweisen und Schwerpunkte. Das MBRE beschreibt Systemzusammenhänge und Anforderungen auf einer übergeordneten Metaebene als essentielles Werkzeug zur Ermittlung, Dokumentation und Verwaltung von Anforderungen. Es eignet sich sehr gut, um sich ändernde Anforderungen und deren Auswirkungen im komplexen Gesamtsystem zu überblicken und zu verfolgen. Der dynamische Charakter und damit der entscheidende Vorteil des MBRE geht allerdings verloren, wenn das Anforderungsmanagement von der konkreten Entwicklungstätigkeit entkoppelt wird. In diesem Fall müssen die Anforderungen in natürlicher Sprache, klassischerweise in Form von Lasten- oder Pflichtenheften, an beteiligte Entwickler weitergeben werden. Der mechatronische Entwurf hingegen fokussiert eher die funktionalen und strukturellen Wechselwirkungen des Systems, die für den anschließenden konkreten Entwurfs- und Absicherungsprozess elementar sind. Man ist sowohl bei der Modularisierung und Hierarchisierung als auch bei der Auslegung jedes Funktionsmoduls auf Anforderungen angewiesen. Zum einen begründet sich die mechatronische Strukturierung auf den Anforderungen. Zum anderen beeinflussen die konzeptionellen Entscheidungen zur Strukturierung aber auch die Anforderungen durch Einschränkungen oder neue Ausprägungen. Gleichzeitig gilt, dass die Komplexität der Anforderungen mit der Komplexität des Systems steigt. Dadurch entstehen Zusammenhänge und Abhängigkeiten, die insbesondere bei interdisziplinären Entwicklungstätigkeiten kaum überschaubar sind. Treten zusätzlich dynamische Anforderungen auf, ergibt sich bei klassischem Lastenheft-basierten Anforderungsmanagement eine Quelle für schwerwiegende Fehler.

Obwohl beide Entwurfsmethoden in ihrer jeweiligen Domäne etabliert sind, besitzen sie Schwächen in der

Schnittstelle zwischen dynamischen Anforderungen und Systemstruktur. Da beide Methoden diese Schnittstelle bedienen, ist es zielführend sie miteinander zu verknüpfen. Dadurch entsteht eine ganzheitliche und durchgängige Entwurf Methodik für komplexe mechatronische Systeme im digitalisierten und vernetzten Umfeld.

2.2 Konzept und Anforderungen

Das MBRE eines Systems lässt sich aus zwei Perspektiven umsetzen: Struktur- und Funktionsperspektive. In der Strukturperspektive werden strukturelle Zusammenhänge und Abhängigkeiten des Systems bzw. dessen Anforderungen aufgeführt. Somit bildet die Strukturperspektive die erste Schnittstelle zur mechatronischen Strukturierung im mechatronischen Entwurf. Um die dynamischen Anforderungen und ihre Auswirkungen auf die mechatronische Strukturierung zu verknüpfen, bedarf es einer gemeinsamen Darstellungs- / Betrachtungsform.

In den fortgeschrittenen Phasen des mechatronischen Entwurfs werden die funktionalen Schnittstellen und Zusammenhänge einzelner Funktionsmodule festgelegt. Die sogenannte Funktionsstruktur beschreibt dabei, welche Ein- und Ausgangsgrößen das zu entwickelnde Modul besitzt und wie es mit anderen Modulen wechselwirkt. Aus der Funktionsperspektive des MBRE adressieren die Funktionsmodule auf höherer Ebene das gewünschte Verhalten des SOI. Gleichzeitig werden auch Anforderungen an die Funktionsmodule und deren Schnittstellen abgebildet. Zum einen bilden diese Anforderungen im Rahmen der mechatronischen Komposition die Grundlage für den Detailentwurf. Zum anderen dienen sie als Vergleichsmaß bei der Absicherung.

Die Kopplung des MBRE mit der mechatronischen Entwurfsmethodik soll Durchgängigkeit infolge einer ganzheitlichen Betrachtung von Anforderungen, Struktur, Funktionen und Absicherung des SOI schaffen. Um die Durchgängigkeit zu gewährleisten, ist es erforderlich diese Zusammenhänge in einem Systemmodell abzubilden und zu verknüpfen. Die Implementierung muss dabei so erfolgen, dass Änderungen von Anforderungen und Struktur möglich sind. Die Auswirkungen, der dynamischen Anforderungen des Gesamtsystems oder einzelner Systembestandteile auf ihre jeweilige Umgebung, müssen ebenfalls abgebildet werden. Das bedeutet, dass Änderungen bzw. Änderungsketten mithilfe des Systemmodells über die gesamte System- und Funktionsstruktur mindestens nachvollziehbar sein müssen. So können Ent-

wicklungspartner, die an demselben oder einem angrenzenden Systembestandteil arbeiten zur Fehlervermeidung informiert werden. Idealerweise lassen sich die Änderungen nicht nur nachverfolgen, sondern durch eine parametrische Implementierung der Anforderungen automatisch an die angrenzenden, abhängigen oder betroffenen Systembestandteile weitergeben. Somit werden Entwicklungspartner nicht nur über Änderungen informiert, sondern es erfolgt eine automatische Aktualisierung der eigenen Anforderungen. Dies erfordert allerdings die Implementierung einer entsprechenden Reglementierung für die Weitergabe im Systemmodell. Im Rahmen der Reglementierung ist auch eine Konsistenzprüfung und Priorisierung der Anforderungen sinnvoll.

2.3 MBRE-basierte mechatronische Entwurfsmethodik

Ausgehend vom Stand des Wissens (Kapitel 1) sowie dem Konzept und den Anforderungen (Abschnitt 2.2), wird in diesem Abschnitt die MBRE-basierte mechatronische Entwurfsmethodik (Abbildung 2) für die modellbasierte Entwicklung mechatronischer Systeme im digitalisierten und vernetzten Umfeld vorgestellt.

Für Anforderungen und Struktur gilt allgemein, dass sich die im Laufe der Entwurfsphase zunächst vom Abstrakten zum Konkreten entwickeln. Während des Detailentwurfs und der anschließenden Absicherung im Rahmen der mechatronischen Komposition werden einzelne Systembestandteile sukzessive zum Gesamtsystem integriert, sodass der Detaillierungsgrad wieder sinkt. Ausgangspunkt sind in jedem Entwicklungsprozess die Kundenbedürfnisse auf der obersten Abstraktionsebene.

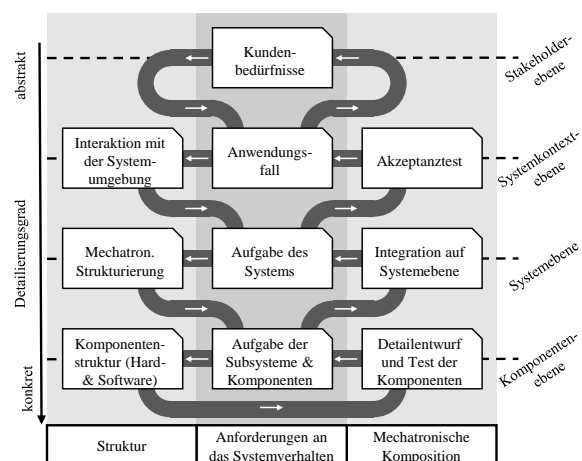


Abbildung 2: Schematische Darstellung der MBRE-basierten mechatronischen Entwurfsmethodik

Diese stehen hier stellvertretend für sämtliche Stakeholder-Anforderungen. Die Stakeholderebene beschreibt folglich nicht nur Nutzerbedürfnisse, sondern auch übergeordnete Anforderungen, wie Kosten, Qualität, Nachhaltigkeit, usw. Unter weiterer Betrachtung von Normen, gesetzlichen Richtlinien oder auch Erfahrungswerten aus bestehenden Systemen, lassen sich Anwendungsfälle definieren, die auf das gewünschte Verhalten des SOI auf der Systemkontextebene abzielen. Daraus lassen sich Art, Gegenstand und Wirkung der Interaktion des SOI mit seiner Umgebung ableiten. Auf Grundlage dessen bestimmt sich die konkrete Aufgabe des SOI, die in Form von Anforderungen auf der Systemebene beschrieben wird. Hieraus kann nun wiederum eine hierarchische mechatronische Strukturierung (Abbildung 1) für das SOI abgeleitet werden. Anschließend müssen auf der Komponentenebene Anforderungen für jedes gekapselte Funktionsmodul definiert werden. Daraus entsteht eine Funktionsstruktur, die beschreibt welche Schnittstellen das zu entwickelnde Modul besitzt und wie es mit anderen Modulen wechselwirkt. Für Softwarekomponenten und Funktionen beinhaltet das logische und physikalische Zusammenhänge der intelligenten Informationsverarbeitung. Bei Hardwarekomponenten sind beispielsweise kinematische und geometrische Wirkungsketten sowie physische Aspekte von Bedeutung. Auf Basis dieser Komponentenstrukturen, die Top-Down aus dem Wechselspiel von Anforderungen und Strukturen abgeleitet wurden, erfolgen Detailentwurf und Test der Hard- und Softwarekomponenten im Rahmen der mechatronischen Komposition. Das Entwicklungsergebnis wird gegen die Anforderungen auf der Komponentenebene abgesichert. Ist der Mindesterfüllungsgrad der Anforderungen erreicht, kann das System sukzessive und Bottom-Up zum Gesamtsystem integriert und auf Systemebene validiert werden. Der Absicherungsprozess im Rahmen der MBRE-basierten mechatronischen Entwurfsmethodik ist iterativ. Bleiben gestellte oder abgeleitete Anforderungen unerfüllt, müssen die Mängel durch einen erneuten Durchlauf der Methodik ab dem betreffenden Punkt korrigiert werden. Erfüllt das SOI auf der Systemebene die definierten Anforderungen kann ein Akzeptanztest für die zu Beginn definierten Anwendungsfälle des SOI in seiner Umgebung stattfinden. Hieraus lassen sich Rückschlüsse auf den Erfüllungsgrad der Kunden- bzw. Stakeholderbedürfnisse und etwaige Konsequenzen ziehen.

Die Durchgängigkeit der MBRE-basierten mechatronischen Entwurfsmethodik wird durch die Pfeile in Ab-

bildung 2 verdeutlicht. Ermöglicht wird sie durch die logische und modellbasierte Vorgehensweise mit der softwaretechnischen Umsetzung im Systemmodell. Die Entwurfsmethodik illustriert und gewährleistet die gegenseitige Abhängigkeit und Beeinflussung von Anforderungen, Struktur und Funktion des SOI. Aufgrund der Verknüpfung dieser Perspektiven und Systembestandteile, wird auch das automatische Übergeben und Anpassen dynamischer Anforderungen und Strukturen abgebildet. Voraussetzung hierfür ist eine entsprechende parametrische Implementierung in SysML. Darüber hinaus sind Ansätze zur Toolkopplung von SysML und den konkreten Entwicklungstools denkbar. So können im Systemmodell dokumentierte und verwaltete Anforderungen, Strukturen, Parameter oder Ergebnisse toolübergreifend genutzt werden.

3 Validierung anhand der konfliktfreien Trajektorienplanung

3.1 Beschreibung des Anwendungsbeispiels

Die hergeleitete ganzheitliche Entwurfsmethodik für die modellbasierte Entwicklung mechatronischer Systeme im digitalisierten und vernetzten Umfeld, soll anhand eines Anwendungsbeispiels exemplarisch demonstriert und validiert werden. Das Beispiel stammt aus dem Bereich der Fertigungsindustrie. Der Schlüssel zum Erhalt der Wettbewerbsfähigkeit sind neue I4.0-Technologien zur Flexibilitätssteigerung. [9] Fahrerlose Transportfahrzeuge (FTF) sind essentielle Bestandteile einer intelligenten Fabrik und müssen ebenfalls zur Erfüllung der Flexibilitätsanforderungen beitragen. Daher wird an der Ostfalia Hochschule ein intelligentes, autonom agierendes und bewegungsflexibles FTF entwickelt und die zugehörigen Funktionen modellbasiert ausgelegt.

So auch eine Funktion zur konfliktfreien, selbstoptimierten Trajektorienplanung, welche in [10] detailliert beschrieben wird. In diesem Beitrag soll der Entwurfsprozess anhand der vorgestellten Entwurfsmethodik vom Anforderungsmanagement bis hin zur Validierung beschrieben werden.

3.2 Anwendung der MBRE-basierten mechatronischen Entwurfsmethodik

Ausgangspunkt des Anwendungsbeispiels ist die Stakeholderebene, auf der übergeordnete, abstrakte Wünsche und Bedürfnisse der Fabrikleitung, wie der Erhalt der

Wettbewerbsfähigkeit, Flexibilität oder die Verbesserung des logistischen Flusses liegen. Denkbare Anwendungsfälle die sich hierfür definieren lassen, zielen auf ein flexibles Produktionssystem mit hohem Automatisierungsgrad ab. Neben reinen Fertigungstätigkeiten ist hierfür auch das Intralogistikmanagement entscheidend. Aus den Anwendungsfällen werden die Interaktionen des FTF als SOI, mit seiner Umgebung, z.B. Maschinen, Lagersystemen oder dem internen Verkehrssystem deutlich. Hieraus lassen sich Anforderungen ableiten und neu definieren. Beispielsweise muss sich das FTF an die zulässige Höchstgeschwindigkeit von 1 m/s halten. Konstruktionstechnische Anforderungen ergeben sich zusätzlich z.B. aus dem Grundriss sowie der verfügbaren Spurbreite von 0,5 m auf den Fabrikstraßen oder auch der Geometrie der Transportgüter. Des Weiteren müssen drahtlose Kommunikationstechnologien und -standards vorgesehen werden, damit das FTF mit seiner Umgebung, beispielsweise anderen FTF, dem Intralogistik-, Warenhaus- oder Instandhaltungsmanagement kommunizieren kann. Aus der Rolle des FTF in seinem Systemkontext ergeben sich im nächsten Schritt Aufgaben bzw. Anforderungen auf der Systemebene. Einige dieser Anforderungen, wie das Einhalten zulässiger Geschwindigkeiten oder die Abmaße des FTF ($B \times L = 0,2 \text{ m} \times 0,3 \text{ m}$) werden direkt Top-Down abgeleitet. Andere ergeben sich indirekt durch Vorgaben oder Restriktionen auf Systemkontextebene. Um beispielsweise den Anforderungen an Flexibilität in der engen Produktionsumgebung gerecht zu

werden, muss sich das FTF losgelöst von vorgegebenen Spuren und omnidirektional bewegen können. Daraus wiederum resultiert, dass das FTF zur selbstständigen Navigation z.B. eine Trajektorienplanung benötigt. Des Weiteren ist es möglich auf der Systemebene neue Anforderungen zu definieren. Beispielsweise soll das FTF Kollisionen mit anderen FTF nicht erst lokal, sondern bereits während der Routenplanung mit einer Konflikterkennungs- und Lösungsfunktion vermeiden. Führt man dieses Vorgehen fort und ordnet die sich ergebenden Anforderungen und Systembestandteile hierarchisch an, entsteht die mechatronische Strukturierung. Diese wurde gemeinsam mit den parametrischen dynamischen Anforderungen in SysML implementiert und ist exemplarisch in Abbildung 3 dargestellt.

Die SysML ermöglicht die Darstellung von Struktur- und Verhaltensdiagrammen, wie den Blockdefinitionsdiagrammen aus Abbildung 3. Sie bietet darüber hinaus den Diagrammtyp der Anforderungsdiagramme. Das Anforderungsdiagramm wird zunächst als separates Diagramm erstellt und anschließend den Strukturebenen und -elementen zugeordnet. Bei der Beziehung der Anforderungen untereinander besteht ein Unterschied, ob sich beispielsweise eine Systemanforderung von einer Systemkontextanforderung ableiten lässt (*derive*) oder eine Anforderung sich aus mehreren Anforderungen zusammensetzt (*containment*). Um die Beziehung zwischen einem Strukturelement und einer Anforderung zu beschrei-

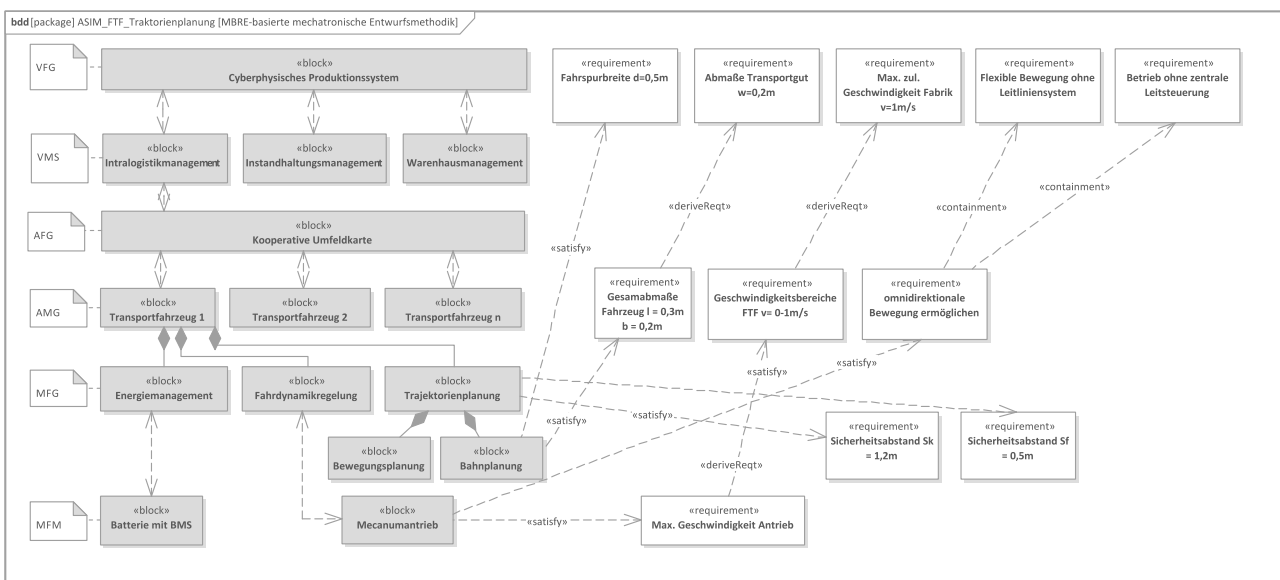


Abbildung 3: Exemplarisches Blockdefinitionsdiagramm der mechatronischen Strukturierung des FTF im CPS mit den verknüpften Anforderungen in SysML

ben, existiert die “*satisfy*”-Beziehung. Zur Verdeutlichung dieser Zusammenhänge, wurde eine Auswahl der Anforderungen direkt in das Blockdefinitionsdiagramm in Abbildung 3 implementiert.

Anschließend werden Anforderungen auf Komponentenebene abgeleitet und definiert. Da sich dieses Anwendungsbeispiel mit der konfliktfreien Trajektorienplanung befasst, wird diese exemplarisch vorgestellt. Um eine Kollision in einem Kreuzungskonflikt zu vermeiden, muss beispielsweise ein Sicherheitsabstand S zwischen den FTF eingehalten werden. Dieser berechnet sich zu $S = f(v_{max}, G, d)$ in Abhängigkeit der maximalen Geschwindigkeit v_{max} der FTF, dessen Geometrie G und der verfügbaren Fahrspurbreite d .

Mit den restlichen Informationen aus dem Systemmodell kann man für das Modul Trajektorienplanung die in Abbildung 4 dargestellte Funktionsstruktur aufstellen, die alle Wirkungszusammenhänge, Schnittstellen und das Verhalten dieser Funktion vom Erhalt eines Auftrags, über die Zielführung, Trajektorienberechnung und Konfliktlösung bis hin zur Fahrdynamikregelung beschreibt. Nach Abbildung 2 folgt nun der Detailentwurf und Test der Komponente bzw. die sukzessive Integration.

Die modellbasierte Auslegung des Funktionsmoduls zur Trajektorienplanung sowie die Simulationsergebnisse bei dynamischen Anforderungen werden im folgenden Abschnitt vorgestellt.

3.3 Auswertung der Simulationsergebnisse

Die Funktion zur Trajektorienplanung wurde auf Basis der hergeleiteten Funktionsstruktur nach Abbildung 4 unter Anwendung der in diesem Beitrag erarbeiteten MBRE-basierten mechatronische Entwurfsmethodik ausgelegt. Die Auswertung der Simulationsergebnisse soll den durchgängigen Charakter der Entwurfsmethodik und deren Vorteil im Umgang mit dynamischen Anforderungen exemplarisch zeigen. Aus den Anforderungen für Kreuzungskonflikte ergibt sich für die ursprüngliche Konfiguration von v_{max} , G und d ein Mindestsicherheitsabstand von $S = 1,2$ m. Ein solcher Konflikt wird gelöst,

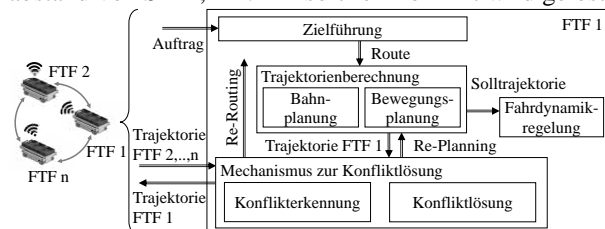


Abbildung 4: Funktionsstruktur der konfliktfreien Trajektorienplanung

indem das FTF mit dem niedriger priorisierten Auftrag seine Geschwindigkeit reduziert. Abbildung 5 zeigt ein Simulationsergebnis, bei dem ein Kreuzungskonflikt vorliegt. Das FTF1 hat in diesem Fall seine Geschwindigkeit so reduziert, dass der kleinste Abstand zwischen den FTF 1,4 m beträgt. Das bedeutet, dass die Funktion so ausgelegt wurde, dass die Anforderung an den Mindestabstand erfüllt wurde.

Nun wird angenommen, dass die Fabrikleitung die zulässige Geschwindigkeit aus Sicherheitsgründen auf 0,6 m/s reduziert. Durch die parametrische Implementierung der dynamischen Anforderungen im Systemmodell, wird diese Änderung automatisch an die unterlagerten und verknüpften Anforderungen und Systembestandteile weitergeleitet. Aufgrund dessen beträgt S nun nur noch 0,6 m. Die Kopplung dieser Anforderung mit den Parametern des modellbasierten Detailentwurfs erlaubt eine einfache Aktualisierung der Funktion. Ein entsprechendes Simulationsergebnis ist in Abbildung 6 illustriert. Es ist zu sehen, dass das FTF1 seine Geschwindigkeit um ein geringeres Maß reduziert. Der Mindestabstand liegt mit 0,7 m immer noch über dem geforderten Wert.

Dieses Anwendungsbeispiel demonstriert den ganzheitlichen und durchgängigen Charakter der MBRE-basierten Entwurfsmethodik. Es zeigt, dass durch die Kopplung von MBRE und mechatronischer Entwurfsmethodik ein echter Mehrwert entsteht, der aus den Abhängigkeiten und Wechselwirkungen von Anforderungen, Struktur und Funktionen eines Systems resultiert. Die Sinnhaftigkeit und Funktionsfähigkeit dieses Ansatzes wurde in dieser Anwendung von abstrakten übergeordneten Anforderungen bis zum Detailentwurf und Test validiert.

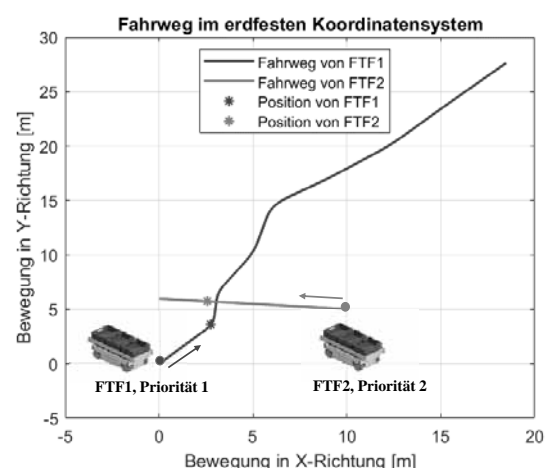


Abbildung 5: Simulation eines Kreuzungskonflikts mit $v_{max}=1$ m/s und $S=1,2$ m

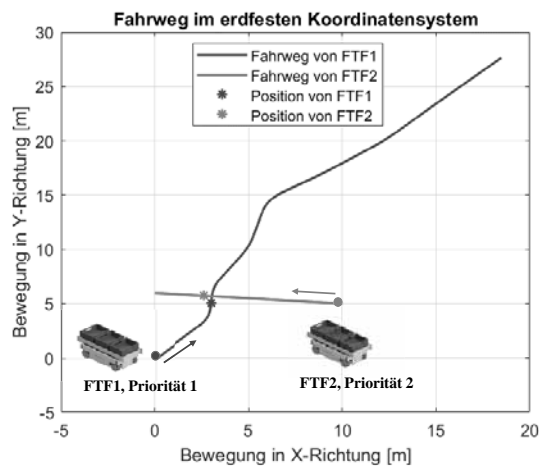


Abbildung 6: Simulation eines Kreuzungskonflikts mit $v_{\max}=0,6 \text{ m/s}$ und $S=0,6 \text{ m}$

4 Zusammenfassung und Ausblick

Anforderungsmanagement und mechatronischer Entwurf bieten unterschiedliche Perspektiven und Schwerpunkte, sind aber beide essentiell für die Entwicklung komplexer Systeme. Das MBRE beschreibt Systemzusammenhänge und Anforderungen auf einer übergeordneten Metaebene, wohingegen der mechatronische Entwurf funktionale und strukturelle Wechselwirkungen des Systems fokussiert. Die MBRE-basierte mechatronische Entwurfsmethodik ermöglicht eine ganzheitliche Betrachtung von Anforderungen, Struktur, Funktionen und Absicherung des SOI. Aufgrund der Verknüpfung dieser Perspektiven und Systembestandteile, wird auch das automatische Übergeben und Anpassen dynamischer Anforderungen und Strukturen unterstützt.

Anhand der konfliktfreien Trajektorienplanung konnte das Vorgehen aufgezeigt und die Durchgängigkeit und Funktionsfähigkeit der Methodik mit Simulationen demonstriert und abgesichert werden.

Ausblickend sind weitere Arbeiten zur Kopplung von SysML-basierten und beispielsweise numerischen Entwicklungstools denkbar. Somit ließe sich eine interdisziplinäre Entwicklungsumgebung realisieren, in der ausgehend von den im Systemmodell dokumentierten und verwalteten Anforderungen, auch Strukturen, Parameter oder Ergebnisse in tool-übergreifenden Anwendungen genutzt werden können.

5 Danksagung

Diese Publikation wurde gefördert durch:

- Europäischer Fond für regionale Entwicklung (EFRE): Verbundprojekt "autoMoVe" (*Dynamisch*

konfigurierbare Fahrzeugkonzepte für den nutzungsspezifischen autonomen Fahrbetrieb) (ZW 6-85030889)

- EFRE: Verbundprojekt "Synus" (*Methoden und Werkzeuge für die synergetische Konzipierung und Bewertung von Industrie 4.0-Lösungen*) (ZW 6-85012454)



Literatur

- [1] Czichos, H. Mechatronik. Springer Vieweg, Wiesbaden, 2019.
- [2] Huth T, Vietor T. Systems Engineering in der Produktentwicklung: Verständnis, Theorie und Praxis aus ingenieurwissenschaftlicher Sicht. *Gruppe. Interaktion. Organisation. Zeitschrift für Angewandte Organisationspsychologie (GIO)*, Jg. 51, Nr. 1, S. 125–130, 2020.
- [3] Haberfellner, R., Daenzer, W.F.: Systems Engineering: Methodik und Praxis. Verl. Industrielle Organisation, Zürich (1999)
- [4] Friedenthal, S. and Oster, C. Architecting spacecraft with SysML: A model-based systems engineering approach, CreateSpace Independent Pub. Platform, s.l., 2017.
- [5] Pohl K., Rupp C. Basiswissen Requirements Engineering, dpunkt.verlag 2015.
- [6] Gilz T. Requirements Engineering und Requirements Management in Modellbasierte virtuelle Produktentwicklung, S 59. Springer, Heidelberg 2014.
- [7] Liu-Henke X. *Mechatronische Entwicklung der aktiven Feder-Neigetechnik für das Schienenfahrzeug RailCab*. VDI Fortschritt-Berichte, Reihe 12, Verkehrstechnik / Fahrzeugtechnik, Nr. 589. Düsseldorf: VDI-Verlag 2005.
- [8] Liu-Henke X, Yarom OA, Scherler S. Virtual Development and Validation of a Function for an Automated Lateral Control using Artificial Neural Networks and Genetic Algorithms. *91st IEEE Vehicular Technology Conference (VTC2020-Spring)*, Antwerpen, Belgien, 25. – 28. Mai, 2020.
- [9] Bauernhansel T, Hompel M, Vogel-Heuser B, *Industrie 4.0 in Produktion, Automatisierung und Logistik*, Springer Vieweg, München, 2014.
- [10] Zhang J., Liu-Henke X. Konfliktfreie, selbstoptimierte Trajektorienplanung für ein fahrerloses Transportfahrzeug zur Durchführung des autonomen Gütertransportes im Produktionsumfeld, 25. Symposium Simulationstechnik – ASIM 2020 – Virtuelle Tagung, 14 -15. Oktober, 2020. (tbp)

Method for the computer-aided design and simulation of hydrogel-based microfluidic chips

Andreas Voigt^{1*}, Jörg Schreiter², Christian Mayr², Andreas Richter¹

¹Chair of Microsystems, Institute of Semiconductors and Microsystems, Faculty of Electrical and Computer Engineering, Technische Universität Dresden, Dresden, Germany, *andreas.voigt@tu-dresden.de

²Chair of Highly-Parallel VLSI-Systems and Neuromorphic Circuits, Institute of Circuits and Systems, Faculty of Electrical and Computer Engineering, Technische Universität Dresden, Dresden, Germany

Abstract Microfluidic chips facilitate the manipulation of nanoliter fluid volumes in order to perform chemical analysis or synthesis and biological cell manipulation. On-chip valves are employed when complex flow control schemes are desired. Stimuli-sensitive hydrogels can be used to create transistor-like microfluidic valves whose opening or closing behavior is determined by the content of chemicals in the liquid. Here, we present a design and simulation method that simplifies the development of hydrogel-based microfluidic chips by reducing the amount of experiments that need to be performed in the lab. Cadence Virtuoso, a tool commonly used for electronic circuit design, is employed as a framework for the implementation of the method. Like in electronics, the microfluidic circuit consists of basic components (channels, valves, pumps) that can be placed and connected with each other in the schematic design interface. The physical-mathematical behavior of these components is implemented in VerilogAMS, a hardware description language. The method was successfully employed in the top-down design of a chemofluidic oscillator. In addition, parts of the design process of the photomasks for chip fabrication were automated by the use of parameterized cells and the SKILL programming language.

This article is a summary of the work presented in [1].

Introduction

Microfluidics is a scientific and engineering discipline that covers the behaviour of fluids (liquids and gases) at the micrometre scale. One endeavour in the field is the construction of labs-on-chips that perform automated chemical, biological or medical laboratory processes on a miniaturized chip. A lab-on-a-chip usually consists of a network of channels and chambers, and inlets and outlets for the liquids and reagents. In addition, the chip can contain microvalves in order to achieve more complicated liquid flow control mechanisms.

Hydrogels are porous polymers that, similar to a sponge, can suck up water and change their size accordingly. Some hydrogels are stimulus-sensitive: their swelling or shrinking depends on the physical or chemical properties of the surrounding liquid, e.g. temperature, pH value or

the presence of specific chemicals. Smart valves (“chemofluidic transistors”) can be built by placing a chemosensitive hydrogel in a chamber (figure 1) [2] [3], or by utilizing a membrane that closes an adjacent channel when the hydrogel swells (figure 2) [4]. These valves have the advantage that they do not require any external control and facilitate direct on-chip feedback mechanisms. Employing this internal feedback, chemofluidic transistors have been used to build a microfluidic oscillator [5] and microfluidic logic circuits [6] [7].

Computer-aided design has been a key factor contributing to the development of the advanced state of current microelectronics. Therefore, it seems natural to establish similar methods for microfluidics. Computer-aided microfluidic design, like computer-aided electronic design, can be roughly divided into two categories: Simulation of the physical processes on the chip, and automated drawing of the layout of the masks that are needed for the fabrication of the chip. One approach in simulation is the use of computational fluid dynamic tools that describe the fluid transport in three dimensions and utilize numerical discretisation, e.g. by the finite volume method. Due to the high amount of degrees of freedom, this comes at the cost of a high computational time. Therefore, simplified descriptions have been developed based on the electrohydraulic analogy (pressure corresponds to voltage, flowrate to current) and dimensional reduction [8].

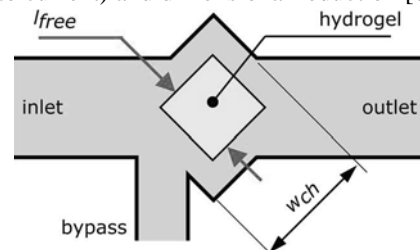


Figure 1: Planar chemofluidic transistor. The hydrogel in the chamber responds to the chemical content in the liquid by swelling or shrinking, leading to opening or blocking of the valve since water can only be transported in the small channels around the gel.

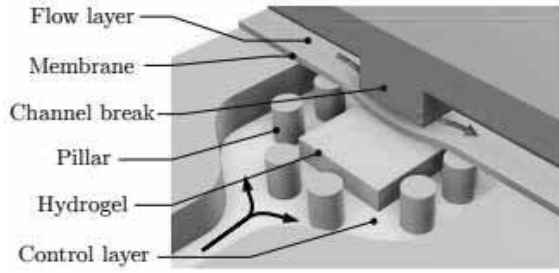


Figure 2: Chemofluidic transistor employing a membrane for the separation of flow layer and control layer. In the swollen state, the gel pushes the membrane against the channel break, blocking the flow in the flow layer. In the shrunken state of the gel, the fluid in the flow layer can pass under the channel break.

Microfluidic chips are then described as circuits of connected basic components, similar to electronic circuits. Another advantage, aside from the significant increase in simulation speed, is the visual representation of the chip functionality and the intuitive usability when designing microfluidic chips on the computer. Modelling languages like Modelica, Simscape or VerilogAMS can be employed for the mathematical description of the microfluidic components. In addition, a design and simulation framework supporting the according language is needed to draw the circuits and compute their behaviour. The Cadence Virtuoso framework with VerilogA was first used by Wang et.al [9] for microfluidic simulations. In this work, we also employ Cadence Virtuoso. However, as the modelling language we use VerilogAMS as it allows combined digital-analogue implementations of component models (AMS = analogue mixed signal) while VerilogA only provides analogue functionality. Compared to Wang et al., the main novelty of our work lies in the time-dependent simulation of chemical transport, in the inclusion of chemofluidic transistors, and in the fact that our method was successfully employed in the top-down design of a chemofluidic oscillator [5].

Microfluidic layout automation has been largely driven by the goal to achieve “full automation”, the generation of the complete mask layout from an abstract description of the functionality of the chip, e.g. in the Columba 2.0 method by Tseng et al. [10]. However, a survey among microfluidic chip designer conducted by McDaniel et al. [11] indicated that the designers prefer to have more control during all steps of the design process instead of using push-button solutions. Therefore, “semi-automation” of certain steps during layout design seems to be a more adequate approach at the current status of lab-on-a-chip technology. In this line, we have employed the parameterized cells (pCells) and the SKILL language of Cadence

Virtuoso to automate small repetitive and cumbersome steps in the layout process.

1 Principles and Design Environment

Like in electronics, to facilitate a schematic circuit representation of a microfluidic chip, a dimensional reduction (1d representation) of the components has to be performed (figure 3). In microchannels, only the effect of the longitudinal direction is taken into account and the pressures are considered constant along the whole cross-section of the channel. Therefore, the channel can be regarded as a component with two terminals. Similarly, all other components are characterized as blocks with terminals, with the inner behaviour of the block described by a VerilogAMS file.

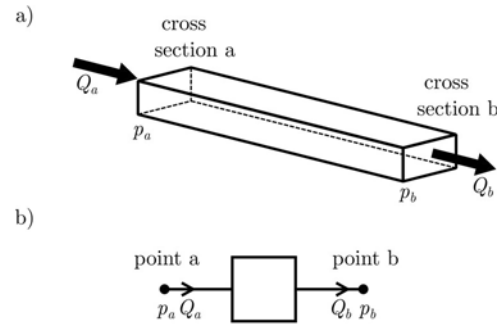


Figure 3: 1d approximation. Microchannels are treated as components with two terminals, each with an applied pressure and a corresponding flowrate.

One main difference between electronics and microfluidics are the chemicals transported in the flow. Therefore, in addition to the pressure-flowrate pair, chemical concentrations need to be implemented in the simulation. We have included chemical concentration as a signal-flow quantity, a feature offered by VerilogAMS and Cadence Virtuoso.

Figure 4 shows a microfluidic schematic. The terminals of components always consist of one input port for the incoming chemical concentration, one output port for the outgoing chemical concentration and one input/output port for the pressure-flowrate pair. Channels are components on their own. In contrast to electronics, the thin “wires” only indicate connectivity and do not correspond to physical entities.

The circuits are entered using the schematic editor of the Cadence Virtuoso design framework. Component models have free parameters that are set when an instance of the component is created. The time-dependent simulation of a circuit is performed by calling the Spectre simulator.

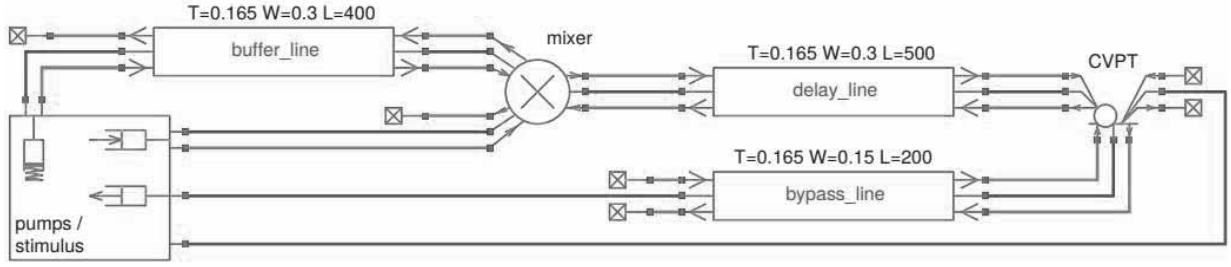


Figure 4: Schematic of the chemofluidic oscillator consisting of pumps, channels (delay line, bypass line and buffer line), a mixer and a chemofluidic transistor (CVPT). The thin lines show the connection of the components and do not correspond to real objects. Blue lines indicate the fluidic quantities (pressure, flowrate); purple lines indicate incoming or outgoing chemical concentrations. Buffer line, delay line and bypass line are all instances of the same component (a microchannel) but with different component parameters.

2 Components

Aside from water (which acts as the solvent), the current component models describe the transport of one single chemical species. In the case of the chemofluidic oscillator this species is propan-1-ol, acting as a stimulus for the employed hydrogel.

2.1 Channels

Microfluidic channels (figure 5) have two effects on fluid transport. First, they act as resistors with the hydrodynamic resistance R_h depending on the viscosity of the fluid and the channel dimensions (length, height, width). When a pressure difference p is applied at the two terminals of the channel, a flow rate of

$$Q = \frac{p}{R_h} \quad (1)$$

results. The second effect is the mechanism by which concentration is transported through the channel. In our model, we neglect diffusion and only account for convection, i.e. the shifting of the position of chemical concentration due to the movement of the fluid.

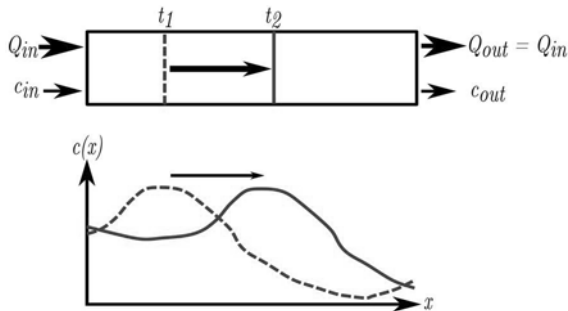


Figure 5: Microfluidic channel. There is an incoming chemical concentration c_{in} at the inlet and an outgoing concentration c_{out} at the outlet. The concentration is shifted along the channel in time due to the flow.

In the VerilogAMS model, a combination of analogue and digital modelling is used to account for the two effects. The analogue part describes the resistive behaviour of the channel while the digital part is employed for discretizing the concentration values and for bookkeeping of the internal state of the concentration distribution inside the channel.

2.2 Mixing Junction

In a mixing junction (figure 6), two incoming flows of flowrate Q_{in1} and Q_{in2} carrying concentrations c_{in1} and c_{in2} are joined. We assume that the outgoing channel is long enough to achieve complete mixing. The outgoing flowrate is simply the sum $Q_{in1} + Q_{in2}$ of the incoming flowrates and the resulting concentration is calculated by a weighted average:

$$c_{out} = \frac{Q_{in1}c_{in1} + Q_{in2}c_{in2}}{Q_{in1} + Q_{in2}}. \quad (2)$$

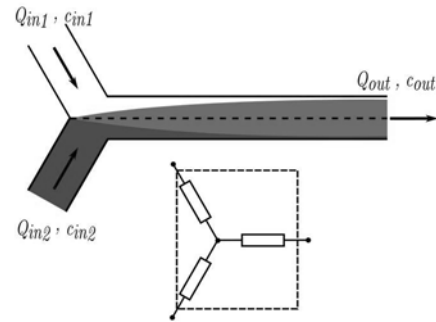


Figure 6: Mixing junction. Two incoming flows of flowrate Q_{in1} and Q_{in2} carrying concentrations c_{in1} and c_{in2} are mixed. The outlet delivers the resulting flowrate Q_{out} and concentration c_{out} .

2.3 Chemofluidic Transistor

Here, we focus on the planar chemofluidic transistor (figure 1). The gel in the chamber acts as a hydrodynamic resistor, where the resistance depends on the chamber width w_{ch} and the current size of the gel. For a free

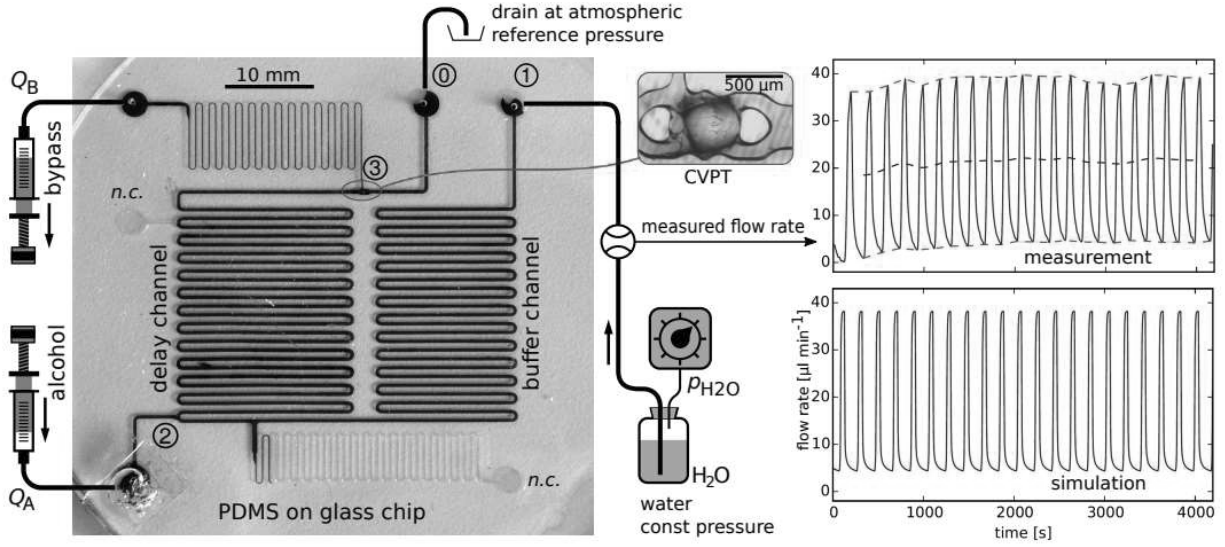


Figure 7: The chemofluidic oscillator is operated by a constant pressure source, by two constant flowrate sources and shows oscillations of flowrates and chemical concentration. This is achieved by the combination of a chemofluidic transistor (CVPT) and a delay channel, providing time-delayed negative feedback. The plots show the measured and the simulated flowrate at the inlet of the buffer channel.

swelling gel (outside of a chamber), the equilibrium size $l_{\text{eq}}(c)$ depends on the concentration of the surrounding liquid (figure 8).

If the current size l_{free} deviates from this value, the gel will swell or shrink according to a first order differential equation

$$\frac{dl_{\text{free}}}{dt} = \gamma(c) \cdot (l_{\text{eq}}(c) - l_{\text{free}}) \cdot K(l_{\text{free}}), \quad (3)$$

where γ is the concentration-dependent swelling rate and K is a size-dependent correction factor that only significantly deviates from 1 for large gel sizes.

For the gel in the chamber we keep track of the current l_{free} and calculate the hydrodynamic resistance accordingly, where the transistor is blocked for $l_{\text{free}} \geq w_{\text{ch}}$.

The bypass of the chemofluidic transistor is needed to allow the transport of fluid (and hence a chemical signal) to the gel, when the transistor is in its blocked state.

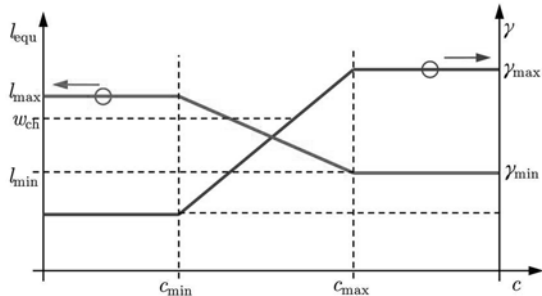


Figure 8: Concentration-dependent equilibrium size l_{free} and swelling rate γ of the hydrogel. The linear piece-wise graphs approximate the experimental behaviour.

3 Example System: The Chemofluidic Oscillator

The chemofluidic oscillator (figures 4 & 7) is a microfluidic circuit that is driven by constant fluidic sources and shows periodic oscillations in flowrate and chemical concentration [5]. At the heart of the circuit is a long microfluidic channel (delay channel) leading to a planar chemofluidic transistor (3). The delay channel is fed by a constant flowrate source providing a propan-1-ol solution (2) and a constant pressure source providing pure water (1). Depending on the state of the chemofluidic transistor either the propan-1-ol source or the pure water source dominates. When a solution with high propan-1-ol concentration reaches the transistor this will lead to a shrinking of the gel, which in turn will result in a solution with low propan-1-ol concentration being fed into the delay line at (2). After a certain delay time this low concentration will reach the hydrogel, leading to a blocking of the transistor and therefore a solution of high propan-1-ol concentration being fed into the delay channel. After a delay time this high concentration will reach the hydrogel, starting a new period of the oscillation.

The chemofluidic oscillator was first conceived conceptually. Then, simulations were used to determine geometric and operational parameters that would lead to a safe oscillation regime. With the determined parameters, the

mask was drawn for fabrication, and the fabricated microfluidic chip was put into operation. Figure 7 shows the measured and the simulated flowrate at a characteristic position on the chip.

4 Towards Layout Semi-automation

As noted by McDaniel et al. [11], microfluidic chip designers prefer to have full control of the design process and, at the current state of the technology, do not trust fully or highly automated layout solutions. Instead, it seems advisable to support them in cumbersome and repetitive day-to-day routines. Along this line, we have developed first steps towards layout semi-automation to alleviate time-consuming processes we encountered during chip design.

The parametric cells (pCells) of Cadence Virtuoso provide the function to define generic layout definitions with a pre-defined shape but with a number of free parameters. In instantiating a pCell, the parameters have to be set and the accordingly generated layout block can then be freely placed. This functionality proved especially helpful for the generation of meanders (figure 9) where the channel width, pitch, curvature, the length of the straight lines, the number of meanders, the entrance length and the exit length are set as free parameters.

Additional functionality is provided when augmenting the functionality of pCells by the use of the SKILL programming language. This facilitates the construction of complex, regular (e.g. highly serialized or parallelized) microfluidic structures in lieu of drawing the layout manually [1].

5 Summary and Outlook

Our methods allows the design of microfluidic chips employing chemofluidic transistors by placing and routing instances of basic components. The components are modelled by the VerilogAMS hardware description language that provides the flexibility of implementing both analogue and digital behaviour. The simulation shows the time-dependent fluidic quantities (pressure, flowrate), concentration transport and internal states of the components. The method proved useful in the top-down design and parametric dimensioning of a chemofluidic oscillator. In addition, first small steps were taken towards a semi-automation of the layout process by parametric cells.

In the future, we will enhance the component library by

including the single-use valves developed in [12]. In addition, it might be worthwhile to investigate whether our method can be transferred to other microfluidic platform technologies, e.g. based on pneumatic valves [13]. For the layout automation, in addition to “little helper” pCells or SKILL scripts, it seems valuable to pursue a closer coupling of the schematic design to the layout process, with the final goal of generating the mask layout directly from the circuit sketch.

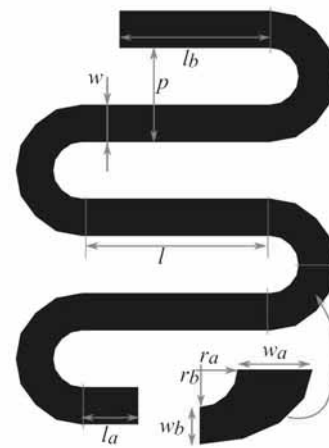


Figure 9: Layout of a parameterized meander channel pCell. The pCell can be instantiated with a chosen set of parameters, providing a flexible and quick meander drawing routine.

References

- [1] A. Voigt, J. Schreiter, P. Frank, C. Pini, C. Mayr, A. Richter, *Method for the Computer-aided Schematic Design and Simulation of Hydrogel-based Microfluidic Systems*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 39, no. 8, pp. 1635–1648, Aug. 2020. DOI: 10.1109/TCAD.2019.2925354
- [2] D. J. Beebe, J. S. Moore, J. M. Bauer, Q. Yu, R. H. Liu, C. Devadoss, B.-H. Jo, *Functional hydrogel structures for autonomous flow control inside microfluidic channels*, Nature, vol. 404, no. 6778, pp. 588–590, Apr. 2000.
- [3] K.-F. Arndt, D. Kuckling, A. Richter, *Application of sensitive hydrogels in flow control*, Polymers for Advanced Technologies, vol. 11, nos. 8–12, pp. 496–505, Aug. 2000.
- [4] P. Frank, D. Gräfe, C. Probst, S. Haefner, M. Elstner, D. Appelhans, D. Kohlheyer, B. Voit, and A. Richter, *Autonomous Integrated Microfluidic Circuits for Chip-Level Flow Control Utilizing Chemofluidic Transistors*, Advanced Functional Materials, vol. 27, no. 30, p. 1700430,

Aug. 2017.

- [5] G. Paschew, J. Schreiter, A. Voigt, C. Pini, J. P. Chávez, M. Allerdissen, U. Marschner, S. Siegmund, R. Schüffny, F. Jüllicher, and A. Richter, *Autonomous Chemical Oscillator Circuit Based on Bidirectional Chemical-Microfluidic Coupling*, Advanced Materials Technologies, vol. 1, no. 1, p. 1600005, Apr. 2016..
- [6] A. Voigt, R. Greiner, M. Allerdissen, A. Richter, S. Henker, M. Völz, *Towards computation with microchemomechanical systems*, International Journal of Foundations of Computer Science, vol. 25, no. 4, pp. 507–523, Jun. 2014.
- [7] P. Frank, D. Gräfe, C. Probst, S. Haefner, M. Elstner, D. Appelhans, D. Kohlheyer, B. Voit, A. Richter, *Autonomous integrated microfluidic circuits for chip-level flow control utilizing chemofluidic transistors*, Advanced Functional Materials, vol. 27, no. 30, p. 1700430., Aug. 2017
- [8] R. Zengerle, M. Richter, *Simulation of microfluid systems*, Journal of Micromechanics and Microengineering, vol. 4, no. 4, pp. 192–204, Dec. 1994.
- [9] Y. Wang, Q. Lin, T. Mukherjee, *Composable Behavioral Models and Schematic-Based Simulation of Electrokinetic Lab-on-a-Chip Systems*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 2, pp. 258–273, Feb. 2006.
- [10] T. Tseng, M. Li, D. N. Freitas, T. McAuley, B. Li, T. Ho, I. E. Araci, and U. Schlichtmann, *Columba 2.0: A Co-Layout Synthesis Tool for Continuous-Flow Microfluidic Biochips*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 8, pp. 1588–1601, Aug. 2018.
- [11] J. McDaniel, W. H. Grover, and P. Brisk, *The case for semi-automated design of microfluidic very large scale integration (mVLSI) chips*, in Design, Automation & Test in Europe Conference Exhibition (DATE), 2017, Mar. 2017, pp. 1793–1798.
- [12] R. Greiner, M. Allerdissen, A. Voigt, A. Richter, *Fluidic microchemomechanical integrated circuits processing chemical information*, Lab on a Chip, vol. 12, no. 23, 5034-5044, 2012
- [13] T. Thorsen, S. J. Maerkl, S. R. Quake, *Microfluidic large-scale integration*, Science, vol. 298, no. 5593, pp. 580–584, Oct. 2002.

Simulation Study on Various Double Pendulum Configurations

Milena Sipovac^{1*}, Stefanie Winkler¹, Andreas Körner¹

¹Institute of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8–10, 1040 Vienna, Austria; *milena.sipovac@tuwien.ac.at

Abstract. This paper offers an introductory overview of some methods used for first principle modeling of different structures for the double pendulum. The mathematical models are based on the Euler-Lagrange equations. The basic simulation study is focused on the planar double pendulum. Chaotic and periodic behavior is investigated, together with an influence of the external force. For initial conditions close to zero a periodic motion is observable. This model is further extended to a simplified model of a church bell with a double clapper. The clapper is modeled as a double pendulum with limitation of movement inside the sides of the bell. Periodicity under different initial conditions is investigated.

Motivation

The pendulum, together with its variations, is a simple mechanical system that is suitable for applying white-box modeling techniques, since we know a lot about the system and its movement. In this paper, we will also extend the simple pendulum to a double pendulum. The planar double pendulum is a mechanical system that shows chaotic behavior even though it is a simple system, but its simple construction allows us to observe and understand the behavior of the chaotic systems more closely. We are observing a deterministic system, which means it is defined by its initial conditions, there are no random elements involved, yet because of its chaotic nature, it is still not predictable. The definition of *chaos* used here is the sensitivity to the change in initial condition. In this paper, we concentrate on the distinction between chaotic and periodic behavior of the system and the initial conditions that influence this, as well as on influence of an external force to its behavior.

The second model observed in this paper is a model of a simplified church bell, with a double pendulum as a clapper. Its purpose is a demonstration of the law of conservation of momentum, since the energy is trans-

ferred and distributed between the bell sides and the clapper. It is implemented through modeling the sides of the bell as a simple pendulum, and they are acting as a moving constraint to a double pendulum. The possibility of rhythmic motion is investigated by some parameter studies. This is all implemented by using the MATLAB ODE solver `ode45` and by incorporating an event function, which would stop the solver each time an impact happens, and continue with new initial conditions based on the law of conservation of momentum and energy.

1 Introductory example

As an introduction to the work, we will start with a simple pendulum in two dimensions. This is a pendulum with a point-mass m attached to a massless, rigid rod of the length l , and the equations of motion are derived using the Euler-Lagrange formalism. The system is constrained by the fixed length l of the rod, so the points of mass with coordinates (x_m, y_m) have to satisfy $x_m^2 + y_m^2 = l^2$.

This means the mass point follows a circular path, and we can use polar coordinates. This is applied on all the further models in this paper. The position is therefore uniquely defined by a position angle φ . With this information we can obtain the Lagrangian and the equation of the motion of this system, with g as the gravity acceleration constant.

$$\ddot{\varphi} = -\frac{g \sin \varphi}{l} \quad (1)$$

The motion is then fully described by the equation (1).

2 The Double Pendulum

By following the principle described in the introduction, we can derive the equations of motion for the dou-

ble pendulum. The setup for this model can be seen on a sketch in Figure 1.

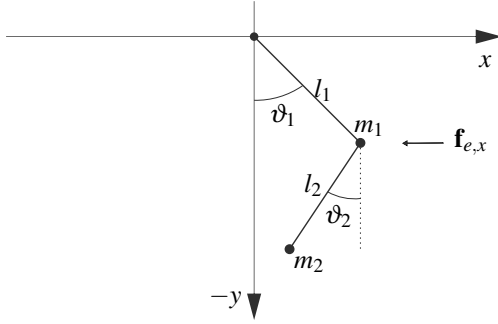


Figure 1: Schematic sketch of a double pendulum.

The positions of the two mass points m_1 and m_2 are uniquely described by their position angles ϑ_1 and ϑ_2 , since they are both attached to rigid rods of fixed length l_1 and l_2 respectively. The external force is applied on the first mass point m_1 . The motion is described by a system of two non-linear ordinary differential equations of second order.

$$\begin{aligned}\ddot{\vartheta}_1 &= -\frac{g}{l_1} \sin \vartheta_1 - \frac{m_2}{m_1 + m_2} \frac{l_1}{l_2} \left(\cos(\vartheta_1 - \vartheta_2) \ddot{\vartheta}_2 \right. \\ &\quad \left. + \sin(\vartheta_1 - \vartheta_2) \dot{\vartheta}_2^2 \right) \\ \ddot{\vartheta}_2 &= -\frac{g}{l_2} \sin \vartheta_2 - \frac{l_1}{l_2} \frac{l_1}{l_2} \left(\cos(\vartheta_1 - \vartheta_2) \ddot{\vartheta}_1 \right. \\ &\quad \left. + \sin(\vartheta_1 - \vartheta_2) \dot{\vartheta}_1^2 \right)\end{aligned}\quad (2)$$

2.1 Autonomous case

Before applying any external forces, we model the behavior of the system without the external forces applied. We solve the system (2) by using ode45 from MATLAB, and investigate different initial conditions.

For the initial conditions less than $\vartheta_{i,0} = \frac{\pi}{3}$, $i = 1, 2$, a periodic motion is visible, whereas a non-periodic motion is visible for initial conditions larger than $\vartheta_{i,0} = \frac{\pi}{3}$, $i = 1, 2$, see Figure 2. This can be explained with the small-angle approximations, which lead to linearization of equations in the system (2). These linearized equations have periodic solutions.

In the Figure 3 we can see a demonstration of the chaotic motion, with a small change in initial condition with no initial velocity. This property is better visible in the motion of the second mass point, which is also be seen on the graph.

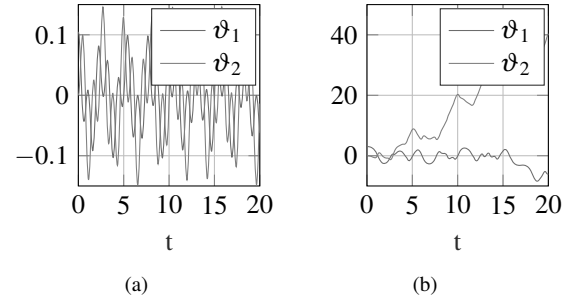


Figure 2: Motion for initial conditions (a) $\vartheta_{1,0} = \frac{\pi}{30}$ and $\vartheta_{2,0} = 0$, (b) $\vartheta_{1,0} = \frac{29\pi}{30}$ and $\vartheta_{2,0} = 0$.

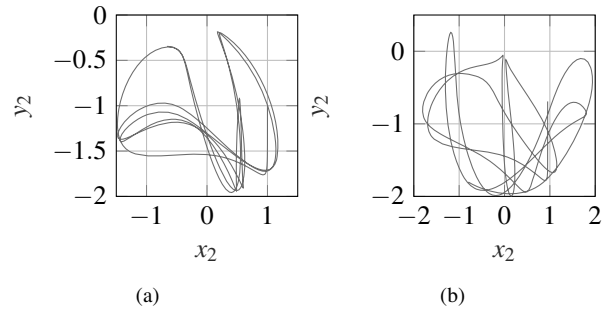


Figure 3: Motion for initial conditions (a) $\vartheta_{1,0} = \frac{\pi}{30}$ and $\vartheta_{2,0} = 0$, (b) $\vartheta_{1,0} = \frac{29\pi}{30}$ and $\vartheta_{2,0} = 0$.

2.2 Motion with external Stimulation

In this subsection we are observing the behavior of the double pendulum when an external force is applied, in the direction of the x -axis. Therefore, the external force is given by $\mathbf{f}_e = (f_{e,x}, 0, 0)^T$. Since the generalized coordinates are $(\vartheta_1, \vartheta_2)$, the external force also has two coordinates, \mathbf{f}_{ϑ_1} and \mathbf{f}_{ϑ_2} . With this, we obtain an inhomogeneous equation system, where the inhomogeneities correspond to

$$\mathbf{f}_{\vartheta_1} = \mathbf{f}_{e,x} l_1 \cos \vartheta_1, \quad \mathbf{f}_{\vartheta_2} = \mathbf{f}_{e,x} l_2 \cos \vartheta_2. \quad (3)$$

In the Figure 4(a) we can see a trajectory of the motion of the double pendulum with an external force applied, and compared to the Figure 4(b), where no external force is applied, we can observe a limited motion in the first figure, since the external force represents a pull in the positive x -direction.

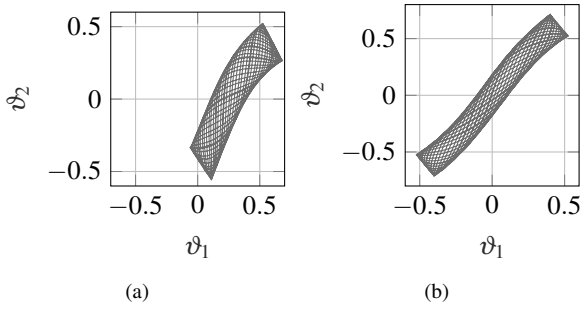


Figure 4: Trajectories for initial conditions $\vartheta_{1,0} = \frac{\pi}{6}$ and $\vartheta_{2,0} = \frac{\pi}{6}$, $t \in [0, 40]$. (a) External force $\mathbf{f}_e = (3, 0, 0)^T$ applied. (b) No external force applied.

3 Bell With a Double Clapper

The basic model of a double pendulum is further extended to a simplified bell with a double clapper, as illustrated in Figure 5.

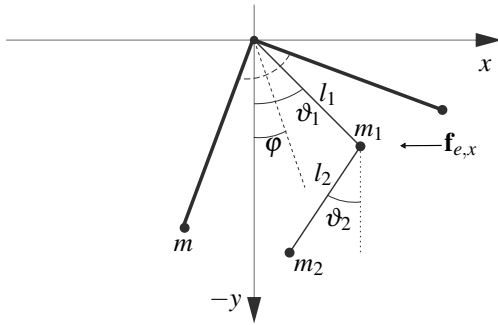


Figure 5: Mechanical setup for the model of a bell.

The outer part of the bell is observed as two coupled simple pendulums with a fixed angle between the rods. The mass of the whole bell is considered concentrated in the mass points m and the rods are considered to be massless. The clapper of the bell is modeled as a double pendulum, with two mass points m_1 and m_2 . An external force is applied to the mass m_1 . The equations used here correspond to a system with coupled independent equations (1) and (2). The angle φ determines the position of the bell sides, and ϑ_1 and ϑ_2 determine the positions of the respective mass points of the bell clapper, all labeled in Figure 5. Depending on which of the two mass points hits the wall of the bell for $i = 1, 2$, we are going to use the law of conservation of energy and momentum, to obtain the new conditions after each

collision, assumed the system is not damped.

$$\begin{aligned} m\dot{\varphi}(t^-) + m_i\dot{\vartheta}_i(t^-) &= m\dot{\varphi}(t^+) + m_i\dot{\vartheta}_i(t^+) \\ \frac{1}{2}m\dot{\varphi}(t^-)^2 + \frac{1}{2}m_i\dot{\vartheta}_i(t^-)^2 &= \frac{1}{2}m\dot{\varphi}(t^+)^2 + \frac{1}{2}m_i\dot{\vartheta}_i(t^+)^2 \end{aligned} \quad (4)$$

The behavior of the bell is investigated by some parameter studies. Each time the normal distance between the mass point and the outer part of the bell would equal zero, an event is triggered, which would stop the solver. The mass of the outer part of the bell is held fixed, and significantly greater than the mass of the clapper. The opening angle is held fixed at $\alpha = \frac{\pi}{3}$.

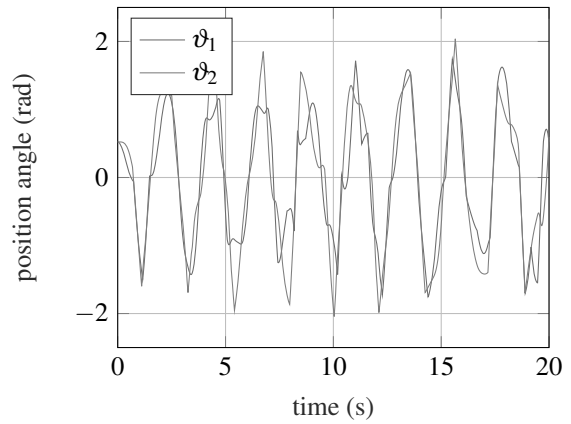


Figure 6: Model of a bell with a double clapper, $m_1 = m_2 = 40\text{kg}$, $m = 2000\text{kg}$, $l = 3\text{m}$, $l_1 = 1\text{m}$, $l_2 = 1.5\text{m}$, position angles of the clapper plotted over time.

On the Figure 6 we can see a bell with no external stimulation, and the bell is brought to ringing by displacing the bell together with the clapper, and letting it swing rather than pushing with an external influence. The initial conditions are $\varphi_0 = \frac{\pi}{3}$, $\vartheta_{1,0} = \frac{\pi}{6}$, $\vartheta_{2,0} = \frac{\pi}{6}$. The events can be seen on the sharp edges on the plot, where the movement was interrupted by an impact on the side of the bell.

Another possibility of ringing a bell, by giving an initial velocity to the bell, but keeping the position angles of the clapper at zero, i.e. giving the bell a push on the side instead of letting it swing. The initial velocity is $\dot{\varphi}_0 = 1.8\text{rad/s}$. The motion induced in this way can be seen on the Figure 7.

The question of the possibility of rhythmic motion can be answered by making a scatter-plot of time differences between consecutive impacts. This can be ob-

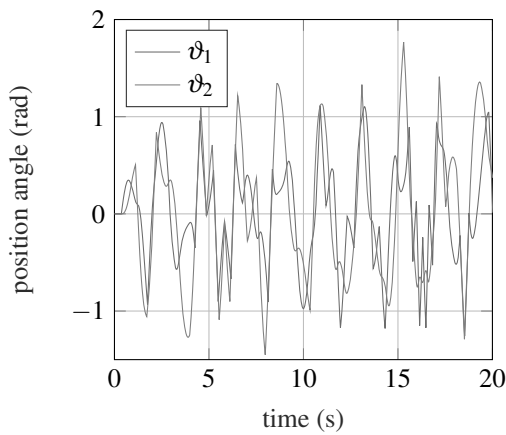


Figure 7: Model of a bell with a double clapper, $m_1 = m_2 = 40\text{kg}$, $m = 2000\text{kg}$, $l = 3\text{m}$, $l_1 = 1\text{m}$, $l_2 = 1.5\text{m}$, position angles of the clapper plotted over time.

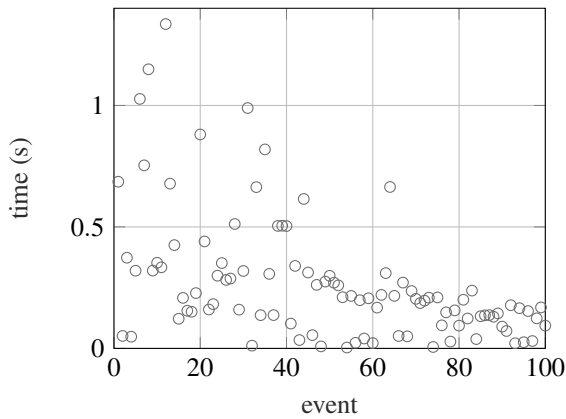


Figure 8: Swinging bell.

served on Figure 8 for the swinging bell, and on Figure 9 for the pushed bell, with initial conditions mentioned above. We can see that the swinging bell, after , and with the pushed bell it does not happen, with the given initial conditions.

4 Conclusion

The main idea of this paper was to investigate the behavior of pendulum systems, by modeling them from first principles. We started with a double pendulum and investigated its chaotic behavior. This model could be extended to a three dimensional model, and by using the external force, or damping, different surroundings

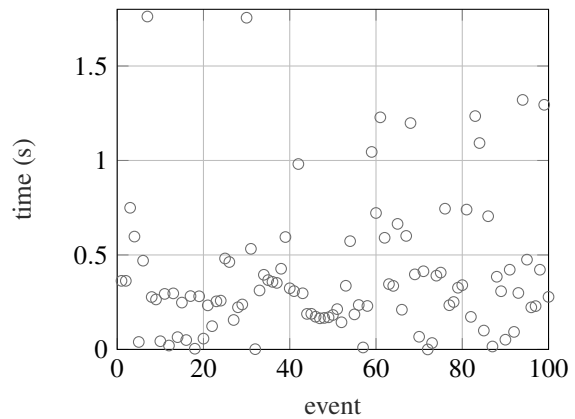


Figure 9: Pushed bell.

could be investigated.

The second model was a simplified model of a bell, where we saw that with a double pendulum as a clapper, periodic motion is not easy to obtain, due to the chaotic behavior of the double pendulum. Taking the material properties of which the bell is made could be an improvement, as well as considering a three dimensional model here as well. Certain parameters were not investigated and this could be included in the future work.

References

- [1] Hasselblatt B., Katok A. *Introduction to the Modern Theory of Dynamical Systems*. Cambridge University Press, 1995.
- [2] Saber N. Elaydi *Discrete Chaos*. Chapman and Hall, 1999.
- [3] Scheck F., *Mechanics - From Newton's Laws to Deterministic Chaos*. Springer-Verlag Berlin Heidelberg, 2010.
- [4] Dudtschenko K., Küpper T., Hosham A.. The dynamics of bells as an impacting system *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*. 2010; 225(10):2436–2443.
- [5] Stachowiak T., Okada T.. A numerical analysis of chaos in the double pendulum. *Chaos, Solitons & Fractals, Elsevier*. 2006; 29(2):417–422.
- [6] Menghetti G., Rossi B.. An analytical model based on lumped parameters for the dynamic analysis of church bells. *Engineering Structures, Elsevier*. 2010; 32(10):3363–3376.

Modulare Entwicklungsplattform für elektrische Luftfahrtantriebe

Markus Henke, Jan Hoffmann*, Lennard Waschke

Institut für Elektrische Maschinen, Antriebe und Bahnen; TU Braunschweig; 38106 Braunschweig

*j.hoffmann@tu-bs.de

Abstract. Im folgenden Beitrag wird eine Simulationsplattform vorgestellt, die im Umfeld der Elektrifizierung von Flugantrieben eine Abbildung des Flugverhaltens und daraus resultierend die Performanceanforderungen an die Energiewandler generiert. Der modulare Aufbau der Simulationsplattform ermöglicht unterschiedliche Detaillierungsgrade und auch Erweiterungen durch z.B. thermische Teilmodelle. Eingegangen wird auf die besondere Rolle der automatisierten Ablaufsteuerung und der Fluglageregelung, die sich selbstständig an veränderte Modell-Parameter und Eigenschaften des Luftfahrzeugs anpasst und so die möglichst realistische Einhaltung des Flugprofils gewährleistet. Dabei werden insbesondere Einflüsse der atmosphärischen Bedingungen und weiterer Systeme wie Auftriebshilfen, Bremsklappen und Fahrwerk berücksichtigt.

1 Einführung

Im Rahmen von Forschungsarbeiten zur Teil- und Voll-elektrifizierung von Antriebssträngen in der Luftfahrt, wurde die modulare Simulationsplattform MoDex.AIR am IMAB in MATLAB/SIMULINK entwickelt, die als Modellierungs- und Simulationsebene zur Auslegung elektrischer Maschinen als Flugantrieb genutzt werden soll. Ziel ist die Abbildung der Flugphysik bis hin zu den auf die elektrische Antriebsmaschine wirkenden physikalischen Größen wie Drehmomente, Drehzahlen und Leistungen. Hierdurch werden die für den thermischen und elektromagnetischen Maschinenentwurf notwendigen Zustandsgrößen über einzelne Flugphasen vorausberechnet und können realitätsgetreu in hierarchisch tiefer liegende Auslegungstools – für z. B. die Elektromagnetik- und für die Energieflussanalyse im Bordnetz einfließen.

Zur ersten Grobauslegung eines elektrischen Antriebs werden Daten zum Leistungs- und Energiebedarf im Flug entlang eines typischen Flugprofils benötigt. Dafür werden der Simulation die wichtigsten Werte des Flugprofils, physikalische – insbesondere aerodynamische – Eigenschaften des Luftfahrzeugs und eine zunächst einfache Repräsentation eines

konventionellen Antriebs vorgegeben. Auf deren Grundlage werden dann unterlagert die physikalischen Prozesse der Energiewandlung modelliert.

Durch Erweiterungen um sehr detaillierte Modelle des elektrischen Antriebskonzeptes, die zusätzlich zum Leistungsfluss beispielsweise auch die erforderlichen Momente und Drehzahlen, sowie das thermische Verhalten simulieren, wird eine präzisere Auslegung möglich.

2 Aufbau und Funktionsweise

Physikalische Grundlagen. Um die physikalischen Prozesse der Energiewandlung während des Fluges adäquat beschreiben zu können, ist zunächst Kenntnis über wirkenden Kräfte erforderlich.

Im stationären Geradeausflug eines Flugzeuges befinden sich dabei folgende vier Kräfte im Gleichgewicht: Die Gravitationskraft F_G , die durch die Triebwerke erzeugte Schubkraft F_T , der aerodynamische Auftrieb F_A und der aerodynamische Widerstand F_W . Vereinfachend nach [1] wird angenommen, dass diese Kräfte alle im Schwerpunkt angreifen. Dabei gilt, wie in Abbildung 1 dargestellt, dass F_W immer entgegen der Fluggeschwindigkeit v wirkt und F_A senkrecht darauf steht. F_T kann um einen Schubeinstellwinkel β von der Flugzeuglängsachse abweichen und ist im Betrag durch die Eigenschaften und Ansteuerungen des Antriebssystems gegeben. F_G wirkt in dieser zweidimensionalen Darstellung senkrecht nach unten.

Sofern sich das Flugzeug nicht in der Horizontalebene bewegt, weicht die Bewegungsrichtung um den Bahnneigungswinkel γ davon ab. Die Längslage des Flugzeugs wird durch den Nickwinkel Θ beschrieben und die Differenz beider Winkel ergibt den Anstellwinkel α .

Die Gravitationskraft ist gegeben durch:

$$F_G = m \cdot g \quad (1)$$

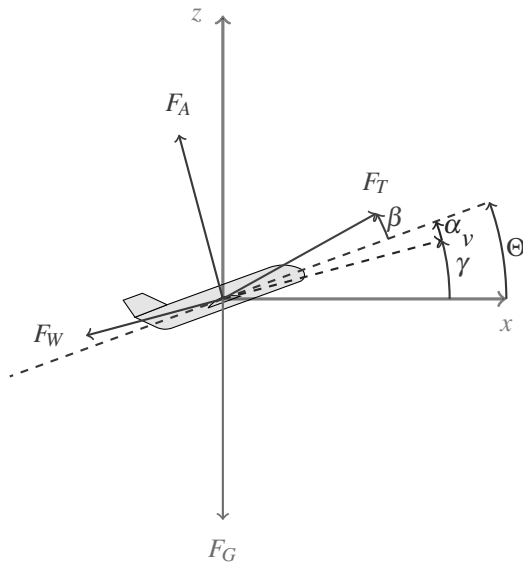


Abbildung 1: Kräfte und Winkel im stationären Geradeausflug in v -Richtung

mit der Erdbeschleunigung g und der Flugzeugmasse m . Auftrieb und Widerstand lassen sich beschreiben durch:

$$F_A = \frac{1}{2} \cdot \rho \cdot S \cdot c_A \cdot v^2 \quad (2)$$

$$F_W = \frac{1}{2} \cdot \rho \cdot S \cdot c_W \cdot v^2 \quad (3)$$

Dabei ist ρ die Dichte der umgebenden Luft, S die Bezugsfläche (typischerweise die Flügelfläche des Flugzeugs) und c_A bzw. c_W sind der Auftriebs- bzw. Widerstandsbeiwert, die eine genauere Betrachtung erfordern:

Beide dimensionslosen Beiwerte dienen zur Beschreibung des komplexen aerodynamischen Verhaltens von Körpern im Luftstrom und werden entweder im Windkanal gemessen oder durch geeignete Simulationen der Strömungsmechanik errechnet. Für jede Körperform ergibt sich ein charakteristisches Verhalten, dass in Abhängigkeit des Anstellwinkels beschrieben werden kann.

Bei Flügelprofilen ergibt sich qualitativ typischerweise das in den Abbildungen 2 und 3 dargestellte Verhalten, das in ähnlicher Form auch für das gesamte Flugzeug gilt. Erwähnenswert ist hierbei, dass der Auftriebsbeiwert innerhalb des typischen Betriebsbereiches näherungsweise linear vom Anstellwinkel abhängt. Er besitzt allerdings bei größeren Anstellwinkeln um etwa

15° bis 25° ein Maximum, was aerodynamisch dem beginnenden Strömungsabriss entspricht. Der Widerstandsbeiwert lässt sich im Betriebsbereich durch ein quadratisches Polynom nähern. Er bleibt in jedem Fall größer Null und besitzt ein absolutes Minimum bei im Betrag kleinen Anstellwinkeln.

Die Schubkraft F_T wird später in der Modellierung des Antriebssystems behandelt.

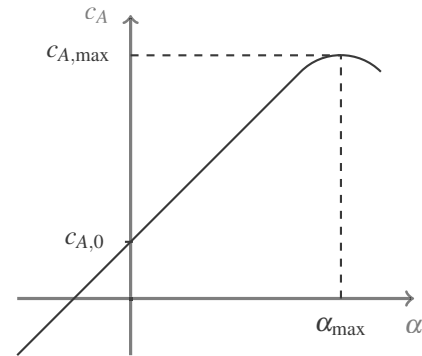


Abbildung 2: Verlauf des Auftriebsbeiwerts c_A in Abhängigkeit des Anstellwinkels α

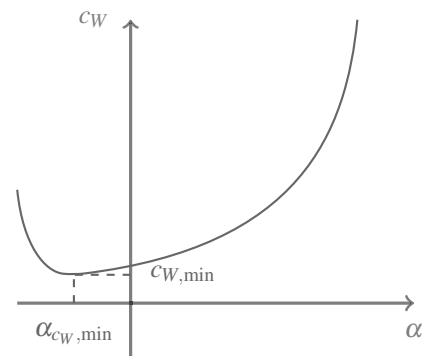


Abbildung 3: Verlauf des Widerstandsbeiwerts c_W in Abhängigkeit des Anstellwinkels α

Aus den genannten Zusammenhängen ergibt sich die Mindestgeschwindigkeit im stationären Horizontalflug durch Gleichsetzen von Auftrieb (Gl. 2) und Gravitationskraft (Gl. 1) und anschließender Umformung:

$$v_{\min} = \sqrt{\frac{2 \cdot m \cdot g}{\rho \cdot S \cdot c_{A,\max}}} \quad (4)$$

Ein Versuch die Fluggeschwindigkeit unter diesen Wert zu reduzieren, würde zum Strömungsabriss führen.

Um für Start und Landung geringere Geschwindigkeiten erreichen zu können, werden Klappensysteme verwendet. Deren Wirkung lässt sich durch ein von der Klappenposition abhängiges Offset der Auftriebs- und Widerstandsbeiwerte beschreiben (vgl. [1] und [2]).

Simulation der Flugphysik. Innerhalb des Simulink-Blocks, der die flugphysikalischen Prozesse simuliert, werden zunächst die genannten Kräfte berechnet. Dabei werden die dafür erforderlichen Auftriebs- und Widerstandsbeiwerte in vektorieller Form und in Abhängigkeit vom Anstellwinkel von einem weiteren Block bereitgestellt, der die Luftfahrzeug-Eigenschaften repräsentiert. Selbiges gilt für Daten wie Flugzeugmasse und Flügelfläche. Auf diese Weise ist es möglich einerseits variable Konfigurationen von z.B. den Klappensystemen direkt zu berücksichtigen, andererseits können die Eigenschaften des Luftfahrzeuges auch grundlegend geändert werden. Analog dazu werden Luftdichte und Erdbeschleunigung von einem Block bereitgestellt, der die Umwelt simuliert. Eine realistische Wahl der Auftriebs- und Widerstandsbeiwerte konnte dabei durch Vergleich mit den Untersuchungsergebnissen von [3] bestätigt werden.

Sind die Kräfte berechnet, werden sie vektoriell addiert und auf die Flugzeugmasse normiert. Durch zweifache Integration und Berücksichtigung der Anfangswerte sind Geschwindigkeit und Position bekannt. Insbesondere die Geschwindigkeit und die mit ihr verknüpften Winkel fließen wieder in die Berechnung der Kräfte ein.

Befindet sich das Flugzeug auf dem Boden (bei Start oder Landung), werden noch zusätzlich die Normalkraft auf der Oberfläche und Reibungskräfte durch Rollwiderstand bzw. Radbremsen berücksichtigt.

Der geschwindigkeitsbezogene Teil der Berechnungen findet im luftfesten Bezugssystem statt, welches sich mit der Windgeschwindigkeit v_w gegenüber dem als Inertialsystem verwendeten erdfesten Bezugssystem bewegt. Daher ist einerseits eine entsprechende Koordinatentransformation (nur translatorisch, nicht rotatorisch) erforderlich, andererseits muss die sich bei Windänderung ergebene Scheinbeschleunigung berücksichtigt werden.

Regelung und Ablaufsteuerung. Um die Fluggeschwindigkeit und Bahnneigung entsprechend des

gewünschten Flugprofils regeln zu können, kommen zwei PI-Regler und eine Vorsteuerung zum Einsatz. Ausgangsgrößen sind dabei die Längsneigung und die Schubkraft, wobei erstere die Geschwindigkeit und letztere die Bahnneigung regelt. Dieses Vorgehen ist zweckmäßig, da erstens auch ohne Schubkraft die gegenüber der Bahnneigung wichtigere Geschwindigkeit geregelt werden kann und zweitens je nach Art des Antriebssystems die Schubkraft stark abhängig von der Geschwindigkeit sein kann. Auf diese Weise ist ein stabiles Verhalten in der Mehrzahl der Anwendungsfälle sichergestellt.

Die Vorsteuerung berechnet zunächst analytisch auf Basis einer Näherung den ungefähren Arbeitspunkt im jeweiligen Flugabschnitt. Sie verwendet dabei eine Lösung des Differentialgleichungssystems, dass sich aus der Flugphysik für den Fall des stationären Horizontalflugs ergibt und zusätzlich eine geeignete Korrektur für Fluglagen mit großer Bahnneigung.

Der erste, schnellere PI-Regler regelt über die Längsneigung die Geschwindigkeit im Arbeitspunkt aus. Der zweite, langsamere PI-Regler regelt die Bahnneigung über den Schubswert aus, welcher in der Simulation des Antriebssystems und Berücksichtigung der Antriebsdynamik in eine vom Flugphysik-Block verwendbare Schubkraft umgesetzt wird. Beide Regler arbeiten in einem festen Verhältnis zueinander und ihre Verstärkung wird in Abhängigkeit vom Arbeitspunkt skaliert, um in jedem Fall stabile Verhältnisse zu erreichen.

Die Verwendung eines in der Literatur zu findenden Regelungskonzeptes ist hier nicht möglich, da diese auf ein nur begrenzt veränderliches Flugzeug abzielen und dabei zusätzlich die Dynamik des Höhenleitwerks und -ruders umfassend berücksichtigen (vgl. [4]). In der Simulation ist allerdings implizit vorausgesetzt, dass die Längslage stabil kontrolliert werden kann und jegliche dadurch entstehenden Auftriebs- und Widerstandskräfte in der tabellarischen Darstellung der entsprechenden Beiwerte berücksichtigt sind. Diese Annahme vereinfacht die Modellierung und verringert die Anzahl der Ursachen für Instabilitäten in der Regelung.

Zur Vorgabe der Sollwerte und allgemein der Steuerung des Simulationsablaufs wird in der aktuellen Modellversion ein in MATLAB/STATEFLOW realisierter Moore-Automat verwendet. Dieser generiert alle erforderlichen Sollwerte für die Regler und steuert außerdem sekundäre Systeme wie Klappen oder Fahrwerk.

Für die realistische Sollwertvorgabe ist außerdem von zentraler Bedeutung, dass die Ablaufsteuerung nach Bedarf die jeweilige Performance berechnet. Beispielsweise ist erfolgt der Steigflug mit zunächst der weg-, dann der zeitoptimalen Steiggeschwindigkeit, welche sich aus dem Schub- bzw. Leistungsüberschuss des Flugzeugs nach [2] berechnet. Die Einhaltung realistischer Geschwindigkeiten orientiert sich außerdem an den Angaben von [5].

Simulation des Antriebssystems. Das Antriebssystem kann in unterschiedlichen Detaillierungsgraden simuliert werden. Für die Berechnung von Leistungsdaten für die noch grobe Erstauslegung des Antriebs kommt eine vergleichsweise einfache Simulation eines konventionellen Antriebs zur Anwendung.

Im nächsten Schritt erfolgt eine Simulation bereits mit einem einfachen Modell einer elektrischen Maschine mit Propeller, so dass neben den Leistungsdaten gegenüber der Luft auch Erkenntnisse zu den erforderlichen Drehzahlen und Momenten, sowie dem Wirkungsgrad des Propellers gewonnen werden können [6]. Dabei ist auch eine Variation von z.B. der Anzahl und Geometrie der Propeller möglich, da das Modell ähnlich wie bei den aerodynamischen Daten auf Basis von Vektoren arbeitet.

Im Rahmen gegenwärtiger Weiterentwicklungen wird das Maschinenmodell so erweitert, dass auch Aussagen über das sehr wichtige thermische Verhalten möglich werden.

3 Ergebnisse

Dank des modularen Aufbaus der Simulationsplattform kann eine Vielzahl unterschiedlicher Flugszenarien simuliert werden. Als anschauliches, kurzes Beispiel wurde hier der Flugverlauf einer vereinfacht modellierten Embraer 170 mit konventionellem Antrieb mit zwei unterschiedlichen Abflugmassen gewählt. Die Flughöhe, Geschwindigkeit und Antriebsleistung ist dazu für 36t und 20t Masse über eine Gesamtflugstrecke von 1000km in Abbildung 4 dargestellt. Als Reiseflughöhe wurde 9000m gewählt und die Reisefluggeschwindigkeit wurde durch die Simulation auf minimalen Energieverbrauch optimiert.

Im Diagramm zur Flughöhe ist erkennbar, dass das leichtere Flugzeug einen Steigflug mit größerer Steigrate ausführen kann, was durch den

größeren Schubüberschuss begründet ist. Durch das geringere Gewicht verschiebt sich die optimale Reisegeschwindigkeit allerdings auch zu niedrigeren Werten - wie im Diagramm zu Geschwindigkeit erkennbar - weshalb sich die Gesamtflugzeit verlängert. In der Darstellung der Antriebsleistung ist deutlich erkennbar, dass das absolute Maximum in der Steigflugphase erreicht wird. Für den Reiseflug ist nur noch ein Bruchteil der maximalen Leistung erforderlich und dies reduziert sich im Sinkflug noch weiter. Ebenfalls aus dem Geschwindigkeitsverlauf ablesbar sind die Landeanfluggeschwindigkeiten, die mit der Masse und der (nicht dargestellten) Landeklappenstellung variieren. Die Landeanfluggeschwindigkeit wurde nach [5] als ein vielfaches der Mindestgeschwindigkeit definiert.

Angemerkt sei, dass durch das hier sehr einfache Triebwerksmodell insbesondere in größeren Flughöhen eine zu große Schubkraft ergibt und daher vom realen Verhalten abweicht. Außerdem wird ein Passagierflug in der Realität meist nicht nur auf geringen Treibstoffverbrauch optimiert, sondern nach [2] werden auch andere Kosten wie beispielsweise für die Besatzung oder die Luftraumnutzung einfließen. Dementsprechend ist die sich im Modell ergebende Flugzeit von über drei Stunden (bei 20t) ebenfalls nicht ganz realistisch.

4 Zusammenfassung und Ausblick

Mit der hier entwickelten Simulationsplattform wurde die Möglichkeit geschaffen eine Vielzahl der für Auslegung und Design von elektrischen Antriebskonzepten für Flugzeuge erforderlichen Daten bereitzustellen. Dank des modularen Aufbaus und den sich daraus ergebenden Erweiterungsmöglichkeiten ist außerdem einfach möglich auch detailliertere und komplexer werdende Maschinenmodelle in ihrer Einsatzumgebung zu simulieren und so weitere Verbesserungen vorzunehmen. Ebenso ermöglicht die Plattform bereits in der frühen Entwicklungsphase den Vergleich unterschiedlicher Ansätze.

Im Rahmen der gegenwärtigen Weiterentwicklungen erfolgt die Modellierung eines elektrischen Antriebsstrangs mit Propeller. Dabei ist auch die Implementierung unterschiedlicher Propeller-Bauformen zum Zwecke des Vergleichs untereinander in Arbeit. Zukünftig wird das Maschinenmodell unter Nutzung von [7] um das thermische Verhalten erweitert, so dass

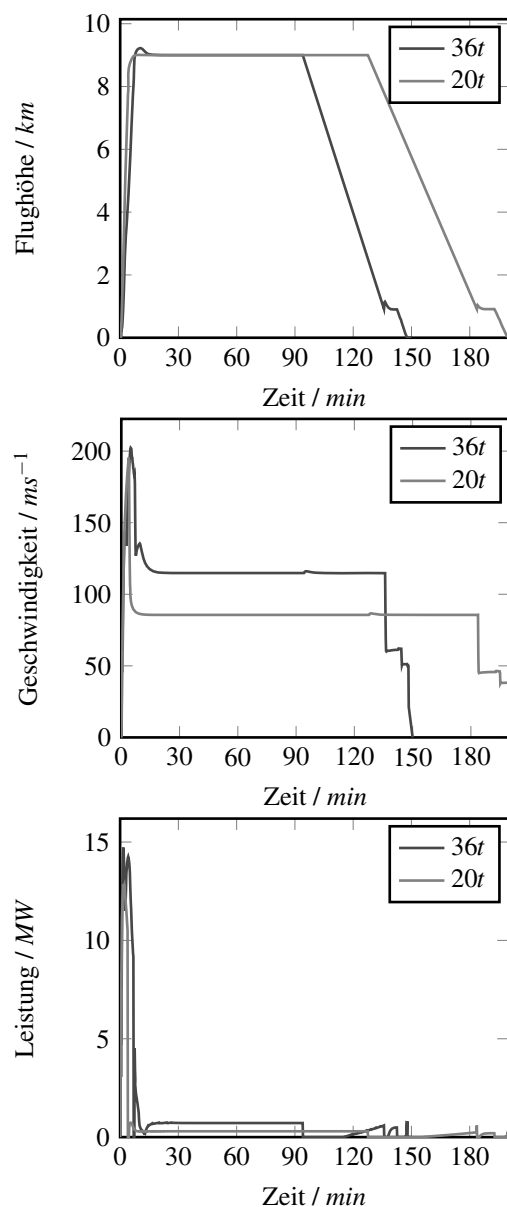


Abbildung 4: Vergleich von Flugprofilen bei unterschiedlichen Abflugmassen

auch dort für die Antriebsentwicklung erforderliche Aussagen getroffen werden können. Denkbar ist auch die Modellierung peripherer Systeme, um Aussagen zum Leistungsbedarf des gesamten Luftfahrzeugs in seinem Flugverlauf treffen zu können. Dabei würden auch die Variationsmöglichkeiten des Flugprofils so erweitert werden, dass kritische Auslegungsfälle wie der Ausfall von einzelnen Triebwerken während des Startlaufs simuliert werden können.

Referenzen

- [1] Prof. Dr.-Ing. C.-C. Rossow, Prof. Dr.-Ing. K. Wolf, Prof. Dr.-Ing. P. Horst (Herausgeber), *Handbuch der Luftfahrzeugtechnik*, Carl Hanser Verlag, München 2014
- [2] J. Schneiderer, *Angewandte Flugleistung - Eine Einführung in die operationelle Flugleistung vom Start bis zur Landung*, Springer-Verlag, Berlin/Heidelberg 2008
- [3] J. Sun, J. M. Hoekstra, J. Ellerbroek, *Aircraft Drag Polar Estimation Based on a Stochastic Hierarchical Model*, Delft University of Technology, Delft 2018
- [4] Universitätsprof. Dr.-Ing. R. Brockhaus, *Flugregelung*, Springer-Verlag, Berlin/Heidelberg/etc. 1994
- [5] Empresa Brasileira de Aeronáutica S.A. (Embraer), *Embraer 190 - airplane operations manual volume 1*, Empresa Brasileira de Aeronáutica S.A. (Embraer), Brasilien 2008 (Revision 4, 2010)
- [6] M. Henke, G. Narjes, J. Hoffmann, C. Wohlers, S. Urbanek, C. Heister, J. Steinbrink, W.-R. Canders, B. Ponick, *Challenges and Opportunities of Very Light High-Performance Electric Drives for Aviation* Energies 2018
- [7] W.-R. Canders, J. Hoffmann, M. Henke, *Cooling Technologies for High Power Density Electrical Machines for Aviation Applications* Energies 2019

Verification of Drogue Detection During Autonomous Aerial Refueling in a Simulation Environment

Oliver Ellis¹ Umut Durak²

¹Clausthal University of Technology, Institute of Informatics, 38678 Clausthal-Zellerfeld, Germany, oliver.ellis@dlr.de

²German Aerospace Center (DLR), Institute of Flight Systems (FT), 38108 Braunschweig, Germany, umut.durak@dlr.de

Abstract

There are use cases that require to extend the airborne time for aircraft. Refueling is one challenge. Intuitively, an option is to do stopovers. Aircraft have to land, refuel and take off again. Another option is aerial refueling, avoiding to land. Autonomous aerial refueling is the refueling process conducted by aircraft in a solely automated manner. A pilot is not interfering during the execution. This process involves a tanker aircraft carrying the fuel and a receiving follower aircraft. The follower enters in a leader-follower formation flight where the leader is the tanker. Then the tanker deploys a cone shaped drogue basket that trails from a flexible hose. The follower can insert a probe into the drogue basket to receive the fuel.

One challenge in autonomous aerial refueling is the automated location of the drogue basket. Object detection is a reasonable option for this task. Recent promising approaches come from the machine learning field. Convolutional Neural Networks (CNN) can be exploited to realize object detection. CNNs are specialized Deep Neural Networks (DNNs) that are comprised of several layers, namely convolutional layers, pooling layers and input and output layers. The input data is taken as matrices of pixels from images. With the output, the location of the drogue basket is predicted.

Like all airborne applications, safety is an essential aspect for aerial refueling. To meet the safety requirements for the automated docking, the detection quality of the object detection needs to be evaluated. However, it is harder than said. Flight tests are expensive and the effort intense. It is almost impossible to provide a broad enough coverage to gain confidence. Here the simulated testing presents a valuable lead.

A challenge is to create a synthetic scenario that is realistic enough and offers a sufficient number of features and variations. Modern game engines with photo-realistic rendering capabilities have been investigated as

options. Pursuing this idea, Unity 3D is applied. To render volumetric clouds and to obtain realistic weather effects, Unity is extended with the real time weather renderer trueSky. Hereby realistic images for aerial refueling scenarios under different conditions are generated (e.g. overcast and precipitation).

It is a combinatorial problem to create enough test cases because of different types of overcast and weather effects and positional factors. The ontology framework System Entity Structure (SES) is utilized to cope with this issue. SESs are used to represent families of systems in simulation environments. This presentation proposes to construct an SES that represent a set of aerial refueling scenarios that are defined by a set of factors (overcast, precipitation, light, relative position and motion). A unique scenario is then derived by pruning (reducing the SES), obtaining a Pruned Entity Structure (PES). To only obtain meaningful scenarios, semantic constraints are applied. For example, if the sky is blue, it cannot rain. Such scenarios are the test cases.

In the simulation the specified test cases are loaded into the context and executed consecutively. During the process image files along with annotation data are generated.

To evaluate the results of the detection the Intersection over Union (IoU) method is used. Here the results are compared against a ground truth (a set of for this purpose prepared annotation data). In such way we obtain for each generated annotation data entry that we categorise with the labels true positive, false positive and false negative. True positive means that the drogue basket is detected with a sufficient accuracy. A false positive case means that detection is not as accurate. If we have a false negative case the CNN detected one or more drogue baskets at the wrong location in an image.

Thus, a desired result would be when above all due the true positives outweigh the other two categories by a large extend. In this case it can be summarised that the simulation based approach for aerial refueling has shown to be promising.

Simulation Based Execution of UAV Missions Sent Through Web Services

Siddhartha Gupta¹, Umut Durak²

¹Clausthal University of Technology, Institute of Informatics, 38678 Clausthal-Zellerfeld, Germany

²German Aerospace Center (DLR), Institute of Flight Systems, 38108 Braunschweig, Germany

Abstract. Drone architectures that support evolution of complex tasks as well as interoperability with other systems are vital for its current application domain. Another important aspect is to do a simulation-based verification of the architecture. This work outlines a system that combines an autonomous architecture, interoperability through web services and a simulation environment to verify the execution of the tasks through web services. This system was tested with the simulation on a PC and the architecture on the Raspberry Pi. The Raspberry Pi can accept mission through web services, process and execute them on the simulation environment remotely.

Introduction

An autonomous Unmanned Air Vehicle (UAV) has the capabilities of accepting a mission and making relevant decisions to achieve the required tasks along the way. High levels of autonomy are needed, especially when the drone is flying beyond the visual line of sight scenario [1]. A suitable feature for UAV's is the ability to work in tandem with other systems/ drones. This interoperability helps in the planning and execution of complex missions. Methodologies such as Service Oriented Architecture (SOA) especially web services, are a popular choice for exposing the functionality as a service while hiding the underlying details of the system [2]. UAV's incorporating an architecture that supports Autonomy, by supporting evolving complexity of mission tasks, and interoperability together are not explored much in the current literature. These become necessary with the advent of many upcoming applications of Drones in the military and civilian domain.

Simulations play a major role in the engineering process as they allow engineers to test designs and prototypes without spending excessive temporal and monetary resources on construction and manufacturing [3]. A computer simulation is relatively simple and convenient to deploy. The cost of any possible failures is minimal, which encourages the developer to be creative

and experiment with new features. The development time also reduces as different environments for validating the drones can be easily created in simulation [4]. It becomes quite important when the software architectures that scale in complexity on many levels. Architectures exploring topics such as Autonomy and Interoperability needs a simulation environment to coexist at development.

A useful methodology for testing and verification using simulations is Software in the Loop (SiL) testing which replaces the real sensors and actuators with simulation models and using the other systems in hardware. Simulating the drone model and its sensors makes it easier to with the architectural changes with simultaneous visualization of the results in simulation.

A popular simulator for robot-based applications is Gazebo [5]. It is a high-fidelity 3D simulator used to simulate robotic models as well as the surrounding environments. It is normally used in conjunction with Robot Operating System (ROS) [6] which is a popular middleware to develop features for autonomous UAVs. Many of the autonomy features for Drones are already available as packages in ROS. ROS and Gazebo work seamlessly with each other and uses the same communication infrastructure. It's relatively simple to create models and build an interface with ROS and Gazebo.

Architecture

This work combines three important aspects of Drone development and testing. The first one is a mechanism to interact with other systems/ drones using RESTful services. The second one is supporting an autonomous architecture that can scale with increasingly complex missions and integrating a simulation Environment to validate the new features.

The main aim here is to have a base to develop these three features independently while coexisting at the same time. The high-level architecture of our system is

shown in Figure 1. Currently, the system can accept missions in the form of JSON scripts from another system, break its complexity down through a three-layer architecture to series of individual tasks, execute those tasks using ROS and validate the tasks in a simulated Gazebo Environment using a ROS/Gazebo communication infrastructure.

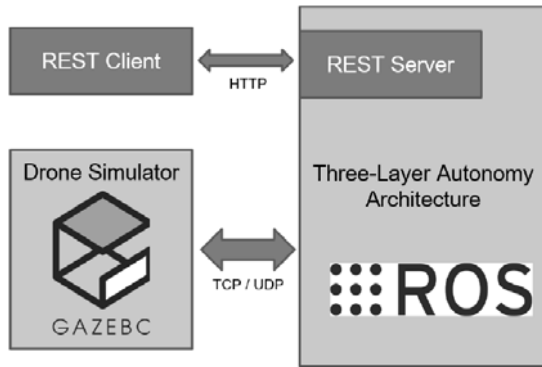


Figure 1 High-Level Architecture

We tested the system using a PC as the REST client to send a mission consisting of popular perception and motion tasks for the Drone. An onboard mission computer in the form of Raspberry Pi was used to accept the mission and another workstation was used to run the simulation.

References

- [1] Viguria, A., "Autonomy Architectures," *Encyclopedia of Aerospace Engineering*, 2016, p. 1–14. doi:10.1002/9780470686652.eae11119.
- [2] Mahmoud, S., Mohamed, N., and Al-Jaroodi, J., "Integrating UAVs into the Cloud Using the Concept of the Web of Things," *Journal of Robotics*, Vol. 2015, 2015, p. 1–10. doi:10.1155/2015/631420.
- [3] [1] Morris, J., Zemerick, S., Grubb, M., Lucas, J., Jaridi, M., Gross, J. N., Ohi, N., Christian, J. A., Vassiliadis, D., Kadiyala, A., et al., "Simulation-to-flight (stf-1): A mission to enable cubesat software-based validation and verification," 2016.
- [4] Ahamed, M. F. S., Tewolde, G., and Kwon, J., "Software-in-the-loop modeling and simulation framework for autonomous vehicles," *2018 IEEE International Conference on Electro/Information Technology (EIT)*, IEEE, 2018, pp. 0305–0310.
- [5] <http://gazebo-sim.org/>
- [6] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., & Ng, A. Y. (2009, May). ROS: an open-

source Robot Operating System. In *ICRA workshop on open source software* (Vol. 3, No. 3.2, p. 5).

Modeling and simulation of a multi-functional high-lift actuation system based on key performance data

Andreas Schäfer^{1*}, René Hollmann², Oliver Bertram¹

¹ German Aerospace Center (DLR), Institute of Flight Systems, Lilienthalplatz 7, 38108 Braunschweig, Germany;
*a.schaefer@dlr.de

² German Aerospace Center (DLR), Institute of Flight Systems, Cornelius-Edzard-Straße 15, 28199 Bremen, Germany

Abstract. The rising complexity of flight control systems leads to a huge amount of additional testing. A major quantity of malfunctions is identified during system integration activities at a late stage in the development process. In order to reach high maturity as early as possible especially on the overall system level, virtual testing methods become increasingly important. This paper describes the development of a flight control actuation systems library based on key performance data. The library is implemented in the modeling language Modelica and enables the simulation and analysis of state-of-the-art high-lift actuation systems including the main mechanical failure cases. In addition, a script-based preprocessing is implemented that minimizes the parameterization effort for different test cases.

Introduction

The development of safety-critical flight control systems requires a high number of physical tests during the entire design process [1,2,3]. Since the system complexity increases due to the implementation of new functions and the development of multi-functional movables, the number of system requirements rises. As a result, the test activities need to be further expanded [2]. In the beginning of the system design process, small component and equipment test benches perform a decisive role. These test benches are required to verify the performance as well as to evaluate aspects such as endurance and fatigue. This verification is usually provided by the suppliers of the components and equipment [4]. The tests are also used to identify the main characteristics such as moments of inertia and frictional losses of the equipment. These identified parameter values are part of the required deliverables and are provided to the original equipment manufacturer (OEM) as so-called *key performance data*.

An essential verification effort is performed by the OEM on so-called *zero-means*. Zero-means are test rigs that represent an entire aircraft system. Up to 90 percent

of all faults are detected on the system level using such test means [3]. However, zero-means are only available late in the development process so that found faults may lead to significant modification costs and delays. The dilemma is enhanced by a growing number of interactions, even beyond system boundaries, due to more complex systems with an increased number of functions. In order to overcome this challenge as well as to minimize the physical testing effort in general, currently an important research focus lies on virtual testing methods [5,6,7]. Reliable models and accurate parameter values are essential prerequisites for virtual testing. The required knowledge can be gained by design and test activities performed on the component or equipment level. This way, valid data can be used to evaluate system level requirements much earlier in the development process than with zero-means. In order to demonstrate such an approach, the modeling of equipment of a high-lift actuation system based on key performance data is presented in this work.

Using key performance data, nonlinear models with lumped parameters are implemented. The equation-based, object-oriented, multi-domain modeling language Modelica [8] is chosen for this purpose. The utilization of Modelica ensures high flexibility in modeling and enables the realization of a high degree of automatization for model generation and parameterization. In contrast to [9], the modeling based on actual key performance data enables, for example, the usage of models independent of the system supplier and minimizes the effort of mapping parameter sets to simulation models, mitigating modeling errors. In order to keep the modeling as well as the verification and validation effort as low as possible, the models of different equipment are based on common component models. A component model specifies a physical characteristic such as frictional losses, backlash or torsional stiffness. Since the

required model structure of an equipment model is equal for similar equipment types embedded in different aircraft, the same equipment model can be used in different aircraft types. The possibility to reuse validated models is a crucial prerequisite to keep the modeling effort low and to exploit the full potential of virtual testing. The key performance data provide a good basis to achieve this goal. In addition to the nominal characteristics, the mechanical faults *disconnection* and *jaming* are implemented in the flight control actuation systems library. Finally, an approach for test case parameterization is introduced and some simulation results are discussed.

1 Flight Control Actuation Systems Library

The model library developed in the context of this work allows the modeling and simulation of a state-of-the-art multi-functional high-lift actuation system. Such a system is illustrated in **Figure 1**. The main difference to a classical high-lift system is the additional drive unit installed in the transmission between the inner and outer flap. The electrically-powered active differential gearbox (ADGB) enables a fully independent motion of the flaps [10]. The differential flap setting is applied to optimize the load distribution of the wing. As a result, the weight of the wing structure can be reduced [11].

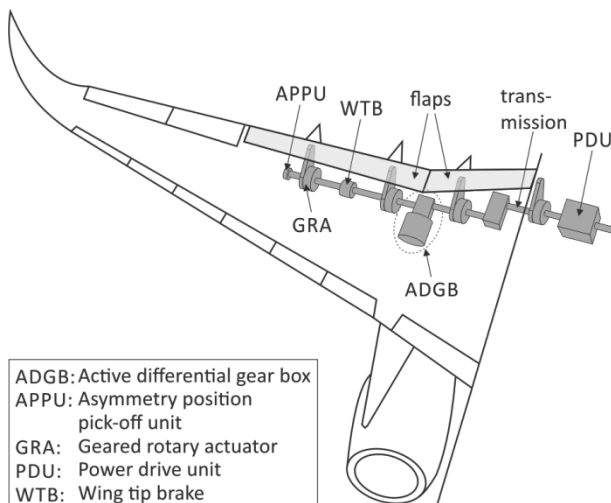


Figure 1: Architecture of a state-of-the-art multi-functional high-lift system [12]

The flight control actuation systems library is implemented in the equation-based and object-oriented modeling language Modelica. A major difference to a block-

oriented approach is that it is not required to specify a certain data-flow direction. Such an acausal feature reduces the modeling effort of physical systems and enables the most possible flexibility and reusability [8]. These and further differences between the two modeling concepts are pointed out in [8] using a simple electric circuit and in [13] modeling a mechanical system. Nevertheless, block-oriented modeling is also supported by the Modelica language. A component-based approach is applied in the implementation of the model library. The required equipment models are created by connecting the developed component models or component models from the *Modelica Standard Library*. Consequently, the flight control library consists of two main packages: the component package and the equipment package. An overview of a selection of relevant component and equipment models of a high-lift actuation system is presented in the following subsections.

1.1 Component Models

A state-of-the-art high-lift actuation system consists of mechanical, hydraulic and electric equipment. Nevertheless, most of the equipment is from the mechanical domain due to the centralized power generation. The central drive unit is mechanically connected to all trailing edge devices via shafts, gearboxes and joints. The key performance data for mechanical equipment can include the following parameters:

- Moment of inertia
- Torsional stiffness
- Mechanical backlash
- Frictional losses

The mechanical equipment is characterized at least by its moment of inertia and torsional stiffness. The torsional stiffness is modeled by a spring-damper system. Since the damping constant is not part of the key performance data yet, an estimated value is assumed based on empirical data for all equipment models. For more accurate simulation results the damping constant should also be part of the key performance data in the future. In addition, mechanical backlash is considered for the most mechanical equipment models. All parameter values are identified with respect to the input (drive) side of the equipment. The required component models are available in the *Modelica Standard Library* (*Inertia*, *Spring-Damper* and *ElastoBacklash*) [14].

Frictional losses are another important characteristic of mechanical equipment of a high-lift system. The

speed-dependent frictional losses are characterized by a breakout torque and a running drag torque. The breakout torque is also referred to as break-away torque and must be overcome to initiate motion [15]. The running drag torque is defined by a drag torque value at referenced low system speed and a drag torque value at nominal system speed. Between those two points a linear increase of the drag torque is assumed approximately. It is presumed that the referenced low system speed is very close to zero. After reaching the nominal system speed, the running drag torque is kept constant. The model of the speed dependent friction torque is illustrated in **Figure 2**.

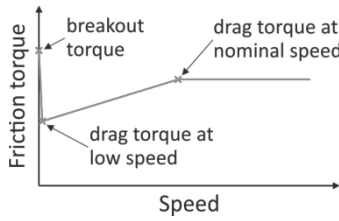


Figure 2: Model of the speed dependent friction torque

In addition to the speed-dependent losses, a load-dependent friction torque characterized by an efficiency value represents, for instance, the meshing friction of a gearbox. If required, an efficiency value for opposing loads and one for aiding loads can be defined. In context of high-lift actuation systems, this distinction is important for the geared rotary actuator (GRA) since its efficiency is strongly dependent on the load case [16].

In order to verify monitoring functions and sensor concepts, the simulation of failure cases is essential. In a transmission system, such as a high-lift actuation system, mechanical disconnection and jamming are, among others, critical failure cases. The disconnection fault model is implemented by adapting the spring-damper component. At a defined failure time both the spring constant and the damping constant are decreased to zero. This value drop is characterized by a time constant as a first-order step response as exemplarily depicted in **Figure 3**. As a result, no torque is transmitted between the flanges of the component and the separated system parts can be driven independently.

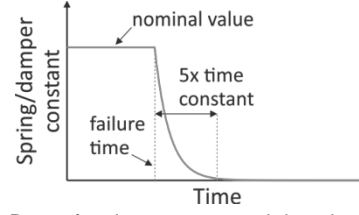


Figure 3: Drop of spring constant and damping constant if disconnection fault is injected

The jamming failure case can be modeled by introducing an additional friction torque. Since the severity of a jamming event may vary, the maximum jamming torque can be defined in the test case specification. Unless this maximum value is exceeded, the jamming event prevents the transmission system from moving. In addition, the jamming fault is characterized by a time constant as illustrated in **Figure 4**.

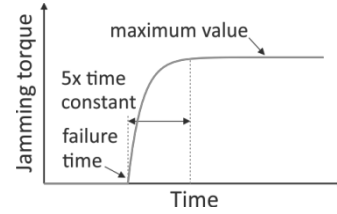


Figure 4: Jamming torque if jamming fault is injected

The interface between the mechanical domain and the electric domain is the electric motor of the ADGB. The transformation between those two domains is defined by the component *EMF* (electromotive force) as follows:

$$v = k_{EMF} \cdot \omega \quad (1)$$

and

$$\tau = -k_{\tau} \cdot i \quad (2)$$

where,

- v : voltage drop across *EMF*
- i : armature current
- ω : angular velocity
- τ : torque
- k_{EMF} : back EMF constant
- k_{τ} : torque constant

If only the back EMF constant is defined, the torque constant is calculated as follows [17]:

$$k_{\tau} = \frac{3}{2} k_{EMF} \quad (3)$$

The interface between the hydraulic domain and the mechanical domain is the secondary controlled variable displacement hydraulic motor (VDHM). In this concept the position of the swashplate of the hydraulic motor is

adjusted to control the motor speed. The implemented component model of the VDHM is based on the linear model presented in [18].

As outlined in the next subsection, equipment models are implemented by connecting corresponding component models.

1.2 Equipment Models

The high-lift system of a long-range aircraft modeled in context of this work consists of 14 different equipment types. The corresponding equipment models are depicted in **Figure 5** and introduced in more detail below.

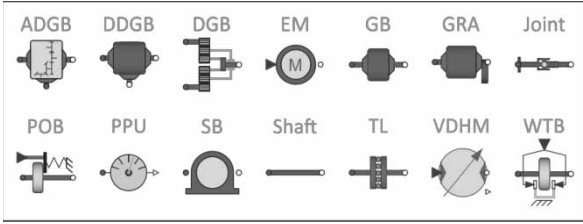


Figure 5: Equipment models of a multi-functional high-lift system

The simplest mechanical equipment model is a shaft consisting of only the components *SpringDamper* and *Inertia* as depicted in **Figure 6** (left). By extending this model with backlash (*ElastoBacklash*) and constant efficiency (*SlopeEfficiency*) the equipment model (transmission) joint results according to the key performance data (**Figure 6**, right).

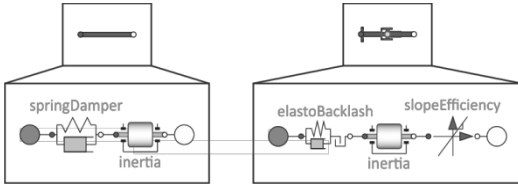


Figure 6: Equipment model of a shaft (left) and a transmission joint (right)

Compared to a joint, a steady bearing (SB) is additionally characterized by speed dependent frictional losses (*SlopeEfficiencyDrag*) as depicted in **Figure 7** (left). The same applies for the equipment model of a gearbox (GB) (**Figure 7**, right). In addition to the frictional losses (*SlopeEfficiencyDrag*) the component model *GearEfficiencyDrag* enables the definition of a gear ratio as well as load case dependent efficiency values. For this reason, the model structure of a geared rotary actuator (GRA) is equivalent to the model structure of a GB. In order to model a down drive gearbox (DDGB), the same component models are used. For the DDGB it is assumed that all frictional losses occur in the down

drive path.

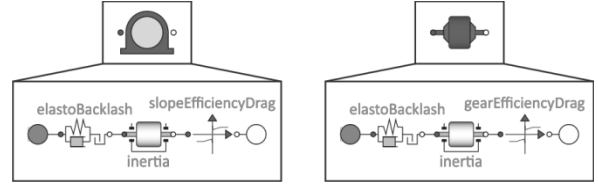


Figure 7: Equipment model of a steady bearing (left) and a gearbox (right)

The fourth type of gearbox is the differential gearbox (DGB) that connects the two hydraulic motors with the transmission system. The present key performance data of the DGB define a moment of inertia at each flange and speed-dependent frictional losses. The general speed summing characteristic of this gearbox with a gear ratio i_{DGB} is specified as follows:

$$\omega_{out} = \frac{\omega_{in1} + \omega_{in2}}{2 \cdot i_{DGB}} \quad (4)$$

$$\tau_{out} = 2 \cdot i_{DGB} \cdot \tau_{in1} = 2 \cdot i_{DGB} \cdot \tau_{in2} \quad (5)$$

where ω is the angular velocity and τ is the torque at input (*in*) and at output (*out*) of the gearbox.

In general, brakes of a high-lift transmission system such as wing tip brakes (WTBs) and power-off brakes (POBs) are supplied by at least one hydraulic system. The hydraulic power is used to generate the brake force (WTB) or to release the brake (POB). This behavior is approximated by means of a signal-based brake command with a defined closing time and opening time (*BrakeSignal*). The brake torque itself is generated by the component *Brake* from the *Modelica Standard Library*. The complete model of a WTB is shown in **Figure 8**.

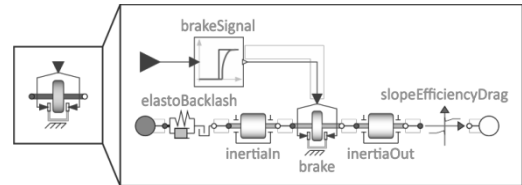


Figure 8: Equipment model of a wing tip brake

In order to mitigate structural damage in case of jamming, each flap drive station is equipped with a torque limiter (TL). In addition, system torque limiters are typically installed on long-range aircraft with high powered drive units. This way, the transmission behind the system torque limiter must be designed to withstand only the threshold torque of the torque limiter and not the maximum torque of the drive unit. The main charac-

teristics of a torque limiter are represented by a nonlinear torsional stiffness and a brake torque [9]. The torsional stiffness value depends on the applied torque and the relative angular displacement. A classical torque limiter with one lockup stage is described by three discrete modes: nominal operation mode, locking up mode and lockup mode. Each mode is defined by a torque limit and a spring constant value as illustrated in **Figure 9**. The brake torque is zero in the nominal operation mode. During locking up, the brake torque increases linearly and reaches its maximum value at lockup.

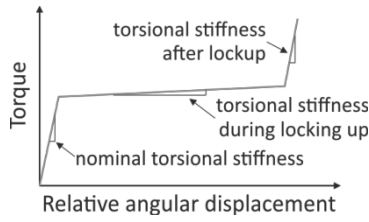


Figure 9: Nonlinear torsional stiffness depending on torque and relative angular displacement

The complete equipment model of a torque limiter is depicted in **Figure 10**. In addition to the nonlinear spring and brake component, the model is composed of spring-damper and inertia components at input and output as well as of a slope efficiency component.

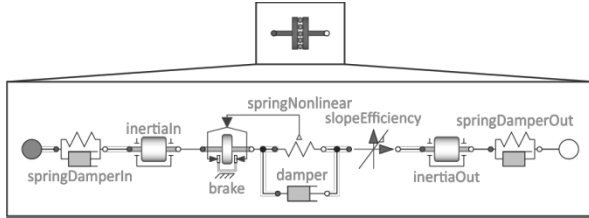


Figure 10: Equipment model of a torque limiter

Depending on the control and monitoring concept of a high-lift system, position sensors at different locations are required. Such position sensors are referred to as position pick-off units (PPUs). The PPU is modeled as an ideal angle sensor with a transmission ratio. The sensor signal is returned in degrees.

As mentioned before, the ADGB enables differential flap setting and represents the main difference compared to conventional high-lift system architectures. This speed summing differential gearbox is driven by the central drive unit when all surfaces should be positioned synchronously (through-drive mode). If only the outer flaps should be moved, the ADGB is driven by an electric motor (EM) and the rest of the transmission system is held by the pressure-off brakes of the central drive unit. For each mode the key performance data

specify identical parameter sets similar to the set of the GB (see **Figure 7**). In contrast to the DGB, the gear ratio of the ADGB depends on which input side is driven. Assuming i_{TH} as defined through-drive gear ratio, the gear ratio i_{EM} , when the EM moves the outer flap, is calculated as follows:

$$i_{EM} = \frac{i_{TH}}{1 - i_{TH}} \quad (6)$$

In order to model the EM in accordance with the key performance data, the above introduced component model *EMF* is extended by a *Resistor*, an *Inductor* and an *Inertia* from the *Modelica Standard Library*. This modeling corresponds to a possible simplified representation of a permanent magnet synchronous motor (PMSM) [12]. The resulting equipment model is depicted in **Figure 11**.

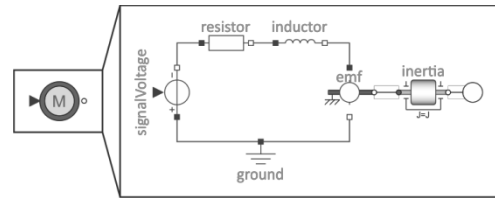


Figure 11: Equipment model of an electric motor (EM)

2 Virtual Testing of a High-Lift System

The developed model library enables the evaluation of all nominal operation conditions as well as of characteristic failure cases. By means of validated equipment models and accurate key performance data, such a virtual testing approach could reduce the testing effort of safety-critical flight control systems and provide important results before system level test rigs are available. In order to fully benefit from simulation-based testing, the system modeling effort and the effort of setting up a test case must be as low as possible. For this reason, the library concept is complemented by a pre-processing with a high degree of automatization regarding the test case specific model parameterization. The general concept of the implemented preprocessing is introduced in the next subsection. Afterwards some selected simulation results are presented exemplarily.

2.1 Preprocessing

The preprocessing is implemented in the programming language Python. As illustrated in **Figure 12**, the implemented process can be divided into two main steps:

1. Creation of a system specific parameter set
2. Creation of a simulation model and executable Modelica scripts

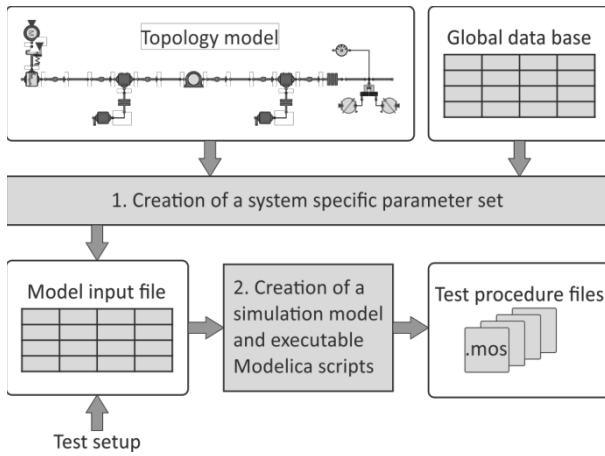


Figure 12: Main steps of the implemented preprocessing

The first step requires the specification of the aircraft and access to the corresponding key performance data. In order to ensure data consistency, it is assumed that the key performance data of different aircraft are stored and managed in one central database. In context of this work, it is termed *global data base*. By means of a Modelica model the expected system architecture is defined. In addition to the topology information, this model provides an interface between the model library and the key performance data. The equipment class and the equipment name both serve as identifiers for the correct parameter set of the specified aircraft. Finally, all information is stored in a file referred to as *model input file*.

The generated model input file is the basis for a test campaign. This file is used to adjust the system if necessary (e.g. insertion of faults) and to set up the test procedures. A test procedure is characterized by a simulation time, value type (nominal, minimum, maximum) and environment temperature among others. Depending on the used topology model, it might also be necessary to define the air loads acting on each GRA or to specify the interfaces if the generated model is used for simulation coupling. Afterwards, the model input file contains all information required to generate a simulation model and to parameterize all defined test procedures. The result of the second step is an executable Modelica script for each test case containing the test case specific parameter set and the simulation setup. By executing such a script the defined test case is simulated or a Functional Mock-up Unit (FMU) for co-simulation is

generated. All generated and required files are stored in a folder so that the test campaign can be repeated any time, regardless of changes in the model library or the global data base, ensuring fully traceability. In context of virtual testing for certification, the assurance of complete traceability is a crucial aspect that should be covered by suitable simulation data and process management (SDPM) [5,19].

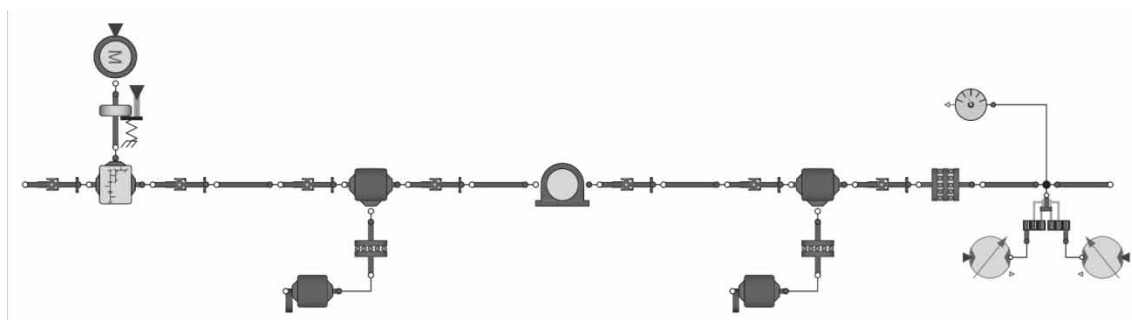
2.2 Simulation

The multifunctional high-lift system considered in this work consists of over 100 equipment models. The general structure of such a system model is illustrated in **Figure 13**. The figure shows the equipment of the left wing between the central drive unit and the ADGB. In the actual simulation model, there are further shafts, joints, steady bearings and gearboxes between the support stations.

In general, the testing of high-lift systems aims at the verification of compliance to the system requirements as well as the correct failure detection and system reaction in such a case. For this purpose, complex system level test rigs are used. Nominal functional tests include, for example, moving the flap to all defined positions at different air loads applied. In order to verify the system behavior in case of a failure, for example, a shaft is replaced by a clutch enabling the simulation of a shaft rupture.

A major advantage of simulation-based testing is the high flexibility. A simulation enables a fast reaction to design changes and the varying of parameter values for a sensitivity analysis or for the evaluation of worst-case scenarios which cannot be normally implemented on a system level test rig. In contrast to a test rig, quantities of interest can be determined at any position enabling, for example, the identification of the most promising sensor position. For this - as discussed earlier - the required data has to be obtained on equipment test benches in the form of key performance data.

The implementation of new monitoring concepts such as an electronic torque limiter requires an accurate knowledge of torque values at all possible operation conditions. As demonstrated in the following, the environment temperature as well as a worst-case scenario might have a major impact on the drive torque and thus on the sensor thresholds. In **Figure 14** the torque measured at the output of the central drive unit for different environment temperatures is depicted. The values are normalized to the maximum torque value. Since the



frictional losses increase with decreasing temperature, the necessary drive torque increases with lower temperature. At higher temperatures (here: +20 °C) on the other hand, the low frictional losses lead to a strong oscillation of the drive torque. Moreover, the worst case scenario characterized by maximum frictional losses and minimum efficiency values impacts the resulting drive torque enormously. Such an analysis is essential for the implementation of a robust sensor concept. Nevertheless, not all scenarios can be carried out on a test bench at a reasonable cost or without the risk of damaging the test rig. Assuming that the models and the data are representative, simulation-based testing extends the test scope and enables a comprehensive verification of the system.

Similar effects can be observed by evaluating failure conditions. As mentioned above, disconnection is a typical failure case of a high-lift transmission system. As illustrated in **Figure 15**, the test setup might have a strong impact on the system behavior and thus the sensor thresholds. The figure shows the measured angle at the central drive unit (FPPU) and at the end of the left

wing (APPU). In the first case, the simulation is performed with key performance data for an environment temperature of 20 degrees Celsius. At a simulation time of six seconds a shaft rupture occurs between the inner flap and the outer flap. The applied air loads push the disconnected part in retraction direction. When the defined asymmetry threshold (here: four degrees) is exceeded, the WTB is applied and the drive unit is shut down. In the second case, the same test procedure is conducted with key performance data for an environment temperature of -55 degrees Celsius. In this case the frictional losses are much higher so that the applied air loads are not able to push the disconnected flap surface back. Consequently, the assumed sensor concept fails to detect the failure in this scenario. The importance of such investigations increases when innovative systems are to be developed and implemented.

In the context of this work, a Modelica library enabling the modeling and simulation of multi-functional high-

lift actuation systems has been developed. Nonlinear models with lumped parameters are implemented based on the so-called key performance data base. In addition to the nominal characteristics, the mechanical faults *disconnection* and *jamming* are implemented in the flight control actuation systems library. By determining the key performance data on equipment test benches, simulation-based verification of requirements on overall system level can be performed before system test rigs are available. Besides, the environmental conditions of small equipment test benches can be varied with less effort in contrast to system level test rigs. As illustrated above, the environment temperature might have a strong impact on the system behavior.

In order to generate more representative simulation results, the transmission system will be simulated with the high-lift structure by means of a co-simulation as the next step. Both the flap surfaces and the flap mechanisms are modeled as elements of a multibody system. The co-simulation will be implemented using the Functional Mock-up Interface (FMI).

Acknowledgments. This work was supported by the German Federal Ministry for Economic Affairs and Energy (BMWi) in the framework of the Federal Aeronautical Research Program (LuFo V-3 Phy-ViTEm under the support code 20X1725C).

References

- [1] Moir, I., Seabridge, A. *Design and Development of Aircraft Systems*. Chichester, UK: John Wiley & Sons; 2012. 312 p.
- [2] Hans, C., Hribernik, K. NFF Special Session – Potentials of Applying Methods, Tools, Processes and Knowledge from Testing in Product Development to the NFF Problem. *Procedia CIRP*. 2014; 22, 53-58. doi: [10.1016/j.procir.2014.07.131](https://doi.org/10.1016/j.procir.2014.07.131).
- [3] Jandaurek, K., Johst, M. Development Trends and Innovations in Aerospace System Testing Using the Example of High-Lift. *55th AIAA Aerospace Sciences Meeting*; 2017 Jan; Grapevine, USA. doi: [10.2514/6.2017-0548](https://doi.org/10.2514/6.2017-0548).
- [4] Langermann, R. *Beitrag zur durchgängigen Simulationsunterstützung im Entwicklungsprozess von Flugzeugsystemen* [dissertation]. Technische Universität Braunschweig; 2009.
- [5] Ulmer, T., Amin, J. Virtual Testing of High Lift Systems. *SAE Technical Paper*. 2013; 2013-01-2280. doi: [10.4271/2013-01-2280](https://doi.org/10.4271/2013-01-2280).
- [6] Valdivia-Guerrero, V., Foley, R., Riveros, S., et al. Modelling and Simulation Tools for Systems Integration on Aircraft. *SAE Technical Paper*. 2016; 2016-01-2052.

doi: [10.4271/2016-01-2052](https://doi.org/10.4271/2016-01-2052).

- [7] Schäfer, A., Hollmann, R., Bertram, O. Process for Virtual Design and Testing of Flight Control Actuation Systems. 68. *Deutscher Luft- und Raumfahrtkongress*; 2019 Oct; Darmstadt, Germany.
- [8] Fritzson, P. *Principles of object-oriented modeling and simulation with Modelica 3.3: A cyber-physical approach*. Piscataway, US: Wiley-IEEE Press; 2014, 1256 p.
- [9] Pfennig, M., Thielecke, F. Implementation of a Modelica Library for Simulation of High-Lift Drive Systems. *6th International Modelica Conference*; 2008 March; Bielefeld, Germany.
- [10] Lulla, C. Functional Flexibility of the A350XWB High Lift System. 60. *Deutscher Luft- und Raumfahrtkongress*; 2011 Sept; Bremen, Germany.
- [11] Strüber, H. The Aerodynamic Design of the A350 XWB-900 High Lift System. *29th International Congress of the Aeronautical Sciences*; 2014 Sept; St. Petersburg, Russia.
- [12] Schäfer, A., Schmid, M. Analysis of the Effects of Modeling Depth and Parameter Uncertainties on the System Behavior of a Multifunctional High Lift Actuation System. *SAE Technical Paper*. 2018; 2018-01-1918. doi: [10.4271/2018-01-1918](https://doi.org/10.4271/2018-01-1918).
- [13] Tiller, M. *Introduction to Physical Modeling with Modelica*. Boston, US: Springer; 2001, 345 p.
- [14] Modelica Standard Library. <https://github.com/modelica/ModelicaStandardLibrary>, viewed 24 August 2020.
- [15] Olsson, H., Åström, K. J., De Wit, C. C., Gäfvert, M., Lischinsky, P. Friction Models and Friction Compensation. *European Journal of Control*. 1998; 4(3), 176-195. doi: [10.1016/S0947-3580\(98\)70113-X](https://doi.org/10.1016/S0947-3580(98)70113-X).
- [16] Wang, A., Gitnes, S., El-Bayoumy, L. The Instantaneous Efficiency of Epicyclic Gears in Flight Control Systems. *Journal of Mechanical Design*. 2011; 133(5). doi: [10.1115/1.4004001](https://doi.org/10.1115/1.4004001).
- [17] Olaf Cochoy. *Investigations for the Synchronized Operation of a Hybrid Actuator Configuration in Redundant Flight Control Systems* [dissertation]. Hamburg University of Technology; 2009.
- [18] Geerling, G. *Entwicklung und Untersuchung neuer Konzepte elektrohydraulischer Antriebe von Flugzeug-Landeklappensystemen* [dissertation]. Hamburg University of Technology; 2003.
- [19] Ulmer, T., Amin, J. Virtual Testing of High Lift Systems. *NAFEMS World Congress 2013*; 2013 June; Salzburg, Austria.

Definitions/Abbreviations

ADGB – Active differential gear box

DDGB – Down drive gearbox

DGB – Differential gearbox
EM – Electric motor
EMF – Electromotive force
FMI – Functional Mock-up Interface
FMU – Functional Mock-up Unit
GB - Gearbox
GRA – Geared rotary actuator
OEM – Original equipment manufacturer
PMSM – Permanent magnet synchronous motor
POB – Power-off brake
PPU – Position pick-off unit
SDPM – Simulation data and process management
SB – Steady bearing
TL – Torque limiter
VDHM – Variable displacement hydraulic motor
WTB – Wing tip brake

Intelligente Zielführung elektrischer Fahrzeuge mit Brennstoffzelle als Range Extender in vernetzten Verkehrssystemen

Sören Scherler*, Xiaobo Liu-Henke

Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau, Institut für Mechatronik (IMEC),
Salzdahlumer Str. 46/48, 38302 Wolfenbüttel, Deutschland; *so.scherler@ostfalia.de

Abstract. Im vorliegenden Beitrag wird die intelligente Zielführung elektrischer Fahrzeuge mit Brennstoffzelle als Range Extender in vernetzten Verkehrssystemen behandelt. Diese Zielführung ermittelt eine Route, welche entweder bzgl. Energieverbrauch, Zeit, Distanz oder einer Gewichtung dieser Zielkriterien optimal ist. Des Weiteren wird eine Schnittstelle zu Informationen aus der V2X-Kommunikation vorgesehen, sodass aktuelle Verkehrsdaten und auch Informationen über die Lade- und Tankinfrastruktur berücksichtigt werden können. Diese Informationen werden zur optimalen Planung von Lade- und Tankstopps auf Fahrten genutzt, für welche die in der Batterie und/oder dem Wasserstofftank gespeicherte Energie nicht ausreicht. Diese Stopps werden bzgl. der zuvor genannten Kriterien geplant.

Einleitung

Im Teilprojekt *Intelligente Elektrofahrzeuge mit Range Extender in Verkehrssystemen mit Fahrzeug 4.0* des vom Niedersächsischen Ministeriums für Wissenschaft und Kultur sowie der VolkswagenStiftung geförderten Verbundprojekts *Zukünftige Fahrzeugtechnologien im Open Region Lab* werden automatisierte Elektrofahrzeuge mit Brennstoffzellen als Range Extender untersucht. Ziel des Teilprojekts ist die Erzielung eines zeit- und energieoptimierten, prädiktiven Fahrbetriebs in vernetzten Verkehrssystemen. Um dieses Ziel zu erreichen, werden drei Optimierungspotentiale in der Zielführung („Welche Route ist optimal?“), der Bahnplanung („Welche Fahrzeugführung ist auf der gewählten Route optimal?“) und dem Energiemanagement („Welche Verteilung der zur Erfüllung der Fahraufgabe notwendigen Leistung auf Batterie und Brennstoffzelle ist optimal?“) fokussiert.

Die Zielführung eines elektrischen Fahrzeugs mit Batterie und Brennstoffzelle als Range Extender ist insofern eine Herausforderung, als dass je nach Betriebsmodus (Batteriebetrieb, Brennstoffzellenbetrieb, Leistungsverteilter Betrieb beider Energiespeicher) bei nicht ausreichender Reichweite zum Ziel unterschiedliche Lade- und Tankinfrastruktur berücksichtigt werden muss. Insbesondere der leistungsverteilte Betrieb, welcher aufgrund einer optimalen Energiebereitstellung durch Batterie und Brennstoffzelle am effizientesten ist, erfordert eine gute Planung der Lade- und Tankstopps, da sowohl Batterie als auch Brennstoffzelle zu jedem Zeitpunkt über ausreichend Energie verfügen müssen, um das Wirkungsgradoptimum erzielen zu können.

Die Problemstellung liegt in der Planung der Route unter Berücksichtigung notwendiger Lade- und Tankstopps bei minimal möglicher Erhöhung der Fahrtdauer, weshalb in diesem Beitrag die modellbasierte Entwicklung einer intelligenten Zielführung elektrischer Fahrzeuge mit Brennstoffzelle als Range Extender in vernetzten Verkehrssystemen zur Lösung der Problemstellung vorgestellt wird. Es wird vorausgesetzt, dass das Fahrzeug sich in einem digital vernetzten Verkehrssystem befindet, in dem Informationen über das aktuelle Verkehrsgeschehen und über den Status von Lade- und Tankinfrastruktur durch V2X-Kommunikation vorliegen.

In diesem Beitrag werden zunächst das methodische Vorgehen (Kap. 1) und der Stand des Wissens (Kap. 2) als Grundlage zur Konzeption der intelligenten Zielführung (Kap. 3) dargestellt. Daraufhin erfolgen die Modellierung des Straßennetzes und der Energiespeicher (Kap. 4) sowie die Auslegung der Zielführung (Kap. 5). Diese wird mittels Model-in-the-Loop-(MiL-), Software-

in-the-Loop-(SiL-) und Hardware-in-the-Loop-(HiL)-Simulationen erprobt (Kap. 6). Abschließend wird ein Resümee gezogen und ein Ausblick gegeben (Kap. 7).

1 Methodik

Zur Entwicklung der intelligenten Zielführung wird nach dem mechatronischen Entwicklungskreislauf (Figure 1), einer Methodik zur durchgängig modellbasierten Entwicklung und -absicherung mechatronischer Systeme nach [1], vorgegangen. Die Methodik zeichnet sich durch Iterationsmöglichkeiten zu jedem Zeitpunkt aus, sodass Fehler frühzeitig behoben werden und Entwicklungszeit sowie Entwicklungskosten gesenkt werden können.

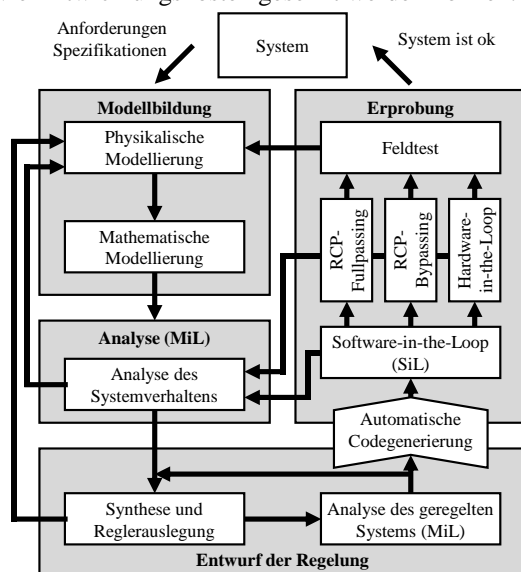


Figure 1: Mechatronischer Entwicklungskreislauf zur Entwicklung mechatronischer Systeme.

Der Entwicklungskreislauf beginnt mit der Definition von Anforderungen und Spezifikationen, die das System erfüllen muss. Mit diesen Kenntnissen wird ein physikalisches Modell des Systems aufgestellt, aus welchem ein mathematisches Modell abgeleitet wird. Die Modellparameter werden aus technischen Unterlagen oder Messungen an realen Komponenten ermittelt. Diesem Modellbildungsprozess folgt die Analyse des Systemverhaltens durch MiL-Simulationen und ggf. eine Änderung des zugrundeliegenden Modells. Nach erfolgreicher Analyse des Systemverhaltens erfolgt die Reglerauslegung mit der Reglersynthese und der Analyse des geregelten Systems ebenfalls durch MiL-Simulationen. Sobald das geregelte System erfolgreich analysiert wurde, erfolgt die automatisierte Codegenerierung des Reglers, sodass der

Regler ohne fehleranfällige manuelle Programmierung als Code vorliegt. Dieser Code wird in SiL-Simulationen abermals untersucht, bevor weitere Verifikationen und Optimierungen unter Echtzeitbedingungen im RCP-Full- oder Bypassing oder in HiL-Simulationen durchgeführt werden. Abschließend erfolgen Feldtests, die den mechatronischen Entwicklungskreislauf bei Erfüllung aller Anforderungen und Spezifikationen beenden.

2 Stand des Wissens

Dieses Kapitel stellt den Stand des Wissens zu Zielführungsverfahren, zur Integration dynamischer Informationen und zur Zielführung mit Zwischenzielen dar.

2.1 Verfahren zur Zielführung

Grundlage der Zielführung sind bewertete und gerichtete Graphen. Es existiert eine Vielzahl von Algorithmen zur Optimierung in Graphen mit unterschiedlichen Zielstellungen, von denen nur jene zum Auffinden optimaler Pfade zwischen zwei Knoten betrachtet werden.

Das bekannteste Verfahren zum Auffinden eines optimalen Pfades in einem bewerteten Graphen ist der von Dijkstra [2] vorgestellte Algorithmus. Es handelt sich um ein Verfahren der Breitensuche, welches immer die optimale Lösung bzgl. eines definierten Bewertungskriteriums findet, sofern der Graph keine negativen Kantenbewertungen beinhaltet. Nachteilig ist der hohe Rechenaufwand. Auf verwandte Algorithmen wie den Bellman-Ford-Algorithmus oder den Floyd-Warshall-Algorithmus soll nicht weiter eingegangen werden. Der Dijkstra-Algorithmus bildet die Grundlage vieler Ansätze, durch die seine Effizienz deutlich gesteigert wird. Für einen Überblick über diese Ansätze sei auf [3] verwiesen.

Neben diesen analytischen Ansätzen existieren auch stochastische Ansätze wie die von [4] beschriebene Ant Colony Optimization (ACO). Das Verfahren arbeitet schneller als exakte Algorithmen, allerdings kann das Auffinden der optimalen Route nicht analytisch nachgewiesen werden. Auch Ansätze der Künstlichen Intelligenz und des Reinforcement Learnings können wie von [5] dargestellt zur Zielführung eingesetzt werden.

Es existiert eine Vielzahl von Anwendungen dieser Verfahren zur Zielführung elektrischer Fahrzeuge, welche den Energieverbrauch ohne Lade- und Tankstopps optimieren. Stellvertretend sei der von [6] fokussierte Ansatz zur energieoptimierten Zielführung eines Elektrofahrzeugs mit Batterie und Superkondensator genannt.

2.2 Integration dynamischer Informationen

V2X-Nachrichten werden i.d.R. durch Anpassung des Graphen ähnlich wie Informationen aus dem Traffic Message Channel (TMC) verarbeitet. Diese können wie von [7] genutzt werden, um die eigene Zielführung zu optimieren und Staus zu umfahren, oder wie von [8] vorgeschlagen, um mithilfe der Zustandsdaten vieler Verkehrsteilnehmer den Verkehrsfluss zu optimieren und Stausituationen im Voraus zu vermeiden.

2.3 Zielführung mit Zwischenzielen

Ansätze der Tourenplanung, wie von [9] für Zustellfahrzeuge oder von [10] und [11] im Allgemeinen für logistische Systeme vorgestellt, ermitteln optimale Touren durch definierte Zwischenziele. Der Ansatz von [12] berücksichtigt die beschränkte Reichweite eines Elektronutzfahrzeugs bei der Tourenplanung, allerdings ohne weitere Ladestopps zu planen. In der Regel handelt es sich bei der Tourenplanung um Offline-Verfahren.

In [13] wird ein Ansatz zum Auffinden energieoptimaler Routen für batterieelektrische Fahrzeuge unter Berücksichtigung notwendiger Ladestopps dargestellt, bei dem es sich um eine Erweiterung eines Multi-Level-Dijkstra-Verfahrens handelt. Ziel ist die Minimierung der durch Ladung aufgenommenen Energie, um eine möglichst kurze Fahrtdauer zu erzielen. Auch Online-Rechner wie GoingElectric oder Apps wie Wattfinder und Next Plug schlagen Ladestopps vor, berücksichtigen allerdings ebenfalls keine Wasserstofftankstellen.

2.4 Zwischenfazit

Insgesamt lässt sich feststellen, dass eine Vielzahl von Ansätzen zur Zielführung existieren. Diese finden einen optimalen Pfad bzgl. eines definierten Bewertungskriteriums wie Fahrtdauer, Distanz oder Energieverbrauch, berücksichtigen aber keine Zwischenziele, sodass sie nicht zur Lösung der Problemstellung geeignet sind. Die im Bereich der Logistik verbreiteten Ansätze der Tourenplanung sind ebenfalls nicht geeignet, da nicht nur das Auffinden einer optimalen Route durch gegebene Zwischenziele, sondern auch die Wahl dieser Zwischenziele die Optimierungsaufgabe zur Lösung der Problemstellung darstellt. [13] stellt einen Ansatz zur Wahl dieser Zwischenziele für ein batterieelektrisches Fahrzeug dar, berücksichtigt allerdings nicht die Planung von Tankstopps für den zweiten Energieträger Wasserstoff.

Keiner der dargestellten Ansätze ist zur vollständigen Lösung der Problemstellung geeignet.

3 Konzeption der intelligenten Zielführung

In diesem Abschnitt werden Anforderungen an die intelligente Zielführung erhoben und ein Konzept der intelligenten Zielführung aufgestellt.

3.1 Anforderungen

1. Das Kartenmaterial des Verkehrssystems muss in einer geeigneten mathematischen Beschreibungsform vorliegen, um die Zielführung zu ermöglichen. Es muss Informationen über Fahrtdauer, Strecke, Höhenprofil, Energieverbrauch sowie Lade- und Tankinfrastruktur beinhalten, um diese bei der Zielführung berücksichtigen zu können.
2. Durch die Zielführung soll je nach Fahrer-/Insassenwunsch die kürzeste, schnellste oder energieeffizienteste Route zum Fahrtziel bestimmt werden.
3. Aktuelle Informationen über das Verkehrsgeschehen und die Verfügbarkeit von Infrastruktur aus der V2X-Kommunikation sollen berücksichtigt werden können.
4. Durch die Zielführung müssen bei Bedarf Lade- und Tankstopps geplant werden, um das Fahrtziel erreichen zu können. Diese Planung soll drei Betriebsmodi (Batteriebetrieb, Brennstoffzellenbetrieb und leistungsverteilten Betrieb) ermöglichen.
5. Für den leistungsverteilten Betrieb muss die Zielführung Lade- und Tankstopps derart planen, dass sowohl Batterie als auch Wasserstoffspeicher immer über ausreichend Energie verfügen, um beide Energiequellen im Wirkungsgradoptimum betreiben zu können.

3.2 Konzept der intelligenten Zielführung

In Figure 2 ist das Konzept der intelligenten Zielführung als Funktionsstruktur dargestellt. Kern der Zielführung ist das statische Kartenmaterial, welches in geeigneter Form bereitgestellt werden muss. Es wird das frei zugängliche Kartenmaterial der OpenStreetMap [14] vorgesehen, welchem Höhendaten und verlässliche Daten über Ladesäulen und Wasserstofftankstellen hinzugefügt werden. Die Informationen aus der Ermittlung von Fahrzeugzuständen, der Umfeldperzeption und der V2X-Kommunikation mit anderen Verkehrsteilnehmern sowie der Infrastruktur werden genutzt, um das statische Kartenmaterial zu aktualisieren und es um dynamische Informationen zu erweitern. Des Weiteren werden GPS-Daten aus der Zustandserfassung sowie die Ergebnisse der Umfeldperzeption in SLAM-Verfahren genutzt, um die

Selbstlokalisierung des Ego-Fahrzeugs durchzuführen, welche den Startpunkt der Zielführung darstellt. Basierend auf der dynamischen Karte, der Ego-Position und dem Fahrtziel wird durch die Zielführung, unter Berücksichtigung eines gewählten Betriebsmodus, eine bzgl. Fahrdauer, Distanz oder Energieverbrauch optimierte Route ermittelt. Etwaige notwendige Lade- oder Tankstopp werden ebenfalls durch die Zielführung geplant.

In diesem Beitrag werden aus der Funktionsstruktur die Bereitstellung des Kartenmaterials (Kap. 4.1 bis 4.3), die Kartenaktualisierung durch V2X-Informationen (Kap. 5.1) sowie die Zielführung (Kap. 5.2 und 5.3) betrachtet.

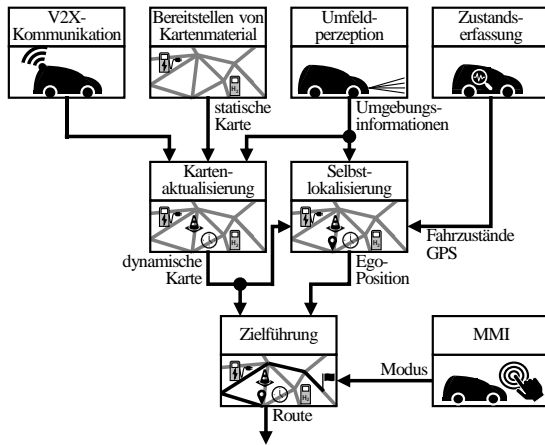


Figure 2: Funktionsstruktur der intelligenten Zielführung.

4 Modellbildung

Dieses Kapitel beschreibt die Modellbildung des Straßennetzes als Datengrundlage der Zielführung und die Modellierung der Energiespeicher, welche zur Planung der Lade- und Tankstops benötigt werden.

4.1 Beschreibung des Straßennetzes

Das Straßennetz wird als Graph $G(N, E)$ beschrieben, welcher nach [15] aus einer Menge N von Knoten n_i und einer Menge E von Kanten $e_{a,b}$, die jeweils zwei Knoten n_a und n_b miteinander verbinden, besteht. Die Kanten des Graphen werden sowohl gerichtet als auch bewertet, da durch die Richtung der Kanten einerseits die Berücksichtigung von Einbahnstraßen oder Sperrungen einzelner Fahrtrichtungen und durch die Bewertung der Kanten andererseits die Berücksichtigung von Straßenlänge, Fahrdauer oder Energieverbrauch ermöglicht wird.

Dem Graphen des Straßennetzes werden die frei zugänglichen Höhendaten der NASA aus der Shuttle Radar Topography Mission (SRTM, [16]) überlagert, um den

Einfluss der Fahrbahnneigung auf den Energieverbrauch abbilden zu können. Zur Veranschaulichung des aus der OSM abgeleiteten Graphen mit überlagerten SRTM-Daten ist dieser im Umfeld des Hauptcampus der Ostfalia Hochschule in Wolfenbüttel in Figure 3 dargestellt.

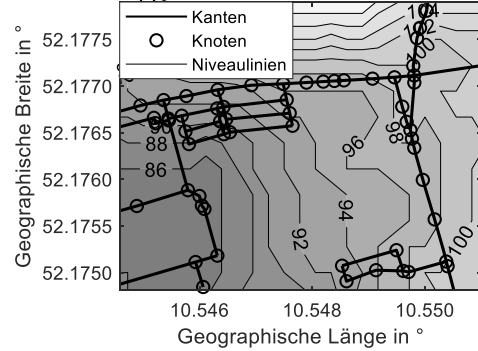


Figure 3: Überlagerung von Graph und Höhenprofil an der Ostfalia in Wolfenbüttel.

4.2 Bewertung des Graphen

Die Kanten des Graphen werden mit der Distanz s , der Dauer t und der Energie E bewertet, welche zum Übergang eines Knotens a zum Knoten b anfallen. Jeder Kante $e_{a,b}$ wird entsprechend eine Bewertung

$$\underline{w}_{a,b} = [s_{a,b} \quad t_{a,b} \quad E_{a,b}]^T \quad (1)$$

zugewiesen. Die Distanz

$$s_{a,b} = 2r \cdot \arcsin(\sqrt{c_{a,b}}) \quad (2)$$

mit

$$c_{a,b} = \sin^2\left(\frac{\Delta\varphi_{a,b}}{2}\right) + \cos(\varphi_a) \cos(\varphi_b) \sin^2\left(\frac{\Delta\lambda_{a,b}}{2}\right) \quad (3)$$

wird nach den von [17] vorgestellten Gleichungen (2) und (3) mithilfe der Längengradsdifferenz $\Delta\lambda_{a,b}$, der Breitengrade φ_a und φ_b , der Breitengradsdifferenz $\Delta\varphi_{a,b}$ sowie dem Erdradius r berechnet. Die Dauer

$$t_{a,b} = t_{\min,a,b} + t_{B,a,b} + t_{W,a,b} + t_{V,a,b} \quad (4)$$

basiert u. a. auf der minimalen Fahrdauer $t_{\min,a,b}$, die sich für die Fahrt mit der zulässigen Höchstgeschwindigkeit ergibt. Des Weiteren werden Zeitaufschläge für Beschleunigungs- und Bremsvorgänge $t_{B,a,b}$, für Wartezeiten $t_{W,a,b}$, bspw. an Kreuzungen oder Lichtsignalanlagen, sowie für verkehrsbedingtes Unterschreiten der Höchstgeschwindigkeit $t_{V,a,b}$ berücksichtigt. Die Energie

$$E_{a,b} = f(v_{a,b}, a_{a,b}, \alpha_{a,b}) + P_{BN} \cdot t_{a,b} \quad (5)$$

wird mithilfe eines inversen Fahrzeug- und Antriebsstrangmodells sowie dem Leistungsbedarf des Bordnetzes P_{BN} und der Dauer $t_{a,b}$ abgeschätzt. Wesentliche Einflussgrößen sind die prognostizierte Geschwindigkeit

$v_{a,b}$, welche verkehrsbedingt nicht zwingend der Höchstgeschwindigkeit entspricht, die Fahrzeugbeschleunigung $a_{a,b}$ sowie die Fahrbahnsteigung $\alpha_{a,b}$.

4.3 Lade- und Tankinfrastruktur

Informationen über die Ladesäulen- und Wasserstofftankstelleninfrastruktur der OSM sind nicht zuverlässig, da sie ohne Qualitätskontrolle von Nutzern gepflegt werden. Für die Zielführung allerdings sind aktuelle Informationen über die Infrastruktur elementar, da fehlerhafte Informationen zum Ausbleiben notwendiger Lade- bzw. Tankstopps und somit zum Liegenbleiben des Fahrzeugs führen können. Aus diesem Grund werden die Ladesäulen der von der Bundesnetzagentur gepflegten Ladesäulenkarte [18] entnommen. Diese enthält deutschlandweit alle Ladesäulen, welche nach der Ladesäulenverordnung als öffentlich zugänglich gemeldet sind. Informationen über die europaweite, vergleichsweise sehr überschaubare, Wasserstofftankinfrastruktur werden von H2 Mobility bezogen [19].

Die Integration der Infrastrukturdaten erfolgt durch Zuordnung zu entsprechenden Knoten des Graphen. Des Weiteren werden diese Knoten und der Status der Infrastruktur in einer Infrastrukturliste eingetragen, welche als Grundlage der infrastruktureloptimierten Zielführung (vgl. Kap. 5.3) dient. Das Ergebnis der Integration der Infrastruktur in den Graphen ist in Figure 4 dargestellt. Der Übersichtlichkeit halber ist nur das Straßennetz der deutschen Bundesautobahn (BAB) mit der nahe der BAB gelegenen Lade- und Tankinfrastruktur eingezeichnet.

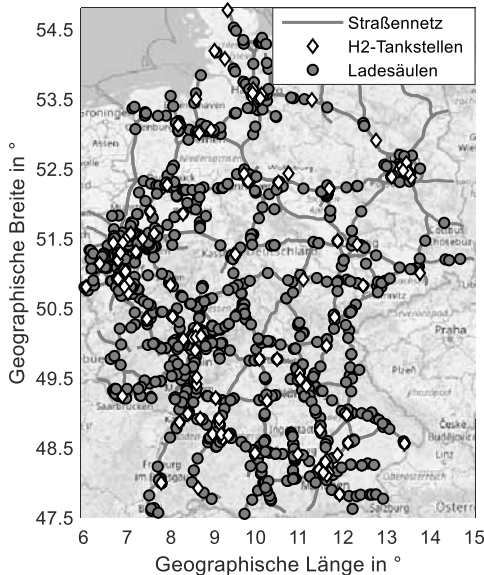


Figure 4: Straßennetz der BAB mit nahegelegener Lade- und Tankinfrastruktur.

4.4 Modellierung der Energiespeicher

Wie in der Motivation beschrieben, kann die Fahrzeugleistung durch eine Batterie und eine Brennstoffzelle aufgebracht werden, sodass sich die für die Fahrt aufgebrauchte Energie für den Übergang von Knoten a zu b

$$E_{a,b} = E_{\text{bat},a,b} + E_{\text{bz},a,b} \quad (6)$$

aus der Energie der Batterie $E_{\text{bat},a,b}$ und der Brennstoffzelle $E_{\text{bz},a,b}$ zusammensetzt. Mithilfe des Faktors

$$k_{\text{mod}} = \begin{cases} 0 & \text{für Brennstoffzellenbetrieb} \\]0,1[& \text{für leistungsverteilten Betrieb} \\ 1 & \text{für Batteriebetrieb} \end{cases} \quad (7)$$

werden die Energie von Batterie

$$E_{\text{bat},a,b} = k_{\text{mod}} E_{a,b} \quad (8)$$

und Brennstoffzelle

$$E_{\text{bz},a,b} = (1 - k_{\text{mod}}) E_{a,b} \quad (9)$$

aus der Gesamtenergie berechnet. Zur Abschätzung der Ladestandsänderung der Batterie und der Massenänderung im Wasserstofftank wird vereinfachend angenommen, dass sowohl Batterie als auch Brennstoffzelle konstant mit Nennspannung betrieben werden. Der Ladestand am Knoten b

$$SOC_{\text{bat},b} \cong SOC_{\text{bat},a} + \frac{E_{\text{bat},a,b}}{\eta_{\text{bat}} C_{\text{bat}}} \quad (10)$$

ergibt sich somit in Abhängigkeit des Startladestandes SOC_a , der Energie $E_{\text{bat},a,b}$, des Coloumb'schen Wirkungsgrades η_{bat} sowie der Nennkapazität C_{bat} . Analog ergibt sich die vorhandene Wasserstoffmasse

$$m_{\text{H}_2,b} \cong m_{\text{H}_2,a} + a_{\text{bz}} \frac{E_{\text{bz},a,b}}{u_{\text{bz}}} \quad (11)$$

abhängig von der Energie $E_{\text{bz},a,b}$, der Nennspannung u_{bz} sowie der Konstante a_{bz} , welche den Massebedarf nach dem Faraday'schen Gesetz sowie die Zellanzahl des Brennstoffzellenstacks und die Stöchiometrie beinhaltet.

5 Auslegung der Zielführung

Dieses Kapitel stellt die Auslegung der Zielführung vor.

5.1 Integration dynamischer Informationen

Die Einbindung dynamischer Informationen aus der V2X-Kommunikation wird durch eine temporäre Anpassung der Kantenbewertungen im Graphen oder eine Statusaktualisierung in der Infrastrukturliste realisiert.

Zur Anpassung der Kantenbewertungen wird aus der GPS-Position der V2X-Nachricht zunächst ermittelt,

welcher Kante die Informationen zuzuweisen ist. Anschließend erfolgt die Änderung der Kantengewichte je nach Art der Information. Tritt bspw. eine Straßensperre infolge einer Baustelle auf, werden für die entsprechende Kante sowohl Distanz als auch Dauer und Energieverbrauch auf unendlich gesetzt, sodass die Kante unabhängig der Gewichtungsfaktoren nicht mehr Bestandteil der optimalen Route sein kann. Tritt hingegen eine Verzögerung durch stockenden Verkehr auf, werden nur Dauer und Energieverbrauch der Kante geändert, da sich die Streckenlänge nicht ändert.

5.2 Optimale Zielführung

Der grundlegende Algorithmus zur optimalen Zielführung sucht einen optimalen Pfad durch den Graphen des Straßennetzes vom Start- zum Zielknoten. Als Optimierungsverfahren zur Bestimmung dieses optimalen Pfades wird der Suchalgorithmus nach Dijkstra gewählt. Mithilfe der Kostenfunktion

$$J_b(J_a, w_{a,b}) = J_a + [g_s \quad g_t \quad g_E] w_{a,b} \quad (12)$$

werden die Kosten J_b nach einem Übergang von Knoten a zu Knoten b abhängig von den Kosten J_a und der Kantenbewertung $w_{a,b}$ bestimmt. Mithilfe der Gewichtungsfaktoren g_s , g_t und g_E wird definiert, ob die Route bzgl. der Distanz, der Fahrtdauer, des Energieverbrauchs oder eines Kompromisses dieser Kriterien optimiert werden soll. Die Gesamtkosten jedes Knotens werden mit Start des Algorithmus bis auf Ausnahme des Startknotens, welcher mit null initialisiert wird, mit unendlich initialisiert. Der Algorithmus bestimmt ausgehend vom Startknoten die Kosten der unbesuchten Nachbarknoten nach Gl. 12 und aktualisiert die Gesamtkosten der Nachbarknoten, wenn die neu berechneten Kosten kleiner als die vorherigen Kosten sind. Alle betrachteten Nachbarknoten werden in eine Warteliste aufgenommen, der Startknoten wird als besucht markiert und der Knoten mit den geringsten Kosten wird aus der Warteliste ausgewählt. Von diesem Knoten ausgehend wiederholt sich das beschriebene Vorgehen, bis der Zielknoten erreicht und der optimale Pfad ermittelt wurde.

5.3 Infrastrukturoptimierte Zielführung

Die infrastrukturoptimierte Zielführung wird relevant, wenn das Fahrtziel nicht ohne Lade- oder Tankstopps erreicht werden kann.

In Figure 5 ist der Programmablaufplan (PAP) der infrastrukturoptimierten Zielführung für alle Betriebsmodi dargestellt. Zunächst erfolgt eine Aktualisierung

des Kartenmaterials und der Infrastrukturliste durch V2X-Informationen, bevor die in Kap. 5.2 vorgestellte Zielführung vom Start zum Ziel durchgeführt wird. Das Ziel kann erreicht werden, wenn im Batteriebetrieb der Ladestand am Ziel größer als null ist, im Brennstoffzellenbetrieb die verfügbare Wasserstoffmasse im Tank am Ziel größer als null ist und im leistungsverteilten Betrieb sowohl der Ladestand als auch die verfügbare Wasserstoffmasse im Tank am Ziel größer als null sind.

Sind diese Bedingungen nicht erfüllt, kann das Ziel nicht ohne zusätzliche Energieaufnahme erreicht werden. Deshalb wird zunächst die unbesuchte Lade- und Tankinfrastruktur innerhalb der Reichweite ermittelt. Zur Reduktion der Menge an Ladesäulen und Tankstellen werden nur diejenigen betrachtet, an denen Restladestand bzw. Restwasserstoffmasse definierte Grenzwerte unterschreiten, sodass eine möglichst hohe Energieaufnahme erzielt wird. Die übrig gebliebene Menge der Lade- und Tankinfrastruktur wird, wenn nicht bereits geschehen, in eine Warteliste eingetragen. Hierbei wird den Kosten ein Zeitaufschlag für den Lade- oder Tankvorgang hinzugefügt. Aus der Warteliste wird ein Lade- oder Tankstopp als neuer Startpunkt ausgewählt, von dem aus die Zielführung zum ursprünglichen Ziel erfolgt. Dieses Prozedere wiederholt sich bis die Warteliste leer ist. Abschließend wird die beste Route bzgl. eines definierten Zielkriteriums (vgl. Gl. 12) ausgewählt.

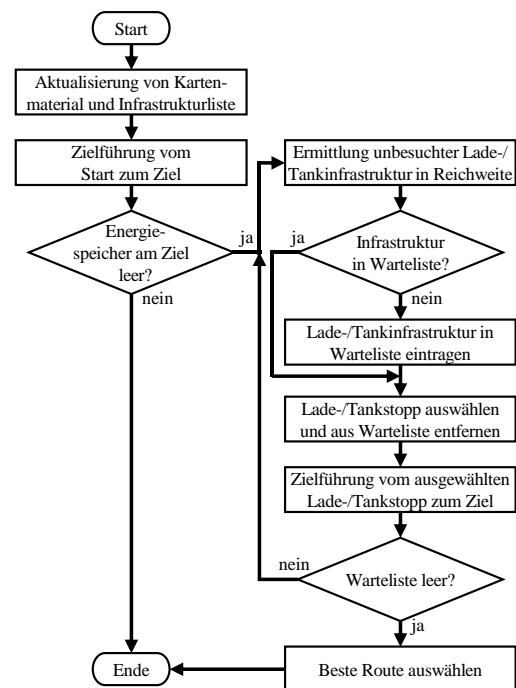


Figure 5: PAP der infrastrukturoptimierten Zielführung zur Planung von Tank- und Ladestops.

6 Erprobung

Dieses Kapitel stellt exemplarische Ergebnisse der Erprobung in MiL-, SiL- und HiL-Simulationen vor.

6.1 MiL- und SiL-Simulationen

In diesem Abschnitt werden offline erzielte Ergebnisse aus MiL- und SiL-Simulationen vorgestellt.

In Figure 6 sind die infrastrukturoptimierten Routen von Hamburg nach München für die drei Betriebsmodi mit geplanten Lade- und Tankstopps für ein Fahrzeug mit einer Batteriekapazität von 42 kWh und einem Speichervermögen von 2,5 kg Wasserstoff dargestellt.

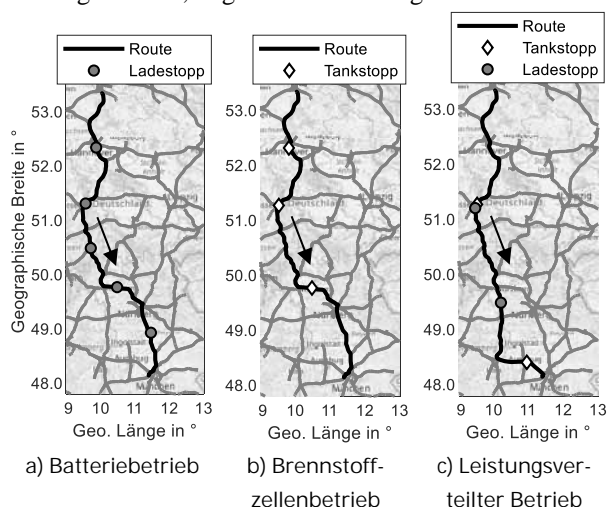


Figure 6: Gegenüberstellung infrastrukturoptimierter Routen von Hamburg nach München für den Batteriebetrieb (a), den Brennstoffzellenbetrieb (b) und den leistungsverteilten Betrieb (c).

Im Batteriebetrieb wurde eine Strecke von 761 km in insgesamt 11,2 h, im Brennstoffzellenbetrieb eine Strecke von 770 km in 8,1 h und im leistungsverteilten Betrieb eine Strecke von 790 km in 9,4 h zurückgelegt.

Im Batteriebetrieb ergibt sich erwartungsgemäß aufgrund der langen Ladezeiten die längste Fahrtdauer, allerdings kann aufgrund der verhältnismäßig hohen Anzahl an Ladesäulen die kürzeste Strecke ohne Umwege zurückgelegt werden. Im Brennstoffzellenbetrieb wird die gleiche Route wie im Batteriebetrieb gewählt, allerdings werden aufgrund der geringen Anzahl an Wasserstofftankstellen kleine Umwege zu diesen nötig. Die Fahrtdauer ist am geringsten, da der Tankvorgang deutlich schneller als das Laden vonstattengeht. Im leistungsverteilten Betrieb resultiert die Fahrzeit aus der Kombi-

nation schneller Tankvorgänge mit langsameren Ladevorgängen. Insgesamt wurde in diesem Modus eine längere Strecke zurückgelegt, allerdings wurden die Energieverluste in der Leistungsbereitstellung um 52,81 % verglichen zum Batteriebetrieb und um 44,59 % verglichen zum Brennstoffzellenbetrieb reduziert. Weitere Informationen zu Einsparpotentialen durch den leistungsverteilten Betrieb können [20] entnommen werden.

6.2 HiL-Simulation

Mithilfe eines mobilen HiL-Prüfstandes (vgl. [21]) wurde die Zielführung unter Einbezug dynamischer Informationen aus der V2X-Kommunikation online unter Echtzeitbedingungen mit realer Hardware getestet.

Die Versuche wurden ohne und mit Störung durch eine Baustelle auf dem Innenhof der Ostfalia Hochschule in Wolfenbüttel durchgeführt. Die Ergebnisse sind in Figure 7 dargestellt. Es ist offensichtlich, dass die Route ohne Baustelle die optimale Route ist. Es lässt sich feststellen, dass die Zielführung in der Echtzeitumgebung des mobilen HiL-Prüfstandes korrekt funktioniert.

Daraufhin wird eine V2X-Nachricht empfangen, welche eine Sperrung der optimalen Route durch eine Baustelle mitteilt. Die Zielführung verarbeitet diese Information und wählt die kürzeste Alternativroute. Dieses trivial wirkende Ergebnis zeigt, dass zum einen die V2X-Kommunikation und zum anderen die Integration der V2X-Nachricht in die Zielführung (vgl. Kap. 5.1) funktioniert und eine Neuberechnung der Route einleitet.

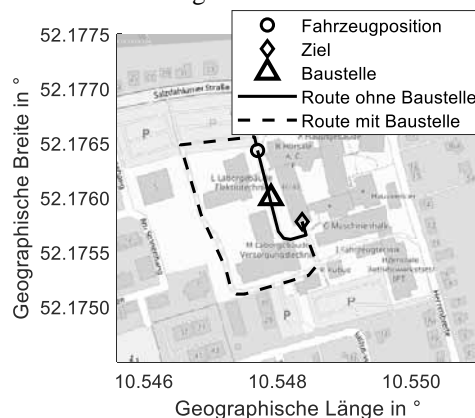


Figure 7: Ergebnisse der Online-Zielführung unter Echtzeitbedingungen auf dem Innenhof der Ostfalia in Wolfenbüttel.

7 Resümee und Ausblick

Dieser Beitrag stellt die Auslegung einer intelligenten Zielführung für Elektrofahrzeuge mit Brennstoffzelle als

Range Extender in vernetzten Verkehrssystemen dar.

Es wurde eine Schnittstelle zur Integration dynamischer Informationen aus der V2X-Kommunikation vorgesehen, durch welche das Kartenmaterial stetig aktualisiert wird. Die Zielführung ermöglicht einen optimierten Betrieb bzgl. Fahrtdauer, Fahrtstrecke, Energieverbrauch und einem Kompromiss dieser Kriterien. Je nach Betriebsmodus (Batteriebetrieb, Brennstoffzellenbetrieb, Leistungsverteilter Betrieb) werden notwendige Lade- oder Tankstopps eingeplant, sollte die Energie zum Erreichen des Ziels nicht ausreichen. Die komplette Zielführung wurde offline im Rahmen von MiL- und SiL-Simulationen erprobt. Die V2X-Schnittstelle und die Zielführung ohne Berücksichtigung der Infrastruktur wurden des Weiteren online in einer HiL-Simulation unter Echtzeitbedingungen erprobt.

Zukünftige Arbeitsschwerpunkte liegen zum einen in weiteren Untersuchungen und Optimierungen mittels MiL-, SiL- und HiL-Simulationen und zum anderen in der Realisierung der infrastruktureloptimierten Zielführung als online-Verfahren in einer Echtzeitumgebung. Hierzu werden Optimierungspotentiale in der Reduktion des zugrundeliegenden Graphen des Straßennetzes, in Maßnahmen zur Effizienzsteigerung des Dijkstra-Algorithmus zur Zielführung und in einer Reduktion der Zielführungsaufrufe durch die infrastruktureloptimierte Zielführung untersucht.

Danksagung

Dieser Beitrag wurde im Rahmen des Verbundprojekts *Zukünftige Fahrzeugtechnologien im Open Region Lab* durch das Niedersächsische Ministerium für Wissenschaft und Kultur sowie die VolkswagenStiftung unter dem Förderkennzeichen VWZN3236 gefördert.



Niedersächsisches Ministerium
für Wissenschaft und Kultur



VolkswagenStiftung

Referenzen

- [1] Liu-Henke X. *Mechatronische Entwicklung der aktiven Feder-/Neigetechnik für das Schienenfahrzeug RailCab*. VDI-Fortschritt-Berichte, Reihe 12, Nr. 589, VDI-Verlag, Düsseldorf, Germany, 2004.
- [2] Dijkstra E W. *A Note on Two Problems in Connexion with Graphs*. Numerische Mathematik. S. 269-271, 1959.
- [3] Bast H et al. *Route Planning in Transportation Networks*, Lecture Notes in Computer Science, Vol. 9220, S. 19-80, 2016.
- [4] Dorigo M et al. *Ant algorithms for discrete optimization*. Artificial Life, Vol. 5, Nr. 2, S. 137-172, 1999.
- [5] Yu J, Yu W, Gu J. *Online Vehicle Routing With Neural Combinatorial Optimization and Deep Reinforcement Learning*. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, Nr. 10, October 2019.
- [6] Jurik T et al. *Energy Optimal Real-Time Navigation System*. IEEE Intelligent Transportation Systems Magazine Vol. 6, Nr. 3, S. 66-79, 2014.
- [7] Wang J et al. *Dynamic Route Choice Prediction Model Based on Connected Vehicle Guidance Characteristics*. Journal of Advanced Transportation, 2017.
- [8] Backfrieder C et al. *Increased Traffic Flow Thorough Node-Based Bottleneck Prediction and V2X Communication*. IEEE Transactions on Intelligent Transportation Systems, Vol. 18, Nr. 2, S. 349-363, February, 2017.
- [9] Yu J, Yu W, Gu J. *Online Vehicle Routing With Neural Combinatorial Optimization and Deep Reinforcement Learning*. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, Nr. 10, October, 2019.
- [10] Fleischmann B, Kopfer H. *Transport- und Tourenplanung*. In: Tempelmeier H. *Planung logistischer Systeme*. Springer Vieweg, Berlin, 2018.
- [11] Wenger W. *Multikriterielle Tourenplanung*. Dissertation, Universität Hohenheim. Gabler, Wiesbaden, 2010.
- [12] Witte C, Marner, T. *Tourenplanung mit Elektronutfahrzeugen – ein GAMS-Modell*. In Proff H, Fojcik T. *Nationale und internationale Trends in der Mobilität*. Springer Gabler, Wiesbaden, 2016.
- [13] Baum M et al. *Energy-Optimal Routes for Battery Electric Vehicles*. Algorithmica, Vol. 82, Nr. 5, S. 1490-1546, 2020.
- [14] OpenStreetMap Foundation. *OpenStreetMap*. Stand: 06.09.2020. <https://www.openstreetmap.org>
- [15] Turau V, Weyer C. *Algorithmische Graphentheorie*. Auflage, De Gruyter, 2015.
- [16] Farr T G. *The Shuttle Radar Topography Mission*. Reviews of Geophysics, 45, RG2004, 2007.
- [17] Gade K. *Non-singular Horizontal Position Representation*. Journal of Navigation, Vol. 63, Nr. 3, S. 395-417, 2010.
- [18] Bundesnetzagentur. *Ladesäulenkarte*. Stand: 07.08.2020.
- [19] H2 Mobility Deutschland GmbH und Co. KG. *H2.live*. Stand: 06.09.2020. <https://h2.live/tankstellen>
- [20] Scherler S et al. *Predictive Energy Management for an Electric Vehicle with Fuel Cell Range Extender in Connected Traffic Systems*. 19th IEEE Mechatronika, Prague, Czech Republic, December 2 - 4, 2020. (tbp)
- [21] Scherler S, Liu-Henke X. *Conception and Realization of a Mobile HiL Test Bench for V2X Communication*. IEEE 91st Vehicular Technology Conference, Antwerp, Belgium, May 25-28, 2020.

Konfliktfreie, selbstoptimierte Trajektorienplanung für ein fahrerloses Transportfahrzeug zur Durchführung des autonomen Gütertransportes im Produktionsumfeld

Jie Zhang*, Xiaobo Liu-Henke

Institut für Mechatronik, Fakultät Maschinenbau, Ostfalia Hochschule für angewandte Wissenschaften, Salzdhallumer Str. 46/48, 38302 Wolfenbüttel, Deutschland; *jie.zhang@ostfalia.de

Abstract. Im vorliegenden Beitrag wird ein Ansatz zur Trajektorienplanung, die aus Bahnplanung und Bewegungsplanung besteht, für ein fahrerloses Transportfahrzeug (FTF), welches den autonomen Gütertransport im Produktionsumfeld konfliktfrei durchführen kann, dargestellt. Anhand von Wegpunkten aus einer Navigationsfunktion wird zuerst ein kontinuierlicher Fahrweg generiert, welche der optimalen Route in Form von geradlinigen Strecken zwischen den Wegpunkten möglichst nahekommt. Anschließend wird die Bewegungsplanung des FTF unter Berücksichtigung von Geschwindigkeitsbeschränkungen bestimmt. Auf Basis von omnidirektionaler Manövrierbarkeit des FTF wird ein Mechanismus zur Konflikterkennung und -lösung entwickelt. Die aus diesem Ansatz generierte konfliktfreie Trajektorie wird als Führungsgröße zur Fahrdynamikregelung übertragen, sodass der Gütertransport ausgeführt werden kann.

Einleitung

Im vom Europäischen Fonds für regionale Entwicklung (EFRE) geförderten Verbundprojekt *Methoden und Werkzeuge für die synergetische Konzipierung und Bewertung von Industrie 4.0-Lösungen (Synus)*, an dem insgesamt fünf Professoren der Technischen Universitäten Braunschweig und Clausthal sowie der Ostfalia Hochschule für angewandte Wissenschaften beteiligt sind, wird ein modellgestütztes Werkzeug entwickelt, mit welchem finanzieller Aufwand und Nutzen von Industrie 4.0-Lösungen in kleinen und mittleren Unternehmen (KMU) bewertet werden sollen, um KMU bei deren Einführung beratend zu unterstützen. Schwerpunkt im Teilprojekt *Modellbasierte Konzeption und Bewertung von Industrie 4.0-Lösungen zur Vernetzung mechatronischer Komponenten in Produktionsanlagen durch Digitalisierung (MiMec)* der Ostfalia Hochschule ist die Modellierung und Simulation der aus vernetzten mechatronischen

Komponenten bestehenden Industrieanlagen und verfügbaren I4.0-Lösungen sowie die systematische Integration autonomer Transportfahrzeuge durch vollständige digitale Vernetzung in intelligenten, cyber-physischen Produktionsanlagen.

Im vorliegenden Beitrag wird zuerst der Ansatz zur konfliktfreien Trajektorienplanung auf Basis des neuen Konzeptes eines fahrerlosen Transportfahrzeuges (FTF) der Ostfalia ausführlich dargestellt.

1 Motivation

Eine der größten zukünftigen Herausforderungen der Produktion ist die zuverlässig zu terminierender Herstellung von variantenreichen bis hin zu kundenindividuellen Produkten in kleinsten Serien oder gar in Einzelfertigung [1]. Die Fertigung muss flexibel auf diese kleinen Losgrößen reagieren können [2]. Als essenzieller Teil einer gesamten Produktionsanlage sollen die FTF zum Gütertransport auch die Anforderungen bezüglich der Flexibilität erfüllen. Diese Anforderungen werden sowohl im mechanischen Konzept als auch in der modellbasierten Funktionsentwicklung berücksichtigt.

Die meisten heutzutage in der Industrie eingesetzten FTF navigieren mit Hilfe von im Boden verbauten induktiven, magnetischen oder optischen Leitlinien. Es ist diesen FTF somit nicht möglich beliebige befahrbare Bereiche des Produktionsumfeldes zu erreichen und die Bewegungsflexibilität solcher FTF wird beträchtlich begrenzt. Für die Routenplanung des FTF wird das Verfahren „ter Mors“ [3] verwendet. Dieses Verfahren überprüft zuerst den freien Zeitslot eines Wegs, für den keine Reservierungen vorliegen und der groß genug ist, damit ein FTF den Abschnitt innerhalb dieses Zeitintervalls passieren kann. Anschließend wird die Route in Form von Zeitslots in bestimmter Reihenfolge an das entsprechende FTF

übergeben [4]. Der große Nachteil dieses Verfahrens liegt darin, dass der Auslastungsgrad des Wegs niedrig und die Durchführungszeit eines Auftrags nur schwierig zu senken ist, weil der Weg von maximal einem FTF innerhalb eines bestimmten Zeitintervalls durchfahren werden darf.

Um die Bewegungsflexibilität des FTF zu optimieren und den autonomen Gütertransport zu untersuchen, wird im Rahmen des Projekts an der Ostfalia ein FTF als Forschungsträger entwickelt und die entsprechenden Funktionen modellbasiert ausgelegt. Basierend auf den Funktionsmodulen kann das FTF die Trajektorie als Sollwert für die Fahrdynamikregelung generieren. Mithilfe IoT-basierter Kommunikationstechnologie z.B. WLAN lassen sich Bewegungskonflikte der FTF lösen.

2 Stand des Wissens

Zentraler Bestandteil des autonomen Fahrbetriebs ist neben dem Folgen einer ausgewählten Route, die effiziente Trajektorienplanung. Auf Basis einer Folge von Knotenpunkten, die von Navigationsalgorithmen erzeugt werden, wird eine kontinuierliche Route erzeugt, die dann als Führungsgröße einer Folgeregelung dient. Eine Trajektorienplanung beinhaltet sowohl die Bahn- als auch die Bewegungsplanung. Während es bei der Bahnplanung um das reine Finden eines zulässigen Weges geht, so berücksichtigt die Bewegungsplanung die dynamischen Restriktionen des sich bewegenden Objekts und den zeitlichen Verlauf der Bewegung [5]. Hierbei lässt sich die Problemstellung der Trajektorienplanung in zwei Teilprobleme zerlegen: Erzeugung eines Fahrwegs mithilfe der Knotenpunkte aus den Navigationsalgorithmen und Bewegungsplanung unter Berücksichtigung der Geschwindigkeitsbeschränkung des FTF.

2.1 Bahnplanung

Ausgang der Navigationsalgorithmen ist in jedem Falle eine Liste von Knotenpunkten, die den optimalen, meist kürzesten, Weg darstellt und in einer bestimmten Reihenfolge durchfahren werden soll. Der Folgeregler könnte nun diese Liste als Reihe von Sprungfunktionen in die entsprechenden Raumrichtungen als Sollwerte bekommen, was aber zu einem schlechten dynamischen Übergangsverhalten führen würde, da der Folgevorgang immer die systemdynamikabhängige Sprungantwort des Regelkreises wäre. Besser ist es, die Sollgröße kontinuierlich nachzuführen und so z.B. für das Durchfahren von

zwei Knoten eine Rampenfunktion und keine Sprungantwort zu bekommen. Die Abfolge der Knotenpunkte muss vorab über eine Funktion angenähert werden.

Eine einfache Form des Fahrwegs kann durch Kombination von Geraden und Kreisbögen erzeugt werden. In [6, 7] wird ein exaktes Verfahren vorgeschlagen, um für einen Verbindungspunkt den kürzesten, aus Geraden und Kreisbögen mit festem Radius r bestehenden, Weg vom Start zum Ziel zu finden. Eine weitere Form zur Berechnung des Fahrwegs kann durch Polynomfunktionen erfolgen. In [8] wird ein exaktes Verfahren zur Erzeugung eines Fahrwegs vorgeschlagen, bei dem mehrere Knotenpunkte mit beliebigem Winkel zueinander zwischen Start und Ziel durchfahren werden.

2.2 Bewegungsplanung

Es ist nicht verwunderlich, dass über die Zeit eine Bandbreite unterschiedlicher Lösungsansätze für die Problemstellung der Bewegungsplanung autonomer Fahrzeuge entwickelt worden ist. In [9, 10] wird ein modellprädiktives Verfahren verwendet, um die Bewegungsplanung genauer zu gestalten und die Realisierbarkeit der Trajektorie zu gewährleisten. Durch die Integration des dynamischen Modells in die Planung können die Geschwindigkeiten des Fahrzeugs entlang des Fahrwegs bestimmt und die Stellgrößen in einigen Fällen direkt im Sinne einer modellprädiktiven Regelung auf die Regelstrecke angewandt werden.

Gegenüber dem modellprädiktiven Verfahren, für welches bei der Realisierung eine präzise Modellierung des dynamischen Verhaltens des Systems notwendig ist und damit zum hohen Rechenaufwand führt, kann die Planung der Geschwindigkeit bzw. der Bewegung durch diskrete Bahnanalyse einfacher realisiert werden. Nach der Diskretisierung des bereits bekannten Fahrwegs wird das Geschwindigkeitsprofil unter Berücksichtigung der Beschränkung der Geschwindigkeit bestimmt.

3 Methodik

Um die Komplexität bei der Entwicklung vernetzter mechatronischer Systeme zu beherrschen, ist eine klare Strukturierung des Gesamtsystems gemäß der mechatronischen Entwicklungsmethodik [11] erforderlich.

Die Modularisierung und Hierarchisierung stehen im Mittelpunkt der Strukturierung. Das bedeutet ein großes System wird in intelligente Teilsysteme mit mechatronischen Komponenten geteilt, die hierarchisch angeordnet

sind und über definierte Schnittstellen miteinander, sowie mit der Umgebung kommunizieren können.

Für eine strukturierte Übersicht des komplexen Gesamtsystems mit definierten Schnittstellen in horizontaler und vertikaler Richtung erfolgt zunächst die hierarchische Strukturierung in einem Top-Down-Prozess. Dabei werden die Hauptfunktionen des Gesamtsystems in Teilfunktionen zerlegt und hierarchisch in sechs Ebenen gegliedert, wie in der Abbildung 1 dargestellt. In [12] werden die einzelnen Ebenen ausführlich beschrieben.

neriert den Fahrweg aus der Route in Form von Knotenpunkten. Anschließend wird die Bewegungsplanung unter Berücksichtigung des dynamischen Verhaltens, beispielsweise der maximalen Beschleunigung und Geschwindigkeit, des FTF bestimmt. Mittels WLAN basierter Vernetzung können sich die FTF im Produktionsumfeld die eigenen Trajektorien untereinander gegenseitig mitteilen. Der Mechanismus zur Konfliktlösung überprüft zuerst das Vorhandensein von Konflikten, identifiziert diesen und wählt eine geeignete Lösung aus.

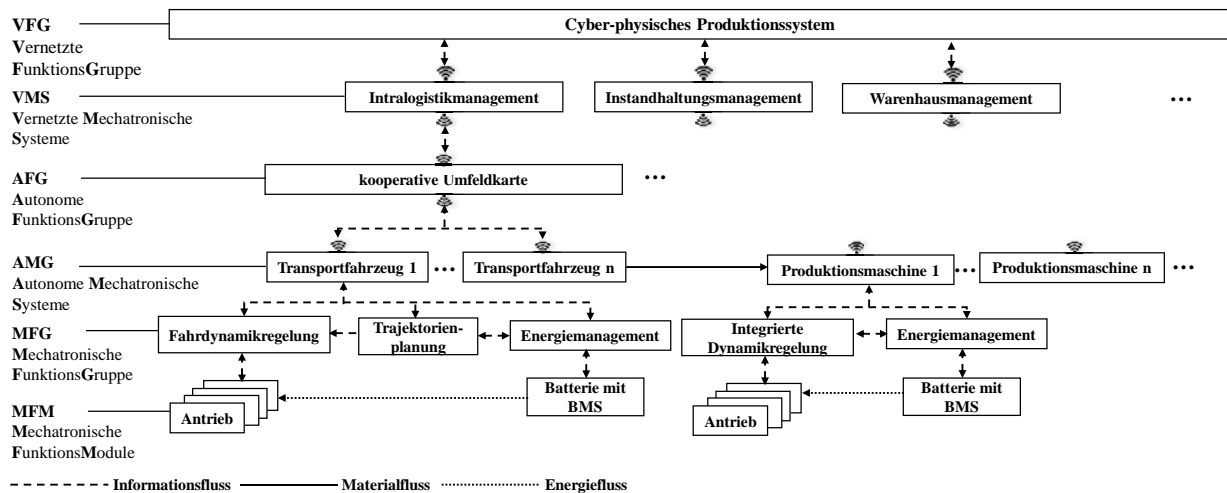


Abbildung 1: Mechantronische Strukturierung des FTF im cyber-physischen Produktionssystem

Das einzelne Funktionsmodul wird modellbasiert ausgelegt und in einem durchgängigen Prozess bestehend aus Model-in-the-Loop (MiL), Software-in-the-Loop (SiL) und Hardware-in-the-Loop (HiL) validiert und optimiert [11]. Durch diesen Prozess werden die entwickelten Funktionen frühzeitig an Modellen in der Simulationsumgebung sowie an dem Prototyp unter Echtzeitbedingungen erprobt und abgesichert.

In diesem Beitrag wird die Trajektorienplanung als Sollwertgenerator der Fahrzeugdynamikregelung mittels MiL-Simulationen vorgestellt.

4 Konzeption und Entwicklung

Abbildung 2 zeigt einen Überblick über die Funktionsstruktur des FTF zur Durchführung der ebenen Bewegung. Dieses FTF navigiert frei mittels des Funktionsmoduls Zielführung, welche auf dem Suchalgorithmus nach Dijkstra die optimale Route für einen Transportauftrag von einer Start- zu einer Zielposition in einem gerichteten Graphen festlegen kann. Die Trajektorienplanung ge-

Schließlich wird die konfliktfreie Trajektorie der Fahrdynamikregelung als Sollwert übergeben und realisiert.

Im Rahmen dieses Beitrags werden die zwei Teilfunktionen Bahnplanung und Bewegungsplanung der Trajektorienplanung als Übergang zwischen Zielführung und Fahrdynamikregelung unter Berücksichtigung des dynamischen Verhaltens des FTF sowie ein Mechanismus zur Konfliktlösung während des Gütertransportes entwickelt.

4.1 Trajektorienplanung

Wie bereits erwähnt, beinhaltet die Trajektorienplanung zwei Teilfunktionen, die Bahnplanung zur Generierung eines kontinuierlichen Fahrwegs aus Knotenpunkten in einer bestimmten Reihenfolge aus der Zielführung und die Bewegungsplanung zur Erstellung des zeitlichen Verlaufs von Fahrweg wobei die Beschränkung von Beschleunigung und auch Geschwindigkeit des FTF berücksichtigt wird und damit die Regelgüte der Fahrdynamikregelung zum Folgen der Trajektorie verbessert wird.

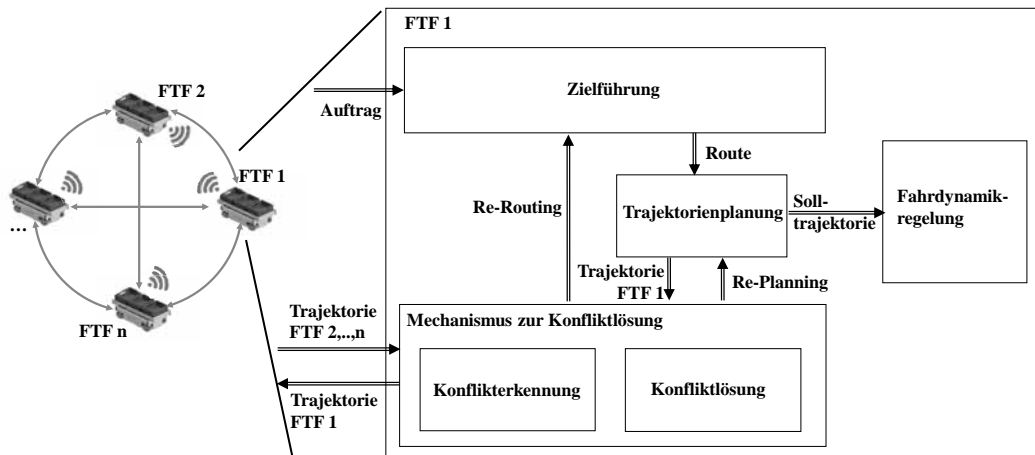


Abbildung 2: Ein Überblick der Funktionsstruktur zur konfliktfreien Trajektorienplanung

4.1.1 Generierung des Fahrwegs mittels Bahnplanung

Der Übergabewert aus der Zielführung ist die Route \underline{l} von einer Start- zu einer Zielposition in der Form von Knotenpunkten (x_{ei}, y_{ei}) in Reihenfolge der abzufahren- den Richtung.

$$\underline{l} = \begin{bmatrix} x_{e1} & x_{e2} & \dots & x_{ei} \\ y_{e1} & y_{e2} & \dots & y_{ei} \end{bmatrix} \quad (1)$$

Die mögliche Route ist die reine Verbindung aus Knotenpunkten, welche nur aus geraden Linien besteht. Zur Generierung des kontinuierlichen Fahrwegs, der sich der Route bestmöglich annähern soll, wird die Bahnplanung mittels Kombination von Geraden und Kreisbögen entworfen. Die Abbildung 2 stellt den Fahrweg \underline{l} , der durch Geraden zwischen den Knotenpunkten erzeugt wird, den Fahrweg \underline{l}_{G+K} , der aus Geraden und Kreisbögen besteht, und den durch Interpolation mit Polynomen bestimmten Fahrweg \underline{l}_{poly} dar.

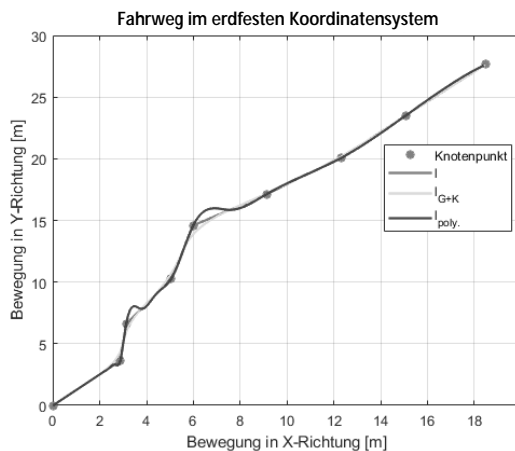


Abbildung 3: generierte Fahrwege

Es ist auffällig, dass die Abweichung zwischen Fahrweg \underline{l}_{poly} und dem Fahrweg \underline{l} groß ist und damit die Fahrtrichtung, als Tangente des Fahrwegs, oft und stark verändert werden muss, was sich ungünstig auf die Fahrndynamikregelung auswirkt.

Die Tabelle 1 stellt die Länge des jeweiligen Fahrwegs und die Abweichung der Länge im Vergleich zu dem Fahrweg \underline{l} dar. Die Längenabweichung von Fahrweg \underline{l}_{G+K} ist viel kleiner als die von Fahrweg \underline{l}_{poly} und durch Einstellung des Radius der jeweiligen Kreisbögen kann die Abweichung weiter verkleinert werden. Demzufolge wird hier die Fahrwegberechnung mittels Kombination von Geraden und Kreisbögen gewählt.

	Länge [m]	Abweichung [%]
\underline{l}	34,3194	0
\underline{l}_{G+K}	33,9946	-0,9464
\underline{l}_{poly}	35,5104	3,4703

Tabelle 1: Länge des jeweiligen Fahrwegs und Abweichung zwischen Fahrweg und kürzester Route

4.1.2 Generierung des zeitlichen Verlaufs mittels Bewegungsplanung

In der Bewegungsplanung wird der in der Bahnplanung gefundene Fahrweg in einen zeitlichen Verlauf umformuliert. In diesem Schritt wird die Beschränkung der Beschleunigung und Geschwindigkeiten anhand des Konzeptes des FTF berücksichtigt, sodass die Position des FTF im Produktionsumfeld in Abhängigkeit der Zeit bestimmt werden kann. Diese ist Grundlage der Konflikter-

kennung. Außerdem können Transportzeit und Energieverbrauch, die wichtigsten Faktoren für die Auftragsvergabe des Intralogistiksystems, für den aktuellen Transportauftrag mittels des zeitlichen Verlaufs abgeschätzt werden.

Wie in der Abbildung 4 veranschaulicht, wird anhand der Kinematik der Bewegungszustand des FTF durch den Vektor $[v_{fx} \ v_{fy} \ \dot{\psi}]^T$ bzw. die Geschwindigkeit des Schwerpunktes in der X- und Y-Richtung sowie die Giergeschwindigkeit um die Hochachse im fahrzeugfesten Koordinatensystem dargestellt. Deswegen soll zuerst der Geschwindigkeitsvektor $[v_{ex} \ v_{ey} \ \dot{\psi}]^T$ im erdfesten Koordinatensystem bestimmt und anschließend in das fahrzeugfeste Koordinatensystem transformiert werden.

$$\begin{bmatrix} v_{fx} \\ v_{fy} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_{ex} \\ v_{ey} \\ \dot{\psi} \end{bmatrix} \quad (2)$$

Die entsprechenden Geschwindigkeiten in der X- und Y-Richtung im erdfesten Koordinatensystem können durch die Durchschnittsgeschwindigkeit innerhalb eines Diskretisierungsintervalls ΔT ersetzt werden.

$$v_{ex} = \frac{x_{ei+1} - x_{ei}}{\Delta T} \quad (3)$$

$$v_{ey} = \frac{y_{ei+1} - y_{ei}}{\Delta T}$$

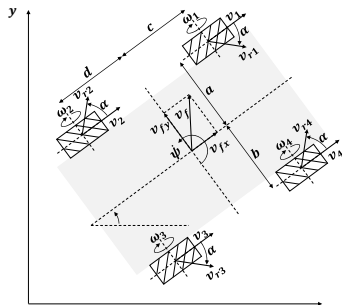


Abbildung 4: kinematisches Verhalten des FTF

Mithilfe der Geschwindigkeitsbeschränkungen $[v_{fxmax} \ v_{fymax} \ \dot{\psi}_{max}]^T$ werden die jeweils nötigen Zeitintervalle bestimmt. Das Maximum der drei Werte ΔT_{mod} entspricht der Berechnungsschrittweite.

$$\Delta T_{vx} = \frac{|x_{fxi+1} - x_{fxi}|}{v_{fxmax}}$$

$$\Delta T_{vy} = \frac{|x_{fyi+1} - x_{fyi}|}{v_{fymax}} \quad (4)$$

$$\Delta T_{\dot{\psi}} = \frac{|\psi_{i+1} - \psi_i|}{\dot{\psi}_{max}}$$

$$\Delta T_{mod} = \max(\Delta T_{v_x}, \Delta T_{v_y}, \Delta T_{\dot{\psi}})$$

4.2 Mechanismen zur Konfliktlösung

Im vorliegenden Konzept erfolgt die Zielführung und Trajektorienplanung dezentral im eigenen Rechner des FTF. Konventionelle Ansätze hingegen arbeiten mit einer zentralen Lensteuerung. Dadurch werden mögliche Konflikte, die beim Gütertransport zwischen den FTF entstehen können, nicht im Vorfeld berücksichtigt und müssen entsprechend vor der Ausführung der ausgelegten Trajektorie durch die Fahrdynamikregelung überprüft und ggf. gelöst werden.

Die Abbildung 3 zeigt den Ablaufplan des Mechanismus zur Konfliktlösung. Über das Kommunikationsmodul können die FTF die generierten Trajektorien untereinander austauschen. Durch den Vergleich der Trajektorien miteinander, kann ein FTF mögliche Konflikte mit anderen FTF erkennen. Falls Konflikte vorhanden sind, wird die Priorität des auszuführenden Transportauftrags ermittelt. Ein FTF mit Transportauftrag höherer Priorität soll die aktuelle Trajektorie nicht umdisponieren. Hat ein Transportauftrag eine niedrige Priorität so wird die Lösbarkeit des Konfliktes überprüft. Kann der Konflikt durch Umdisponieren nicht gelöst werden, so soll die Trajektorienplanung die Route neu berechnen. Falls der Konflikt lösbar ist, soll die geeignete Lösung ausgewählt und die Trajektorie modifiziert werden.

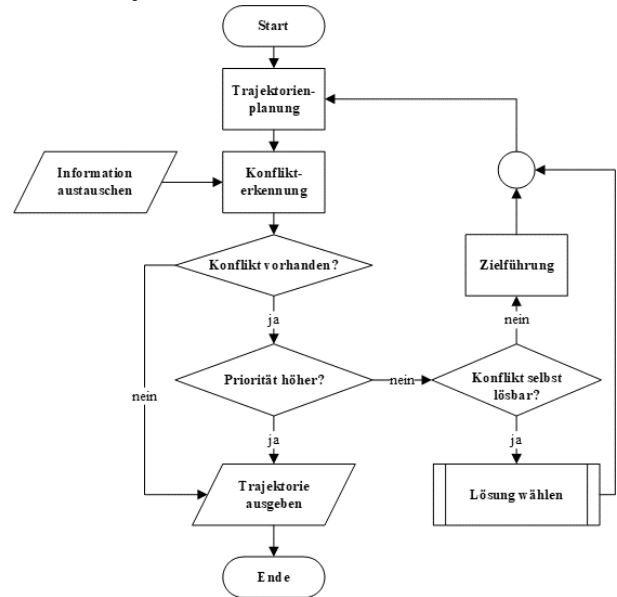


Abbildung 5: Ablaufplan des Mechanismus zur Konfliktlösung

5 Funktionsabsicherung

In diesem Kapitel wird die Funktionalität der Trajektorienplanung und des Mechanismus zur Konfliktlösung in

der Simulationsumgebung betrachtet und abgesichert. Hierbei werden lediglich die Bewegungskonflikte zwischen den FTF betrachtet.

In der Abbildung 6 wird ein Kreuzkonflikt von FTF1 und FTF2 dargestellt. Zuerst wird der Konflikt nach Vergleich des zeitlichen Verlaufs untereinander erkannt und identifiziert. Anschließend trifft das FTF nach Priorität eigenständig eine Entscheidung, welches FTF zum Überwinden des lösbaren Konfliktes eine neue Trajektorie planen soll. Hierbei sei angenommen, dass das FTF mit niedrigerer Priorität die Trajektorie neu berechnen muss, während das FTF mit höherer Priorität entlang der originalen Trajektorie weiterfahren darf. In diesem Fall soll FTF2 die Trajektorie modifizieren. Durch die Konflikterkennung ist klar, wann und wo der Kreuzkonflikt passiert. Deswegen soll FTF2 um ΔT zur Erreichung der Konfliktposition verzögern, sodass das FTF1 während des Zeitintervalls die Konfliktposition durchfahren kann. Die Zeitverzögerung ΔT wird in der Abhängigkeit der Geschwindigkeit und auch der Abmaße der FTF ermittelt. Durch Beschränkung der Geschwindigkeit des FTF2 für die Trajektorie vor dem Konflikt kann die Zeitverzögerung in der Trajektorienplanung ganz einfach realisiert werden. Sobald das FTF2 die Konfliktposition durchfährt, wird die Geschwindigkeit des FTF2 wieder erhöht.

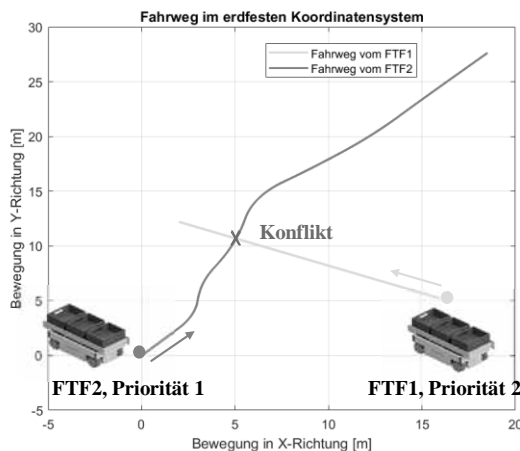


Abbildung 6: Kreuzkonflikt zwei FTF im Produktionsumfeld

Die Abbildung 7 stellt das Simulationsergebnis des in Abbildung 6 dargestellten Falls vor. Hierbei wird die Zeitverzögerung ΔT mit einem Wert von 2 s eingestellt. Der Fahrweg des FTF2 verändert sich nicht. Der Kreuzkonflikt wird durch den schwarzen Kreis auf dem Fahrweg markiert und dessen Informationen für die Zeit und auch die Position im erdfesten Koordinatensystem wer-

den durch einen Vektor ausgewiesen. Wie in der Abbildung 7 verdeutlicht, passiert der Konflikt nach 12 s in der Lage $[5,0520 \ 10,6723]^T$. Durch die neue Trajektorienplanung liegt das FTF2 zum gleichen Zeitpunkt 12 s in der Lage $[4,4032 \ 9,0473]^T$, während das FTF1 in der Konfliktposition liegt. Die Distanz dazwischen ist 1,75 m zu diesem Zeitpunkt und damit ist der Konflikt gelöst.

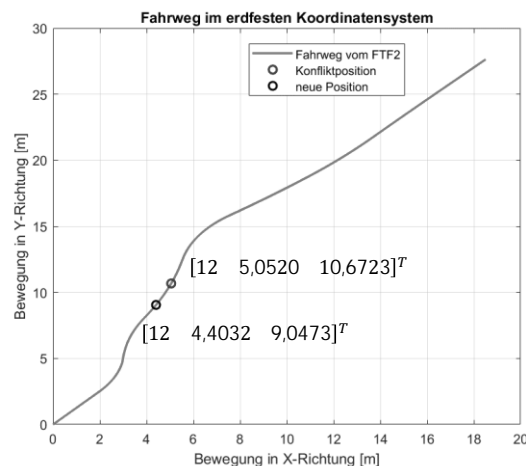


Abbildung 7: Simulationsergebnis der neuen Trajektorienplanung

Ein weiterer Konflikt neben dem Kreuzkonflikt liegt vor, wenn sich zwei FTF entlang des gleichen Fahrwegs aufeinander zu bewegen, wie in der Abbildung 8 veranschaulicht. Analog zur Lösung des Kreuzkonfliktes wird zuerst die Lage des Konfliktes im zeitlichen Verlauf der Trajektorie markiert. Anschließend wird der Fahrweg des FTF2 mit niedrigerer Priorität zum Ausweichen mittels Sigmoid-Funktion neu generiert und anhand der Geschwindigkeitsbeschränkung die entsprechende Trajektorie neu geplant.

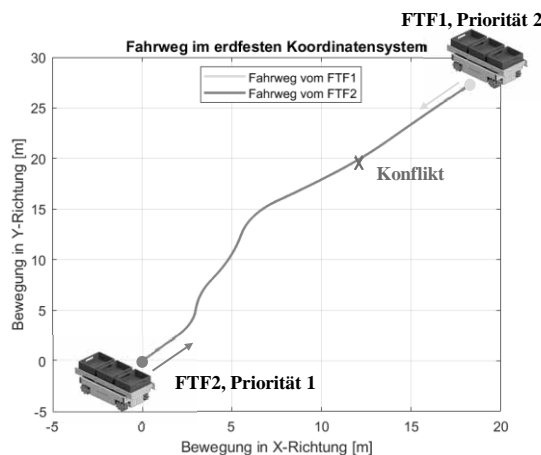


Abbildung 8: Konflikt zwei FTF mit einem gleichen Fahrweg

In der Abbildung 9 wird das Simulationsergebnis gezeigt. Der Fahrweg des FTF1 mit höherer Priorität verändert sich nicht, während der Fahrweg des FTF2 mit niedrigerer Priorität um einen Ausweichfahrweg erweitert wird. Der Konflikt passiert zum Zeitpunkt 25 s in der Lage $[12,7833 \ 20,6782]^T$, wie in der Abbildung 9 durch den blauen Kreis markiert. Nach Generierung des neuen Fahrwegs und der entsprechenden Anpassung der Geschwindigkeit bei der Trajektorienplanung liegt das FTF2 zu diesem Zeitpunkt in der Lage $[12,4592 \ 21,0950]^T$. Die Distanz dazwischen beträgt 0,53 m und unter Berücksichtigung der Breite des FTF existiert in dieser Lage kein Konflikt mehr.

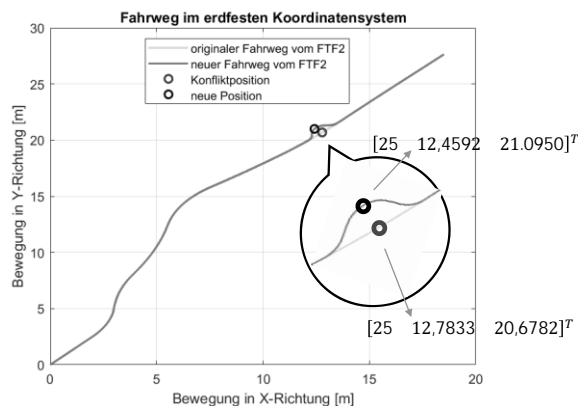


Abbildung 9: Simulationsergebnis des Ausweichens zur Konfliktlösung

6 Fazit und Ausblick

In diesem Beitrag wurden die Funktionsmodule „Trajektorienplanung“ und „Mechanismus zur Konfliktlösung“ auf Basis des kinematischen und dynamischen Verhaltens des FTF entworfen, welche als Übergang vom Funktionsmodul Zielführung zur Fahrdynamikregelung dienen, sodass ein konfliktfreier Gütertransport realisiert werden kann.

Bei dem Entwicklungsprozess beider ausgelegter Funktionsmodule wird die Interaktion von anderen Produktionsakteuren z.B. der Arbeiter und des Raumplans der Produktionshalle noch nicht betrachtet. Um den autonomen Gütertransport in der Zukunft unter Echtzeitbedingungen durchzuführen, soll die Interaktion der anderen Produktionsakteure in den Funktionsmodulen integriert werden.

7 Danksagung

Dieser Beitrag wurde im Rahmen des Teilprojekts *Modellbasierte Konzeption und Bewertung von Industrie 4.0-Lösungen zur Vernetzung mechatronischer Komponenten in Produktionsanlagen durch Digitalisierung (MiMec)* des Verbundprojekts *Methoden und Werkzeuge für die synergetische Konzipierung und Bewertung von Industrie 4.0-Lösungen (Synus)* durch den Europäischen Fonds für regionale Entwicklung (EFRE) unter dem Förderkennzeichen ZW 6 85012454 gefördert. Die Verantwortung für den Inhalt liegt bei den Autoren.



EUROPÄISCHE UNION
Europäischer Fonds für
regionale Entwicklung



Literatur

- [1] Spath D, Ganschar O, Gerlach S, Hämmerle M, Krause T, Schlund S, *Produktionsarbeit der Zukunft-Industrie 4.0*, Fraunhofer Verlag, 2013.
- [2] Bauernhansel T, Hompel M, Vogel-Heuser B, *Industrie 4.0 in Produktion, Automatisierung und Logistik*, Springer Vieweg, Münschen, 2014.
- [3] Mors AW ter, Zutt J, Witteveen C, *Context-Aware Logistic Routing and Scheduling*, Proceedings of the Seventeenth International Conference on Automated Planning and Scheduling, 328-335, 2007.
- [4] Schwarz C, Schachmanow J, Sauer J, Overmeyer L, Ullmann G, *Selbstgesteuerte Fahrerlose Transportsysteme*, Logistics Journal, 2013.
- [5] Frese C. *Planung kooperativer Fahrmanöver für kognitive Automobile*. Dissertation. Karlsruhe, Hannover: KIT Scientific Publishing; Technische Informationsbibliothek u. Universitätsbibliothek, 2012.
- [6] Isermann R. *Fahrdynamik-Regelung: Modellbildung, Fahrerassistenzsysteme, Mechatronik*. Wiesbaden, Vieweg Verlag, 2006.
- [7] Jacobs P, Canny J, *Planning smooth paths for mobile robots*, In: Zexiang Li und J. F. Canny, Hrsg., *Nonholonomic Motion Planning*. Kluwer Academic Publishers Group, 1992.
- [8] Habenicht, S. *Entwicklung und Evaluation eines manöverbasierten Fahrstreifenwechselassistenten*. Dissertation. Technische Universität Darmstadt, Fortschrittsberichte VDI, Reihe 12, Nr. 756, VDI Verlag GmbH, Düsseldorf, 2012.

- [9] Keller M. *Trajektorienplanung zur Kollisionsvermeidung im Straßenverkehr*. DiSSERTATION. Technische Universität Dortmund, 2017.
- [10] Yi B. *Integrated Planning and Control for Collision Avoidance Systems*. Dissertation. Karlsruher Institut für Technologie, 2017.
- [11] Liu-Henke X. *Mechatronische Entwicklung der aktiven Feder-/ Neigetechnik für das Schienenfahrzeug RailCab*, VDI-Fortschritt-Berichte, Reihe 12, Nr. 589, VDI-Verlage, Düsseldorf, 2004.
- [12] Liu-Henke X, Scherler S, Göllner M, Jacobitz S, Zhang J, Yarom O, *A holistic methodology for model-based development of mechatronic systems in digitized and connected system environments*, IEEE ISSE 2020, Vienna, Austria, October 12 - 14, 2020. (tbp)

Parameter-Optimierung eines Brake-by-Wire-Pedals

Jennifer Werner¹, Martin Düsing^{1*}, Bernhard Bachmann², Ali Kemal Kücükayavuz¹

¹HELLA GmbH & Co. KGaA Beckumer Straße 130, 59555 Lippstadt, Deutschland; *martin.duesing@hella.com

²Fachhochschule Bielefeld, Fachbereich Ingenieurwissenschaften und Mathematik, Interaktion 1, 33609 Bielefeld

Kurzfassung. Das hydraulische Bremspedal wird aufgrund verschiedener Vorteile bei modernen Fahrzeugen durch Brake-by-Wire-Pedale ersetzt, die keine mechanische Verbindung zum Bremssystem haben, sondern nur noch elektronische Signale vom Pedalsensor zum Bremssteuergerät übertragen. Da sich das hydraulische und das Brake-by-Wire-Pedal für den Fahrer weitestgehend gleich bedienen und anfühlen sollen, ist die Kennlinie Pedalkraft über Weg eine entscheidende Anforderung. Diese Anforderung ist je nach Fahrzeughersteller und Fahrzeug unterschiedlich. In dieser Arbeit wird ein Simulationsmodell vorgestellt, dass während der Entwicklung eines Brake-by-Wire-Pedals zur Anwendung kam. Der Fokus dieser Arbeit liegt auf einer mathematischen Optimierung des ursprünglichen Modells zur Erreichung einer neuen Kundenkennlinie.

Einleitung

Brake-by-Wire-Pedale stellen eine grundlegende Änderung in Bremssystemen von Automobilen dar. In klassischen hydraulischen Bremssystemen betätigt ein Bremspedal mechanisch einen Hydraulikkolben. In Brake-by-Wire-Systemen gibt es nur noch elektronische Signale, die vom Bremspedalsensor an das Steuergerät zur Bremsung übergeben werden.

Diese Technik bietet verschiedene Vorteile. So lässt sich die Ansprechzeit verkürzen und dadurch der Bremsweg verringern. Da Komponenten wie Bremskraftverstärker, Hauptbremszylinder und ABS wegfallen oder Funktionen durch Software realisiert werden, können Kosten gespart werden. In Elektro- oder Hybridfahrzeugen kann das Betätigen des Brake-by-Wire-Pedals auch als Auslöser für die Rekuperation verwendet werden, so dass bei leichtem Auslösen Energie zurückverwandelt wird, anstatt sie als Wärme abzugeben.

Während der Entwicklung eines Brake-by-Wire-Pedals bei der Firma Hella wurden zu unterschiedlichen Konzepten jeweils Modelle zur Simulation verwendet. Die Modelle werden mit der Sprache Modelica in der

3DExperience Plattform und in Dymola entwickelt.

Ziel der Simulation ist die möglichst präzise Vorhersage der Kennlinie Pedalkraft über Pedalweg oder synonym über Pedalwinkel für festgelegte Geschwindigkeiten des Pedals. Diese Kennlinien werden von den Fahrzeugherstellern vorgegeben und sind eine entscheidende Anforderung an das Produkt.

Nach der erfolgreichen Entwicklung und Produktion des Brake-by-Wire-Pedals richtet sich der Fokus stärker auf die Frage der Weiterentwicklung, die sich besonders auf die Erreichung anderer Kennlinien bezieht. Dazu wird hier eine mathematische Optimierung eingesetzt.

Ziel dieser Arbeit ist die Parameter-Optimierung eines Modelica-Modells, um vorgegebene Kennlinien zu treffen. Die Optimierung findet in MATLAB statt.

Die Arbeit gliedert sich in fünf Teile. Im ersten Teil wird das Modelica-Modell des Pedals beschrieben, während im zweiten Teil Grundlagen der Parameteroptimierung zusammengefasst werden. Der dritte Teil befasst sich mit einer neuen MATLAB-Klassenumgebung, die zur praktischen Umsetzung genutzt wird. Im vierten Teil wird die Optimierung durchgeführt und die Ergebnisse präsentiert. Abschnitt 5 fasst die Ergebnisse der Arbeit zusammen.

1 Modell des Brake-by-Wire-Pedals

Zur Modellierung des Brake-by-Wire-Pedals wird die objektorientierte Modellierungssprache Modelica verwendet. Als Entwicklungsumgebung wird sowohl Dymola als auch die 3DExperience Plattform von Dassault Systèmes verwendet. Die 3DExperience Plattform bietet den Vorteil, dass die Geometrie und damit die Hebellängen, Volumen, Dichten und Massenträgheitsmomente automatisch vom CAD-Modell in das Modelica-Modell übernommen werden können. Nur die Kinematik, Reibung und Kontakte müssen getrennt, speziell modelliert

werden. Liegt allerdings noch kein fertiges CAD-Modell vor, so kann dieser Weg nicht beschritten werden. Ist das CAD-Modell noch nicht fertig, die kinematisch wichtigen Abmessungen liegen aber bereits vor und sollen auch durch eine Simulation auf ihre Funktionalität hin überprüft werden, so lässt sich mit Hilfe der Modelica Standard Library auch effizient ein Modell aufbauen. Mit `Modelica.Mechanics.MultiBody.Parts.BodyBox`, `Modelica.Mechanics.MultiBody.Joints.Revolute` und `Modelica.Mechanics.MultiBody.Forces.Spring` lassen sich große Teile des Pedals beschreiben.

In diesem Fall wurden die für die Kinematik wichtigen Massenträgheitsmomente nur grob angenähert. Hoch dynamische Lastfälle lassen sich damit nicht simulieren. Quasistatische Versuche können aber mit sehr kurzen Simulationszeiten durchgeführt werden.

Die Verifizierung des Modells mit einer im Entwicklungsprozess deutlich später durchgeführten Messung zeigt eine hervorragende Übereinstimmung. Abbildung 1 zeigt die Messdaten in blau und die Simulationsergebnisse in rot. Die Kraft in N wird über den Pedalwinkel in ° dargestellt. Es sind zwei Kurven mit Hin- und Rückweg zu sehen. Die oberen Kurven sind die Hin- und die unteren die Rückkurven. Die Hysterese ist typisch für Bremspedale.

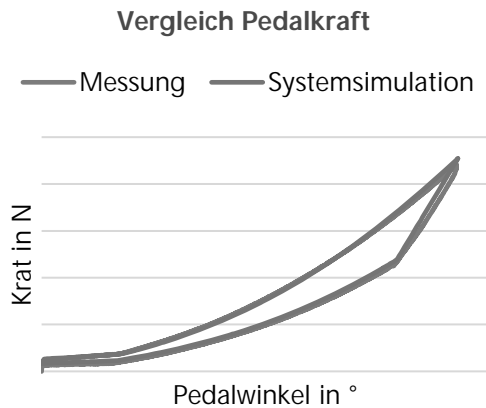


Abbildung 1: Vergleich von Messung und Simulation

2 Grundlagen der Parameteroptimierung

Während einer Optimierung werden verschiedene Alternativen bezüglich eines Zielkriteriums verglichen und unter allen betrachteten Alternativen wird die beste ge-

sucht [1]. Die Alternativen unterscheiden sich durch verschiedene Parametereinstellungen des Modells, die variiert werden.

Die nichtlineare Optimierung kommt bei unterschiedlichen Modellen aus Natur-, Ingenieur- und Wirtschaftswissenschaften zum Einsatz, beispielsweise bei geometrischen Problemen, mechanischen Problemen, Parameter-Fitting-Problemen, Schätzproblemen, Approximationsproblemen und bei der Sensitivitätsanalyse [1]. Unterschieden wird zwischen unrestringierten und restringierten nichtlinearen Optimierungsproblemen.

Optimierungsprobleme ohne Nebenbedingungen sind unrestringierte Optimierungsprobleme [2]. Ein nichtlineares Optimierungsproblem ohne Restriktionen besitzt die Form

$$\min_{x \in \mathbb{R}^n} f(x) \quad (1)$$

mit $f(x): \mathbb{R}^n \rightarrow \mathbb{R}$ [1].

Ein nichtlineares Optimierungsproblem mit Nebenbedingungen wird als restringiertes nichtlineares Optimierungsproblem bezeichnet. Es ist gegeben durch:

$$\min_{x \in \mathbb{R}^n} f(x) \quad (2)$$

so dass

$$g(x) \leq 0 \quad (3)$$

$$h(x) = 0 \quad (4)$$

und es wird angenommen, dass die Funktionen $f(x): \mathbb{R}^n \rightarrow \mathbb{R}$, $g(x): \mathbb{R}^n \rightarrow \mathbb{R}^r$, $h(x): \mathbb{R}^n \rightarrow \mathbb{R}^m$ zweimal stetig differenzierbar sind [3]. Die Funktion $f(x)$ wird dabei Zielfunktion genannt. Die Funktion $g(x)$ beschreibt die Ungleichheitsbedingungen und die Funktion $h(x)$ die Gleichheitsbedingungen des restringierten nichtlinearen Optimierungsproblems.

Ein nichtlineares Least-Square-Problem (NLLSP) ist ein Minimierungsproblem der Form

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} \sum_{i=1}^m f_i(x)^2, \quad m \geq n \quad (5)$$

wobei jedes $f_i(x)$, $i = 1, \dots, m$ eine nichtlineare Funktion im \mathbb{R}^n ist [4].

Ein wichtiger Einsatzbereich für NLLSPs ist der Bereich des Data Fitting. Hier ist es das Ziel einen gegebenen Datensatz (y_i, t_i) , $i = 1, \dots, m$ an eine Modell-Funktion $g(x, t)$ anzupassen. Sei

$$f_i(x) = y_i - g(x, t_i), \quad i = 1, \dots, m, \quad (6)$$

so führt dies auf ein Least-Square-Problem der Art (5) (siehe [4]).

3 MATLAB-Klassenumgebung

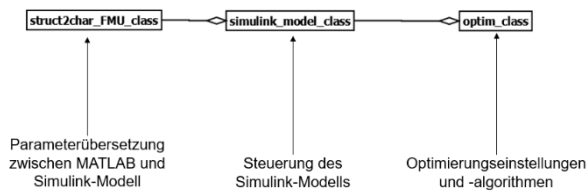


Abbildung 2: UML-Diagramm MATLAB-Klassenumgebung

Das in Abschnitt 2 beschriebene Modelica-Modell wird als FMU aus Dymola exportiert und in MATLAB Simulink importiert, siehe Abbildung 3. Der Block ist der FMU-Import aus Dymola, welcher das Modelica-Modell des Brake-by-Wire-Pedals beinhaltet. Dieser hat mehrere Ausgänge, allerdings sind für die Parameteroptimierung nur die Ausgänge *FootForce* (entspricht der Pedalkraft) und der Ausgang *PedalTravel* (entspricht dem Pedalweg) notwendig. Zur Optimierung soll das Simulink-Modell mit der FMU direkt aus MATLAB gestartet werden. Die MATLAB-Klassenumgebung dient zum einfachen Starten der Modellsimulation, zur Veränderung der Parameter des Modells und zum Starten der Optimierung. Abbildung 2 zeigt den Aufbau der Klassenumgebung mit den drei Klassen *struct2char_FMU_class*, *simulink_model_class* und *optim_class*.

struct2char_FMU_class

Diese Klasse dient als Übersetzer zwischen den MATLAB-Parametern und den Simulink-Parametern. Dies ist notwendig, da die Parameter aus dem Simulink-Modell in Form eines Strings vorliegen. Für die MATLAB-Klasse *simulink_model_class* muss dieser String in eine Struktur umgewandelt. Damit die Parameter wieder in das Simulink-Modell eingegeben werden können, wird in der Klasse *struct2char_FMU_class* die Struktur wieder in einen String umgewandelt. Die Struktur kann dabei beliebig oft verschachtelt sein oder auch eine mehrdimensionale Struktur sein und die Felder der Struktur können verschiedene Datentypen besitzen.

simulink_model_class

Diese Klasse ist zur Steuerung des Simulink-Modells programmiert worden. Es ist möglich, die Parameter des Simulink-Modells anzusprechen. Außerdem ist es möglich, Default-Werte für die Parameter festzulegen und das Simulink-Modell mit einer Funktion auf die entsprechenden Default-Werte zurückzusetzen. Die Simulation des Simulink-Modells kann ebenfalls mit der Klasse gestartet

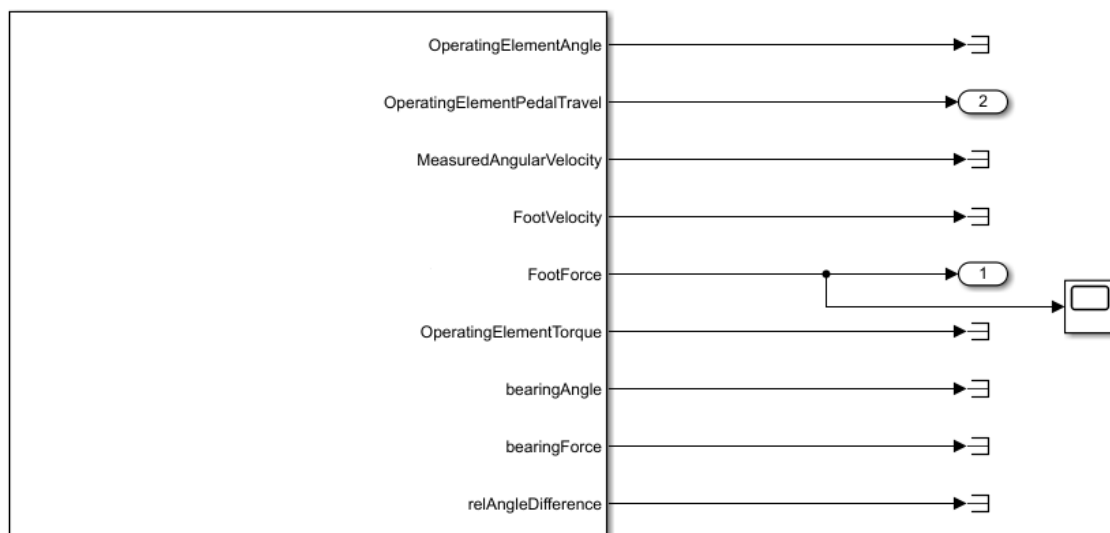


Abbildung 3: FMU-Import in Simulink des Modelica-Brake-by-Wire-Pedals

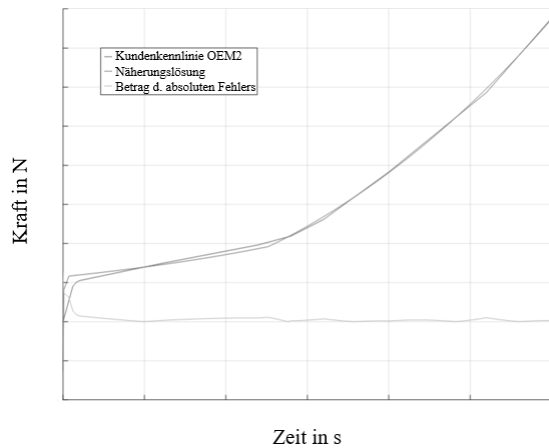


Abbildung 4: Ergebnis der Optimierung für OEM2

werden und das Ergebnis der Simulation wird automatisch in einem Attribut der Klasse abgespeichert.

Nach verschiedenen Tests in Simulink zeigt sich, dass der Solver ode23s (stiff/Mod. Rosenbrock) am schnellsten und robustesten im Vergleich zu den anderen verfügbaren Solvern ist. Dabei ist eine relative Fehlertoleranz von 10^{-4} ausgewählt. Die absolute Fehlertoleranz wird auf den Wert *auto* gestellt. Bei dem Bremspedal-Modell handelt es sich um ein sehr steifes System und an den Kontaktstellen kommt es zu numerischen Schwierigkeiten. In den Tests hat sich gezeigt, dass diese Stellen mit einer sehr kleinen minimalen Schrittweite überwunden werden können.

optim_class

Mit dieser Klasse ist die Steuerung des Optimierungsprozesses des Simulink-Modells möglich. In der Klasse wird zur Optimierung die MATLAB-Funktion *lsqcurvefit* aufgerufen. Dabei handelt es sich um ein nichtlineares Least-Square-Data-Fitting-Problem. Diese Funktion ist Teil der Optimization Toolbox und wird beschrieben durch

$$\min_x \sum_{k=1}^n (F(x, xdata_k) - ydata_k)^2 \quad (7)$$

Dabei ist die Funktion $F(x, xdata)$ vom Anwender definiert. In diesem Fall ist es das Simulationsergebnis des Bremspedal-Modells, konkret die diskreten Werte der Pedalkraft. Die Größe der Vektoren $xdata$ und $ydata$ entspricht n und ist gleich der Anzahl der Zeitschritte. Der Vektor $xdata$ enthält die diskreten Zeitpunkte, der Vektor $ydata$ die diskreten Zielwerte der Optimierung. Die Tuner-Parameter befinden sich im

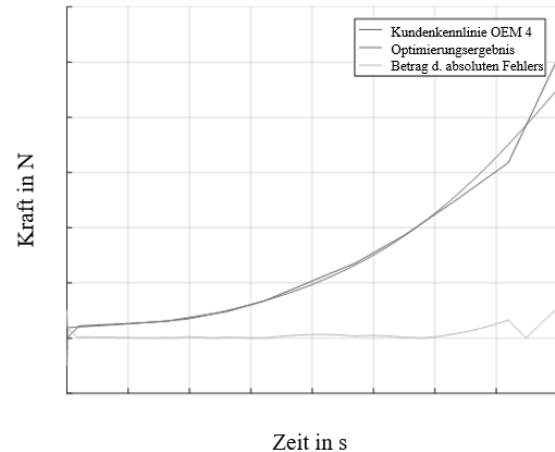


Abbildung 5: Ergebnis der Optimierung für OEM4

Vektor x der Länge m , die der Anzahl aller Tuner-Parameter entspricht.

Der Funktion *lsqcurvefit* müssen die Funktion $F(x, xdata)$, die Startwerte x_0 sowie die unteren und oberen Grenzen *lowerBound* und *upperBound* für die Tuner-Parameter übergeben werden. Außerdem können noch Einstellungen hinsichtlich des von der Funktion verwendeten Optimierungsalgorithmus übergeben werden. Diese Übergabeparameter sind in der Klasse *optim_class* als Attribute definiert. Die MATLAB-Funktion *lsqcurvefit* stellt den Levenberg-Marquardt-Algorithmus (Details siehe [4] und [5]) sowie ein Trust-Region-Verfahren [5] zur Lösung des NLLSP zur Verfügung.

4 Ergebnisse der Optimierung

Abbildungen 4 und 5 zeigen Ergebnisse der Optimierung für zwei verschiedene Kundenkennlinien mit dem gleichen Pedalmodell. Zehn verschiedene Parameter des Modells werden während der Optimierung mit dem Ziel angepasst, die jeweils blau dargestellte Kundenkennlinie im Kraft-Zeit-Diagramm zu erreichen. Anstelle des Kraft-Weg-Diagramms wird in der Optimierung auf das Kraft-Zeit-Diagramm zurückgegriffen, da dieses Vorgehen für die Optimierung einfacher ist. Die Pedale werden mit einer niedrigen konstanten Geschwindigkeit bewegt. Zeit und Position des Pedals sind also linear abhängig, sodass beide Varianten die gleiche Aussage haben.

Die roten Kurven in Abbildungen 4 und 5 sind die Näherungslösungen, die mit der Klassenfunktion *optimize_simulink_model* erreicht werden. Der Betrag des

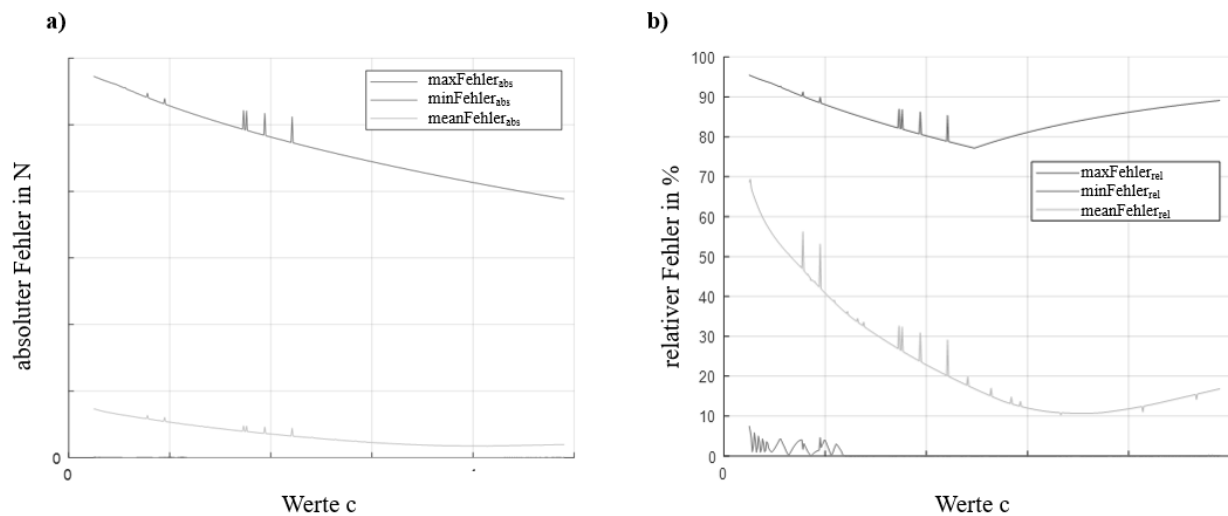


Abbildung 6: Absoluter (a) und relativer Fehler (b) einer Variation über den Parameter c

absoluten Fehlers ist in gelb eingetragen. Für die Optimierung in Abbildung 4 ist der Betrag des absoluten Fehlers im Zeitpunkt 0 s am höchsten während der Durchschnittsfehler bei nur 3,11 % liegt. Das Ergebnis für die Optimierung in Abbildung 5 hat einen Durchschnittsfehler von 8,80 %.

Die Ergebnisse zeigen, dass mit einem Pedal, nur durch die Änderung einzelner Komponenten eine deutlich andere Kennlinie erreichbar ist. Die Abweichung im hinteren Bereich in Abbildung 5 ist auf den Kontakt des Pedals mit dem Anschlag zurückzuführen. Dieser Kontakt wurde im untersuchten Modell vernachlässigt und kann daher hier nicht besser getroffen werden.

Abbildung 6.a zeigt den Betrag des absoluten Fehlers in N über der Federkonstante c einer Feder für verschiedene Werte von c . Dargestellt werden der maximale Fehler, der minimale Fehler und der Durchschnittsfehler über die Zeit oder äquivalent den Weg. Die Werte der Federkonstanten sind in N/m dargestellt. Der minimale Fehler ist immer Null. Das bedeutet, dass für jeden getesteten Parameter zumindest zu einem Zeitpunkt die Zielkurve perfekt erreicht wird. Der maximale Fehler zeigt wiederum, dass zu einem Zeitpunkt der Fehler sehr hoch ist. Der Durchschnittsfehler zeigt den Fehler als Durchschnitt über den gesamten Zeitraum an. Abbildung 6.b zeigt die relativen Fehler der gleichen Parametervariation in % an.

Um die Fehler besser analysieren zu können bieten sich dreidimensionale Darstellungen an, die die Zeitkomponente zusätzlich aufnehmen.

Die Diagramme in Abbildung 7 zeigen eine entsprechende Darstellung der gleichen Parametervariation. Es wird deutlich, dass wie auch in Abbildung 5 zu erkennen ist, die großen Fehler erst zum Ende der Simulation auftreten. An dieser Stelle ist das Pedal maximal ausgelenkt und trifft auf einen Anschlag. Dieser Anschlag wurde hier nicht speziell modelliert und führt daher zu den hier dargestellten Abweichungen. Die maximale Fehlerkurve aus Abbildung 6.a lässt sich in Abbildung 7.a gut als die oberste Kante identifizieren.

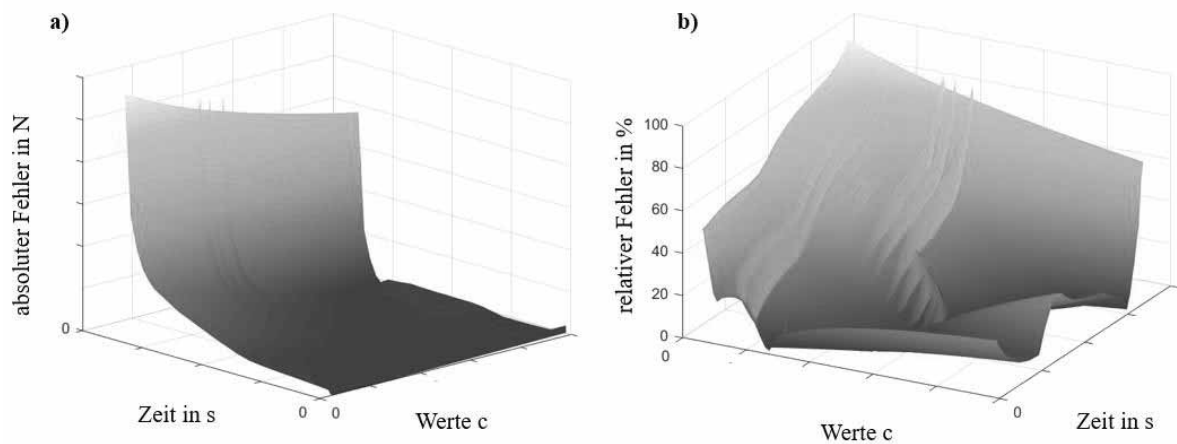


Abbildung 7: Absoluter (a) und relativer Fehler (b) einer Variation über den Parameter c_4 und die Zeit

5 Zusammenfassung

Die Ergebnisse der Arbeit zeigen, dass ein vorhandenes und verifiziertes Simulationsmodell genutzt werden kann, um die Weiterentwicklung des Produktes massiv zu unterstützen. Ein fertig entwickeltes Pedal mit einer charakteristischen Kennlinie zwischen Kraft und Auslenkung lässt sich mit Hilfe der Parameteroptimierung durch den Austausch einzelner Komponenten, wie zum Beispiel der Federn, auf eine andere Kennlinie hin anpassen.

Um die neuen Werte für die Parameter zu finden wird ein nichtlineares Optimierungsproblem formuliert und gelöst. Ein vorhandenes und verifiziertes Modelica-Modell wird als FMU aus Dymola exportiert und mit einem Least-Square-Ansatz in MATLAB optimiert.

Literaturverzeichnis

- [1] Stein, Oliver. *Grundzüge der Nichtlinearen Optimierung*. Springer-Verlag, 2017.
- [2] Reinhardt, Rüdiger; Hoffmann, Armin; Gerlach, Tobias. *Nichtlineare Optimierung: Theorie, Numerik und Experimente*. Springer-Verlag, 2012.
- [3] Biegler, Lorenz T. *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. Society for Industrial and Applied Mathematics, 2010.
- [4] Björck, Åke. *Numerical methods for least squares problems*. Society for Industrial and Applied Mathematics, 1996.
- [5] Nocedal, Jorge; Wright, Stephen. *Numerical optimization*. Springer Science & Business Media, 2006.

Modelling and Simulation of All-Electric Machines and Renewable Electric Power Systems in Agricultural Operations

Felix Klabunde^{1*}, Christian Reinhold¹, Bernd Engel¹

¹elenia Institute for High Voltage Technology and Power Systems, Technische Universität Braunschweig, Schleinitzstrasse 23, 38106 Braunschweig, Germany; *f.klabunde@tu-braunschweig.de

Abstract. Today's operation of agricultural machines is still characterised by the use of fossil fuels. In the future, all-electric agricultural machines can be used, whose energy requirements must be covered by local renewable energies or the power grid. The occurrence of limit value violations in the power grid can slow down and hinder this change in agricultural operations. For this reason, this paper describes the modelling and simulation of all-electric agricultural machines, renewable energy systems and rural power grids and the evaluation of suitable energy supply strategies for the machines and needed investments in the power grid.

Introduction

The restructuring of the German energy system in the context of the Energiewende requires the interaction and commitment of private and commercial stakeholders. As one of the main players in decentralized energy production, today's agriculture already has a special role in the Energiewende. In addition to the widespread photovoltaic plants, agriculturally operated biogas plants can be used to provide weather-independent electrical and thermal power. The combination of different generation and storage systems leads to a variety of energy structures with different technical and economic potential. In combination with new electricity applications in agriculture, such as all-electric agricultural machines, these energy structures have the potential to make agriculture sustainable in the long term. To determine the technical-economic potential and to derive suitable recommendations for action for the involved stakeholders, it is necessary to carry out holistic modelling and simulation of the involved technologies. Therefore, this paper describes the modelling and simulation of agricultural influenced energy sys-

tems and power grids with a focus on all-electric agricultural machines. In a simulation of different scenarios, suitable energy supply strategies are derived.

1 All-electric agricultural machines

More and more manufacturers of agricultural machinery have turned their attention to all-electric agricultural machines in recent years. Given the further development of power electronics and electric motors, as well as new requirements in efficiency and environmental protection, a future use of all-electric agricultural machines is realistic and comprehensible. For many processes in field cultivation, there are currently only prototypes and concepts and no market-ready products, which is why the modelling approach of all-electric agricultural machines, as shown in Figure 1, is based on empirical values and consumption profiles of diesel-powered agricultural machines. The

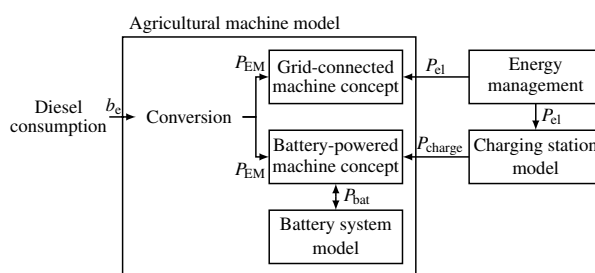


Figure 1: Modelling approach for the all-electric agricultural machines

diesel consumption profiles b_e are converted to electrical power values P_{EM} , which represent the power requirement of the electric drive train. These electrical consumption profiles serve as input data for the two

modelled machine concepts: the battery-powered agricultural machine and the grid-connected agricultural machine. The battery-powered agricultural machine is recharged by the charging station with the charging power P_{charge} . The charging station as well as the grid-connected agricultural machine are connected to the energy management system via their electrical power consumption P_{el} . The conversion process as well as both machine concepts are explained in more detail below.

1.1 Conversion of the diesel consumption profiles

The diesel consumption profiles used in this paper were created in the EkoTech research project [1] and represent a typical agricultural process chain in field cultivation. The process chain includes the agricultural processes soil tillage, sowing, plant protection, fertilisation and harvesting. While three agricultural machines are used simultaneously for harvesting (harvester and two tractors for transporting the harvest), only one agricultural machine is used at a time for the other processes. The use of diesel consumption profiles offers the advantage of using both simulated consumption profiles as well as measured consumption profiles. By using time series instead of static data, the varying power requirements of the individual processes can be realistically represented and process-dependent statements can be made regarding the machine concept to be used and the necessary parameterisation of the agricultural machine. The conversion to an electric power profile is done by using the calorific value of diesel H_i and overall efficiency of the diesel-powered drive train η_{CM} with P_{EM} representing the power consumption of the electric drive train and b_e representing the diesel consumption rate:

$$P_{\text{EM}} = b_e \cdot H_i \cdot \eta_{\text{CM}} \quad (1)$$

With the calorific value of diesel of 9.86 kWh/l and an assumed static efficiency of the diesel-powered drive train of 30 %, the electric motor output, shown in Figure 2 for the process soil tillage, is obtained.

Depending on the agricultural process, the energy and power requirements of the agricultural machines vary greatly. Particularly in field cultivation, many processes are associated with a high energy input. Machines used for soil tillage, sowing and as harvester in harvesting have a high energy consumption of up to 1 MWh per day, while other processes have a comparatively low energy consumption (e.g. the resulting elec-

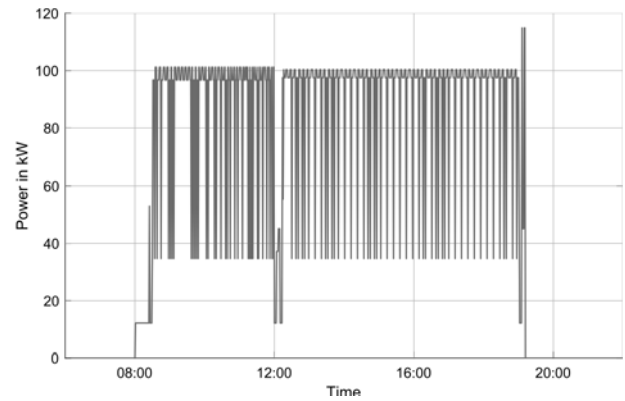


Figure 2: Resulting power consumption P_{EM} of the electric drive train during soil tillage

trical energy consumption for one tractor for transporting the harvest is 115 kWh per day). Due to the different requirements it is essential to consider different machine concepts with different parameterization for all electric agricultural machines.

1.2 Battery-powered agricultural machine

Battery-powered agricultural machines draw their energy from a battery system, which can be installed in the machine or as exchangeable front/back weight. The advantage of the battery-powered concept lies in its unlimited mobility and the possibility of bidirectional charging.

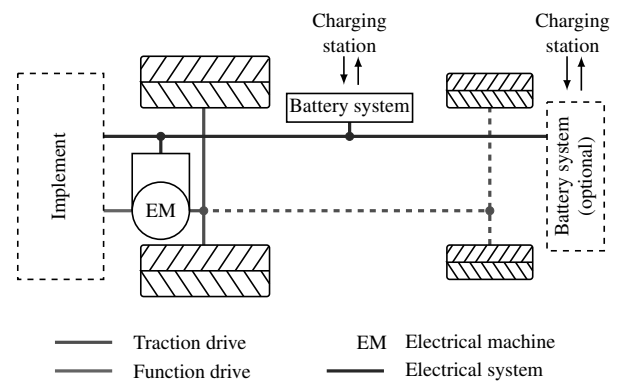


Figure 3: Possible machine concept of a battery-powered agricultural machine

The battery-powered agricultural machine is modelled by a battery system which is discharged by the specified electrical consumption profile and recharged by a model of a charging station. The presence of the agri-

cultural machine and the maximum possible recharging time at the charging station is determined by the power consumption of the electric drive train P_{EM} . If no power is needed ($P_{EM} = 0$) it is assumed for simplification that the agricultural machine is at the charging station and has the possibility of recharging at the charging station's full nominal power. The presence of the agricultural machine at the charging station is described by the parameter appearance:

$$P_{EM} \begin{cases} > 0 \rightarrow \text{appearance} = 0 \\ = 0 \rightarrow \text{appearance} = 1 \end{cases} \quad (2)$$

The charging power at the charging station P_{charge} is calculated depending on the chosen charging strategy (*Charging with maximum power*, *Charging with minimum power*) and the maximum charging power of the battery system:

$$P_{\text{charge}} = \min \begin{cases} P_{\text{bat,charge,max}} \\ P_{\text{charge,strategy}} \end{cases} \quad (3)$$

$P_{\text{bat,charge,max}}$ is the maximum charging power of the battery system and $P_{\text{charge,strategy}}$ is the charging power set by the charging strategy. $P_{\text{bat,charge,max}}$ is calculated depending on the state-of-charge of the battery and is based on typical battery charging profiles. In the event that the actual state-of-charge is smaller than the state-of-charge at cut-off voltage, the charging power is calculated using the following equation:

$$P_{\text{bat,charge,max}} = \frac{C \cdot E_{\text{bat,nom}}}{\eta_{\text{bat,charge}}} \quad (4)$$

C is the charging/discharging rate, $E_{\text{bat,nom}}$ is the nominal battery capacity, E_{bat} is the available battery capacity and $\eta_{\text{bat,charge}}$ is the charging efficiency of the battery system. If the battery voltage reaches the cut-off voltage, the charging current is reduced at constant battery voltage, resulting in a reduced charging power. This effect is implemented by adjusting equation 4:

$$P_{\text{bat,charge,max}} = \frac{C \cdot E_{\text{bat,nom}}}{\eta_{\text{bat,charge}}} \cdot \left(\frac{\text{SOC}_{\text{max}} - \text{SOC}}{\text{SOC}_{\text{max}} - \text{SOC}_{\text{cov}}} \right) \quad (5)$$

SOC is the available state-of-charge, SOC_{max} is the maximum state-of-charge and SOC_{cov} is the state-of-charge at cut-off voltage.

In the case of the loading strategy *Charging with maximum power*, the agricultural machine is reloaded

with the nominal charging power of the charging station $P_{\text{charge,nom}}$ after completion of the field cultivation:

$$P_{\text{charge,strategy}} = P_{\text{charge,max}} = P_{\text{charge,nom}} \quad (6)$$

This charging strategy enables the agricultural machine to be recharged quickly, reducing the required idle time of the agricultural machine and increasing the potential operating time. The second charging strategy, *Charging with minimum power*, calculates the minimum charging power as a function of the total idle time $T_{\text{appearance}}$ as shown in the following equations and Figure 4.

$$E_{\text{bat,dest}} = E_{\text{bat,nom}} - E_{\text{bat}} \quad (7)$$

$$P_{\text{charge,strategy}} = P_{\text{charge,min}} = \frac{E_{\text{bat,dest}}}{T_{\text{appearance}} \cdot \eta_{\text{charge}}} \quad (8)$$

$E_{\text{bat,dest}}$ is the required battery capacity to be recharged, E_{bat} is the available battery capacity and $P_{\text{charge,min}}$ is the minimum charging power. This charging strategy is dependent on the next operating time of the agricultural machine and potentially reduces load peaks by recharging evenly over the entire idle time of the agricultural machine.

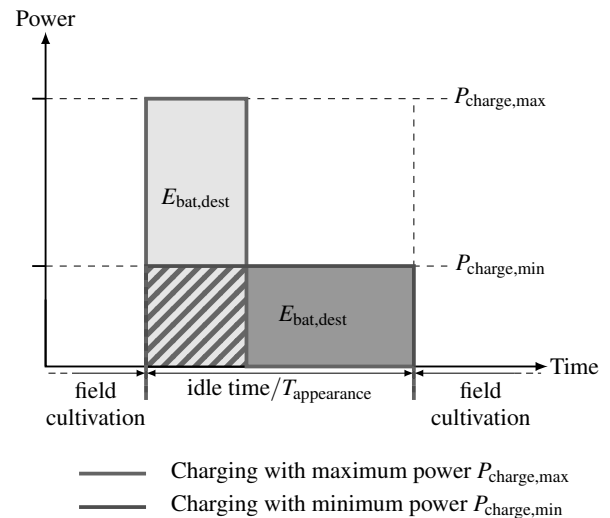


Figure 4: Influence of the charging strategies on charging power and time

For the discharge power of the battery system, it is assumed for simplification that the power consumption P_{EM} can be covered at any time. This assumption is sufficient, as it can be assumed that only agricultural machines that can provide the required electrical power are used.

1.3 Grid-connected agricultural machine

Grid-connected agricultural machines are continuously supplied with energy via a power cable and do not have their own energy storage. Higher theoretical engine power is advantageous than with battery-powered agricultural machines, but mobility is restricted due to the cable, which is why this concept is not suitable for every agricultural process. The concept is particularly suitable for the power-intensive soil tillage and for the harvester in harvesting. In contrast to the battery-powered agricultural machine, the engine power P_{EM} is directly transmitted as output power P_{el} to the energy management.

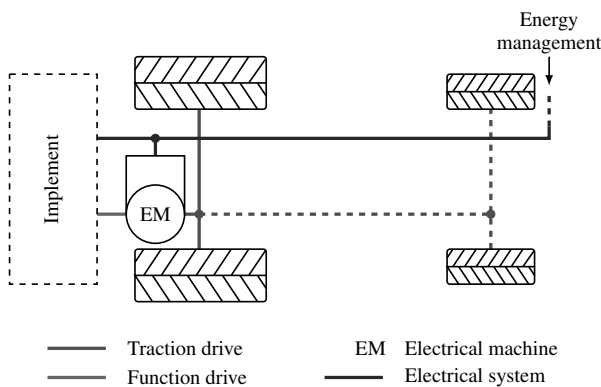


Figure 5: Machine concept of a grid-connected agricultural machine

2 Modelling of the renewable electric power system

All-electric agricultural machines can be supplied with energy from renewable energy plants operated by agricultural operations or from the power grid. To assess the effects of the integration of these machines and to derive suitable energy supply concepts, the modelling of the plants under consideration is necessary. In the following, the modelling of the photovoltaic plant, wind power plant and biogas plant as considered renewable energies, as well as the energy management and power grid, is briefly explained.

2.1 Renewable Energies

The photovoltaic model builds on the work already done in [2] and is based on a physics-based modelling

approach. Weather data (solar radiation, solar azimuth, solar altitude, outdoor temperature and cloud cover) from the test reference year of the German weather service [3] and serve as input data.

A data-based modelling approach as shown in Figure 6, is chosen for the wind turbine model, in which the generation capacity is calculated based on weather data from [4] and wind power curves from product data sheets of wind turbine manufacturers (e.g. [5])

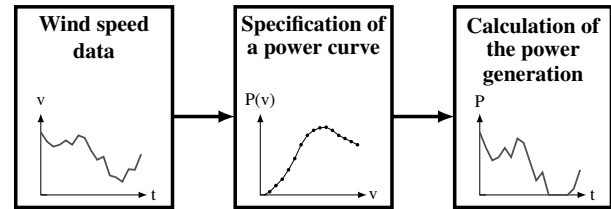


Figure 6: Data-driven modelling approach of the wind power plant

The biogas plant is modelled in a simplified way using the load profile E0 of E-Control (Austrian government regulator for electricity and gas) [6]. This simplification is possible if biogas plants are operated at constant generation capacity without a flexibility option.

2.2 Energy management and power grid

The energy management serves as a superordinate control system of the power flows between the individual models on the farm and provides the interface to the power grid. The control process aims to maximise the internal consumption of the energy generated by the renewable energies and thus reduce the amount of energy supplied by the grid.

The grid calculation is carried out with the help of MATPOWER [7]. MATPOWER calculates the power flow with the grid structure and generator and consumption power as input data. The grid results can then be analysed and visualised in MATLAB and required and suitable grid optimisation and reinforcement measures can be determined.

3 Simulation environment

The modelling of the technical components (Chapter 2) and the simulation studies (Chapter 4) are carried out in the institute's simulation environment eSE (elenia Simulation Environment) [8]. eSE is a MATLAB-based simulation environment for the scientific investigation

of electrical and thermal systems and their behavior. In addition, a flexible signal coupling of individual devices enables the investigation of connected systems and the testing of control concepts.

The simulation environment is divided into different modules, which can be used independently or in combination. The connection between all modules is the Simulator, which is represented by a single MATLAB class. It controls the information flow between the modules and takes over the central data management. The result of a simulation is a collection of time series of individual information flows and physical parameters of the devices, which can then be analyzed and evaluated. Figure 7 shows the three paths of the simulation process in eSE.

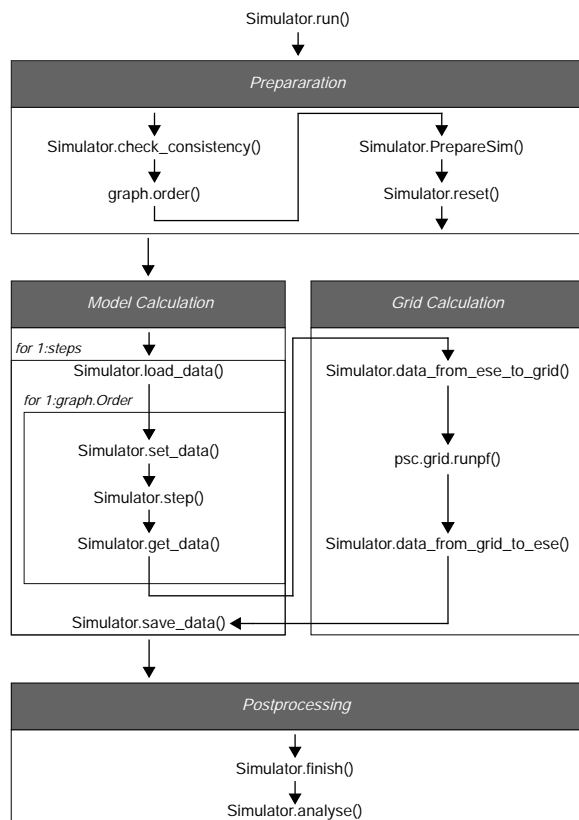


Figure 7: Simulation process in eSE

In the preparation step, the simulation is checked for consistency and possible errors and automatically repaired if necessary. Afterwards a directed graph is created, which indicates the simulation order of the models by means of a topological search algorithm. Before all models are reset to a valid initial state, the simulation is prepared. During the preparation, model relationships

are stored temporarily in a local data structure to avoid database queries during the simulation.

At the beginning of each simulation step, stored external data sets are reloaded from an HDF5 file. Subsequently, three methods are executed for each model step. Before the model calculation, data from other models or external data sets are assigned to the model. Then the model with the functional relationships is calculated and result values are queried afterwards. The result values for active and reactive power are used as input values for the grid calculation to perform the load flow calculation with MATPOWER. The resulting values of the grid resources are finally queried and stored together with the model results in the HDF5 file for time series.

In the final phase, defined key indicators of time series and model properties are calculated and stored in an SQLite database. In addition, economic calculations of stakeholders and models are executed.

4 Simulation scenarios and results

To assess the power grid integration for all-electric agricultural machines, several simulation scenarios are considered and compared with each other. The simulation scenarios differ according to the power grid connection level of the agricultural enterprise (low voltage or medium voltage) and the selected charging strategy (*Charging with minimum power*, *Charging with maximum power*), resulting in four simulation scenarios. Due to the differentiation at the power grid connection level, different compositions of renewable energies are considered for the scenarios. The selected parameterization is shown in Table 1.

Model	Low voltage scenarios	Medium voltage scenarios
Photovoltaic plant	30 kW	100 kW
Wind power plant	-	100 kW
Biogas plant	-	75 kW
Charging Station	50 kW	300 kW
Battery-powered machines	1x260 kWh 1x600 kWh	1x260 kWh 1x600 kWh
Grid-connected machines	2	2

Table 1: Parameterization of the simulation scenarios

For all scenarios, four all-electric agricultural machines are considered, whereby two are battery-powered with built-in batteries (used in fertilization, sowing and harvest) and two are grid-connected (used in soil tillage and harvest). The inclusion of four agricultural machines in the scenario enables a realistic representation of agricultural machines needed for a typical process chain in field cultivation. The battery-powered agricultural machines can be recharged after completion of the field cultivation at one charging station. A realistic annual load profile from a measurement campaign on a dairy farm with 120 cows is used as base load to model the remaining electricity consumption of the agricultural operation [9].

The low voltage distribution grid is represented by Kerber's *Landnetz Kabel 1* benchmark power grid and supplies five agricultural operations and three households [10]. The medium voltage distribution grid used is the CIGRE benchmark grid, where at load nodes 8 and 11 an agricultural operations is considered [11]. The remaining load nodes are mapped with standard load profiles for households and businesses [12].

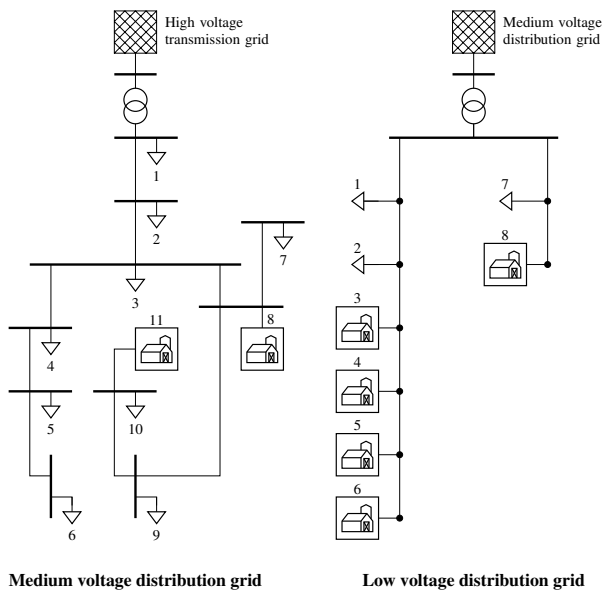


Figure 8: Structure of the modelled distribution grids

The simulation is carried out using nine randomly selected type days distributed over an entire year, which allows seasonally dependent fluctuations in generation and consumption to be depicted and meaningful results to be obtained for the entire year. The type days cover arbitrarily selected agricultural processes with each agricultural machine being used at least once. The

Season	Day	Frequency per year	Agricultural process
Winter	13.03.	100	Fertilization
	23.11.	20	-
	03.02.	20	Plant protection
Summer	14.08.	88	Harvest
	03.08.	18	Soil tillage
	26.05.	17	-
Transition	09.04.	73	-
	20.04.	14	Fertilization
	15.09.	15	Sowing

Table 2: Chosen type days for the simulation

frequency of the type days does not necessarily correspond to the whole duration of the agricultural processes, but since the agricultural machines are mostly used for further agricultural work, the selection of processes and allocation to the type days is assumed to be sufficiently accurate.

4.1 Technical results

The simulated agricultural operation with low voltage power grid connection shows an average own consumption of 39 % and an average self-sufficiency of 17 % when using the charging strategy *Charging with maximum power*. On days without the use of all-electric

	Charging max. Power	Charging min. Power
Own consumption	39 %	39 % ➡
Self-sufficiency	17 %	18 % ⬆
max. power grid supply	212 kW	188 kW ⬇
max. power grid feed-in	22 kW	22 kW ➡

Table 3: Technical results for the agricultural operations with low voltage power grid connection

agricultural machines, but with a high level of renewable energy production, there are feed-in capacities of up to a maximum of 22 kW. On days with usage of all-electric agricultural machines and low regenerative energy production, a grid supply of up to 212 kW is required. By using the charging strategy *Charging with minimum power*, the own consumption and self-sufficiency do not increase significantly, as the charging

times of the battery-powered agricultural machines are outside the generation times of the photovoltaic plant for both charging strategies.

	Charging max. Power	Charging min. Power
Own consumption	17 %	18 % ➡
Self-sufficiency	70 %	98 % ⬆
max. power grid supply	233 kW	123 kW ⬇
max. power grid feed-in	178 kW	178 kW ➡

Table 4: Technical results for the agricultural operations with medium voltage power grid connection

In the two scenarios with grid connection to the medium voltage distribution grid, the degree of self-sufficiency can be significantly increased, which can be attributed to the additional consideration of a biogas plant and wind power plant and the increased nominal capacity of the photovoltaic plant. Using the charging strategy *Charging with minimum power* instead of *Charging with maximum power*, the degree of self-sufficiency can be increased to nearly 100 %, since the generation capacity of the biogas plant and wind power plant can cover the entire consumption capacity more frequently (cf. Figure 9).

For the scenarios considered, possible voltage band violations and thermal overloads in the power grid are examined. The voltage applied to the network nodes in the power grid changes depending on the connected loads, generation plants and the location in the power grid. The permitted deviations from the nominal voltage are defined in DIN EN 50160 [13].

The thermal load capacity of the power grid elements used is regarded as a further key figure for determining the condition of the power grid. The nominal load capacity of the equipment is often specified in the associated data sheets and is assumed to be the limit value in the following.

	Charging max. Power	Charging min. Power
Voltage range deviations	18 %	18 % ➡
Thermal overload	21 %	18 % ⬇

Table 5: Frequency of limit violations in the low voltage power grid per year

Table 5 shows the results for the low voltage distribution

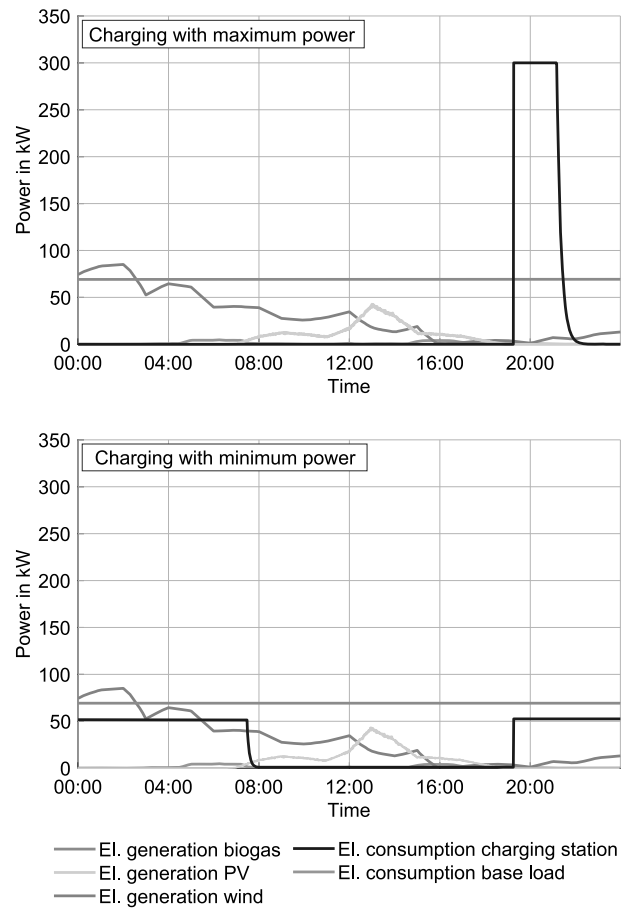


Figure 9: Electrical generation and consumption of different models on 15. September during sowing

grid. In both scenarios with connection to the low voltage distribution grid, voltage band violations and equipment overloads occur on almost every power grid element, whereby the frequency of equipment overloads per year is slightly reduced by using the charging strategy *Charging with minimum power* compared to *Charging with maximum power*. Due to the limit value violations in the low voltage power grid and the maximum power grid supply of the individual agricultural operations, it can be useful to change the grid connection level of these operations to medium voltage. The medium voltage distribution grid did not show any limit value violations in the two scenarios with connection to the medium voltage distribution grid and is therefore suitable for agricultural operations with a previous connection to the low voltage distribution grid as a possible grid integration solution for all-electric agricultural machines.

5 Conclusion and future plans

This paper describes preliminary studies for the research project “Concept and modeling of agricultural systems with renewable energy supply - Energy-4-Agri”. The development and evaluation of energy supply concepts for all-electric agricultural machines requires a holistic modelling of the involved technologies. This paper therefore provides information on the modelling and simulation of the technologies and energy systems.

Diesel consumption profiles were chosen as input data set for the all-electric agricultural machines. The conversion to electric power values can only be seen as a first approximation given a dynamic efficiency of combustion engines. Further research is therefore needed in the development of electric power profiles for all-electric agricultural machines, which take into account dynamic efficiency of the internal combustion engine and can thus provide more accurate simulation results. It has been shown that the charging strategy *Charging with minimum power* can already reduce the grid load in contrast to *Charging with maximum power* and increase the degree of self-sufficiency. In the future, further charging strategies can be modelled to determine the optimal charging times based on forecast data of the generation capacities of the renewable energies or depending on the power grid condition.

Acknowledgement

The work of this paper is originally based on the research project “Energy-4-Agri” (FKZ 03EI1013A). The authors acknowledge the support of the project within the energy research programme of the Federal Ministry of Economics and Energy and the Projektträger Jülich. The responsibility for the content of this publication lies with the authors and does not necessarily reflect the opinion of the project consortium Energy-4-Agri.

Supported by:



Federal Ministry
for Economic Affairs
and Energy

on the basis of a decision
by the German Bundestag

References

- [1] Frerichs L, Hanke S, Steinhaus S, Tröskén L. EKOtech - A Holistic Approach to Reduce CO2 Emissions of Agricultural Machinery in Process Chains. In: *9th AVL International Commercial Powertrain Conference 2017*, edited by SAE Mobilus, SAE Technical Paper Series. SAE International. 2017; .
- [2] Diekmann S, Reinhold C, Engel B. Centralized energy management for the optimization of residential districts. In: *International ETG Congress 2017*. ETG, Berlin and Offenbach: VDE Verlag GmbH. 2017; .
- [3] *Ortsgenaue Testreferenzjahre von Deutschland für mittlere, extreme und zukünftige Witterungsverhältnisse*. Offenbach: Deutscher Wetterdienst und Bundesamt für Bauwesen und Raumordnung. 2017.
- [4] Erichsen G. *DWD weather model data for energy system simulation: 2017*. 2020.
- [5] *NPS 100C-24: Datasheet*. Northern Power Systems.
- [6] *Sonstige Marktregeln Strom: Kapitel 6: Zählwerte, Datenformate und standardisierte Lastprofile*. Österreich: E-Control. 2015.
- [7] Zimmermann R, Murillo-Sanchez C, Thomas R. MATPOWER: Steady-State Operations, Planning and Analysis Tool for Power Systems Research and Education. *Power Systems, IEEE Transactions on*. 2011;26(1):12–19.
- [8] Reinhold C, Engel B. Simulation environment for investigations of energy flows in residential districts and energy management systems. In: *International ETG Congress 2017*. ETG, Berlin and Offenbach: VDE Verlag GmbH. 2017; .
- [9] Neiber J. *Strombedarf und Eigenstromversorgung in der Nutztierhaltung*. Mannheim. 2020.
- [10] Kerber G. *Aufnahmefähigkeit von Niederspannungsverteilnetzen fuer die Einspeisung aus Photovoltaikkleinanlagen*. Dissertation, Technische Universitaet Muenchen, Muenchen. 2010.
- [11] Rudion K, Orths A, Styczynski ZA, Strunz K. Design of benchmark of medium voltage distribution network for investigation of DG integration. In: *IEEE Power Engineering Society general meeting, 2006*. IEEE, Canada: IEEE Operations Center. 2006; .
- [12] Meier H, Fünfgeld C, Adam T, Schieferdecker B. *Repräsentative VDEW-Lastprofile*. Frankfurt am Main: VDEW. 1999.
- [13] DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE. *Voltage characteristics of electricity supplied by public distribution networks*. 50160. Berlin: Beuth Verlag GmbH. 2010.

Simulationsbasierte Dimensionierung von Regeneratoren für eine volatile Hochtemperatur-Abwärmeverstromung

Wolfgang Schlüter^{1*}, Jack Hanna¹, Konstantin Zacharias¹

¹Hochschule Ansbach, Residenzstraße 8, 91522 Ansbach; *wolfgang.schluter@hs-ansbach.de

Abstract. In Industrieprozessen, wie dem Schmelzen von Metallen, fällt Abgas mit sehr hoher Temperatur an, das oft nicht energetisch verwertet wird. Eine Möglichkeit, die thermische Energie zu nutzen, ist die Verstromung mittels einer Mikrodampfturbine. Aufgrund der entstehenden fluktuierenden Abwärme bei schwankenden Abgastemperaturen und Massenströmen ist dabei eine Zwischenspeicherung der Energie erforderlich, um eine konstante thermische Leistung am Dampferzeuger bzw. der Dampfturbine zu gewährleisten.

Aus verfahrenstechnischen Gründen bietet sich an, die Abwärme in zwei Regeneratoren zu speichern und zu entladen. Eine erste Dimensionierung kann anhand der verfahrenstechnischen Parameter und der Schwankungsbreite des Abwärmestroms durchgeführt werden. Eine realistische Dimensionierung muss jedoch auch das dynamische Verhalten berücksichtigen. Dazu wird ein thermodynamisches Regeneratormodell entwickelt, welches mit einer auf einem endlichen Automaten basierenden ereignisdiskreten Steuerung die Anlage in einem hybriden Simulationsmodell nachbildet. CFD-Simulationen liefern dabei den für den Wärmeübergang entscheidenden und analytisch schwer zu ermittelnden Wärmeübergangskoeffizienten. Bei der Simulation einer gleichmäßigen Aufladung zeigen thermodynamisches Regeneratormodell, CFD-Simulation und das aus der Literatur bekannte Stufenmodell nur geringe Abweichungen. Mit dem entwickelten hybriden Simulationsmodell kann die analytisch erfolgte Dimensionierung der Regeneratoren überprüft und verbessert werden, indem in einer dynamischen Simulation der Gesamtanlage der volatile Abwärmestrom berücksichtigt wird. Damit steht ein Werkzeug zur Verfügung, mit dem sich die Regeneratoren und die Anlagensteuerung genauer spezifizieren lassen.

1 Einleitung

Energieintensive Industriebranchen, wie Aluminiumgussbetriebe, erzeugen hochenergetisches Abgas, dessen Energie nur teilweise zur Erzeugung der benötigten Prozesswärme verwendet wird. Ein Großteil des Abgases verlässt die Anlage in Form von ungenutzter Abwärme im Rauchgas. Diese Energie im Abgas gilt es energetisch zu nutzen. Die energetische Nutzung von

industrieller Abwärme führt zur Energieeffizienzsteigerung und infolgedessen zu einer Erhöhung des Gesamtnutzungsgrades der zugeführten Energie. Neben der Rückführung der Abwärme in den Prozess (z.B. zur Materialvorwärmung), welche nicht immer problemlos integrierbar ist, oder der betriebsinternen Verwendung (z.B. Gebäudebeheizung) bietet sich vor allem bei einer hohen Abgastemperatur und hoher thermischen Leistung die Umwandlung der ungenutzten Abwärme in elektrische Energie mittels einer Mikrodampfturbine an.

Im Rahmen eines Forschungsprojektes soll eine Anlage zur Hochtemperatur-Abwärmeverstromung für einen Aluminiumgussbetrieb entwickelt und getestet werden. Der Kern der geplanten Anlage zur Abwärmeverstromung besteht aus einer Mikrodampfturbine mit einer thermischen Leistung von 1,2 MW. Für eine lange Lebensdauer der Mikrodampfturbine ist es notwendig, diese unter konstanten Betriebsbedingungen zu betreiben, d.h. eine konstante thermische Leistung am Dampferzeuger bzw. der Dampfturbine zu gewährleisten. Um den zeitlich stark schwankenden Abgasstrom eines Aluminiumschmelzofens zu vergleichmäßigen, eignen sich Regeneratoren für die Zwischenspeicherung der thermischen Energie (Abbildung 1).

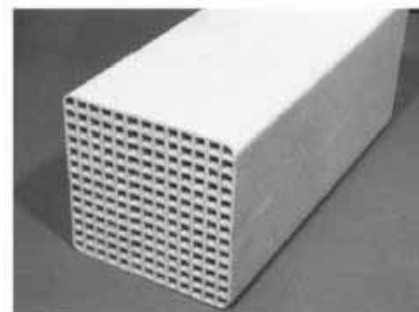


Abbildung 1: Regeneratorgeometrie [2]

2 Anlagenbeschreibung

Das hier verfolgte Konzept sieht vor, im Wechselbetrieb zwei Regeneratoren für die Zwischenspeicherung der thermischen Energie zu nutzen (Abbildung 2).

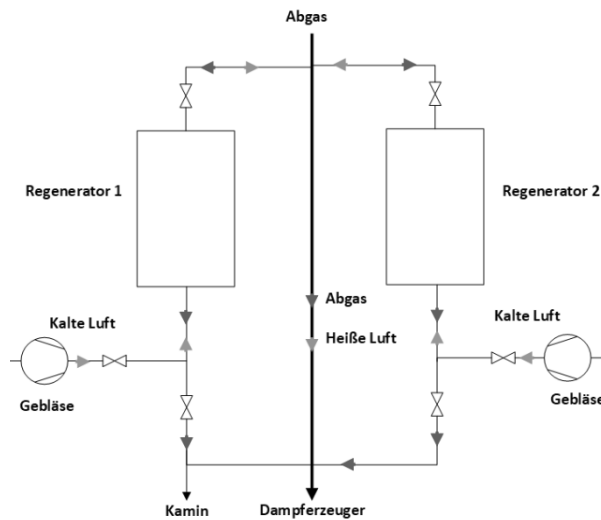


Abbildung 2: Verfahrensfließschema der Anlage

Dabei soll bei einer thermischen Leistung des Abgases über der benötigten Leistung für die Dampfturbine (Sollleistung) die zusätzliche Energie in einem der beiden Regeneratoren eingespeist werden. Sinkt die thermische Leistung des Abgases unter die Sollleistung, so wird die zusätzlich benötigte Energie aus einem geladenen Regenerator entnommen. Die Regeneratoren arbeiten im Wechselbetrieb, d.h. ein Regenerator dient zum Speichern von überschüssiger Energie, der andere zur Versorgung der Dampfturbine bei einem Mangel an thermischer Energie. In Abbildung 3 ist der gemessene Zeitverlauf für die stark fluktuierende thermische Leistung eines Aluminiumschmelzofens in 24 Stunden dargestellt.

Die Mikrodampfturbine benötigt eine konstante Wärmeleistung von 1,2 MW. Während der jeweils 6 1/2 Stunden dauernden normalen Betriebszeit des Schmelzofens fällt mehr Wärme an, als zum Betrieb der Dampfturbine notwendig ist. Problematisch sind die alle 8 Stunden auftretenden Freischmelz- und Reinigungszeiten, in denen die Brenner heruntergefahren bzw. ausgeschaltet werden. Diese Zeiten müssen für den Betrieb der Dampfturbine von den Regeneratoren überbrückt werden.

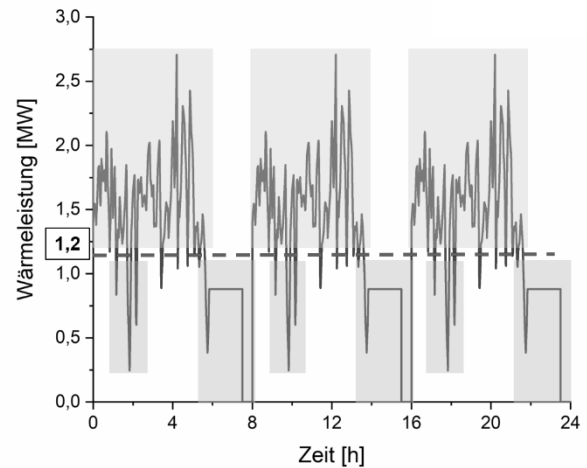


Abbildung 3: Typischer Leistungsverlauf des Abgases aus einem Aluminiumschmelzofen

Die Regeneratoren laden sich in der normalen Betriebszeit beide auf, um dann in der Freischmelz- und Reinigungsphase fast ihre komplette gespeicherte Energie wieder abzugeben. Dieser Vorgang wiederholt sich in jeder der 3 acht Stunden dauernden Arbeitsschichten.

Es stellt sich nun die Frage, wie die beiden Regeneratoren abhängig von Speichermaterial und Geometrie dimensioniert werden müssen, um den Betrieb der Dampfturbine über den kompletten Zeitverlauf sicherzustellen. Dabei gilt es aus Kosten- und Effizienzgründen eine Überdimensionierung zu vermeiden. Im Folgenden wird dafür ein simulativer Ansatz gewählt, der die dynamischen Schwankungen in der thermischen Leistung berücksichtigt.

3 Simulationsmodell

Das Simulationsmodell der in Abbildung 2 dargestellten Anlage besteht aus einem dynamischen Modell für die beiden Regeneratoren und der Steuerung der Anlage. Dabei erhält die Steuerung als Eingangsparameter die Sensordaten, die den aktuellen Regeneratorzustand beschreiben und liefert unter Berücksichtigung des Steuerungsalgorithmuses die Aktordaten, die die Abgasströme zu den beiden Regeneratoren steuern. Der Steuerkreis, der während der Simulation im Abstand von einer Sekunde durchlaufen wird, ist in Abbildung 4 dargestellt.

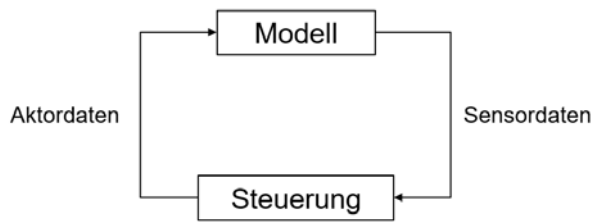


Abbildung 4: Steuerkreis der Anlage

3.1 Thermodynamisches Regeneratormodell

Um das thermodynamische Verhalten der Regeneratoren simulativ abzubilden wird ein thermodynamisches Modell (TDM) entwickelt, dass durch eine örtliche Diskretisierung der Speichermasse und des Volumens im Strömungskanal die Leistungsbilanzen berücksichtigt. Durch die gleichmäßige Geometrie genügt es, den Wärmeübergang für ein einzelnes Rohr mit umgebender Speichermasse zu modellieren (Abbildung 5).

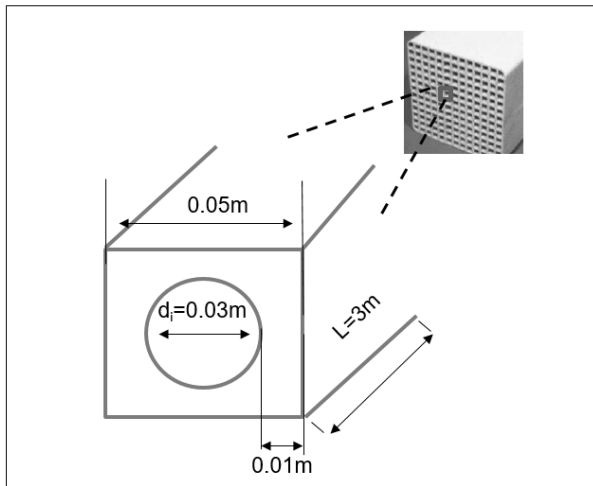


Abbildung 5: Rohr mit umgebender Speichermasse

Das mathematische Modell wird anhand einer Diskretisierung in Längsrichtung in 5 Segmente sowohl des Rohres (Gasseite) als auch der Speichermasse (Solidseite) erläutert (Abbildung 6).

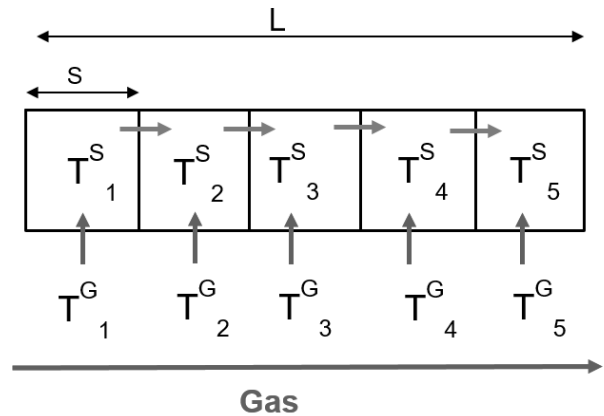


Abbildung 6: Räumliche Diskretisierung

Die Änderung der inneren Energie in den einzelnen Segmenten des Regenerators auf der Solidseite resultiert aus der Differenz zwischen zu- und abgeführter Wärmeleistung.

$$\frac{dE}{dt} = P_{zu} - P_{ab} \quad (1)$$

Die zugeführte Leistung eines Segmentes berechnet sich aus der abgeführten Leistung des vorhergehenden Segmentes und der durch Konvektion abgeführten Leistung P_{konv} des Abgases. Die abgeführte Leistung der einzelnen Segmente berechnet sich aus der Wärmeleitung P_{kond} in das folgende Segment. Wärmeverluste über die Regeneratorwand sowie die radiale Wärmeleitung im Segment werden nicht berücksichtigt. Letztere trägt durch die dünne axiale Segmentdicke und der geringen Wärmeleitfähigkeit des Regeneratormaterials weniger als 5% zum Wärmetransport bei. Die Regeneratorgeometrie geht bei der Bestimmung der konvektiven Wärmeleistung P_{konv} neben dem Wärmeübergangskoeffizienten k durch die wärmetauschende Fläche A und in die Bestimmung der konduktiven Wärmeleistung P_{kond} mit der Segmentstirnfläche F ein.

$$P_{kond} = \frac{\lambda}{S} \cdot F \cdot (T_s^{aus} - T_s^{ein}) \quad (2)$$

$$P_{konv} = k \cdot A \cdot (T_G - T_S) \quad (3)$$

Die Energieerhaltung (Gleichung (1)) ist auch die Basis für die Modellierung auf der Gasseite. Die zugeführte Leistung besteht hier aus der einströmenden Wärme des

Abgases P_{ein} , die abgeführte Leistung bestimmt sich aus der ausströmenden Wärme des Abgases P_{aus} und der durch Konvektion an die Solidseite übertragene Wärme P_{konv} .

$$P_{\text{ein/aus}} = c \cdot \dot{m} \cdot T_G^{\text{ein/aus}} \quad (4)$$

Im Fall von 5 Segmenten müssen 10 Differentialgleichungen gekoppelt gelöst werden. Die Simulation eines Speichervorganges mit Matlab/Simulink zeigt die Aufwärmung des Regenerators bei heißem Abgas von 500°C innerhalb von einer Stunde (Abbildung 7).

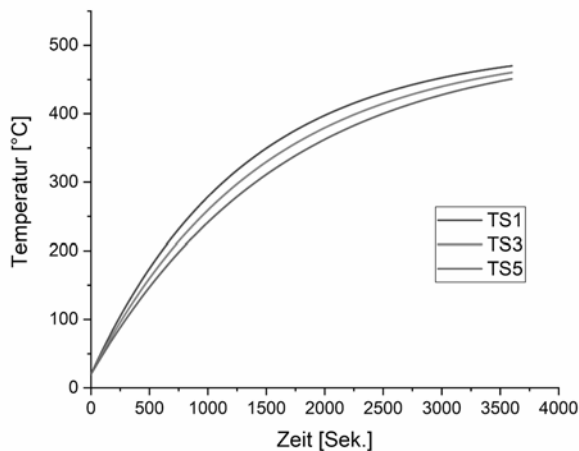


Abbildung 7: Aufheizvorgang

Die Ausspeicherung von Energie erfolgt durch das Zuführen von Luft auf der Abgasauslassseite. Für die Simulation der Ausspeicherung bedeutet dies, dass bei gleicher mathematischer Modellierung lediglich die Strömungsrichtung und die Gaseingangstemperatur geändert werden müssen.

Für einen einfachen Auf- bzw. Abkühlungsvorgang, wie er in Abbildung 7 gezeigt ist, eignet sich das Stufenverfahren (SV), mit dem sich der zeitliche Temperaturverlauf in Längsrichtung von Regeneratoren bestimmen lässt. Bei dem von Hausen [1] entwickelten Stufenverfahren 2 wird zunächst stufenweise die querschnittsgemittelte Temperatur der Speichermasse berechnet und daraus dann die Gastemperatur der jeweiligen Stufe. Mit diesem Verfahren, das auf der Diskretisierung der Fourierschen Wärmeleitungsgleichung beruht, kann man Regeneratoren verhältnismäßig einfach und hinreichend genau berechnen, sofern diese gleichmäßig über einen längeren Zeitraum auf- oder entladen

werden.

Das Stufenverfahren eignet sich gut zur schnellen Implementierung bei konstanten oder auch gering variierenden Gaseintrittstemperaturen und zeigt dabei eine große Genauigkeit. Für stark volatile Gaseintrittstemperaturen, wie im hier untersuchten Fall, ist das gewählte Differenzenverfahren zu ungenau. Weiterhin erweist sich die Implementierung in ein übergeordnetes Modell, das die komplette Anlage (Abbildung 2) unter den Betriebsbedingungen (Abbildung 3) abbildet, als sehr schwierig. Sowohl das Stufenverfahren wie auch das hier entwickelte thermodynamische Prozessmodell benötigen zur Simulation den Wärmeübergangskoeffizienten k . Zu dessen Bestimmung werden numerische Strömungssimulationen durchgeführt.

3.2 Numerische Strömungssimulation

In der numerischen Strömungssimulation (CFD) werden die Gaskanäle in ihrer exakten geometrischen Form modelliert. Da es sich um einen instationären Wärmetransportvorgang handelt, wird neben den Navier-Stokes-Gleichungen auch die Energiegleichung gelöst. Somit wird in jedem Zeitschritt das Temperaturfeld für die Speichermasse und das Heißgas berechnet. Im Vergleich zum Stufenverfahren und dem dynamischen Modell wird der Wärmeübergangskoeffizient zeitlich und örtlich für die reale Strömungsgeometrie bestimmt und somit kann das Ergebnis aus der CFD-Simulation als Referenz für das thermodynamische Modell dienen. Der entscheidende Nachteil der Strömungssimulation sind die langen Rechenzeiten, die die Simulation eines Regenerators für 24 Stunden aktuell unmöglich machen. In Abbildung 8 werden die drei Möglichkeiten, den Aufwärmvorgang in einem Regenerator mit der Länge 0.5 m zu simulieren, miteinander verglichen.

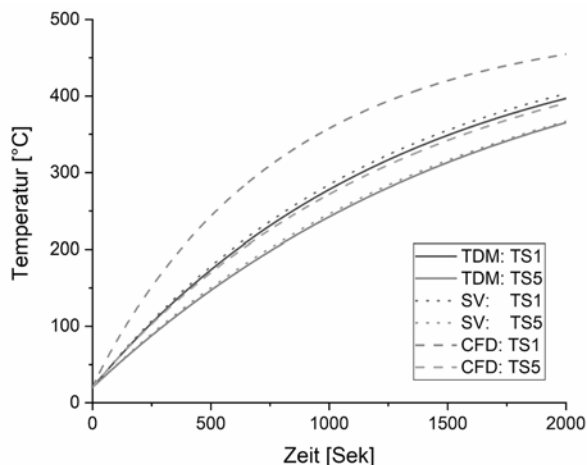


Abbildung 8: Simulierter Aufheizvorgang des Regenerators

Es zeigt sich, dass durch die Berücksichtigung der realen Strömungsverhältnisse in der CFD-Simulation der Wärmeübergang höher als im thermodynamischen Modell ist. Tabelle 1 zeigt für die drei Verfahren die Aufheizdauer bis zu einer Zieltemperatur von 300°C.

Segment	DM	SV	CFD	Einheit
TS1	1138	1103	705	Sekunden
TS5	1402	1376	1187	Sekunden

Tabelle 1: Aufheizdauer für Zieltemperatur 300°C

Über eine Kopplung aus CFD-Simulation und dynamischen Modell kann man sich die Vorteile des jeweiligen Verfahrens zu Nutze machen. Aus der CFD-Simulation können die Wärmeübertragungskoeffizienten für die einzelnen Segmente berechnet werden. Dadurch kann eine höhere Genauigkeit im dynamischen Modell erzielt werden.

Das dynamische Modell berücksichtigt nur die wärmeübertragende Fläche und das Speichervolumen. Dadurch sind die Simulationsergebnisse des thermodynamischen Modells für die drei in Abbildung 9 im Querschnitt gezeigten Kanalgeometrien identisch.

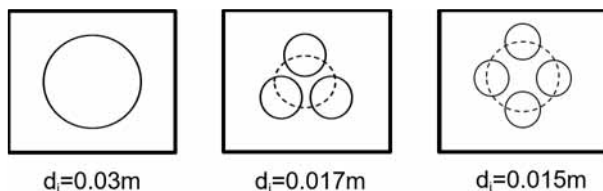


Abbildung 9: Mögliche Regeneratorgeometrien

Die CFD-Simulation kann hier die Unterschiede aufzeigen und die Geometrie mit dem besten Wärmeübergang identifizieren, wobei dabei auch die Durchlässigkeit der Kanäle und der Druckverlust zu berücksichtigen sind (Abbildung 10).

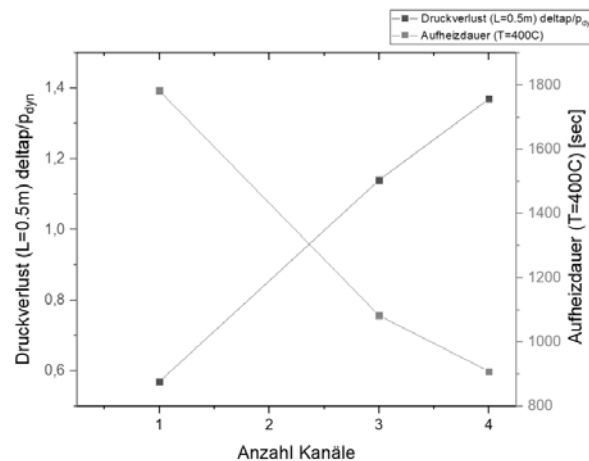


Abbildung 10: Aufheizdauer und Druckverlust bei steigender Anzahl an Strömungskanälen

3.3 Steuerung

Während es sich bei dem Modell für die beiden Regeneratoren um ein dynamisches Prozessmodell handelt, liegt bei der Steuerung ein ereignisgesteuertes diskretes System vor. Für die mathematische Modellierung der Steuerung wird ein Standardautomat [3] bestimmt. Tabelle 2 zeigt die Zustände des Standardautomaten, die insbesondere den Betriebszustand der beiden Regeneratoren beinhalten. Im Zustand 1 und Zustand 6 werden die Regeneratoren nicht betrieben, der Abgasstrom wird komplett vorbeigeleitet.

Nummer	Zustand
1	Aus (Leer)
2	Aufladen Regenerator 1
3	Aufladen Regenerator 2
4	Entladen Regenerator 1
5	Entladen Regenerator 2
6	Aus (Voll)

Tabelle 2: Zustände der Regeneratoren

Der Automat reagiert auf bestimmte Ereignisse, die aus Temperatur- und Volumenstromsensoren ermittelt werden können. Betrachtet werden der Wechsel von einem Defizit an thermischer Leistung zu einem Überschuss,

der umgekehrte Fall, sowie das Erreichen einer minimalen oder maximalen Temperaturgrenze in einem Regenerator (Tabelle 3).

σ	Ereignis
1	Leistungsdefizit => Leistungsüberschuss
2	Leistungsüberschuss => Leistungsdefizit
3	Temperatur im Regenerator 1 erreicht Maximalwert
4	Temperatur im Regenerator 2 erreicht Maximalwert
5	Temperatur im Regenerator 1 sinkt auf Minimalwert
6	Temperatur im Regenerator 2 sinkt auf Minimalwert

Tabelle 3: Sensorbasierte Ereignisse

Der Standardautomat, der die Steuerung beschreibt, bestimmt abhängig vom aktuellen Zustand der Steuerung und dem neuen Zustand den Folgezustand. Die Funktionsweise der Steuerung ist in Abbildung 11 als Automatengraph dargestellt.

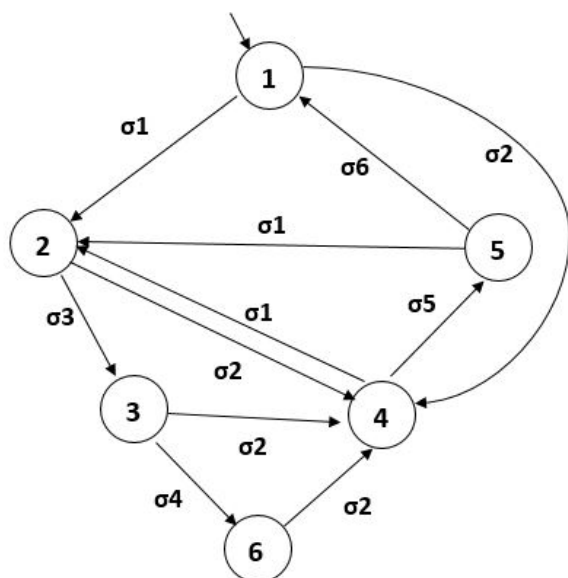


Abbildung 11: Funktionsweise der Steuerung

Die Steuerung ist so aufgebaut, dass der erste Regenerator bevorzugt aufgeladen wird. Sind beide Regeneratoren aufgeladen, was durch das jeweilige Erreichen der Maximaltemperatur detektiert wird, wird der Zustand 6 (Aus, voll) erreicht. Bei einem Leistungsdefizit wird dann der Zustand 4 angenommen, d.h. der erste Regenerator wird bevorzugt entladen.

Nach der simulationstechnischen Realisierung des hybriden Gesamtmodells, bestehend aus der diskreten

Steuerung und dem kontinuierlichen thermodynamischen Modell der beiden Regeneratoren, kann bei gegebener Abgastemperatur und Abgasmassenstrom der Zeitverlauf der eingespeicherten Energiemengen bestimmt werden.

4 Dimensionierung

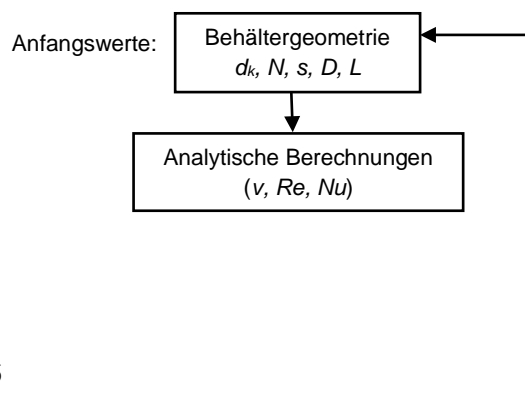
Eine erste Dimensionierung kann anhand der verfahrenstechnischen Parameter und der Schwankungsbreite des Abwärmestroms durchgeführt werden. Der Vergleich mit dem entwickelten komplexen thermodynamischen Modell zeigt jedoch, dass dabei das dynamische Verhalten des Abwärmestroms nur unzureichend berücksichtigt wird. Auch ein stark vereinfachtes Modell, welches die Regeneratoren als Speicher für die überschüssige thermische Energie betrachtet, ohne die thermodynamischen Gegebenheiten abzubilden, führt nicht zu den Ergebnissen der komplexen Simulation.

4.1 Analytische Dimensionierung

Die Geometrie des Regenerators kann durch eine analytische Dimensionierung berechnet, bewertet und entsprechend ausgewählt. Ausgehend von einem angenommenen Behältervolumen wird die Speichermasse berechnet. Aus der Masse m , der spezifischen Wärmekapazität und der Temperaturänderung im Festkörper kann mit

$$Q = m \cdot c \cdot \Delta T \quad (5)$$

die maximal im Regenerator speicherbare Energie Q bestimmt werden. Mittels bekannter Korrelationen wird über physikalische Kennzahlen wie Reynolds- und Prandtlzahl der Wärmeübergang und der Druckverlust berechnet. In einem rekursiven Verfahren lassen sich damit die geometrischen Abmessungen des Regenerators mit dem mittleren Wärmeübergang und dem Druckverlust über das Regeneratorrohr bestimmen (Abbildung 12). Ziel des heuristischen Optimierungsverfahrens ist dabei eine Regeneratorgeometrie mit möglichst hohem mittlerer Wärmeübergang bei gleichzeitig möglichst niedrigem Druckverlust.



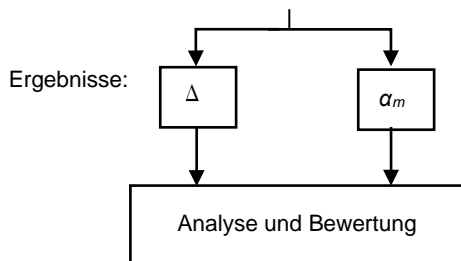


Abbildung 12: Analytische Regenerator-Auslegung

Nach erfolgter analytischer Auslegung liegen die folgenden Auslegungsparameter des Regenerators vor:

- Speichermassenvolumen V_s
- Durchmesser der Strömungskanäle d_k
- Anzahl der Kanäle N
- Wanddicke zwischen den Kanälen s
- Druckverlust über das Regeneratorrohr ΔP
- Mittlerer Wärmeübergang im Regenerator α_m

Bei dem geschilderten Verfahren geht die Volatilität der anliegenden thermischen Leistung des Abgases (siehe Abbildung 3) jedoch nur durch einige wenige Kenngrößen ein. So wird als Regeneratortemperatur der Mittelwert der Abgastemperaturen und als Massenstrom der maximale Abgasmassenstrom, der in den Regenerator strömen kann, angenommen. Die analytische Auslegung des Regenerators liefert mit den getroffenen Annahmen und dem gemessenen Leistungsverlauf des Abgases folgende Ergebnisse (Tabelle 4).

	Wert	Einheit
Breite	0,5	m
Länge	8	m
Kanaldurchmesser	0,03	m
Wanddicke	0,018	m
Mittlerer Wärmeübergang	39,19	W/m ² K
Druckverlust	7,9	mbar

Tabelle 4: Ergebnis der analytischen Regenerator - Auslegung

Es stellt sich die Frage, ob die analytische Auslegung die volatilen Abgasströme in ausreichendem Maße berücksichtigt, so dass die Anlage die notwendige thermische Leistung für die Mikrodampfturbine zur Verfügung

stellen kann, ohne dass die Regeneratoren überdimensioniert sind.

4.2 Simulationsbasierte Dimensionierung

Grundlage der simulationsbasierten Dimensionierung sind die gemessenen Temperaturen und Massenströme während einer 8-Stunden-Schicht. Die Zeitverläufe in Abbildung 13 zeigen, dass das einstündige Freischmelzintervall und das halbstündige Reinigungsintervall am Ende einer Schicht die Herausforderung für den Regeneratorbetrieb darstellen. In den vorhergehenden 6½ Stunden muss genügend Energie in den Regeneratoren gespeichert werden, so dass diese die 1½ Stunden überbrücken und den Betrieb der Mikrodampfturbine sicherstellen können.

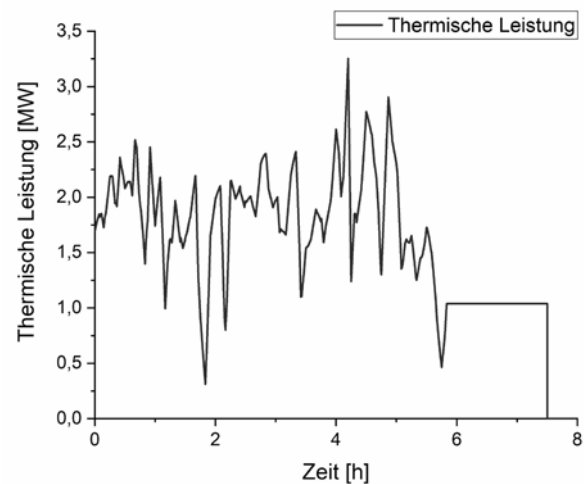


Abbildung 13: Thermische Leistung des Abgases

Die Simulation liefert zu den eingelesenen Abgaswerten die Energieinhalte der Regeneratoren, die sich bei der beschriebenen Steuerung einstellen. Bei einer Länge von 3 m zeigt Abbildung 14, dass am Ende einer Schicht lediglich der erste Regenerator komplett entladen ist. Damit kann auch mit deutlich verkürzten Regeneratoren der Betrieb der Mikrogasturbine sichergestellt werden.

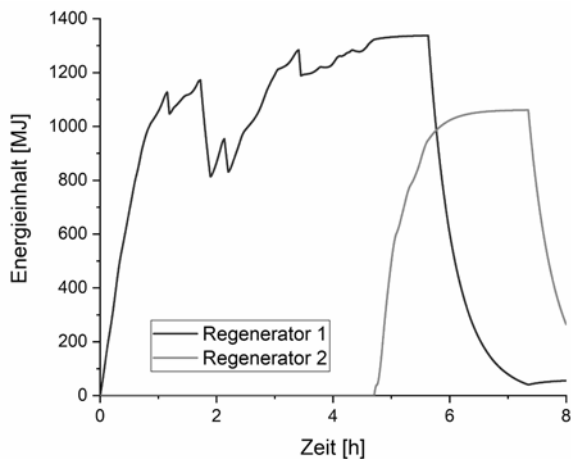


Abbildung 14: Energieinhalt der Regeneratoren

Betrachtet man lediglich die ein- bzw. ausgespeicherte Leistung in den Regeneratoren ohne die detaillierte Simulation des Wärmeübergangs, so reduziert sich das Modell für die Regeneratoren auf Gleichung (1). Der Vergleich mit dem thermodynamischen Modell in Abbildung 15 zeigt jedoch, dass die Modellierung des Wärmeübergangs eine entscheidende Rolle spielt: Im Unterschied zum thermodynamischen Modell werden bei dem einfachen Modell noch vor dem Ende der betrachteten Schicht beide Regeneratoren entladen. Folgt man dem Ergebnis der Simulation mit dem einfachen Speichermodell, so reicht die Länge von 3 m für den Betrieb der Mikrodampfturbine nicht aus. Da die thermische Trägheit des Systems nicht berücksichtigt wird, kommt es zu einer zu schnellen Lade- und Entladezeit für die Regeneratoren, woraus die Abweichung zum komplexen thermodynamischen Modell resultiert.

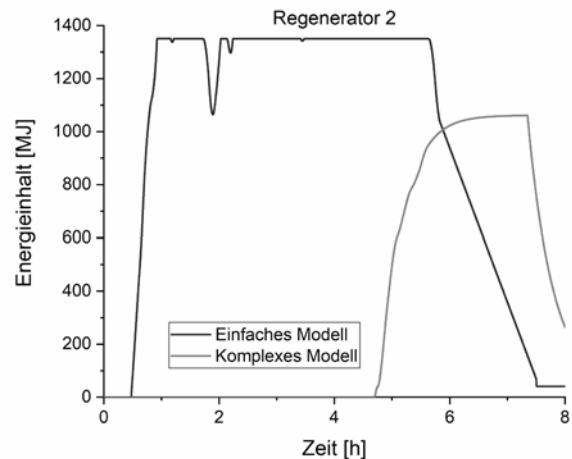
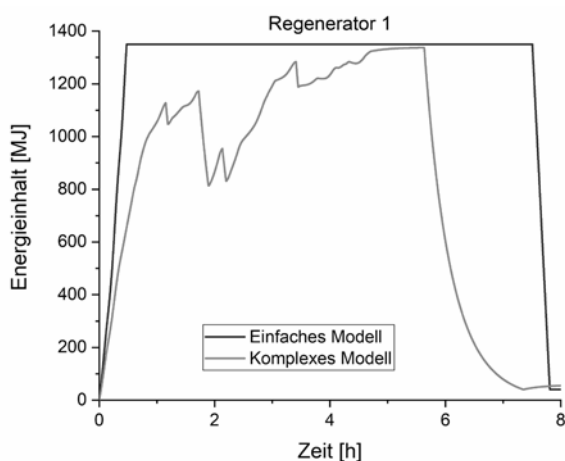


Abbildung 15: Vergleich des einfachen Speichermodells mit dem thermodynamischen Modell

5 Ausblick

Das thermodynamische Modell des Regenerators berücksichtigt zwar die Wärmetransportmechanismen Konvektion und Leitung, nicht aber die Wärmeübertragung durch Strahlung. Da das Abgas Temperaturen von 900°C aufweist, beträgt der Strahlungsanteil bei der übertragenen Wärme etwa 35% und sollte für eine genaue Modellierung nicht vernachlässigt werden. Daher wird in einem nächsten Schritt der Einfluss der Strahlung mitberücksichtigt.

Aktuell ist der Aufbau einer Laboranlage auf dem Energiecampus Nürnberg geplant, in der das Auf- und Entladerverhalten eines Modellregenerators getestet werden kann. Damit kann die Simulation mit Messdaten validiert werden. Mit dem validierten Simulationsmodell der Anlage lassen sich dann verschiedene Materialien und Regeneratorgeometrien im Hinblick auf ihre Eignung für den Betrieb sowohl der Labor- als auch der Pilotanlage untersuchen. Das Simulationsmodell gestattet durch seinen modularen Aufbau den einfachen Test von verschiedenen Steuerungsstrategien, so dass auch dieses Optimierungspotential für die geplante Anlage genutzt werden kann.

Literatur

- [1] Hausen, H. *Wärmeübertragung im Gegenstrom, Gleichstrom und Kreuzstrom*. Zweite Auflage. Springer-Verlag Berlin Heidelberg New York; 1976
- [2] VDI-Gesellschaft. *VDI-Wärmeatlas*. 11. Auflage. Springer-Verlag Berlin Heidelberg; 2013
- [3] J. Lunze, *Ereignisdiskrete Systeme*, De Gruyter 2012

Simulation eines MPR-basierten Energiemanagementsystems

Sebastian Schwarz^{1*}, Andreas Rehkopf¹

¹Institut für Automatisierungstechnik, TU Bergakademie Freiberg, Bernhard-von-Cotta-Str. 4
09599 Freiberg, Deutschland; *sebastian.schwarz@aut.tu-freiberg.de

Abstract. Dieser Beitrag beschreibt die Entwicklung eines MPR-basierten EMS für ein Netzwerk hybrider Energiesysteme. Zunächst wird das grundlegende Energiesystem beschrieben und modelliert. Basierend auf dem so gewonnenen Zustandsraummodell wird eine MPR und das daraus resultierende Optimierungsproblem formuliert. Letztendlich werden verschiedene Simulationsszenarien präsentiert und ihre Ergebnisse diskutiert.

Einleitung

Bedingt durch die Energiewende kommt es zu einer zunehmenden Dezentralisierung der Energieerzeugung. Gleichzeitig erhöht die Zunahme privat betriebener Energieanlagen und die staatlich garantierte Abnahme überschüssiger Leistung die Bedeutung des Netzes als temporären Handelsplattform. Der parallel dazu stattfindende Einsatz von KWK-Anlagen, etwa Nano- und Mikro-Blockheizkraftwerken ($P^{el} \leq 10 \text{ kW}$), zur Reduktion von CO₂-Emissionen führt zu einer Kopplung der Sektoren Wärme und Strom. Dies stellt im ländlichen Raum ohne Wärmenetz eine besondere Herausforderung dar [1]. So muss die elektrische Leistung wärmegeführter KWK-Anlagen mit der privater, regenerativer Anlagen, wie Photovoltaik und Windkraft, in Einklang gebracht werden.

Ein Energiemanagementsystem muss somit in der Lage sein, sowohl die Deckung der thermischen und elektrischen Last für jeden an das Niederspannungsnetz angeschlossenen Verbraucher, als auch den wirtschaftlichen Betrieb der privat geführten Anlagen zu gewährleisten. Der Verbraucher tritt dabei zusätzlich als Erzeuger in Erscheinung. Gleichzeitig müssen die wirtschaftlichen Interessen des Netzbetreibers, über dessen Netz der Austausch vollzogen wird, berücksichtigt werden.

Dieser Beitrag beschreibt die Entwicklung eines EMS für einen Verbund von Bestandsanlagen, sowohl von KWK- als auch RES-Anlagen, innerhalb eines Niederspannungsnetzes. Zunächst wird ein Modell für eine beispielhafte Laboranlage generiert. Dieses wird anschließend für verschiedene, mögliche Konfigurationen variiert. Die verschiedenen Konfigurationen werden beispielhaft zu einem Netzabschnitt zusammengefasst und ein Zustandsraummodell synthetisiert und in eine modellprädiktive Regelung überführt. Anschließend werden verschiedene Testszenarien definiert und ihre Ergebnisse diskutiert.

1 Systembeschreibung

Die grundlegende Anlagenkonfiguration besteht aus Batterie- (BES) und thermischen Energiespeicher (TES), sowie M-BHKW, Solarthermie- (ST), Photovoltaik- (PV) und Windkraftanlage (WK).

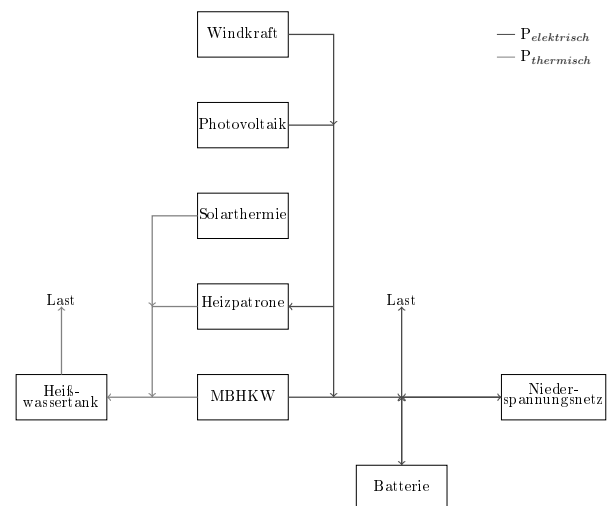


Abbildung 1: Schema der Laboranlage am Institut für Automatisierungstechnik, TU Bergakademie Freiberg

Zusätzlich ist eine Heizpatrone (HP) zur Erhöhung der Flexibilität, in Form einer dynamischen Wärmequelle aber auch als dynamischer Verbraucher, installiert. Die Konfiguration stellt sich somit schematisch gemäß Abb. 1 dar.

Die Deckung der thermischen Last erfolgt gemäß

$$P_{Last}^{th}(t) = \beta P_{TES}^{th}(t) \quad (1)$$

, wobei β als Komfortfaktor eine Deckung der thermischen Last innerhalb eines Toleranzbereiches erlaubt. Für die Deckung der elektrischen Last (vgl. 1) ergibt sich

$$P_{Last}^{el}(t) = P_{System}^{el}(t) + P_{Netz}^{el}(t) \quad (2)$$

Dabei gilt

$$P_{System}^{el}(t) = P_{BES}^{el}(t) + P_{BHKW}^{el}(t) - P_{HP}^{el}(t) + P_{PV}^{el}(t) + P_{WK}^{el}(t) \quad (3)$$

Die Einträge von PV- und WK-Anlage werden im Weiteren als nicht regelbar angenommen.

Für die Ladegleichung des TES gilt [6]

$$EC_{t_1} = EC_{t_0} + \int_{t_0}^{t_1} -P_{TES}^{th}(t) + P_{BHKW}^{th}(t) + P_{HP}^{th}(t) + P_{ST}^{th}(t) dt \quad (4)$$

Das Ladeverhalten des BES kann mittels

$$SOC_{t_1} = SOC_{t_0} + \int_{t_0}^{t_1} \mu_c^{BES} P_{BES,c}^{el}(t) - \frac{P_{BES,d}^{el}(t)}{\mu_d^{BES}} - P_{BES,sdc}^{el}(t) dt \quad (5)$$

mit

$$P_{BES}^{el}(t) = \begin{cases} (1 - \delta_{BES}(t)) P_{BES,d}^{el}(t) \\ -\delta_{BES}(t) P_{BES,c}^{el}(t) \end{cases} \quad (6)$$

beschrieben werden. Weiterhin gilt für die maximale Lade- und Entladeleistung

$$P_{BES,c}^{el}(t) \leq P_{c,max}^{el}(SOC(t)) \quad (7)$$

$$P_{BES,d}^{el}(t) \leq P_{d,max}^{el}(SOC(t)) \quad (8)$$

Die Kosten für die Nutzung des BES [7] ergeben sich

gemäß

$$c_{BES}^{abn} = \frac{c_{BES}^{inv}}{SOC_{BES}^{max} * L} \quad (9)$$

$$c_{BES} = c_{BES}^{abn} \int_{t_0}^{t_1} P_{BES,c} + P_{BES,d} dt \quad (10)$$

Gl. 9 beschreibt die Abnutzungskosten des BES bei ursprünglichen Investitionskosten c_{BES}^{inv} und maximaler Zyklenzahl L . Daraus resultierend lassen sich Kosten für die Nutzung des BES ableiten (vgl. Gl. 10).

Der Betrieb des MBHKWs erfolgt in binären Betriebszuständen mit einem elektrischen Eigenbedarf im Standby $P_{BHKW_s}^{el}$

$$P_{BHKW}^{el}(t) = \begin{cases} P_{BHKW_s}^{el} & , \text{ für } q(t) = 0 \\ P_{BHKW,max}^{el} & , \text{ für } q(t) = 1 \end{cases} \quad (11)$$

$$P_{BHKW}^{th}(t) = \begin{cases} 0 & , \text{ für } q(t) = 0 \\ P_{BHKW,max}^{th} & , \text{ für } q(t) = 1 \end{cases} \quad (12)$$

Die Berücksichtigung des Betriebszustandsübergangs des BHKWs kann bei definierter Anfahrtszeit t_{up} und Abschaltzeit t_{down} mittels

$$E_{BHKW}^{el}([t_0, t_1]) = E_{BHKW,up}^{el} + \int_{t_0+t_{up}}^{t_1} P_{BHKW}^{el}(t) dt \quad (13)$$

beziehungsweise

$$E_{BHKW}^{el}([t_0, t_1]) = E_{BHKW,down}^{el} + \int_{t_0}^{t_1-t_{down}} P_{BHKW}^{el}(t) dt \quad (14)$$

erfolgen. $E_{BHKW,up}^{el}$ und $E_{BHKW,down}^{el}$ werden dabei als konstant vorausgesetzt. Analog erfolgt die Betrachtung für $E_{BHKW}^{th}([t_0, t_1])$ innerhalb eines Zustandswechsels. Bei Berücksichtigung der Kopplung zwischen elektrischer und thermischer Leistung ergeben sich resultierend aus der Verbindung zwischen Brennstoffbedarf und thermischer Leistung Kosten für den Betrieb des BHKWs mit

$$c_{BHKW} = c_{Gas} \mu_{BHKW_M} \mu_{BHKW_G} E_{BHKW}^{el}([t_0, t_1]) + c_{BHKW}^{inv} (t_1 - t_0) q(t_1) \quad (15)$$

Der erste Term beschreibt die Kosten bedingt durch den Brennstoffverbrauch, der zweite berücksichtigt den Verschleiß basierend auf Wartungs- und zu erwartenden Reparaturkosten für die Anlage.

Für die HP gilt

$$P_{HP}^{th}(t) = \mu_{HP} P_{HP}^{el}(t) \quad (16)$$

und es ergeben sich die Betriebskosten mittels

$$c_{HP} = c_{HP}^{abn} \int_{t_0}^{t_1} P_{HP}^{el} dt \quad (17)$$

Der durch das EMS zu verwaltende Netzabschnitt kann mittels der im Abschnitt befindlichen Menge an Energiesystemen \mathcal{H} , im Weiteren als Haushaltssysteme H_i , beschrieben werden. Die Komponentenkonfiguration der einzelnen Systeme H_i kann dabei von der grundlegenden in Abb. 1 dargestellten abweichen. Die Systeme können somit in die Klassen \mathcal{H}_{VER} , \mathcal{H}_{RES} und \mathcal{H}_{BHKW} eingeteilt werden. \mathcal{H}_{VER} sind reine Verbraucher, die keine eigene Erzeugung aufweisen. \mathcal{H}_{RES} sind Systeme, die ein oder mehrere regenerative Energiesysteme besitzen. Bei Existenz einer ST-Anlage wird zudem ein TES vorausgesetzt. Für eine PV- oder WK-Anlage ist ein BES innerhalb des Systems möglich. Die Installation einer HP wird nur bei elektrischer Erzeugung und zusätzlich zu einer ST-Anlage angenommen. \mathcal{H}_{BHKW} bilden Systeme mit einem MBHKW und somit mit einem TES und einer HP. Insgesamt ergibt sich somit

$$\mathcal{H} = \mathcal{H}_{VER} \cup \mathcal{H}_{RES} \cup \mathcal{H}_{BHKW} \quad (18)$$

mit

$$\mathcal{H}_{VER} \cap \mathcal{H}_{RES} \cup \mathcal{H}_{BHKW} = \emptyset \quad (19)$$

Der Austausch elektrischer Leistung innerhalb des Netzabschnitts kann als gerichteter, vollständiger Graph zwischen den einzelnen H_i modelliert werden. Für $P_{Netzi}^{el}(t)$ (vgl. Gl. 2) des H_i gilt dann

$$P_{Netzi}^{el}(t) = P_{Ni}^{el}(t) + \sum_{\forall j \in \mathcal{H} \setminus i} \mu_{L,j} * P_{Nji}^{el}(t) - \sum_{\forall j \in \mathcal{H} \setminus i} P_{Nij}^{el}(t) \quad (20)$$

Dabei berücksichtigt $\mu_{L,j}$ etwaige Leitungsverluste bei Bezug $P_{Nji}^{el}(t)$ aus anderen Systemen H_j . $P_{Nij}^{el}(t)$ modelliert die Leistungsübertragung an andere Systeme H_j . $P_{Ni}^{el}(t)$ beschreibt den Bezug $P_{imp_i}^{el}(t)$ und die Abgabe $P_{exp_i}^{el}(t)$ von Strom über die Grenzen des betrachteten Netzabschnitts hinweg

$$P_{Ni}^{el}(t) = \delta(t) P_{imp_i}^{el}(t) - (1 - \delta(t)) P_{exp_i}^{el}(t) \quad (21)$$

Bei einem Strompreis c_{Strom} , einer Einspeisevergütung für Strom c_{EEG} und einem verringerten Preis für Strom c_{VStrom} , der direkt im Netzabschnitt bezogen wird, ergeben sich die Kosten für die Netzinteraktion c_{Netzi} gemäß

$$\begin{aligned} c_{Netzi} = & \int_{t_0}^{t_1} c_{Strom} P_{imp_i}^{el}(t) - c_E P_{exp_i}^{el}(t) dt \\ & + \int_{t_0}^{t_1} c_{VStrom} \sum_{\forall j \in \mathcal{H} \setminus i} \mu_{L,j} * P_{Nji}^{el}(t) dt \\ & - \int_{t_0}^{t_1} c_E \sum_{\forall j \in \mathcal{H} \setminus i} P_{Nij}^{el}(t) dt \end{aligned} \quad (22)$$

2 EMS-Synthese

Die Zustände des grundlegenden Systems bilden SOC , EC und q . Für die einzelnen Klassen der Systeme H_i ergeben sich allgemeine Zustandsvektoren

$$x_{VER} = \begin{bmatrix}] \\] \end{bmatrix} x_{BHKW} = \begin{bmatrix} EC \\ q \end{bmatrix} x_{RES} = \begin{bmatrix} SOC \\ EC \end{bmatrix} \quad (23)$$

Basierend auf der Zugehörigkeit zu einer Klasse kann x_i formuliert werden. Dabei weisen Systeme aus \mathcal{H}_{VER} , bedingt durch den gewählten Modellierungsansatz, einen leeren Zustandsvektor auf. Für die Steuervektoren kann ein analoger Ansatz gewählt werden:

$$u_{VER} = \begin{bmatrix} P_{imp}^{el} \\ P_{exp}^{el} \end{bmatrix} \quad u_i = \begin{bmatrix} u_{Komp} \\ P_{imp}^{el} \\ P_{exp}^{el} \end{bmatrix} \quad (24)$$

Zusätzlich zu diesen Steuervektoren, basierend auf der Konfiguration der Systeme, wird bei $|\mathcal{H}| = n$ ein Steuervektor u_N der Länge $|u_N| = n(n-1)$ eingeführt. Dieser resultiert aus der Modellierung des Austauschs der Systeme H_i mittels eines vollständigen Graphs. Für das Netzwerk ergibt sich somit bei entsprechender Diskretisierung ein Modell [2]

$$\begin{aligned} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} (k+1) = & \begin{bmatrix} A_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & A_n \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} (k) \\ & + \begin{bmatrix} B_1 & \dots & 0 & B_{N_1} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & B_n & B_{N_n} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_n \\ u_N \end{bmatrix} (k) \end{aligned} \quad (25)$$

Der Ausgangsvektor y_i der Systeme wird über die Kosten für den Betrieb der einzelnen Komponenten zur Deckung der thermischen und elektrischen Last in H_i und ein Bewertungsmaß für die über den Netzabschnitt hin-

aus ausgetauschte Leistung gebildet.

$$y_i = \begin{bmatrix} c_{sys_i} \\ c_{netz_i} \end{bmatrix} \quad (26)$$

Damit ergibt sich die Ausgangsgleichung zu

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} (k+1) = \begin{bmatrix} C_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_n \end{bmatrix} x(k) + \begin{bmatrix} D_1 & \dots & 0 & D_{N_1} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & D_n & D_{N_n} \end{bmatrix} u(k) \quad (27)$$

Auf Basis dieses Modells kann eine MPR für eine beliebige Menge von Haushaltssystemen synthetisiert werden [5]. Im Weiteren wird eine Abtastezeit von $\Delta t = 300s$, ein Prädiktionshorizont $n_p = 72$ und Kontrollhorizont $n_K = 12$ festgelegt. Es ergibt sich

$$Y(k) = Fx(k) + \Phi U(k) \quad (28)$$

Anschließend wird ein lineares Gütefunktional definiert [3].

$$J(k) = eY(k) = \sum_{j=k}^{k+N_p-1} \sum_{i \in \mathcal{H}} c_{sys_i}(j) + c_{netz_i}(j) \quad (29)$$

Die sich aus der Modellierung ergebenden Nebenbedingungen lassen sich wie folgt definieren:

$$x_i^{min} \leq x_i(k) \leq x_i^{max}, \quad \forall i \in \{1, \dots, |x(k)|\} \quad (30)$$

$$u_j^{min} \leq u_j(k) \leq u_j^{max}, \quad \forall j \in \{1, \dots, |u(k)|\} \quad (31)$$

3 Simulationsszenarien

Die Simulationsumgebung besitzt eine Auflösungsrate von 1s. Der Eintrag regenerativer Energiesysteme wird auf Basis von normalverteilt gestörten Basisprofilen vorgenommen. Die Basisprofile berücksichtigen jahreszeitliche Schwankungen bezüglich der Peakleistung von ST- und PV-Anlagen, sowie sich verändernde Tageslängen.

Die elektrische Last der einzelnen Systeme H_i bilden Lastprofile aus [8], dabei finden sowohl Jahreszeit, als auch Wochentag Berücksichtigung. Die thermische Last wird auf Basis von Standardlastprofilen gebildet, analog zur elektrischen Last finden Wochentag und

Jahreszeit Berücksichtigung. Für die Prognosedaten der Lasten und regenerativen Energieeinträge werden der MPR entsprechend abweichende Profile zur Verfügung gestellt.

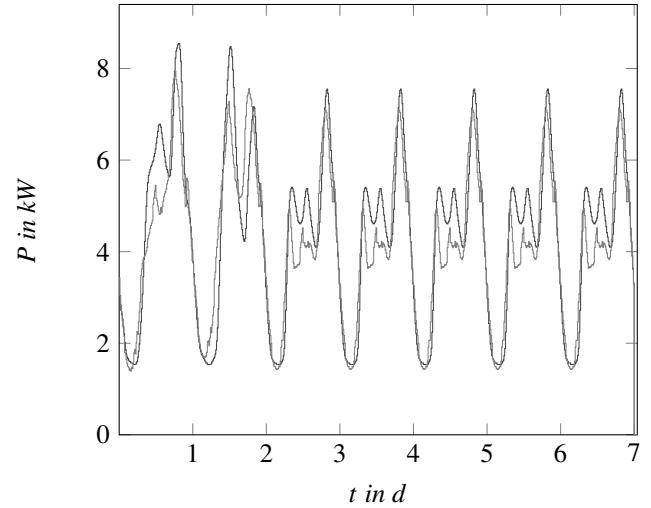


Abbildung 2: Beispielhafter thermischer Bedarf (—) und Prognose (---) im Winter mit Wochengang

Für die Simulation wird ein Netzabschnitt bestehend aus 10 häuslichen Systemen H_i modelliert. Drei der Systeme werden als reine Verbraucher modelliert, die restlichen Systeme gehören mindestens zu einer der Systemklassen \mathcal{H}_{BHKW} und \mathcal{H}_{RES} . Die Auslegung der im betrachteten Netzabschnitt auftretenden Komponenten ist Tab. 1 zu entnehmen.

Komponente	Auslegung
BHKW	6 kW _{el} /14.9 kW _{th}
HP	2 kW _{el} , $\mu_{HP} = 0.98$
PV	8 kW _p
ST	4 kW
WK	2 kW
BES	6.4 kW _h
TES	30 kW _h

Tabelle 1: Auslegung der möglichen Komponenten der häuslichen Systeme

Die Kostenparameter werden entsprechend EEG und KWKG vorgegeben. Die Einspeisevergütung für Systeme mit verschiedenen Erzeugern, die Vergütungsanspruch haben, wird vereinfachend auf die niedrigste

Vergütung für die Einspeisung von Leistung aus dem System festgelegt.

	Szenario 1	Szenario 2
c_{Strom}	$32 \frac{ct}{kWh}$	$32 \frac{ct}{kWh}$
c_{VStrom}	$28 \frac{ct}{kWh}$	$25 \frac{ct}{kWh}$
c_{Gas}	$6.63 \frac{ct}{kWh}$	$6.63 \frac{ct}{kWh}$
c_E^{PV}	$8.77 \frac{ct}{kWh}$	$10 \frac{ct}{kWh}$
c_E^{WK}	$6.04 \frac{ct}{kWh}$	$10 \frac{ct}{kWh}$
c_E^{BHKW}	$8 \frac{ct}{kWh}$	$10 \frac{ct}{kWh}$
c_{BES}^{inv}	$400 \frac{€}{kWh}$	$400 \frac{€}{kWh}$
c_{BHKW}^{inv}	$0.68 \frac{ct}{kWh}$	$0.68 \frac{ct}{kWh}$

Tabelle 2: Parametersetzung basierend auf KWKG/EEG (Szenario 1) und Austausch fördernd (Szenario 2)

4 Ergebnisse

Als Vergleichsszenario wird der Netzabschnitt ohne Interaktion zwischen den Systemen simuliert. Das heißt, ein Ausgleich von Einspeisung und Bezug erfolgt lediglich auf zufälliger Basis. Neben den zu erwartenden Peaks im Bezug in den Abendstunden (hoher elektrischer Bedarf, keine PV-Leistung) zeigen sich Leistungspeaks bei der Einspeisung etwa mittig des Tages (niedriger elektrischer Bedarf, hohe PV-Leistung), welche durch die verbauten BES-Systeme offensichtlich nicht abgefangen werden kann (vgl. Abb. 3). Zusätzlich zeigen die Phasen geringerer Leistungsschwankungen einen Off-Set von 0 kW hin zu einer negativen Leistung, es findet somit tendenziell ein Bezug über den Netzabschnitt hinweg statt.

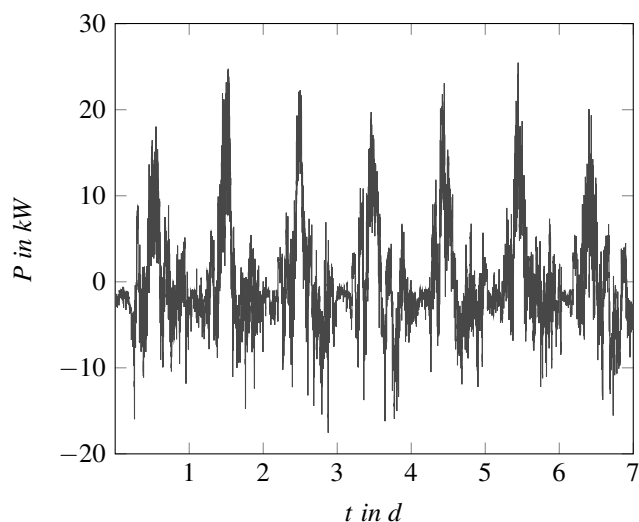


Abbildung 3: Leistungsaustausch über den Netzabschnitt hinweg ohne gezielten internen Austausch

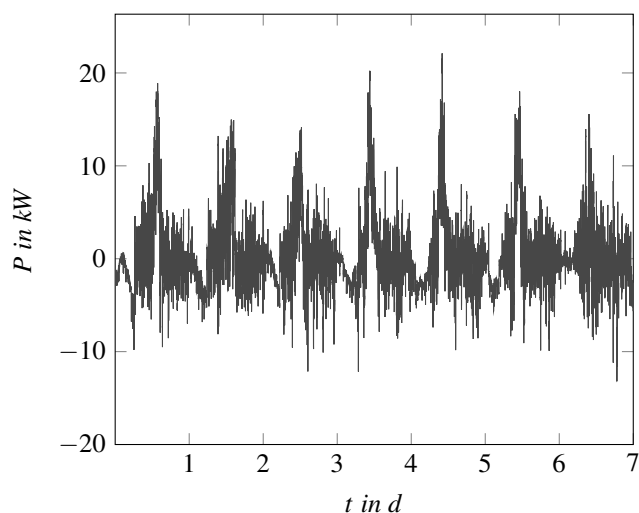


Abbildung 4: Leistungsaustausch über den Netzabschnitt hinweg mit gezielten internen Austausch (Szenario 1)

Bei gleichem Netzabschnitt zeigt sich bei einer Interaktion zwischen den System (vgl. Abb. 4) und somit einer geplanten Balancierung, eine Reduktion der Leistungspeaks, speziell bei der Einspeisung. Gleichzeitig wird die Breite und somit die Dauer der Peaks verringert. In den Bereichen geringerer Schwankungen ist der im vorherigen Fall beschriebene Off-Set behoben.

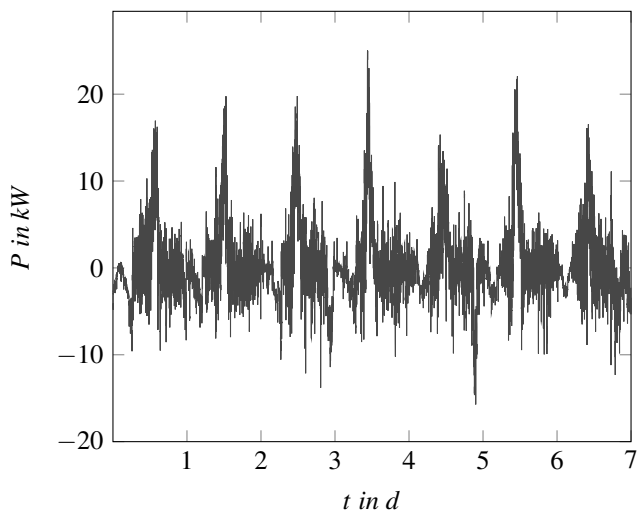


Abbildung 5: Leistungsaustausch über den Netzabschnitt hinweg mit gezielten internen Austausch (Szenario 2)

Abb. 5 zeigt neben den in Szenario 1 auftretenden Verbesserungen im Hinblick auf das Ausgangsszenario nur geringfügige Veränderung. So werden Peaks im Bezug, wie auch in der Einspeisung leicht erhöht. Dies kann auf eine verfrühte Nutzung im Abschnitt befindlicher Flexibilitäten [9] zurückgeführt werden. So werden die BES im Vergleich zum vorherigen Szenario frühzeitig entladen. Dies geschieht zur Deckung des Bedarfs in benachbarten Systemen, mit der Folge, im systeminternen Bedarfsfall diese Flexibilitätsoption nicht mehr zur Verfügung zu haben.

5 Zusammenfassung

Der Beitrag zeigt, dass durch ein geeignetes EMS die Lastdeckung in einzelnen häuslichen Systemen gewährleistet werden kann, bei gleichzeitiger Reduktion der Netzbelastung. Die wirtschaftlichen Interessen der Anlagenbesitzer können dabei gewahrt werden [4]. Die Simulation zur Verifikation der Funktionalität fand unter marktwirtschaftlichen Aspekten, gestützt durch KWKG und EEG statt. Ein vergleichendes Szenario zeigt zudem, dass unausgewogene Anreize zur Nutzung von Flexibilitäten und für den Austausch zwischen Systemen des Netzabschnittes, einen negativen Effekt auf die Netzbelastung hervorrufen können.

Literatur

- [1] Ch. Müller, A. Rehkopf. Optimale Betriebsführung eines virtuellen Kraftwerks auf Basis von gasgetriebenen Mikroblokheizkraftwerken. *at Automatisierungstechnik*. 2011; Band 59 (Heft 3): 180 – 186. doi: <https://doi.org/10.1524/auto.2011.0912>
- [2] S. Schwarz, A. Rehkopf. Modellprädiktive Regelung hybrider Energiesysteme. In T. Meurer et al. Tagungsband. *GMA-Fachausschuss 1.40*; 2018; Anif, Salzburg. 186-195. ISBN: 978-3-9819634-1-0
- [3] Y. Zhang et al. Optimal operation of a smart residential microgrid based on model predictive control by considering uncertainties and storage impacts. *Solar Energy*. 2015; 122: 1052–1065. doi: <http://dx.doi.org/10.1016/j.solener.2015.10.027>
- [4] A. Moser et al. A MILP-based modular energy management system for urban multi-energy systems: Performance and sensitivity analysis. *Applied Energy*. 2020; 261. doi: 10.1016/j.apenergy.2019.114342 <https://doi.org/10.1016/j.apenergy.2019.114342>
- [5] J. Adamy. *Nichtlineare Regelungen*. 1. Auflage. Berlin: Springer Verlag. 2009
- [6] P.O. Kriett, M. Salani. Optimal control of a residential microgrid. *Energy*. 2012; 42: 321–330. doi: 10.1016/j.energy.2012.03.049
- [7] Wencong Su et al. Model predictive control-based power dispatch for distribution system considering plug-in electric vehicle uncertainty. *Electric Power Systems Research*. 2014; 106: 29–35. doi: 10.1016/j.epsr.2013.08.01
- [8] T. Tjaden et al. Repräsentative elektrische Lastprofile für Einfamilienhäuser in Deutschland auf 1-sekündiger Datenbasis. *Datensatz*. Hochschule für Technik und Wirtschaft HTW Berlin. 2015.
- [9] B. Wille-Haussmann, O. Selinger-Lutz. Optimal Control of Distributed Energy Generation & Storages for Flexibility Provision on the Residential Level. *International ETG-Congress*. 2019. ISBN: 978-3-8007-4954-6

Vorgehensmodell zur Simulation von gebündeltem Energiebedarf

Benjamin Jacobsen^{1,2*}, Maximilian Stange³

¹ Professur BWL III - Unternehmensrechnung und Controlling, Technische Universität Chemnitz, Thüringer Weg 7, 09107 Chemnitz, Deutschland; benjamin.jacobsen@wirtschaft.tu-chemnitz

² Professur für Energie- und Hochspannungstechnik, Technische Universität Chemnitz, Reichenhainer Straße 70, 09126 Chemnitz, Deutschland

³ Fraunhofer Institut für Werkzeugmaschinen und Umformtechnik IWU, Reichenhainer Straße 88, 09131 Chemnitz, Deutschland

Abstract. Im Projekt GRIDS – Grüne Energie in industriellen Verbünden werden die Potentiale und Verbesserungsmöglichkeiten innovativer Energieversorgungskonzepte in Gewerbe- und Industrieparks untersucht. Für die Planung neuer (Greenfield) sowie den Aus- und Umbau bereits bestehender Gewerbeparks (Brownfield) werden Leitfäden zur Planung der Energiekonzepte erstellt. Eine elementare Herausforderung ist die Voraussage (Prädiktion) des (gebündelten) Energiebedarfs, die zur ökologisch und ökonomisch langfristig tragfähigen Auslegung von Versorgungskonzepten mit ihren Ressourcen (Betriebsmittel, Rohstoffe, Verlustenergie etc.) notwendig ist. Allein durch Datenerhebungen und Auswertungen vergleichbarer Energienutzer kann nicht umfänglich auf die jeweilige Situation am Ort der Planung rückgeschlossen werden. Denn bei Energiedaten handelt es sich für die meisten (Industrie) Unternehmen um sensible Daten, somit ist eine detaillierte Datenerhebung in aller Regel ausgeschlossen. Daneben besteht speziell bei kleineren Unternehmen keine Transparenz bezüglich ihrer Energiedaten. Diese sind meist sehr grobgranular und stützen sich allein auf monatliche Abrechnungen des Energieversorgers. Weiterhin könnte selbst bei hervorragender Datengrundlage nicht davon ausgegangen werden, dass selbst Betriebe identischer Branche, Größe und Struktur den gleichen Lastgang haben. Damit sollte zur allgemeinen Simulation von Lastgängen von Unternehmen immer eine Methode zum Einsatz kommen, die den Lastgang als stochastische Größe ermittelt, um Verfälschungen zu vermeiden.

Aber das Wissen über den zeitlichen Verlauf des Energie- und Leistungsbedarfs ist eine Grundvoraussetzung für eine nachhaltige Gestaltung des Energieversorgungssystems mit dem Ziel der Erhöhung des Anteils regenerativer Energien. Nur so können Betriebsmittel, insbesondere Speicher und regenerative Erzeugungsanlagen effizient geplant werden, was aufgrund der oft enormen Investitionshöhe sowie der Langfristigkeit der größtenteils irreversiblen Investitionen von hoher Bedeutung ist [1]. Es besteht demnach ein Mangel an Daten für eine verlässliche Planung von Energienetzen. Auf dieser Grundlage wird eine Methodik erstellt die ein Vorgehensmodell zur Simulation des Energiebedarfs von

gebündelten Versorgungskonzepten bereitstellt. Mit Hilfe der Simulation des Energiebedarfs von (heterogenen) Gruppen, wie sie durch Unternehmen in Gewerbeparks repräsentiert werden, kann die Simulation und die Auslegung der elektrischen Anlagen verbessert werden, somit kann die auf Erfahrungswerten basierende Netzplanung durch eine zielführende Methode unterstützt werden [2].

Neben der Unterstützung bei der Planung von Versorgungsnetzen, kann das Vorgehensmodell auch zur Generierung weiterer Energiedaten für andere Anwendungszwecke genutzt werden. Dazu zählt beispielsweise die energieorientierte Materialflusssimulation. Mit deren Hilfe können unter anderem Maßnahmen zur Energieflexibilität auf Unternehmens- und Gewerbebeparkebene untersucht werden. Dadurch können zusätzlich organisatorische Maßnahmen zur Senkung der Spitzenlast untersucht werden und wie sich diese auf die Produktion und damit den Unternehmenserfolg auswirken.

Einleitung

Die Identifikation interner Optimierungspotentiale und die Anforderungen externer Akteure (z.B. Energieversorger) machen eine Energiedatenerfassung für viele Unternehmen notwendig und hilfreich. Beispiele für externe Treiber sind:

- Erfassung des tatsächlichen Lastganges eines Unternehmens ab einen Jahresbedarf von $\geq 100.000 \frac{kWh}{a}$ (nach §12 StromNZV)
- Einführung eines Energiemanagementsystems nach DIN EN ISO 50001 um Erstattungen auf die Stromsteuer zu erhalten (nach §10 StromStG).

Trotzdem verfügen gerade Unternehmen, die keine energieintensiven Prozesse aufweisen, über nur unzureichende Energiedaten. Selbst dort wo prinzipiell Energiedaten in einer ausreichenden Granularität und Qualität vorhanden sind, gestaltet sich der Zugang von Planern und Forschern zu diesen Daten schwierig. Anhand von Energiedaten könnten unter anderem

Rückschlüsse auf Produktionsprozesse oder die allgemeine Geschäftslage gezogen werden, was die Geheimhaltung des detaillierten Energiekonsums vielerorts rechtfertigt.

1 Stand der Technik

Grundsätzlich können drei Arten der Energiedatenerfassung unterschieden werden [3]:

1. Berechnung

Die Berechnung von Energiewerten bedarf einer umfassenden Basis an technischen und betriebswirtschaftlichen Daten, die nicht immer ohne weiteres zur Verfügung stehen. Beispielsweise können mithilfe von Leistungsdaten einer Maschine und deren Betriebszeiten ein Jahresbedarf an Energie abgeschätzt werden. Vorteil dieser Methode ist es, dass keine Messgeräte und Eingriffe in den laufenden Betrieb nötig sind. Nachteilig ist die Komplexität der Berechnungen, um z.B. den Gesamtenergiebedarf eines Gebäudes zu berechnen. Hier ist davon auszugehen, dass gerade kleinere Unternehmen nicht über das notwendige Wissen zum Durchführen dieser Berechnungen verfügen [4].

2. Temporäre Messung

Mithilfe von temporären Messungen kann die Datengrundlage für die Berechnungen von Energiedaten verbessert werden. Die Kosten liegen dabei höher als bei der bloßen Berechnung, jedoch können die Datengrundlage und die Berechnungsergebnisse damit validiert werden. Außerdem liegen die Kosten unter denen von fest installierten Messeinrichtungen [4].

3. Fest installierte Messeinrichtung

Diese Art der Energiedatenerfassung eignet sich vor allem für tieferegreifende Analysen und bietet Automatisierungsmöglichkeiten für die Erfassung energiebezogener Daten. Dem gegenüber stehen jedoch hohe Anschaffungskosten sowie ein hoher Auswertungsaufwand der Datenmengen [4].

Alle Energiesimulationsansätze, seien es Simulationen zur Auslegung von Versorgungsnetzen oder eine energieorientierte Materialflusssimulation zur Untersuchung der Auswirkungen von Energieflexibilitätsmaßnahmen benötigen eine geeignete Datenbasis, um Energiemodelle zu erstellen [5]. Die zugrundeliegenden Ansätze zur Energiedatenerfassung

für die Simulation basieren dabei vor allem auf temporären Energiemessungen [5]. Solche Ansätze lassen sich im Maßstab eines Produktionssystems oder Fabrik noch vertreten, für die energetische Betrachtung eines ganzen Gewerbe- bzw. Industrieparks ist der Aufwand jedoch in den meisten Fällen zu hoch.

Um ohne großen Messaufwand Energiedaten für Gewerbe- bzw. Industrieparks zu generieren, könnten theoretisch Standardlastprofile herangezogen werden. Standardlastprofile sind Darstellungen eines Lastverlaufes in einem definierten Zeitraum. Vom Verband der Elektrizitätswirtschaft (VDEW) wurden solche Verläufe für die Bereiche Gewerbe und Haushalt erstellt. [6] Für Unternehmen der Industrie existieren solche verlässlichen Standardlastprofile nicht, da die Lastprofile hier sehr unterschiedliche Verläufe aufweisen [7]. Um trotzdem Standardlastprofile aus wenigen Messungen zu generieren, entwickelten Emde et al. eine Methode für die energieintensive Industrie [7]. Dabei stellen Sie als einen der Vorteile eine schnellere Simulation von Energieeffizienzmaßnahmen heraus.

Die Ergebnisse, dass die bekannten Standardlastprofile nicht ohne Anpassung für die Simulation des Elektroenergiebedarfs von modernen Industrieunternehmen angewandt werden sollten, konnten auch im Projekt GRIDS getroffen werden. Entgegen dem eben skizzierten Ansatz nach Emde et al. [7] wird in der vorliegenden Untersuchung ein Lastprofil simuliert, welches aus der Synthese verschiedener Standardlastprofile des VDEW [6] entsteht.

2 Methodik

Das Vorgehensmodell beruht auf der Verwendung von Standardlastprofilen. Wie in der Erstellung und Anwendung dieser Lastgänge begründet ist, können allgemeine Aussagen zum Konsumverhalten elektrischer Energie nur für bestimmte Nutzergruppen erstellt werden [6]. Demnach ist es notwendig, dass die in Gruppen zusammengefassten Energiekonsumenten als homogen angesehen werden (können). Nur so lässt sich eine Simulation des Energiebedarfs eines ganzen Gebietes in der notwendigen Form und Güte realisieren. Das Projekt GRIDS hat jedoch gezeigt, dass derartige Annahmen nicht getroffen werden können. Das aus diesem Grund entwickelte Vorgehensmodell zur verbesserten Simulation von Lastgängen, die eine Grundlage für die Planung von Anforderungen an das Elektroenergieversorgungsnetz bilden, verfolgt das Ziel

die Auswahl der geeigneten Standardlastprofile so zu treffen, dass die Versorgungszuverlässigkeit der Netze aufrechterhalten und unterstützt wird, ohne Überdimensionierungen, wie sie durch die klassische Anwendung von Standardlastprofilen bedingt werden, hervorzurufen.

Der Ablauf des standardisierten Vorgehens zur Netzplanung unter Zuhilfenahme von Standardlastprofilen ist in der folgenden Abbildung dargestellt.

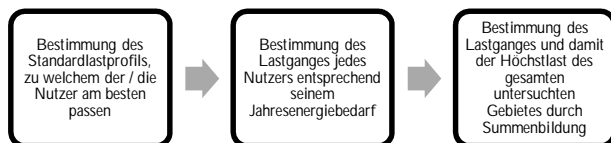


Abbildung 1: Ablauf der Bestimmung des Lastganges sowie der Höchstlasten bei standardmäßiger Verwendung von Standardlastprofilen

Mit Hilfe einer Untersuchung auf Grundlage stochastischer Kenngrößen, wie Korrelationen und weiterführenden Signifikanzanalysen werden Zusammenhänge der einzelnen tatsächlichen und zusammengefassten Lastgänge zu bekannten Standardlastprofilen [8] untersucht.

Dabei beschrieben die Korrelationskoeffizienten (r) den Zusammenhang zweier Größen [9]. Dieser Koeffizient hat einen Wertebereich zwischen -1 und 1 ($W_r = [-1; 1]$). Je näher er sich dem Betrag von 1 nähert, desto deutlicher ist der Zusammenhang der beiden Größen ausgeprägt, wobei ein positiver Korrelationskoeffizient ($r > 0$) auf einen ebenso positiven Zusammenhang hinweist, steigt eine Größe, steigt die andere ebenfalls. Ebenso deutet ein negativer Korrelationskoeffizient auf einen negativen (entgegengesetzten) Zusammenhang ($r < 0$) der untersuchten Größen hin. Im Allgemeinen wird ab einem Korrelationskoeffizient größer dem Betrag von 0,5 ($|r| > 0,5$) von einer hohen Korrelation gesprochen. [10] Für die Berechnung der Korrelation können verschiedene Verfahrensweisen herangezogen werden. Viele in der Praxis untersuchte und simulierte Größen sind normalverteilt, damit wird in der Regel auf das Verfahren nach Pearson zurückgegriffen. Jedoch zeigt sich in vielen Fällen, dass vor der Methodenwahl ein Test der Verteilungsart erfolgen sollte. [11]

Aufgrund der Verteilungsart der Daten wird hier auf die Berechnung der Korrelationskoeffizienten nach Spearman zurückgegriffen. Der Test auf Standardnormalverteilung nach Schiefe und Kurtosis hat

erwartungsgemäß ergeben, dass es sich bei den untersuchten Lastgängen und Standardlastprofilen nicht um normalverteilte Größen handelt. Somit können auf die untersuchten Größen keine parametrischen Verfahren angewandt werden und die Verwendung der Spearman Korrelation ist der nach Pearson vorzuziehen [12].

Bei einer bloßen mathematischen Analyse der Zusammenhänge kann es jedoch im Anwendungsfall auch bei richtiger Methodenwahl zu Fehlern kommen, diese werden in der Literatur oft durch so genannte Scheinkorrelationen beschrieben. Hierbei handelt es sich um mathematisch korrekt berechnete Zusammenhänge, die jedoch bei kausaler Betrachtung nur unzureichenden Rückschluss auf die reale Abhängigkeit zulassen. [13] Ein prominentes Beispiel für solche scheinbaren Zusammenhänge ist die Abhängigkeit der Geburtenrate in einem Gebiet von der Anzahl Störche, die sich dort aufhalten. Mathematisch konnte zwar ein eindeutiger Zusammenhang hergestellt werden [14], aus aktuellem Wissensstand kann hierbei aber kein kausaler Zusammenhang hergestellt werden. Darauf aufbauend kann bei den untersuchten Lastprofilen aber eine Scheinkorrelation ausgeschlossen werden.

Die Erstellung der Standardlastprofile erfolgte auf Grundlage einer repräsentativen Stichprobe. Somit ist es logisch und kausal nachzuvollziehen, dass es zwischen den untersuchten Lastprofilen einen Zusammenhang gibt. Dennoch findet zusätzlich eine grafische Anwendung der Maximum-Likelyhood-Methode statt. Die Notwendigkeit der zusätzlichen graphischen Auswertung wird an Abbildung 1 deutlich.

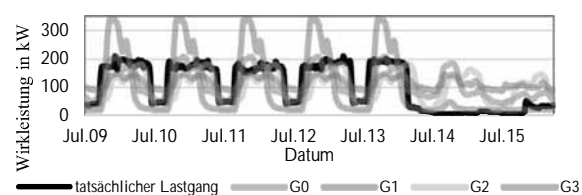


Abbildung 2: Gegenüberstellung der Standardlastprofile mit einem untersuchten tatsächlichen Lastgang innerhalb einer Woche

* Exkurs: Die Standardlastprofile (kurz SLP) G0 bis G3 beschreiben die üblicherweise zugrunde gelegten Lastprofile, differenziert nach Gewerbetyp, wobei G0 für allgemeine Gewerbe (gebildet aus dem Mittelwert der SLP G0 bis G6), G1 Gewerbe mit einer Arbeitszeit von werktags 08:00 Uhr bis 18:00 Uhr, G2 Gewerbe mit einem überwiegenden Energiebedarf in den Abendstunden und G3 durchlaufende Gewerbe beschreibt [15].

Es zeigt sich, dass neben rein mathematischen

Methoden auch grafische Verfahren zur Bestimmung des best fit herangezogen werden sollten, um die Abweichung des eigentlichen Leistungsbedarfs zu bewerten. Denn die Anwendung von Korrelationen untersucht ausschließlich den formalen Zusammenhang der verschiedenen Lastprofile, die tatsächliche Abweichung des (normierten) Energiebedarfs wird dadurch nicht gewährleistet. Dieser zusätzliche Schritt ermöglicht eine enorme Verbesserung der Simulation von Energiebedarfen. Die Verbesserung wird hauptsächlich durch die Separierung der Lastprofile nach Tageszeit sowie nach Wochentag erreicht. Schließlich wird ein angepasstes synthetisches Lastprofil so zusammengesetzt, dass ein best Fit nach visuellem Verlauf (wodurch Korrelation, Energiemenge und Spitzenlast gleichermaßen abgebildet werden) für Tag und Nacht sowie für Werktag und Wochenende zusammengefasst, gewährleistet werden kann. Nachdem angepasste synthetische Lastprofile (es entstehen Lastprofile für Nacht – Werktag; Tag – Werktag; Wochenende) zur Prädiktion des gebündelten Energiebedarfes gebildet und zu einem Lastgang zusammengefasst sind, werden sie auf Grundlage des zu erwartenden Gesamtenergiebedarfs skaliert. Die Skalierung erfolgt in diesem Fall mit dem Ziel der Gleichheit des erwarteten jährlichen Energiebedarfes und des Integrals des modellierten angepassten synthetischen Lastprofils.

3 Ergebnisse

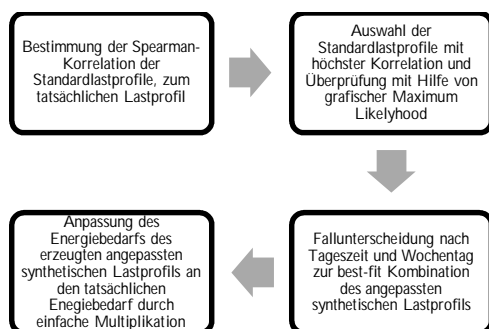


Abbildung 3: Ablauf der Bestimmung des Lastganges sowie der Höchstlasten bei Verwendung von angepassten synthetischen Lastprofilen

Das Vorgehensmodell führt eine Verbesserung der technischen Auslegung von Versorgungsnetzen herbei und unterstützt somit die ökologische und ökonomische Optimierung bei Planung und Simulation. Die Bildung angepasster kombinierter synthetischer Lastgänge

ermöglicht eine effiziente Simulation des Energie- und Leistungsbedarfs von Nutzergruppen elektrischer Energie. Die fehlende Normalverteilung des Energiebedarfs führt zur Verwendung von Spearman Korrelationen und damit zu abweichenden Ergebnissen gegenüber der allgemein üblichen Verwendung von Pearson Korrelationen [16, 17]. Es konnte jedoch gezeigt werden, dass bei (Standard) Lastprofile nicht von einer Normalverteilung gesprochen werden kann. Um das allgemein verwendete Verfahren nach Pearson methodenrichtig anwenden zu können, müssten die (normierte) Schiefe und Kurtosis innerhalb von Grenzen nahe Null liegen. [18] Die Ergebnisse zeigten aber eine deutliche Abweichung der Parameter von einer Normalverteilung. Zur Verdeutlichung der Bedeutung der Wahl der richtigen Methode zur Korrelationsbestimmung, werden die Ergebnisse nach Pearson und Spearman für die untersuchte Stichprobe in der folgenden Tabelle gezeigt.

	Spearman	Pearson
G0	0.43	0.44
G1	0.51	0.29
G2	0.31	0.33
G3	0.37	0.66

Tabelle 1: Gegenüberstellung der Korrelationskoeffizienten nach Spearman und nach Pearson.

Das Ergebnis bei einer Abweichung der Methodenwahl von der Üblichen ist als besonders kritisch zu werten, da es auf Grundlage einer anderen Berechnungsmethode zu abweichenden Ergebnissen der zugrunde zu legenden Standardlastprofile kommt, wie Tabelle 1 zeigt. Dennoch würde diese Unschärfe durch die Zuhilfenahme grafischer Verfahren (Maximum Likelihood) sowie der Aufteilung der Standardlastprofile nach Zeitabschnitten nicht zu unbrauchbaren Ergebnissen führen. Jedoch hat sich gezeigt, dass die Güte des Ergebnisses der Methode zur Erstellung der angepassten synthetischen Lastprofile maßgeblich von der richtigen Korrelationsanalyse abhängt. So ist im dargestellten Beispiel deutlich zu erkennen, dass bei der Wahl der Standardlastprofils G3 (höchste Korrelation entsprechend dem Verfahren nach Pearson – vergleiche Tabelle 1) ein Standardlastprofil gewählt werden würde, dessen Annahmen nicht der tatsächlichen Arbeitsweise der untersuchten Unternehmen entsprechen (G3 gilt für durchlaufende Gewerbe, die Anlieger arbeiten jedoch im Regelfall im

Einschicht-, höchstens aber im Zweischichtbetrieb). Damit bildet sich aus der Kombination der Standardlastprofile G0 bis G3 eine weitaus bessere Simulation der tatsächlichen Verhältnisse. Auch an dieser Stelle soll nochmals darauf hingewiesen werden, dass die Anwendung der falschen Korrelationsanalyse mit den richtigen Schlussfolgerungen (Wahl des SLP G3) schon zu einer erheblichen Verbesserung gegenüber der herkömmlichen Vorgehensweise führt. Entsprechend Abbildung 1 würde für das Gewerbegebiet Standardlastprofil G1 zur Prädiktion des Lastganges mit seiner Spitzenlast herangezogen werden. Jedoch zeigt Abbildung 2, dass die tatsächliche Spitzenlast gerade einmal rund 60 % der Höchstlast von G1 beträgt. In der aktuellen Netzplanung wird jedoch den Standardlastprofilen entsprechend geplant (sodass im Regelfall G1 als Planungsgrundlage zum Einsatz kommt). Es kann also allein durch die Anwendung einer Korrelationsanalyse und der Ableitung richtiger Schlussfolgerungen ein Beitrag zur Optimierung der Netzplanung geleistet werden. Das volle Potential der vorgestellten Methode wird erst durch die Anwendung der Korrelationsanalyse nach Spearman zur Erstellung angepasster synthetischer Lastprofile ausgeschöpft. Es ergibt sich erst mit der so bestimmten Kombination der Standardlastprofile höchster Korrelation (in Tabelle 1 sind nur diese dargestellt) die Grundlage für die verbesserte Simulation. Durch die Kombination dieser Standardlastprofile, kann nicht nur die Last, sondern auch der Lastgang sehr genau vorausbestimmt werden. Schlussendlich können durch die vorgestellte Methode Überkapazitäten des Energienetzes vermieden werden. Die installierte Netzkapazität kann im Untersuchten Beispiel bei der Planung um 30 % reduziert werden. Trotzdem bleibt eine installierte Übertragungsleistung in Höhe von 125 % der tatsächlichen Höchstlast bestehen, sodass mit keinen Kapazitätsengpässen zu rechnen ist. Ebenso gelingt es mit der vorgestellten Methode sehr gut den tatsächlichen Lastgang nachzubilden. Es kommt zu einer Verbesserung der mittleren Abweichung des prognostizierten vom tatsächlichen Energiekonsum von 50 %. Diese Größe ist vor allem für die Planung und Simulation von möglichen Flexibilitätsmaßnahmen und deren Steuerung von großer Bedeutung.

4 Anwendung

Zum Thema der Energieflexibilität auf Verbraucherseite im verarbeitenden Gewerbe existiert eine Vielzahl an

Publikationen. Dabei werden die Betrachtungen jedoch nur auf einzelne Verbraucher gerichtet. Wie Energieflexibilitätsoptionen im Kontext von Industrieverbünden zu modellieren und bewerten sind bis jetzt ein noch wenig beachtetes Teilgebiet. Weeber et al. werfen in ihrem Beitrag die Frage auf, inwiefern Dienstleister in kooperierenden Industrie- und Gewerbeparks von Energieflexibilitätsoptionen profitieren können [19]. Solche Fragestellungen lassen sich durch die Komplexität und dynamischen Einflüsse, die auf Produktionssysteme wirken, nicht mit statischen Verfahren abbilden. Dabei hat sich die Simulation als ein wichtiges Werkzeug herausgestellt [4].

Die Datenbeschaffung stellt bei jedem Simulationsprojekt eine unabdingbare Phase dar, wie beispielsweise im Vorgehensmodell von Rabe et al. beschrieben [20]. Sie beeinflusst die Qualität der Simulationsstudie sehr stark. Auf der anderen Seite bestehen bei den meisten Simulationsprojekten Zeit- und Termindruck, die eine Erhebung von Primärdaten verhindert oder zumindest einschränkt. Problematisch ist besonders die Gewinnung von aussagekräftigen Energiedaten.

Mithilfe des hier vorgestellten Vorgehens können schneller genauere Inputdaten für Simulationsmodelle generiert werden, womit sich die Zeitdauer und Güte für Simulationsprojekte im Bereich Energieeffizienz- und Energieflexibilitätsmaßnahmen in Gewerbe- und Industrieparks erheblich verkürzt bzw. die Güte und Aussagekraft verbessert. Das vorgestellte Verfahren kann in der Planung des Energiebedarfs industrieller und gewerblicher Verbünde Anwendung finden. Es müssen nicht mehr die einfachen Standardlastprofile zum Einsatz kommen, die aus aktueller Sicht weder hinsichtlich der Spitzenlast noch des Lastganges für die detaillierte Planung des Versorgungskonzeptes von Industrieunternehmen (und Gewerbe) allgemein zum Einsatz kommen sollten. Die Folgen wären in der Regel Überkapazitäten bei der Netzplanung und schlecht angepasste Fahrpläne für die Belieferung mit elektrischer Energie. Die vorgestellte Methode leistet damit einen entscheidenden Beitrag zur Simulation von Energieflüssen in Gewerbeparks und -gebieten.

Im weiteren Verlauf des Projekts GRIDS wird das hier vorgestellte Vorgehensmodell weiter Anwendung finden, um einen Teil der Inputdaten für eine energieorientierte Materialflusssimulation des Gewerbegebiets Süd in Limbach-Oberfrohna durchzuführen.

5 Diskussion

Das Vorgehensmodell ist in der Lage Energiebedarfe auf Grundlage des zu erwartenden jährlichen Gesamtenergiebedarf sowie der angenommenen Höchstlast zuverlässig zu generieren. Es ist eine enorme Verbesserung gegenüber der herkömmlichen Verwendung von Standardlastprofilen zu verzeichnen. Ein Nachteil der Methode ist die grafische Kontrollentscheidung für die Wahl des entsprechenden Standardlastprofils innerhalb der einzelnen Zeitabschnitte. Die Güte der Kontrollentscheidung hängt wieder von Erfahrung und subjektivem Empfinden ab. Dieses Verfahren wird im Zuge der Weiterentwicklung der Methodik hin zu einer automatischen Auswahl optimiert, wodurch eine objektive Auswahl gewährleistet wird. Dafür hat sich gezeigt, dass durch ebenjene zusätzlich grafische Kontrolle bzw. Auswahl der Zusammensetzung der angepassten synthetischen Lastprofile die Auswirkungen durch eine möglicherweise falsche Methodenwahl zur Bestimmung der Korrelationen begrenzt werden. Die Erkenntnis, die darauf hindeutet, dass es sich bei (Standard-)Lastprofilen nicht um normalverteilte Größen handelt, zeigt, dass Datenreihen immer hinsichtlich ihrer Verteilungsart untersucht werden sollten und gegebenenfalls (falls keine Verteilungsart ermittelt werden kann) auf nicht-parametrische Methoden zurückgegriffen werden muss. Zusammenfassend leistet die vorgestellte Methode einen Beitrag zur besseren Simulation von Energiebedarfen von Industrieunternehmen und industriellen sowie gewerblichen Verbänden. Die Methode selbst simuliert damit den Lastgang von Unternehmen

6 Verweise

- [1] M. Sauer, *Das Recht der Vergabe von Strom- und Gas-Konzessionsverträgen im EnWG: Legitimität und Anwendung eines Wettbewerbsinstruments im Kontext des Unions- und deutschen Verfassungsrechts*, 1st ed. Baden-Baden: Nomos Verlagsgesellschaft mbH & Co. KG, 2012.
- [2] T. Werth, *Netzberechnung mit Erzeugungsprofilen: Grundlagen, Berechnung, Anwendung*. Wiesbaden: Springer Vieweg, 2016. [Online]. Available: <https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=4504392>
- [3] M. Geilhausen, J. Bränzel, D. Engelmann, and O. Schulze, *Energiemanagement: Für Fachkräfte, Beauftragte und Manager*. Wiesbaden: Springer Vieweg, 2015. [Online]. Available: <http://gbv.eblib.com/patron/FullRecord.aspx?p=3567708>
- [4] S. Wenzel and T. Peter, *Simulation in Produktion und Logistik 2017*. Kassel: Kassel University Press GmbH, 2017. [Online]. Available: <https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=5091317>
- [5] G. Posselt, *Towards Energy Transparent Factories*, 1st ed. s.l.: Springer-Verlag, 2016. [Online]. Available: http://ebooks.ciando.com/book/index.cfm/bok_id/1960675
- [6] Vereinigung Deutscher Elektrizitätswerke - VDEW, "Umsetzung der Analytischen Lastprofilverfahren - Step-by-step," Frankfurt am Main, 2000.
- [7] A. Emde, F. Zimmermann, M. Feil, and A. Sauer, "Erstellung und Validierung von Lastprofilen für die energieintensive Industrie," *ZWF*, vol. 113, no. 9, pp. 545–549, 2018, doi: 10.3139/104.111977.
- [8] Bundesverband der Energie- und Wasserwirtschaft e.V. - BDEW, "Standardlastprofile Strom,"
- [9] U. Held, "Tücken von Korrelationen: die Korrelationskoeffizienten von Pearson und Spearman," (in deu), *Swiss Medical Forum*, vol. 10, no. 38, 652–653–653, 2010. [Online]. Available: <https://www.zora.uzh.ch/id/eprint/46199/>
- [10] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. Hoboken: Taylor and Francis, 2013. [Online]. Available: <http://gbv.eblib.com/patron/FullRecord.aspx?p=1192162>
- [11] P. Sedlmeier, *Jenseits des Signifikanztest-Rituals: Ergänzungen und Alternativen*.

- [Online]. Available: <https://www.dgps.de/fachgruppen/methoden/mpr-online/issue1/art3/article.html> (accessed: Sep. 14 2020).
- [12] P. Schober, C. Boer, and L. A. Schwarte, "Correlation Coefficients: Appropriate Use and Interpretation," *Anesthesia and analgesia*, vol. 126, no. 5, pp. 1763–1768, 2018, doi: 10.1213/ANE.0000000000002864.
- [13] A. Schäfer and T. Schöttker-Königer, *Statistik und quantitative Methoden für Gesundheitsfachberufe*, 1st ed. Berlin: Springer, 2015. [Online]. Available: <http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&AN=1078892>
- [14] R. Matthews, "Storks Deliver Babies ($p = 0.008$)," *Teaching Statistics*, vol. 22, no. 2, pp. 36–38, 2000, doi: 10.1111/1467-9639.00013.
- [15] BDEW Bundesverband der Energie- und Wasserwirtschaft e.V., *Standardlastprofile Strom*.
- [16] L. Myers and M. J. Sirois, "Spearman Correlation Coefficients, Differences between," in *Encyclopedia of statistical sciences*, S. Kotz, C. B. Read, N. Balakrishnan, B. Vidakovic, and N. L. Johnson, Eds., 2nd ed., [Hoboken, N.J.]: Wiley, 2010.
- [17] J. C. F. de Winter, S. D. Gosling, and J. Potter, "Comparing the Pearson and Spearman correlation coefficients across distributions and sample sizes: A tutorial using simulations and empirical data," *Psychological methods*, vol. 21, no. 3, pp. 273–290, 2016, doi: 10.1037/met0000079.
- [18] A. Field, *Discovering statistics using SPSS: (and sex and drugs and rock 'n' roll)*, 3rd ed. Los Angeles, Calif.: Sage, 2011. [Online]. Available: <http://www.uk.sagepub.com/field3e/main.htm>
- [19] M. Weeber, C. Lehmann, J. Böhner, and R. Steinhilper, "Augmenting Energy Flexibility in the Factory Environment,"

Procedia CIRP, vol. 61, pp. 434–439, 2017, doi: 10.1016/j.procir.2016.12.004.

- [20] M. Rabe, S. Spiekermann, and S. Wenzel, *Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodelle und Techniken*. Berlin, Heidelberg: Springer, 2008. [Online]. Available: <http://nbn-resolving.de/urn:nbn:de:1111-2008093035>

Angabe der Fördermittelgeber des Projekt GRIDS – Grüne Energie in industriellen Verbünden



Europäische Union

Europa fördert Sachsen.



Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des vom Sächsischen Landtag beschlossenen Haushaltes.

Anwendung von Batterieanalysemethoden zur Validierung von Alterungsmodellen in Lithium-Ionen Zellen

Steffen Bazlen^{1*}, Markus Blessing^{1*}, Walter Commerell^{1*}

¹Technische Hochschule Ulm, Prittwitzstraße 10, 89075 Ulm, *bazlen@mail.hs-ulm.de, *blessing@mail.hs-ulm.de, *commerell@thu.de

Abstract. Die Leistungsfähigkeit einer Lithium-Ionen Zelle weist neben den elektrochemischen Eigenschaften der initialen Zell- und Materialauslegung eine hohe Abhängigkeit vom Alterungszustand auf. Für die Simulation von Zellen und Batteriesystemen sind Kenntnisse über das Alterungsverhalten notwendig und in der Modellbildung zu berücksichtigen. Die unvermeidbare Zellalterung wird dabei durch eine Vielzahl parallel auftretender Degradationseffekte hervorgerufen. Eine detaillierte Alterungsanalyse ist mit einem hohen Aufwand verbunden und oftmals nur durch eine Post-mortem-Analyse ermittelbar. In dem vorliegenden Artikel wird eine vereinfachte Methode vorgestellt, die es erlaubt, auf die dominanten Degradationseffekte zurückzuschließen. Betrachtet wird der Leitfähigkeitsverlust (conductivity loss, CL), der Rückgang von Lithiuminventar (loss of lithium inventory, LLI) und der Verlust von Aktivmaterial (loss of active material, LAM).

Um die Entwicklungsverläufe der einzelnen Alterungsmechanismen bestimmen zu können, wurden im Rahmen der Untersuchungen zwei voneinander unabhängige Messmethoden angewendet und optimiert. Als Messmethoden wurden die Elektrochemische Impedanzspektroskopie (electrochemical impedance spectroscopy, EIS) und die Messung der Leerlaufspannung (open circuit voltage, OCV) definiert. Aus der einfachen Leerlaufspannungsmessung lässt sich die Inkrementellen Kapazität (incremental capacity, IC) und die Differentielle Spannung (differential voltage, DV) bilden. Durch einen Vergleich der Messergebnisse beider Methoden wurde die Plausibilität und somit eine zulässige Anwendung der Messmethoden für eine detailliertere Alterungsanalyse von Lithium-Ionen Zellen nachgewiesen.

Die Erkenntnisse können nun zur Parametrierung von Simulationsmodellen herangezogen werden.

Einleitung

Der Einsatz von Lithium-Ionen (Li-Ionen) Batterien ist in der heutigen Zeit sehr vielseitig. Sie finden Anwendung in Elektroautos, E-Bikes, Mobiltelefonen, Hausspeichertechnik etc. Grund dafür ist ihre hohe Energiedichte und die große Zyklenfestigkeit. An der Anode werden meist Graphite und kathodenseitig Nickel-Mangan-Cobalt (NMC)-, Nickel-Cobalt-Aluminium (NCA) und Eisenphosphatverbindungen (LFP) als Elektrodenmaterialien eingesetzt. Alle basieren auf dem Interkalationsprinzip von Li-Ionen in ein Kristallgitter [1–4]. Je nach Anwendung eignen sich verschiedene Kombinationen aus Zellmaterialien, um die benötigte Eigenschaft der Zelle (Energiedichte, Strombelastung, Zyklenfestigkeit etc.) möglichst sicher und effizient bereitzustellen. Die derzeit kommerziellen Zellmaterialien haben einen langen und intensiven Entwicklungsprozess durchlaufen. Die Materialeigenschaften sind weitestgehend erforscht [5]. Dennoch erfahren die Zellen über deren Benutzung eine unvermeidbare Alterung, welche zu Kapazitäts- und Leistungsverlust der Batterie führt [6, 7]. Die Intensität der Degradationsentwicklung ist dabei stark abhängig von den verwendeten Zellmaterialien sowie Nutzungsprofilen und weiteren Einflussgrößen wie Umgebungsbedingungen [8]. Um in einem realen Batteriesystem oder einem Simulationsmodell die Alterung geeignet abzubilden, ist eine möglichst genaue Identifikation der Parameter erforderlich. Folglich ist eine anwendungsangepasste Bestimmung der idealen Zellparameter für eine möglichst geringe Zellalterung sehr aufwendig. In heutigen Batteriemanagementsystemen (BMS) wird der Gesundheitszustand (state of health, SOH) ausschließlich über die Gesamtalterung durch den Verlust an nutzbarer Kapazität oder den Anstieg des Zellwiderstandes bestimmt [9–11]. Um die Degradation einer Batteriezelle genauer zu analysieren, existieren verschiedene Alterungsmodelle, um die Schädigungseffekte auf die einzelnen Komponenten der Batterie zurückzuführen [12–20].

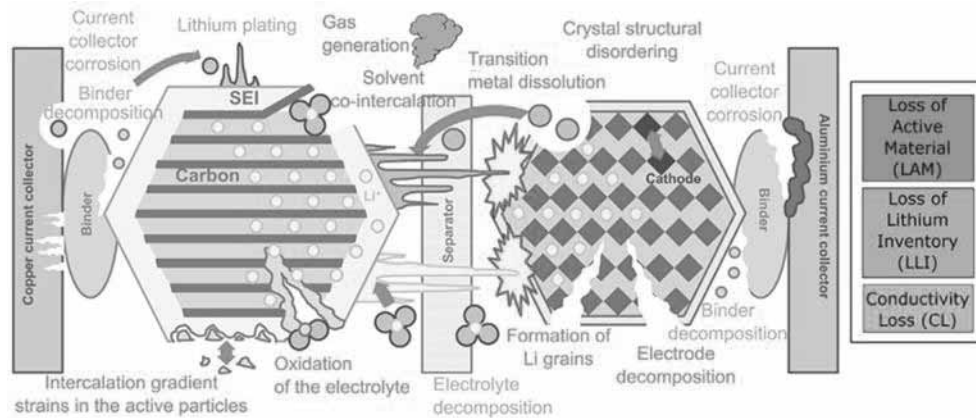


Abbildung 1: Auftretende Schädigungseffekte innerhalb einer Batteriezelle mit farblicher Zuordnung in die drei Alterungsmechanismen Verlust an Aktivmaterial (LAM), Verlust an Lithium Inventar (LLI) und Leitfähigkeitsverlust (CL) [8].

Durch die Anwendung einfacher und etablierter Messmethoden sowie eines universellen Auswertalgorithmus können die relative Alterungsentwicklungen für verschiedene Zellchemien und Belastungsprofile in einzelne Degradationskennzahlen aufgeschlüsselt werden.

1 Alterungsmechanismen in Lithium-Ionen-Zellen

In einer Batteriezelle trägt eine Vielzahl unterschiedlicher Degradationseffekte zur Gesamtalterung der Zelle bei. Für eine genauere Untersuchung der Ursachen, werden sämtliche Schädigungseffekte häufig in drei getrennte Alterungsmechanismen unterteilt. Diese drei Untergruppen werden aus den Verlusten der Leitfähigkeit (CL), des Lithiuminventars (LLI) und des Aktivmaterials der Elektroden (LAM) gebildet [12–17, 20].

Typische Schädigungseffekte, welche innerhalb einer Batteriezelle auftreten, werden in Abbildung 1 dargestellt und farblich zu den drei übergeordneten Alterungsmechanismen zugeordnet.

Ein Rückgang der elektrischen Leitfähigkeit ist direkt proportional mit dem Widerstandsanstieg der Zelle verknüpft. Die Korrosion der Stromableiter und der Abbau der Bindemittel im Elektrolyten gelten als typische Ursachen für den Anstieg des Zellwiderstands [7, 8].

Das Lithiuminventar wird durch die Anzahl der Li-Ionen beschrieben, welche für die Ein- und Auslagerungsprozesse der Elektroden verfügbar sind. Der irreversible Verlust von Lithium führt daher zwangsläufig zu einer Reduktion der nutzbaren Zellkapazität [8, 12]. LLI wird vor allem durch das stetige Anwachsen der SEI-Schicht, Lithium-Plating und Zersetzungsprozesse hervorgerufen [2, 14, 21].

Ein Verlust von Aktivmaterial bedeutet neben dem Kapazitätsrückgang zusätzlich eine verminderte Leistungsfähigkeit der Zelle [2, 7]. Im Bereich der definierten Spannungsgrenzen sind die Aktivmaterialien der Elektroden

enormen mechanischen Belastungen ausgesetzt. Infolgedessen können Teile des Aktivmaterials durch eine Beschädigung der Gitterstrukturen nicht mehr für die Interkalationsprozesse genutzt werden [4, 8, 22]. Weitere Ursachen für LAM sind das Brechen elektrisch leitender Partikel, isolierte Teile des Aktivmaterials aufgrund einer Passivierungsschicht oder die chemische Zersetzung durch den reaktiven Elektrolyten [7, 8, 21].

2 Analysemethoden zur Alterungsbestimmung in Batteriezellen

Zur Charaktisierung von Batteriezellen, besonders der Alterungsanalyse, wird eine Vielzahl an Messmethoden angewendet.

In dieser Arbeit werden zwei Methoden, welche eine zerstörungsfreie, nicht-invasive Messung während des Zellbetriebs ermöglichen, betrachtet und auf deren reliable Aussagekraft untersucht. Bei den angewendeten Messmethoden handelt es sich um die Elektrochemische Impedanzspektroskopie (EIS) und die Aufnahme der Leerlaufspannung (OCV). Anhand des OCV-Verlaufs über der Kapazität können die Kurvenverläufe der Inkrementellen Kapazität (IC) und der Differentiellen Spannung (DV) bestimmt werden.

2.1 Elektrochemische Impedanzspektroskopie

Bei der Elektrochemischen Impedanzspektroskopie handelt es sich um ein nicht-invasives Messverfahren, bei welchem ein sinusförmiges Eingangssignal auf die Zelle eingeprägt wird. Durch die Messung des Ausgangssignals kann das Übertragungsverhalten des Systems ermittelt werden. Wird als Eingangssignal der Strom und als Ausgangssignal die Spannung verwendet, so ergibt das Übertragungsspektrum die elektrische Impedanz des

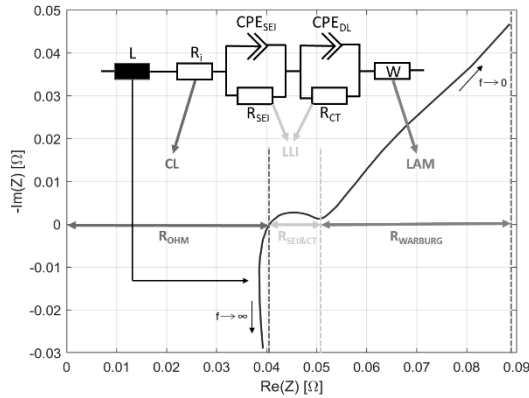


Abbildung 2: Informationsgehalt eines Impedanzspektrums mit Einteilung in die einzelnen Widerstandsbe-reiche und die Darstellung im Ersatzschaltbild.

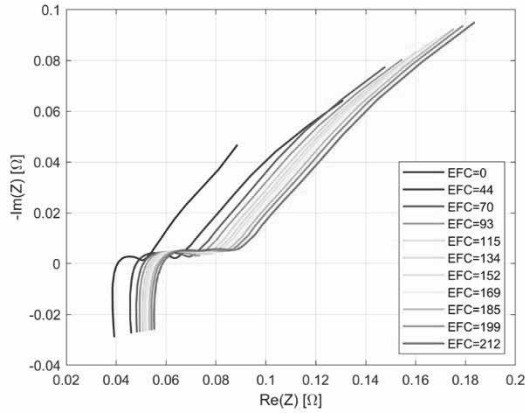


Abbildung 3: Impedanzspektren einer NCA-Graphit-Zelle in Abhängigkeit der durchlaufenen äquivalenten Voll-zyklen (equivalent full cycle, EFC).

Systems [23, 24]. Um ein Impedanzspektrum aufzu-zeichnen, werden einzelne Impedanzen bei unterschied-lichen Frequenzen gemessen und zu einer Ortskurve ver-bunden. Die graphische Darstellung eines Impedanz-spektrums erfolgt in der Regel im Nyquist-Diagramm [2].

Wie in Abbildung 2 dargestellt setzt sich das Spektrum aus einzelnen Bereichen, welche durch ein Ersatzschaltbild modelliert und simuliert werden können, zusammen. Während eine Batteriezelle bei sehr hohen Frequenzen ein induktives Verhalten besitzt, ist die Zellimpedanz hauptsächlich kapazitiv geprägt. Der Schnittpunkt der Ortskurve mit der Realteil-Achse beschreibt den Ohm'schen Widerstand.

Das Verhalten der SEI-Schicht und des Ladungstrfers werden durch zwei charakteristische Halbkreise darge-stellt, welche im Ersatzschaltbild zwei RC-Glieder entsprechen. Für eine präzise Modellierung werden im Ersatzschaltbild anstatt einfacher Kondensatoren häufig Konstant-Phaselemente (constant phase element, CPE) eingesetzt [25]. Das Verhalten durch Diffusionseffekte beschreibt den Ionentransport innerhalb der Elektroden

aufgrund des Konzentrationsgefälles und wird durch ei-nen Diffusionsast im Spektrum repräsentiert. Im Ersatz-schaltbild wird dies als komplexe Warburgimpedanz (Z_W) dargestellt [24, 26]. Da für diese Arbeit nur die Ent-wicklungen der Widerstände aus dem Impedanzspektrum nötig ist, können die Widerstände an der Realteil-Achse abgelesen werden. Die Einteilung der einzelnen Wider-stände ist in Abbildung 2 farblich dargestellt. Die Wider-stände der SEI und des Ladungstrfers können zusam-mengefasst werden [25]. Der Vorteil dieser Methode ist, dass die markanten Punkte aus dem Spektrum durch ei-nen automatisierten Auswertalgorithmus über den kom-pletten Alterungsverlauf zuverlässig erkannt werden [27].

2.2 Identifikation und Quantifizierung der Alterungsmechanismen durch die Elektrochemische Impedanzspektroskopie

Eine Alterung der Batteriezelle ist mit einer Verschie-bung des Impedanzspektrums nach rechts auf der Real-teil-Achse verbunden. In Abbildung 3 sind mehrere EIS-Spektren einer NCA-Graphit-Zelle in Abhängigkeit von äquivalenten Vollzyklen (equivalent full cycles, EFC) dargestellt. Durch die Zunahme des Ohm'schen Wider-stands kann CL berechnet werden. Des Weiteren ist eine Vergrößerung des zusammengesetzten Widerstands aus SEI und Ladungstransfer auf LLI zurückzuführen. Der Warburg'sche Widerstand wird aus der Differenz des ma-ximalen Realteil-Wertes und des Tiefpunktes des Spekt-rums gebildet. Durch die Zunahme des Warburg'schen Widerstands kann auf LAM geschlossen werden [8, 15]. Die relativen Alterungsentwicklungen, bezogen auf den jeweiligen Startwert der ersten EIS-Messung, berechnen sich nach Gleichungen (1) - (3).

$$CL_{EIS,n}[\%] = \frac{R_{Ohm,n} - R_{Ohm,1}}{R_{Ohm,1}} \cdot 100 \quad (1)$$

$$LLI_{EIS,n}[\%] = \frac{R_{SEI+CT,n} - R_{SEI+CT,1}}{R_{SEI+CT,1}} \cdot 100 \quad (2)$$

$$LAM_{EIS,n}[\%] = \frac{R_{Warburg,n} - R_{Warburg,1}}{R_{Warburg,1}} \cdot 100 \quad (3)$$

für $n \in \mathbb{N}$

2.3 Inkrementelle Kapazität/Differentielle Spannung

Die Auswertemethode Inkrementelle Kapazität/Differen-tielle Spannung (IC/DV) basiert auf der Aufnahme der Leerlaufspannung. Dafür wird die Ruhespannung zwis-chen den Spannungsgrenzen mit kleinen Strömen, bei denen die Zelle in mehr als 25h entladen wird ($< C/25$), aufgenommen. Diese kleinen Ströme sind notwendig, um die markanten Interkalationsstufen und Phasenplateaus

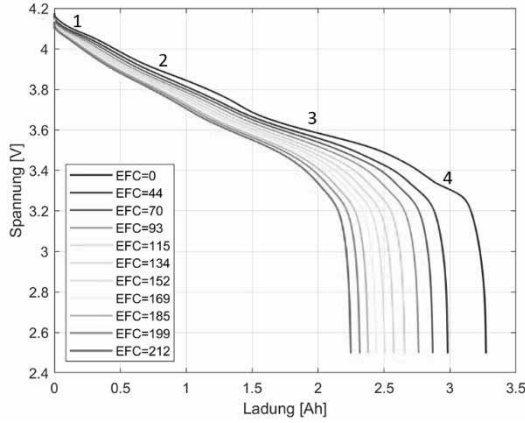


Abbildung 4: Verlauf der Leerlaufspannungen einer NCA-Graphit-Zelle in Abhängigkeit der äquivalenten Vollzyklen (EFC) mit Markierung der markanten Phasenplateaus bei Null EFC.

zu erfassen, da die Einlagerung von Li-Ionen in die Graphitanode stufenartig erfolgt [2, 8]. Aufgrund der konstanten Lade- bzw. Entladeströme tritt eine konstante Ladungsänderung über der Zeit auf.

Um den Verlauf der Inkrementellen Kapazität (IC) zu erhalten, wird die Ladung Q über der Spannung U nach Gleichung (4) abgeleitet. Der Kurvenverlauf der IC-Kurve entspricht der Kapazitätsänderung über der Zellspannung. Demnach resultieren aus den Phasenplateaus der OCV-Kurve markante Peaks im Kurvenverlauf des IC-Graphen [28]. Ein Rückgang der Peaks ist demnach mit einer höheren Potentialänderung während der transferierten Ladung verbunden.

$$IC = \frac{dQ}{dU} \approx \frac{\Delta Q}{\Delta U} \quad (4)$$

Die DV-Kurve wird entsprechend Gleichung (5) durch die Ableitung der Spannung nach der Ladung generiert. In der Ableitung dU/dQ stellen die Peaks Phasenübergänge in der Anode dar. Das bedeutet, um Li-Ionen in eine neue Schicht einzulagern, ist eine gewisse Spannungsänderung nötig [28].

$$DV = \frac{dU}{dQ} \approx \frac{\Delta U}{\Delta Q} \quad (5)$$

In Abbildung 4 sind OCV-Messungen einer NCA-Graphit-Zelle in Abhängigkeit von EFC dargestellt. Dabei ist bei zunehmender Zellalterung ein deutlicher Rückgang der nutzbaren Zellkapazität erkennbar. In der Grafik sind die vier markanten Phasenplateaus numeriert.

2.4 Identifikation und Quantifizierung der Alterungsmechanismen durch die Inkrementelle Kapazität/Differentielle Spannung

Für die Quantifizierung der drei Alterungsmechanismen

werden, aufgrund des unterschiedlichen Informationsgehaltes, beide Ableitungskurven herangezogen. Während der Kurvenverlauf der Differentiellen Spannung (DV) Rückschlüsse auf den Verlust an Lithium Inventar (LLI) gibt, können anhand der Inkrementellen Kapazität der Verlust von Aktivmaterial (LAM) und der Verlust von Letifähigkeit (CL) ermittelt werden. Die relevanten Veränderungen der Kurven, welche mit den Alterungsmechanismen verknüpft werden, sind in Abbildung 5 illustriert.

Laut [12] ist CL direkt proportional zum Rückgang der maximalen Leerlaufspannungen. Dies beruht auf der Annahme, dass diese auftretende Spannungsdifferenz durch den Anstieg des Innenwiderstands der Zelle resultiert. Allerdings führt eine zunehmende Zellalterung neben dem Anstieg des Zellwiderstandes auch zu einem veränderten Relaxationsverhalten [29]. Deshalb muss anstatt dem Rückgang der maximalen Leerlaufspannungen die Betrachtung des CL lediglich über die Änderung des Innenwiderstands der Zelle erfolgen. Der Innenwiderstand der Zelle (R_{Zelle}) kann anhand des senkrechten Spannungsabfalls nach Beendigung eines Ladevorgangs und dem Betrag des Ladestroms nach Gleichung (6) berechnet werden [10, 11]. Dabei entspricht U_1 dem letzten Spannungswert des Ladevorgangs, U_2 dem unteren Spannungswert des senkrechten Spannungsabfalls. Die relative Entwicklung von CL bezieht sich auf den Startwert des ersten Kurvenverlaufs und berechnet sich nach Gleichung (7). LLI ist direkt proportional zur maximal verfügbaren Kapazität und berechnet sich nach Gleichung (8). LAM hingegen wird über den Rückgang des höchsten IC-Peaks nach Gleichung (9) berechnet [12].

$$R_{Zelle,n} [\Omega] = \frac{U_{1,n} - U_{2,n}}{I_{Ladung}} \quad (6)$$

$$CL_{ICDV,n} [\%] = \frac{R_{Zelle,n} - R_{Zelle,1}}{R_{Zelle,1}} \cdot 100 \quad (7)$$

$$LLI_{ICDV,n} [\%] = \frac{\max(Q)_1 - \max(Q)_n}{\max(Q)_1} \cdot 100 \quad (8)$$

$$LAM_{ICDV,n} [\%] = \frac{\max(\frac{dQ}{dU})_1 - \max(\frac{dQ}{dU})_n}{\max(\frac{dQ}{dU})_1} \cdot 100 \quad (9)$$

für $n \in \mathbb{N}$

3 Messablauf und Auswertung

Für aussagekräftige Messergebnisse ist eine Reproduzierbarkeit der Messung zwingend notwendig. Deshalb werden identische, neuartige Rundzellen des kommerziellen Zelltyps Panasonic NCR18650B mit einer Kapazität von 3,2 Ah unter äquivalenten Prüfbedingungen vermessen. Da die Umgebungstemperatur einen beachtlichen Einfluss auf das Zellverhalten nimmt, befinden sich

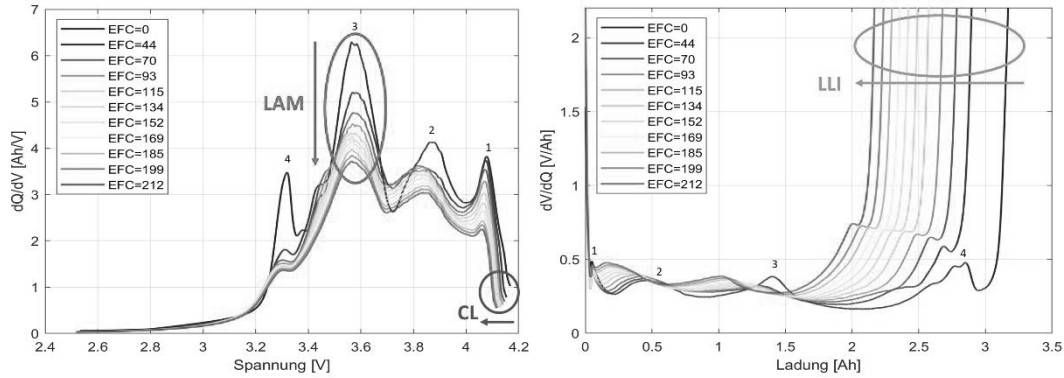


Abbildung 5: Verlauf der Inkrementellen Kapazität (IC-Zelle 1) in Abhängigkeit der Äquivalenten Vollzyklen (EFC) mit Markierung der relevanten Bereiche zur Bestimmung der Alterungsmechanismen Verlust von Aktivmaterial (LAM) und Leitfähigkeitsverlust (CL). Verlauf der Differentiellen Spannung (DV-Zelle 1) in Abhängigkeit der äquivalenten Vollzyklen (EFC) mit Markierung der relevanten Bereiche zur Bestimmung des Alterungsmechanismus Verlust von Lithiuminventar (LLI).

die zu vermessenden Zellen durchgehend in einer Klimakammer mit einer konstanten Temperatur von 25 °C [30]. Da in dieser Arbeit die auf zwei unterschiedlichen Methoden basierenden Alterungsmechanismen verglichen werden sollen, wird ein Messplan mit zusätzlich nachgebildeter Zellalterung im Prüfplan benötigt. Der Prüfplan ist durch die Parametrisierung der Eingabewerte und Abbruchkriterien universell für verschiedene Zellchemien und Belastungsprofile anwendbar. Der Prüfplan für die Messung beider Batteriezellen wird mit dem Batterietestprogramm BaSyTec erstellt. Die Programmierung der elektrochemischen Impedanzspektroskopie erfolgt über ein externes Gamri Reference 3000. Durch ein automatisiertes Durchlaufen des gesamten Messplans können einheitliche Messparameter wie identische Relaxationszeiten gewährt werden.

Dadurch erfahren beide Zellprüflinge von Beginn bis Beendigung der Messung exakt kongruente Bedingungen und können daher zuverlässig und reliabel verglichen werden.

Die Impedanz einer Batteriezelle ist stark abhängig vom Ladezustand während der Messung [8]. Für einen zulässigen Vergleich mehrerer Impedanzspektren ist daher jeweils derselbe SOC nötig. In dieser Arbeit wurde ein Ladezustand von 50 % SOC gewählt. Zusätzlich wird nach Beendigung der Lade-Entlade-Phase auf 50 % SOC der senkrechte Spannungsabfall ermittelt, um den Innenwiderstand der Zelle zu bestimmen. Anschließend wird das Impedanzspektrum für einen Frequenzbereich zwischen 10 kHz und 1 mHz aufgezeichnet. Da die Messmethode der Leerlaufspannung bei möglichst identischen Alterungszuständen der Zelle erfolgen soll, wird im direkten Anschluss zur EIS-Messung die OCV-Aufnahme mit C/25 für einen Lade-/Entladevorgang des gesamten Spannungsbereichs zwischen 2,5 V und 4,2 V durchgeführt, was einer Entladetiefe (depth of discharge, DOD) von 100 % entspricht. Für eine simulierte Zellalterung erfährt die Zelle 100 Lade-/Entladezyklen mit einer konstanten

Stromstärke von 1 C, was der Stromstärke entspricht, um eine Batteriezelle in einer Stunde vollständig zu laden oder zu entladen. Nach Beendigung der 100 Alterungszyklen startet der Messzyklus erneut.

Je nach Auswahl der Prüfparameter und Ladeverfahren wird nur eine bestimmte Ladungsmenge in die Zelle geladen bzw. entnommen. Auch unter realen Bedingungen wird der Zelle nicht immer die volle Ladung entnommen oder zugeführt. Um einen Vergleich über die tatsächliche Anzahl n der Lade-/Entladezyklen zwischen unterschiedlichen Messparametern zu ermöglichen, ist die Bildung von äquivalenten Vollzyklen (EFC) nötig. Die EFC sind nach Gleichung (10) definiert.

$$EFC = \frac{\sum_{i=1}^n Q_{Entladung,n}}{Q_{Nenn}} \quad (10)$$

für $n \in \mathbb{N}$

Die EIS- und IC/DV-Methode liefert, wie bereits in der Veröffentlichung [27] bewiesen, unterschiedliche Werte für die Alterungskennzahlen. Daher muss ein Vergleich über den Korrelationskoeffizient nach Pearson erfolgen. Der Korrelationskoeffizient berechnet die lineare Abhängigkeit von zwei Messgrößen und variiert betragsmäßig zwischen 0 und ± 1 [31].

Durch den Korrelationskoeffizienten kann allerdings noch nicht bestimmt werden, ob die beiden Messreihen dasselbe aussagen. In der Literatur wird hierfür häufig der 2-Stichproben-t-Test angewendet. Dieser ist allerdings nicht zulässig für einen Vergleich zweier Messmethoden mit unterschiedlicher Skalierung [32]. Deshalb wird hierfür ein Verfahren angewendet, welches den Faktor zwischen jedem Messwert der beiden Skalen berechnet. Der Faktor ist nach Gleichung (11) definiert. Dabei ist x der jeweilige Messwert der einzelnen Messmethode.

$$Faktor_{EIS/ICDV,n} = \frac{x_{EIS,n}}{x_{ICDV,n}} \quad (11)$$

für $n \in \mathbb{N}$

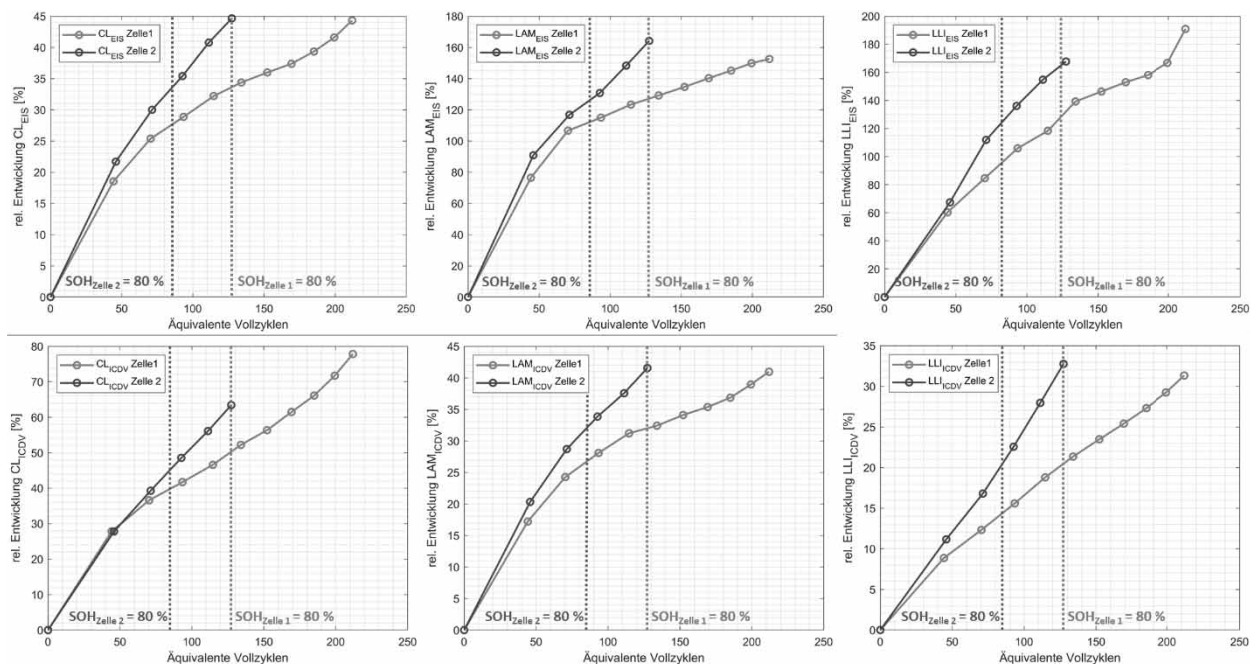


Abbildung 6: Vergleich der relativen Entwicklung der einzelnen Alterungsmechanismen Verlust von Aktivmaterial (LAM), Verlust an Lithium Inventar (LLI) und Leitfähigkeitsverlust (CL) von Zelle 1 und 2 ermittelt durch beide Auswertemethoden Elektrochemische Impedanzspektroskopie (EIS) und Inkrementelle Kapazität/Differentielle Spannung (ICDV).

Anhand mehrerer Messwerte und des daraus gebildeten Faktors werden der Mittelwert und die Standardabweichung des berechneten Faktors berechnet. Dies erfolgt für jeden der drei Alterungsmechanismen (CL, LLI, LAM).

Der Auswertalgorithmus wird in MATLAB programmiert. Für die Auswertemethode IC/DV werden jeweils die Entladekurven der OCV-Messungen verwendet, um die Ableitungen zu bestimmen. Da durch die Bildung der Ableitung ein großes Rauschen entsteht, werden die Kurven durch ein Savitzky-Golay-Filter gefiltert. Vorteil des Savitzky-Golay Filters ist, dass Anteile von hohen Frequenzen nicht abgeschnitten werden [33–35]. Die IC/DV-Verläufe und EIS-Spektren werden anschließend automatisiert analysiert und die Alterungskennzahlen anhand der Gleichungen (1) – (3) und (7) – (9) gebildet. Abschließend erfolgt der Vergleich der beiden Messmethoden durch Gleichung (11).

4 Ergebnisse

In diesem Kapitel werden zunächst die Messergebnisse der IC/DV- und der EIS-Methode separat vorgestellt und anschließend deren Aussagekraft, hinsichtlich der Entwicklung der einzelnen Alterungsmechanismen, verglichen. Die für die Auswertung verwendete Zellen werden nachfolgend als Zelle 1 und Zelle 2 bezeichnet. Für die Bewertung des SOH der Zellen wird der Rückgang der nutzbaren Zellkapazität betrachtet. Als Kriterium für das Lebensdauerende (end of life, EOL) wird der

Wert der nutzbaren Restkapazität von 80 % der Nennkapazität gewählt. Zelle 2 erreicht das EOL-Kriterium bereits nach 250 durchlaufenen Lade-/Entladezyklen, welche 85 EFC entsprechen. Zelle 1 hingegen erreicht das EOL erst nach 450 Zyklen bzw. 125 EFC. Für die weitere Untersuchung der Alterungsentwicklung wurden die Messungen bis zu einem SOH von ca. 70 % fortgeführt. Trotz identischer Zelltypen und äquivalenter Prüfparameter weist Zelle 2 eine deutlich stärkere Alterung auf, deren Ursachen im Folgenden untersucht werden.

4.1 Messergebnisse der Elektrochemischen Impedanzspektroskopie-Methode

In Abbildung 6 wird die relative Entwicklung der einzelnen Alterungsmechanismen in Bezug auf den Startwert der ersten EIS-Messung veranschaulicht. Bei der Betrachtung der einzelnen Entwicklungen fällt auf, dass die jeweiligen Alterungsmechanismen beider Zellen bis etwa 50 EFC ähnliche Verläufe annehmen und darauffolgend bei Zelle 2 schneller ansteigen. Der abweichende Verlauf von Zelle 2 ist auf die stärkere Alterung und die damit geringere Anzahl der erreichten EFC bis zum Lebensdauerende zurückzuführen. Auffällig ist, dass beim Erreichen des jeweiligen EOL die relativen Alterungsentwicklungen der Zellen ähnliche Beträge aufweisen, obwohl Zelle 2 weniger EFC durchlaufen hat. Die geringeren EFC rühren von einem erhöhten Innenwiderstand der Zelle, wodurch die Spannungsgrenzen durch eine Ladung mit konstantem Strom deutlich schneller erreicht

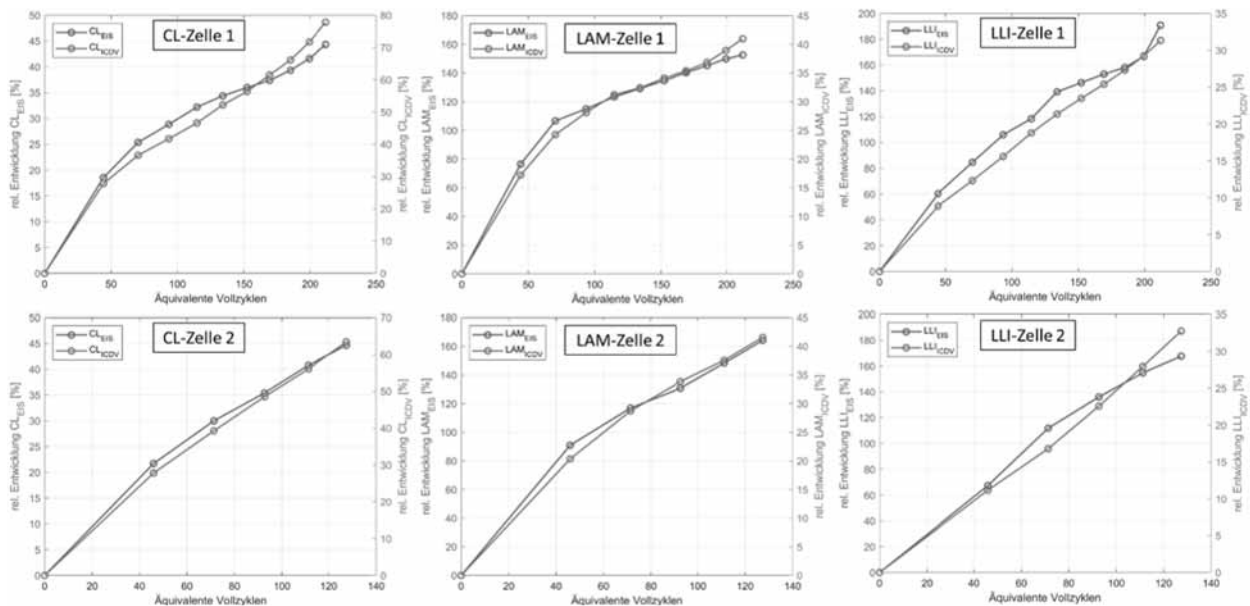


Abbildung 7: Vergleich der relativen Entwicklungstrends der einzelnen Alterungsmechanismen beider Zellen zur Ermittlung einer übereinstimmenden Aussagekraft beider Auswertemethoden Inkrementelle Kapazität/Differentielle Spannung (ICDV) und Elektrochemische Impedanzspektroskopie (EIS).

werden und weniger Ladung transferiert wird. Während beide Zellen zu Beginn der Messung eine hohe relative-Entwicklung des LAM aufweisen, nimmt bei Zelle 1 LAM im weiteren Verlauf ab und verhält sich annähernd linear zu den EFC. Nach dem Erreichen von 80 % SOH der Zelle 1 ist ein deutlicher Anstieg von LLI durch eine höhere Steigung zu erkennen. Dies lässt sich durch die starke Ausbildung des zweiten Halbkreises im Impedanzspektrum von Zelle 1 erklären. Die erste EIS-Messung liefert nahezu identische Impedanzspektren und somit übereinstimmende Startwerte, wodurch ein Vergleich der Alterung über die relative Entwicklung zulässig ist.

4.2 Messergebnisse der Inkrementellen Kapazität/Differentiellen Spannung

Die durch die IC/DV-Methode ermittelten, relativen Entwicklungen der Alterungsmechanismen sind ebenfalls in Abbildung 6 illustriert. Ähnlich wie bei den Entwicklungsverläufen der EIS-Methode korrelieren die Verläufe beider Zellen bis etwa 50 EFC. Bei steigender Zyklenzahl ist bei Zelle 2 eine jeweils stärkere Alterung zu verzeichnen. Bei steigender Anzahl der äquivalenten Voltzyklen (EFC) ist bei Zelle 2 jeweils eine höhere Steigung der Alterungsmechanismen erkennbar, wodurch das EOL schneller erreicht wird. Trotzdem weisen beide Zellen bei dem Erreichen des jeweiligen EOL nahezu äquivalente Beträge der jeweiligen Alterungsentwicklungen auf. Die Bestimmung des Innenwiderstands durch die Pulsstrommethode beinhaltet den Widerstand der SEI-Schicht. Die Effekte des Anwachsens der SEI werden somit zusätzlich im CL berücksichtigt. Nach dem Erreichen des EOL wird

bei Zelle 1 deutlich, dass der CL zunimmt. Bei Betrachtung des LAM wird eine Proportionalität der Abnahme des Peaks der IC-Verläufe, wie in Abbildung 5 ersichtlich, zu dem SOH beobachtet, was bereits in der Veröffentlichung [13] bewiesen wurde. LLI wird in dieser Methode über den Rückgang der nutzbaren Zellkapazität bestimmt und verhält sich bei beiden Zellen annähernd linear. Auch bei dieser Methode nehmen die Startwerte ähnlich Werte an. Damit ist ein Vergleich der Alterungsentwicklung beider Zellen zulässig.

4.3 Vergleich der Methoden

Aufgrund der unterschiedlichen Auswerteverfahren liefern die IC/DV- und EIS-Methode abweichende Startwerte für die einzelnen Alterungsmechanismen. Somit ergibt sich eine abweichende Skalierung der relativen Entwicklungen bezogen auf die unterschiedlichen Startwerte. Ein anschaulicher Vergleich der Entwicklungstrends beider Methoden wird durch eine Faktorbildung nach Gleichung (11) ermöglicht. Abbildung 7 stellt für jeden Alterungsmechanismus die Entwicklungsverläufe beider Messmethoden und einzelnen Zellen dar. Zur Bewertung der Übereinstimmung des Entwicklungsverlaufs beider Messverfahren wird die Standardabweichung der jeweiligen Faktoren aus Gleichung (11) gebildet. Somit kann durch kleine Standardabweichungen eine nötige Übereinstimmung beider Methoden bewiesen werden. In Tabelle 1 sind die Ergebnisse für die Faktormittelwerte und deren Standardabweichung dargestellt. Der optische Vergleich der Entwicklungstrends beider Messmethoden zeigt für beide Zellen eine hohe Ähnlichkeit für alle drei Alterungsmechanismen.

Alterungs- mechanismus		Faktor- mittelwert [-]	rel. Standard- abweichung [%]
CL	Zelle 1	0,64	7,52
	Zelle 2	0,74	4,15
LLI	Zelle 1	6,31	6,71
	Zelle 2	5,88	9,90
LAM	Zelle 1	4,03	5,62
	Zelle 2	4,06	5,92

Tabelle 1: Ergebnisse des Vergleichs über den Faktormittelwert und der Standardabweichung für die einzelnen Alterungsmechanismen CL, LLI und LAM beider Zellen.

Die relativen Standardabweichungen von 7,52 % bei Zelle 1 und 4,15 % bei Zelle 2 verdeutlichen eine hohe Übereinstimmung der ermittelten Trendentwicklungen des CL durch beide Messmethoden. Die Ergebnisse für den Korrelationskoeffizienten ergeben für Zelle 1 einen Wert von 0,987 und für Zelle 2 einen Wert von 0,998. Anhand der Werte für den Korrelationskoeffizienten nahe eins wird eine lineare Abhängigkeit beider Messmethoden bewiesen.

Für LAM ergeben sich relative Standardabweichungen von 5,62 % bei Zelle 1 bzw. 5,92 % bei Zelle 2. Somit ist die Aussagekraft beider Methoden für den Entwicklungsverlauf von LAM annähernd kongruent. Ebenfalls die Werte für den Korrelationskoeffizienten von 0,994 (Zelle 1) und 0,997 (Zelle 2) belegen diese lineare Abhängigkeit.

Die annähernd linearen Verläufe der relativen Entwicklungen von LLI lassen auf eine gute Korrelation der beiden Messmethoden schließen. Dies lässt sich durch die Korrelationskoeffizienten von 0,993 (Zelle 1) und 0,986 (Zelle 2) begründen. Durch die relativen Standardabweichungen von 6,71 % (Zelle 1) und 9,90 % (Zelle 2) lässt sich eine hohe Übereinstimmung der Entwicklungstrends des LLI erkennen.

Anhand des Vergleichs der beiden Messmethoden kann nachgewiesen werden, dass beide Methoden ähnliche Entwicklungstrends aufweisen. Durch die hohe Übereinstimmung der Entwicklungsverläufe ist sowohl die EIS- als auch die IC/DV-Methode für die Ermittlung der einzelnen Alterungsmechanismen zulässig.

5 Diskussion

Aufgrund des entstehenden Mischverhaltens bei der Messung von Batteriepacks ist die Bestimmung der definierten Alterungsmechanismen nur für einzelne Batteriezellen möglich. Da es sich bei dem verwendeten Zelltyp

um Hochenergie-Zellen handelt, werden sehr geringe Ladeströme bis 0,5 C empfohlen. Für eine beschleunigte Alterungssimulation wurde allerdings eine Ladestromstärke von 1 C gewählt. Obwohl die Plausibilität der Messergebnisse dadurch nicht beeinträchtigt wird, wirkt sich eine höhere Ladestromstärke auf eine geringe Zyklenzahl und damit auf eine stärkere Zellalterung aus.

Trotz der unterschiedlichen Zeitverläufe der Zellalterung weisen beide Zellen eine hohe Übereinstimmung bei der Entwicklung der einzelnen Alterungsmechanismen auf. Dabei ist auffällig, dass die getesteten Zellen beim Erreichen von 80 % der nutzbaren Restkapazität nahezu dieselben relativen Werte aufweisen, obwohl Zelle 2 eine deutlich geringere Anzahl der EFC durchläuft. Für eine genauere Betrachtung, besonders zu Beginn der Alterungsentwicklung, ist eine häufigere Messung, beispielsweise bereits nach 50 Lade-/Entladezyklen, empfehlenswert.

Die gesamte Degradation einer Zelle resultiert aus einer Kombination aus CL, LLI und LAM. Allerdings tritt eine beachtliche Überschneidung der einzelnen Mechanismen auf, sodass die Gesamalterung keiner Addition der drei Mechanismen entspricht. Aus diesem Grund müssen die einzelnen Alterungsmechanismen getrennt voneinander analysiert werden. Anhand der Steigung des einzelnen Kurvenverlaufes kann die momentane Intensität des Alterungsmechanismus erkannt und beschrieben werden. Die Bildung von EFC ist jedoch für einen reliablen Vergleich verschiedener Belastungsprofile unabdingbar. Durch die Bestimmung des Innenwiderstandes über die Pulsstrommethode konnte in dieser Arbeit der Leitfähigkeitsverlust bestimmt und zulässig mit dem anhand der EIS-Methode ermittelten CL verglichen werden. Dabei ist zu beachten, dass der Innenwiderstand einer Zelle zusätzlich den Widerstand der SEI-Schicht beinhaltet, während die EIS-Methode den Effekt des SEI-Wachstums lediglich im LLI berücksichtigt.

Aufgrund der unterschiedlichen Auswerteverfahren und diverser Startwerte muss der Vergleich der IC/DV- und EIS-Methode zwingend über eine Faktorbildung erfolgen. Somit kann die unterschiedliche Skalierung der Messmethoden relativiert werden und ein zulässiger Vergleich erfolgen.

Durch weitere Analysemethoden können die Effekte und Ursachen der auftretenden Alterungsmechanismen auf einzelne Zellbauteile zugeordnet werden [36, 37].

6 Zusammenfassung

Die vorliegende Arbeit befasst sich mit der Alterungsentwicklung in NCA Li-Ionen Zellen. Dafür wurde die Gesamalterung einer Batteriezelle in die drei Alterungsmechanismen Verlust von Leitfähigkeit, den Verlust von Lithiuminventar und den Verlust von Aktivmaterial unterteilt. Diese Degradationsmechanismen wurden anhand

zwei voneinander unabhängigen Messmethoden bestimmt und analysiert. Als Messverfahren wurden die Elektrochemische Impedanzspektroskopie und die Aufnahme der Leerlaufspannung, deren Ableitungen die Inkrementelle Kapazität/Differentielle Spannung liefert, gewählt. Beide Methoden dienen zur Detektion aller drei Alterungsmechanismen und wurden in dieser Arbeit auf eine reliable und zulässige Aussagekraft überprüft.

Die Bewertung der einzelnen Alterungsmechanismen erfolgt über die relative Entwicklung, bezogen auf einen definierten Startwert, welcher durch die erste Messung des Prüfplans ermittelt wird. Aufgrund der unterschiedlichen Auswerteverfahren der Alterungsmechanismen ergeben sich für beide Messmethoden verschiedene Startwerte und dadurch eine abweichende Skalierung. Durch den Vergleich der Entwicklungstrends kann nachgewiesen werden, dass die beiden Messmethoden ähnliche Degradationsverläufe und dadurch eine übereinstimmende Aussagekraft besitzen. Demnach sind beide Messmethoden für die Entwicklungsanalyse der drei definierten Alterungsmechanismen (CL, LLI und LAM) geeignet und zulässig. Alterungsmodelle können somit detaillierter aufgebaut und parametrisiert werden. Die Ergebnisse sollen in weiteren Untersuchungen in geeignete Modelle implementiert werden.

Diese Arbeit bildet die Grundlage für eine genauere Untersuchung der auftretenden Alterungseffekte in Li-Ionen Zellen. Unabhängig von der verwendeten Messmethode liefert der in dieser Arbeit entwickelte Mess- und Auswertalgorithmus die grundlegende Datenaufbereitung für die Analyse der Alterungsmechanismen in Li-Ionen Zellen. Dabei kann der Einfluss verschiedener Prüfparameter und Nutzungsprofile auf die Intensität der Alterungsentwicklung untersucht werden. Durch das automatisierte Auswerteverfahren kann eine reliable Datenbereitstellung mit enormer Zeitersparnis gewährleistet werden. Anhand der ermittelten Degradationssentwicklungen kann das optimale Betriebsfenster einer Zelle durch kritische Spannungsbereiche, zulässige C-Raten oder sonstige Einflussparameter erkannt und beispielsweise an das Batteriemanagement im späteren Einsatz übermittelt werden. Die bereitgestellten Daten können außerdem die Basis von Alterungsmodellen von Li-Ionen Zellen bilden. Somit können verschiedene Batteriezellen je nach gewünschtem Einsatzgebiet ausgewählt und die Lebensdauer und Leistungsfähigkeit erheblich optimiert werden.

Literatur

- [1] R. Korthauer, *Handbuch Lithium-Ionen-Batterien*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.
- [2] A. Jossen und W. Weydanz, *Moderne Akkumulatoren richtig einsetzen*, 2. Aufl., 2019.
- [3] M. Sterner und I. Stadler, *Energiespeicher: Bedarf, Technologien, Integration*. Berlin: Springer Vieweg,

- 2014.
- [4] J.-K. Park, *Principles and applications of lithium secondary batteries*. Weinheim: Wiley-VCH, 2012. [Online]. Verfügbar unter: <http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10570738>
- [5] A. Kwade *et al.*, „Current status and challenges for automotive battery production technologies“, *Nat Energy*, Jg. 3, Nr. 4, S. 290–300, 2018, doi: 10.1038/s41560-018-0130-3.
- [6] J. Vetter *et al.*, „Ageing mechanisms in lithium-ion batteries“, *Journal of Power Sources*, Jg. 147, 1-2, S. 269–281, 2005, doi: 10.1016/j.jpowsour.2005.01.006.
- [7] C. R. Birkl, M. R. Roberts, E. McTurk, P. G. Bruce und D. A. Howey, „Degradation diagnostics for lithium ion cells“, *Journal of Power Sources*, Jg. 341, S. 373–386, 2017, doi: 10.1016/j.jpowsour.2016.12.011.
- [8] C. Pastor-Fernández, K. Uddin, G. H. Chouchelamane, W. D. Widanage und J. Marco, „A Comparison between Electrochemical Impedance Spectroscopy and Incremental Capacity-Differential Voltage as Li-ion Diagnostic Techniques to Identify and Quantify the Effects of Degradation Modes within Battery Management Systems“, *Journal of Power Sources*, Jg. 360, S. 301–318, 2017, doi: 10.1016/j.jpowsour.2017.03.042.
- [9] N. Watrin, B. Blunier und A. Miraoui, „Review of adaptive systems for lithium batteries State-of-Charge and State-of-Health estimation“ in *2012 IEEE Transportation Electrification Conference and Expo (ITEC)*, Dearborn, MI, USA, 2012, S. 1–6, doi: 10.1109/ITEC.2012.6243437.
- [10] S. Zhao, F. Wu, L. Yang, L. Gao und A. F. Burke, „A measurement method for determination of dc internal resistance of batteries and supercapacitors“, *Electrochemistry Communications*, Jg. 12, Nr. 2, S. 242–245, 2010, doi: 10.1016/j.elecom.2009.12.004.
- [11] A. Barai, K. Uddin, W. D. Widanage, A. McGordon und P. Jennings, „A study of the influence of measurement timescale on internal resistance characterisation methodologies for lithium-ion cells“ (eng), *Scientific reports*, Jg. 8, Nr. 1, S. 21, 2018, doi: 10.1038/s41598-017-18424-5.
- [12] C. Pastor-Fernández, W. D. Widanage, G. H. Chouchelamane und J. Marco, „A SoH Diagnosis and Prognosis Method to Identify and Quantify Degradation Modes in Li-ion Batteries using the IC/DV technique“ in *6th Hybrid and Electric Vehicles Conference (HEVC 2016)*, London, UK, 2017, 6 (6.)-6 (6.), doi: 10.1049/cp.2016.0966.
- [13] M. Bercibar *et al.*, „SOH Estimation and Prediction for NMC Cells Based on Degradation Mechanism Detection“ in *2015 IEEE Vehicle Power and Propulsion Conference (VPPC)*, Montreal, QC, Canada, 2015, S. 1–6, doi: 10.1109/VPPC.2015.7353020.
- [14] D. Ansean *et al.*, „Lithium-ion battery degradation indicators via incremental capacity analysis“ in *2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, Milan, Italy, 2017, S. 1–6, doi: 10.1109/EEEIC.2017.7977776.
- [15] C. Pastor-Fernandez, W. Dhammika Widanage, J. Marco, M.-A. Gama-Valdez und G. H. Chouchelamane,

- „Identification and quantification of ageing mechanisms in Lithium-ion batteries using the EIS technique“ in *2016 IEEE Transportation Electrification Conference and Expo (ITEC)*, Dearborn, MI, 27.06.2016 - 29.06.2016, S. 1–6, doi: 10.1109/ITEC.2016.7520198.
- [16] M. Lewerenz, A. Marongiu, A. Warnecke und D. U. Sauer, „Differential voltage analysis as a tool for analyzing inhomogeneous aging: A case study for LiFePO₄/Graphite cylindrical cells“, *Journal of Power Sources*, Jg. 368, S. 57–67, 2017, doi: 10.1016/j.jpowsour.2017.09.059.
- [17] S. Schindler und M. A. Danzer, „A novel mechanistic modeling framework for analysis of electrode balancing and degradation modes in commercial lithium-ion cells“, *Journal of Power Sources*, Jg. 343, S. 226–236, 2017, doi: 10.1016/j.jpowsour.2017.01.026.
- [18] M. Bercibar, M. Dubarry, N. Omar, I. Villarreal und J. van Mierlo, „Degradation Mechanism Detection for NMC Batteries based on Incremental Capacity Curves“, *WEVJ*, Jg. 8, Nr. 2, S. 350–361, 2016, doi: 10.3390/wevj8020350.
- [19] M. Dubarry, V. Svoboda, R. Hwu und B. Yann Liaw, „Incremental Capacity Analysis and Close-to-Equilibrium OCV Measurements to Quantify Capacity Fade in Commercial Rechargeable Lithium Batteries“, *Electrochem. Solid-State Lett.*, Jg. 9, Nr. 10, A454, 2006, doi: 10.1149/1.2221767.
- [20] M. Dubarry, C. Truchot und B. Y. Liaw, „Synthesize battery degradation modes via a diagnostic and prognostic model“, *Journal of Power Sources*, Jg. 219, S. 204–216, 2012, doi: 10.1016/j.jpowsour.2012.07.016.
- [21] C. Birkl, „Diagnosis and Prognosis of Degradation in Lithium-Ion Batteries“, University of Oxford. [Online]. Verfügbar unter: <https://ora.ox.ac.uk/objects/uuid:7d8ccb9c-1469-4209-9995-5871fc908b54>. Zugriff am: 12. August 2019.
- [22] M. K. Gulbinska, Hg., *Lithium-ion battery materials and engineering: Current topics and problems from the manufacturing perspective*. London, Heidelberg, New York, Dordrecht: Springer, 2014.
- [23] J. P. Schmidt, „Verfahren zur Charakterisierung und Modellierung von Lithium-Ionen Zellen“.
- [24] B. Bedürftig, „Erstellung eines Li-Ionen Zellmodells unter Berücksichtigung physikalischer und chemischer Zelleffekte“, Otto-von-Guericke-Universität Magdeburg, Magdeburg, 2015.
- [25] Y. Olofsson, J. Groot, T. Katrasnik und G. Tavcar, „Impedance spectroscopy characterisation of automotive NMC/graphite Li-ion cells aged with realistic PHEV load profile“ in *2014 IEEE International Electric Vehicle Conference (IEVC)*, Florence, 2014, S. 1–6, doi: 10.1109/IEVC.2014.7056095.
- [26] M. Oldenburger *et al.*, „Investigation of the low frequency Warburg impedance of Li-ion cells by frequency domain measurements“, *Journal of Energy Storage*, Jg. 21, S. 272–280, 2019, doi: 10.1016/j.est.2018.11.029.
- [27] S. Käbitz, „Investigation of the aging of lithium-ion batteries using electroanalysis and electrochemical impedance spectroscopy“.
- [28] P. Keil, *Aging of lithium-ion batteries in electric vehicles*. München, 2017.
- [29] M. Messing, T. Shoa und S. Habibi, „Lithium-Ion Battery Relaxation Effects“ in *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, Detroit, MI, USA, 6/19/2019 - 6/21/2019, S. 1–6, doi: 10.1109/ITEC.2019.8790449.
- [30] P. Kurzweil und O. K. Dietlmeier, *Elektrochemische Speicher*. Wiesbaden: Springer Fachmedien Wiesbaden, 2015.
- [31] A. Roach, *Statistik für Ingenieure: Wahrscheinlichkeitsrechnung und Datenauswertung endlich verständlich ; [Beispielaufgaben mit ausführlichen Lösungen]*. Berlin: Springer Spektrum, 2014.
- [32] R. L. Wasserstein und N. A. Lazar, „The ASA Statement on p -Values: Context, Process, and Purpose“, *The American Statistician*, Jg. 70, Nr. 2, S. 129–133, 2016, doi: 10.1080/00031305.2016.1154108.
- [33] A. Savitzky und M. J. E. Golay, „Smoothing and Differentiation of Data by Simplified Least Squares Procedures“, *Anal. Chem.*, Jg. 36, Nr. 8, S. 1627–1639, 1964, doi: 10.1021/ac60214a047.
- [34] H. Azami, K. Mohammadi und B. Bozorgtabar, „An Improved Signal Segmentation Using Moving Average and Savitzky-Golay Filter“, *JSIP*, Jg. 03, Nr. 01, S. 39–44, 2012, doi: 10.4236/jsip.2012.31006.
- [35] J. & O. Guiñón, Emma & García-Antón und V. José & Pérez-Herranz, „Moving Average and Savitzki-Golay Smoothing Filters Using Mathcad“, *International Conference on Engineering Education*, 2007.
- [36] K. Richter *et al.*, „Surface Film Formation and Dissolution in Si/C Anodes of Li-Ion Batteries: A Glow Discharge Optical Emission Spectroscopy Depth Profiling Study“, *J. Phys. Chem. C*, Jg. 123, Nr. 31, S. 18795–18803, 2019, doi: 10.1021/acs.jpcc.9b03873.
- [37] N. Delpuech *et al.*, „Mechanism of Silicon Electrode Aging upon Cycling in Full Lithium-Ion Batteries“ (eng), *ChemSusChem*, Jg. 9, Nr. 8, S. 841–848, 2016, doi: 10.1002/cssc.201501628.

Einsatz von Simulationen beim Entwurf leistungselektronischer Systeme

Robert Rohn^{1*}, Thorben Schobre¹, Günter Tareilus¹, Regine Mallwitz¹

¹Institut für Elektrische Maschinen, Antriebe und Bahnen; Technische Universität Braunschweig; Hans-Sommer-Str. 66; 38106 Braunschweig, Deutschland; *r.rohn@tu-braunschweig.de

Kurzfassung

Dieser Beitrag soll eine Einsatzmöglichkeit der Simulationssoftware LTspice im Entwurfsprozess leistungselektronischer Systeme vorstellen. Dabei geht es um den Entwurf des Modells eines Silizium-Karbid-Wechselrichters mit Sinusfilter für lange Motorleitungen bei hohen Schaltfrequenzen. Es wird beschrieben, wie die Festlegung geeigneter Parameter erfolgt und geeignete Vereinfachungen getroffen werden. Abschließend wird am Beispiel der Halbleiter ein Vergleich gezogen, ob es sinnvoll ist, alle physikalischen Eigenschaften abzubilden oder ob getroffene Vereinfachungen das Ergebnis zu stark verfälschen.

Einleitung

Bei Wechselrichtern handelt es sich um Systeme, die zum Antrieb von Motoren mit variabler Drehzahl in einer Vielzahl von Industrieanlagen eingesetzt werden. Im Allgemeinen werden für den Anschluss von elektrischen Maschinen an Wechselrichtern geschirmte Motorleitungen eingesetzt. Diese weisen im Vergleich zu ungeschirmten Leitungen gleichen Querschnitts einen insgesamt größeren Durchmesser bei mehr Gewicht auf und verursachen zusätzlich durch ihre komplexe Handhabung ein Vielfaches der Kosten. Die Entwicklungen im Bereich der Wide-Bandgap-Halbleiter erlauben durch steile Schaltflanken hohe Schaltfrequenzen, was zu kapazitiven Umladeströmen auf dem Schirm der Motorleitung und in der Maschine führt. Gerade bei großen Leitungslängen, wie sie beispielsweise in explosionsgeschützten Bereichen häufig anzutreffen sind, können hohe Umladeströme fließen, wodurch sich die Systemeffizienz verringert.

Im BMWi geförderten Projekt Ide3Al soll daher ein Silizium-Karbid-Wechselrichter mit Sinusfilter entwickelt werden. Der Sinusfilter glättet die Ausgangsspan-

nung des Wechselrichters, wodurch auf den Leitungsschirm verzichtet werden kann. Dieser Verzicht birgt ein erhebliches Potential zur Effizienzsteigerung, da weniger Energie zum Umladen der Schirmkapazität aufgewendet werden muss. Auch die Verluste in der Maschine verringern sich durch die Spannungsglättung erheblich. Der Fokus der folgenden Betrachtungen liegt hierbei auf Leitungslängen von rund 100 m, da die beschriebenen Effekte hier besonders zum Tragen kommen.

Um eine Einschätzung des Einsparpotentials zu erhalten und die Auslegung von Filter und Wechselrichter zu unterstützen, wird im Projekt ein Simulationsmodell des Systems erstellt. Mit dessen Hilfe lassen sich der Einfluss von Flankensteilheit und Ableitkapazitäten auf das Gesamtsystem untersuchen.

Im Folgenden werden die notwendigen Schritte zum Erstellen des Modells beschrieben.

Wechselrichter

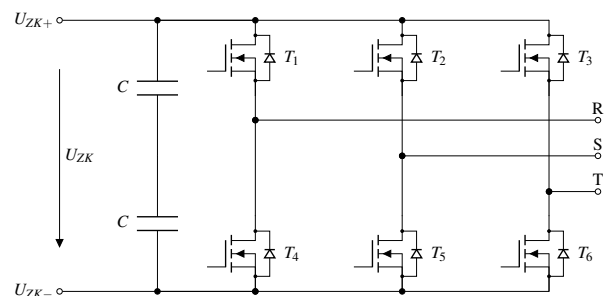


Figure 1: B6-Leistungsteil eines Wechselrichters mit eingangsseitigem Spannungszwischenkreis und drei Halbbrücken

Im Leistungsteil des Wechselrichters (Fig. 1) kommt eine klassische B6-Anordnung der Halbleiter T_1 - T_6 zum Einsatz. Eingangsseitig befindet sich ein Gleichspannungszwischenkreis, dem die Energie zum Antrieb der

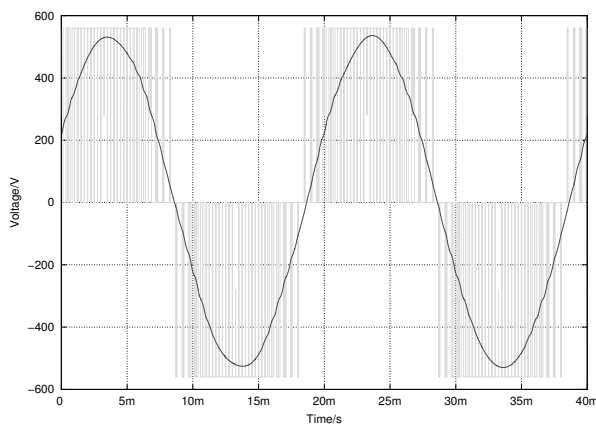


Figure 2: Spannungen am Sinusfilter (grün: Wechselrichter-ausgangsspannung an Klemme R,S, blau: Filter-ausgangsspannung an Klemme F1,F2)

Maschine entnommen wird. An den Ausgangsklemmen R, S, T kann die Wechselspannung für den Motor abgegriffen werden.

Als Vereinfachung werden die Halbleiter jedoch nicht mit ihren physikalischen Eigenschaften dargestellt, sondern idealisiert durch spannungsgesteuerte Schalter. Diese nehmen abhängig von der an ihren Steuereingängen anliegenden Spannung den Zustand leitend oder sperrend ein, was sich mittels Spice-Direktive festlegen lässt. Darüber lassen sich die Durchlassverluste und Leckströme der realen Halbleiter näherungsweise nachbilden. Zusätzlich lässt sich das Modell durch eine Streuinduktivität erweitern. Im vorliegenden Fall wird allerdings hierauf verzichtet.

Die Halbleiter werden mittels Raumzeigermodulation so angesteuert, dass sich an den Klemmen R, S, T des Lastabgangs in Fig. 1 eine dreiphasige, gepulste, im Mittel sinusförmige Ausgangsspannung ergibt, dargestellt in Fig. 2. Pro Halbbrücke darf dabei allerdings immer nur einer von beiden Halbleitern eingeschaltet sein. Andernfalls fließt in der betreffenden Halbbrücke ein sehr großer Strom, was die Betrachtung unrealistisch macht und in der Realität zur Zerstörung der betroffenen Halbleiter führt. Daher ist beim Umschalten der Halbbrücken eine Totzeit einzuhalten, eine kurze Phase, in der keiner von beiden Halbleitern eingeschaltet ist.

Der Laststrom kann sich jedoch bei induktiven Lasten wie elektrischen Maschinen nicht sprunghaft ändern, sondern fließt zunächst unverändert weiter. Dadurch steigt die Spannung an den Schaltern während

der Totzeit stark an. Um dies zu verhindern, muss ihm ein alternativer Pfad zur Verfügung gestellt werden, in dem er freilaufen kann, was mit einer antiparallelen Diode zu jedem Schalter erreicht wird. Hierdurch wird dem Strom ermöglicht, auf die Diode zu kommutieren, was die am Halbleiter anliegende Spannung auf die Vorwärtsspannung der Diode begrenzt [1, p. 87].

Viele kommerzielle B6-Module integrieren diese Freilaufdioden standardmäßig. Dioden- und Halbleitermodell lassen sich über folgende Spice-Direktiven parametrieren:

```
.model Mosfet T(Ron=10m...
...Roff=100Meg Vt=4V)
.model Diode D(Ron=10m...
...Roff=100Meg Vfwd=0.6V)
```

In diesem Beispiel werden der Widerstand im eingeschalteten Zustand R_{on} und im ausgeschalteten Zustand R_{off} , die Einschaltwellenspannung des Mosfets V_t und die Vorwärtsdurchbruchspannung der Diode V_{fwd} parametrisiert [2].

Sinusfilter

Wie bereits erwähnt ergibt sich an den Ausgangsklemmen des Wechselrichters eine gepulste Wechselspannung. Wird an den Klemmen eine elektrische Maschine angeschlossen, sorgt das integrierende Verhalten der Motorinduktivität für eine Glättung des Stroms, so dass dieser einen nahezu sinusförmigen Verlauf hat. Allerdings kann der Wechselrichter selten direkt an die Maschine angeschlossen werden. In der Regel kommt zwischen Wechselrichter und Maschine eine Motorleitung zum Einsatz.

In der Norm DIN VDE 0160-103 sind Grenzwerte für die maximale Flankensteilheit auf ungeschirmten Leitungen festgelegt. Bei Wechselrichtern mit Schaltfrequenzen zwischen 8 kHz und 20 kHz ist es daher üblich, eine geschirmte Leitung zu verwenden. Durch den geringen Abstand zwischen den Leitern und dem Leitungsschirm ergibt sich für die geschirmte Motorleitung eine deutlich größere Ableitkapazität, als sie eine vergleichbare ungeschirmte Leitung aufweist. Wegen der Spannungspulsation am Wechselrichterausgang wird die Ableitkapazität mit Schaltfrequenz umgeladen, was im Betrieb zur Ausbildung von Ableitströmen führt. Diese können eine erhebliche Verlustleistung zur Folge haben.

Mit dem Einzug von Wide-Band-Gap-Halbleitern wie Silizium-Karbid-Mosfets (SiC-Mosfets) sind jedoch

sehr steile Schaltflanken und damit hohe Schaltfrequenzen von 100 kHz und mehr möglich. Diese hohen Schaltfrequenzen führen zu einem proportional größeren Ableitstrom, da die Ableitkapazitäten häufiger umgeladen werden. Daher ist der Einsatz eines Sinusfilters (Fig. 3) sinnvoll [3].

In Fig. 4 und Fig. 5 sind die Spektren der Wechselrichterausgangsspannung und der Filterausgangsspannung bei 100 kHz Schaltfrequenz gegenübergestellt. Die eingezeichnete Normgrenzwertkennlinie wird am Wechselrichterausgang nicht mehr eingehalten, wohl aber am Sinusfilterausgang. Die Auslegung der Filterelemente muss entsprechend der Anwendung angepasst werden.

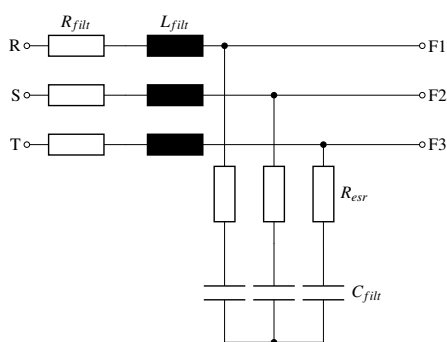


Figure 3: Ersatzschaltbild des Sinusfilters

Die Filterinduktivität L_{filt} führt zusammen mit der Filterkapazität C_{filt} , wie bereits für die Maschine beschrieben, zu einer Glättung des Laststroms, wodurch sich an den Ausgangsklemmen F1 bis F3 in Fig. 3 eine sinusförmige Ausgangsspannung ergibt. Der Dämpfungswiderstand R_{esr} bildet den äquivalenten Ersatzwiderstand des Kondensators ab, R_{filt} repräsentiert die Wicklungs- und Eisenverluste der Filterinduktivität. Zur Bestimmung der Parameter der Filterelemente wird zunächst der maximal zulässige Stromrippel ΔI festgelegt. Mit diesem kann zusammen mit der Zwischenkreisspannung U_{ZK} und der Schaltfrequenz f_{sw} nach Gl. (1) die erforderliche Größe der Induktivität bestimmt werden [1, p. 191].

$$L_{filt} = \frac{U_{ZK}}{4 \cdot \Delta I \cdot f_{sw}} \quad (1)$$

Anschließend wird die Grenzfrequenz f_0 des Filters festgelegt, die unterhalb der Schaltfrequenz der Halbleiter und oberhalb der Regelfrequenz liegen sollte. Als weiteres Kriterium ist die Dämpfung entscheidend, die das Filter bei der Schaltfrequenz der Halbleiter bie-

ten soll. Die höchste Frequenz der Grundwelle findet hier keine Betrachtung, da diese unterhalb der Regelfrequenz liegt. Ein Filter 2. Ordnung bietet eine Dämpfung von $20 \frac{dB}{dekade}$. Mit der Grenzfrequenz lässt sich daraus der Kapazitätsbedarf des Filters nach Gl. (2) bestimmen [4].

$$C = \frac{1}{4 \cdot \pi^2 \cdot L_{filt} \cdot f_0^2} \quad (2)$$

Durch die ausbleibende Spannungspulsation am Filterausgang (Fig. 2) werden die Ableitkapazitäten nur noch mit der Grundwellenfrequenz umgeladen, wodurch vollständig auf die Schirmung der Motorleitung verzichtet werden kann, wie aus Fig. 5 hervorgeht [5]. Daher wird im weiteren Verlauf die Anordnung einer einfachen dreiadrigen Leitung betrachtet.

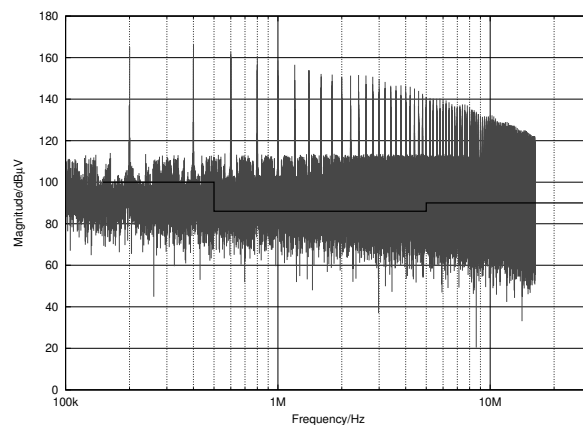


Figure 4: Spannungsspektrum an den Wechselrichterklammern R,S,T mit Grenzwertkurve nach DIN VDE 0160-103

Motorleitung

Die Motorleitung bildet wie das Sinusfilter ein System 2. Ordnung (Fig. 3). Es finden der Ohmsche Leitungswiderstand R_{cab} , die Leitungsinduktivität L_{cab} und die Leitungskapazität C_{cab} mit dem Dämpfungswiderstand R_{esr} Berücksichtigung.

Zusätzlich wird die Kabelisolation durch den Widerstand G_{iso} dargestellt. Die Ersatzschaltbildparameter der Leitung setzen sich aus den Leitungsbelägen und der -länge zusammen und sind etwa eine Größenordnung kleiner als die des Sinusfilters. Anders als z.B.

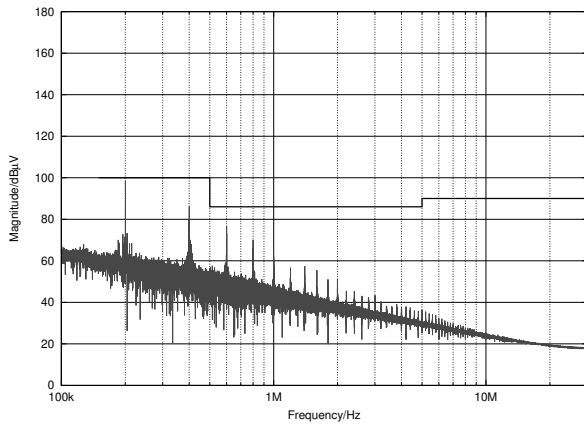


Figure 5: Spannungsspektrum an den Filterausgangsklemmen F1,F2,F3 mit Grenzwertkurve nach DIN VDE 0160-103

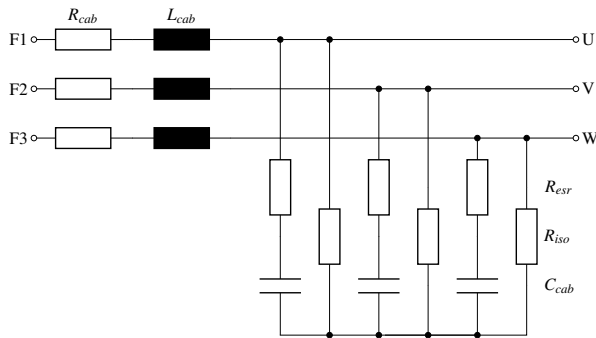


Figure 6: Ersatzschaltbild der Motorleitung

in [6] finden bei diesem Modell keine Reflexionen Berücksichtigung.

Maschine

Abschließend findet die Maschine Anschluss an das Leitungsmodell. Dabei kommt das einfache Ersatzschaltbild einer Synchronmaschine in Sternschaltung zum Einsatz, dargestellt in Fig. 7 [1, p. 340]. Durch den Widerstand R_M wird die in der Maschine umgesetzte Wirkleistung abgebildet. Dies umfasst die abgegebene mechanische Leistung wie auch interne Verluste durch Ummagnetisierung, Skin- und Proximityeffekt sowie Kupferverluste. Die Maschineninduktivität wird durch L_M repräsentiert. Mittels Spannungsquelle lässt sich eine Gegenspannung U_G einstellen, die zur Simulation generatorischer Arbeitspunkte notwendig ist. Da-

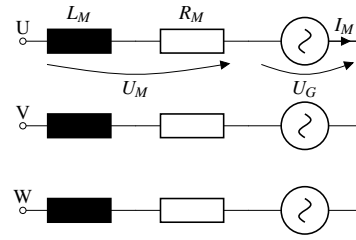


Figure 7: Ersatzschaltbild einer Synchronmaschine

mit lassen sich alle vier Quadranten darstellen. Da bestimmte Arbeitspunkte simuliert werden sollen, lassen sich Motorinduktivität L_M und -widerstand R_M nach Gl. (3) und (4) über Strangspannung U_M und -strom I_M einstellen. Der Strom in Phase N ist bei einem dreiphasigen System um 0° , 120° und 240° zu verschieben.

$$L_M = \frac{\hat{U}_M}{2\pi \cdot f_G \cdot \sqrt{2} \cdot I_M \cdot \sqrt{1 - \cos^2(\varphi)}} \quad (3)$$

$$R_M = \frac{\hat{U}_M}{\sqrt{2} \cdot I_M \cdot \cos(\varphi)} \quad (4)$$

Um Einschwingvorgänge abzukürzen sollte auch der initiale Strom, der in der Motorinduktivität fließt, festgelegt werden. Dies erfolgt nach Gl. (5) mit einer .ic-Direktive.

$$I_{M_N}(t_0) = I_M \cdot \sqrt{2} \cdot \sin\left(-\varphi - \frac{N \cdot 2\pi}{3}\right) \quad (5)$$

Auch in diesem Fall ist die Phasenverschiebung für alle Motorphasen entsprechend zu berücksichtigen.

Ergebnisse

Dieses Modell soll eine Basis für weitergehende Untersuchungen bilden. So lässt sich der Filterbedarf bei verschiedenen Flankensteilheiten anhand des auftretenden Spektrums an den Ausgangsklemmen des Wechselrichters vorhersagen. In Fig. 4 und 5 sind die Spektren für den Betrieb ohne Sinusfilter und mit Sinusfilter für den Fall des idealen Schaltens gegenüber gestellt. Es lässt sich unschwer erkennen, dass die Grenzwertkennlinie ohne Sinusfilter deutlich überschritten wird, weshalb ein Betrieb ohne geschirmte Motorleitung unzulässig ist. Mit Sinusfilter werden die Grenzwerte hingegen eingehalten. Weiterhin spielt die Anbindung der Sternpunkte von Filter und Maschine an den Zwischenkreis eine Rolle bei der Filterwirkung. Mit dem Modell las-

sen sich verschiedene Variationen untersuchen.
Auch der Einfluss des Schaltverhaltens der Halbleiter

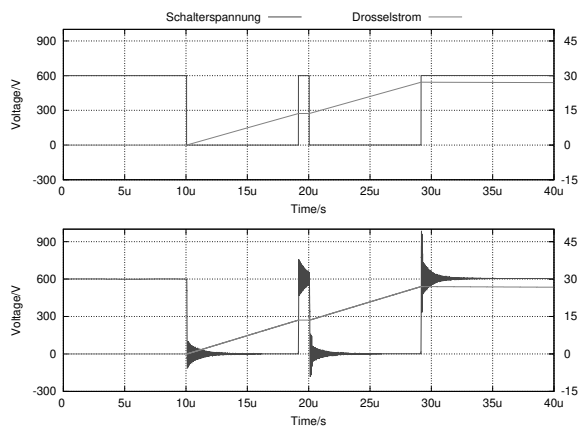


Figure 8: Unterschiede im Schaltverhalten idealisierter Schalter (oben) gegenüber eines exakten Halbleitermodells (unten)

lässt sich betrachten. Viele Hersteller liefern entsprechende Simulationsmodelle die sich einbinden lassen. In Fig. 8 ist das Schaltverhalten des idealisierten B6-Modells mit dem Schaltverhalten des Halbleitermodells eines Herstellers gegenübergestellt. Es lassen sich deutliche Unterschiede erkennen, die von der Ausgangskapazität der Halbleiter hervorgerufen werden. Durch die hochfrequenten Umladevorgänge dieser Kapazitäten verkürzt sich der Simulationszeitschritt allerdings erheblich.

Die Simulationsdauer von zwei Grundwellen verlängert sich auf der genutzten Hardware (Intel Xeon E3-1230v3, 32 Gb Ram) bei ansonsten gleichbleibenden Simulationsparametern in diesem Fall von 0,6 h auf 6,6 h Stunden. Die Unterschiede im Systemverhalten sind allerdings vernachlässigbar, wie sich aus Fig. 9 ablesen lässt, weshalb sich der zusätzliche Rechenaufwand hier nicht auszahlt.

Um hingegen eine Bestimmung von Schaltverlusten der Halbleiter vorzunehmen, ist es notwendig mit einem exakten Modell zu simulieren. In diesem Beispiel führen die beschriebenen Vereinfachungen zu großen Abweichungen von 2,27 W bei idealisierter Betrachtung zu 44,6 W mit exaktem Halbleitermodell.

Für die Auslegung einer Kühllösung ist die idealisierte Betrachtung somit viel zu Ungenau.

Es ist also Abwägungssache, ob sich die Investition von Rechenzeit in der Leistungselektronik auszahlt. Betrachtungen auf Systemebene lassen sich i.d.R. gut an

einem einfachen Modell abschätzen, Detailbetrachtungen dagegen müssen anhand eines exakten Modells getroffen werden.

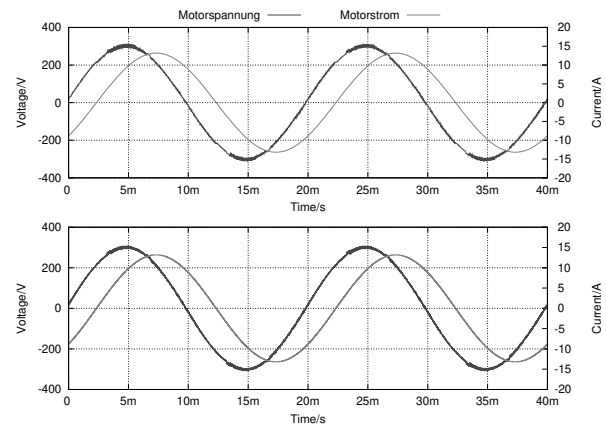


Figure 9: Unterschiede im Simulationsverlauf von Motorspannung (blau) und dem Motorstrom (rot) für die Idealisierte Annahme (oben) und ein reales Halbleitermodell (unten)

References

- [1] SPECOVIVUS, JOACHIM: *Leistungselektronik und EMV*. Springer Fachmedien Wiesbaden, Wiesbaden, 2017.
- [2] BROCARD, G.: *Simulation in LTSpice IV: Handbuch, Methoden und Anwendungen*. Würth Elektronik. Swiridoff, 2013.
- [3] WEIS, BENNO: *Kompakter 690V-Umrichter mit SiC-Schottkydioden für sinusförmige Ausgangsspannung*. doctoralthesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 2008.
- [4] PICATOSTE, R., M. BUTCHER A. MASI: *Sine wave filter for stepper motor drives working with long cables*. 2012 International Conference on Electrical Machines, 1075–1081, Sep. 2012.
- [5] WEIS, B.: *Ein kompakter 690-V-Umrichter mit integriertem Sinus-Ausgangsfilter und SiC-Schottky-Freilaufdioden*. VDE Kongress'08 Zukunftstechnologien Innovationen – Märkte – Nachwuchs. Siemens AG, VDE Verlag, Nov. 2008.
- [6] VON JOUANNE, A., P. ENJETI W. GRAY: *The effect of long motor leads on PWM inverter fed AC*

motor drive systems. Proceedings of 1995 IEEE Applied Power Electronics Conference and Exposition - APEC'95, 2, 592–597 vol.2, 1995.

Kurzbiographie



Robert Rohn erwarb an der FH Aachen 2015 den Bachelor of Science und 2017 den Master of Engineering im Studiengang Elektrotechnik. Seine Bachelorarbeit verfasste er bei der Firma devolo über ein Thema aus dem Bereich Energy Harvesting, die Masterarbeit bei der Firma futavis, wo er einen interleaved SiC-Synchronous-Buck-Converter entwickelte. Seit November 2017 arbeitet er als wissenschaftlicher Mitarbeiter am IMAB der TU-Braunschweig und befasst sich mit Stromwandlern und Ladegeräten auf Basis von SiC-Halbleitern.

Lifetime modelling of electrical machines using the methodology of design of experiments

Lucas Vincent Hanisch^{1*}, Markus Henke¹

¹Inst. for Electrical Machines, Traction and Drives, Technische Universität Braunschweig, Hans-Sommer Str. 66, 38118 Braunschweig, Germany; *l-v.hanisch@tu-braunschweig.de

Abstract. In the coming years, electromobility will be confronted with increasing demands regarding the reliability of electrical machines. In this paper a modeling methodology is presented, which allows to estimate the reliability and lifetime of the insulation system of electrical machines. Different statistical and physical modeling methods are presented, which are transformed for the later multiple regression. The methodology of Design of Experiments (DoE) is used to describe the insulation system. Since the effort for the experimental design of the DoE varies strongly with the number of effects to be investigated and the statistical accuracy, different experimental designs are presented, which can be considered for different numbers of factors. Depending on the research question, a suitable experimental design can be selected. For the calculation of the lifetime, Miner's rule is used in addition to the multiple regression, so that the percentage lifetime consumption due to a load spectrum can also be calculated.

Introduction

In the future, technologies such as electrified aircraft, trolley wire trucks or autonomous driving will be established in the field of electric mobility. Reliability, durability and safety are important criteria for the acceptance of new technologies in society. In addition to increased safety requirements, these technologies must also be able to cope with new, more challenging boundary conditions.

Whereas bearing damage used to be the most frequent cause of failure of electrical machines, the increased requirements and new boundary conditions lead to a more varied error pattern [1], [2]. A deep understanding of the causes of failure and the relevant damage mechanisms is necessary to design electrical machines for these new applications. In addition to the design of durable machines that operate under increased environmental conditions, lifetime models, on the other hand, can help electrical machines achieve a minimum target lifetime for a given load. This offers potential to save resources and reduce costs. Lifetime models and reliability analyses are therefore becoming increasingly important in the design process of electrical machines.

1 Damage mechanisms

The damage of electrical machines is caused by various mechanisms from different physical disciplines and there are various modelling approaches to model these damage processes. Since the use of wide bandgap semiconductors and the increasing electrical load, the cause of failure of electrical machines is increasingly based on faults and breakdowns of the electrical insulation system [2]. Because failures of the electrical insulation system will occur more frequently in the future, research concentrates on lifetime models of these insulation systems.

Basically, with regard to the aging effects that damage the insulation system, a distinction can be made between constant stresses and transient stresses. The probability of a fault in the insulation system at constant stresses is proportional to the number of operating hours and at transient stresses proportional to the number of transient effects. Figure 1 shows as examples for constant stresses the ambient temperature and the voltage slope oval and for transient stresses partial discharges rectangular.

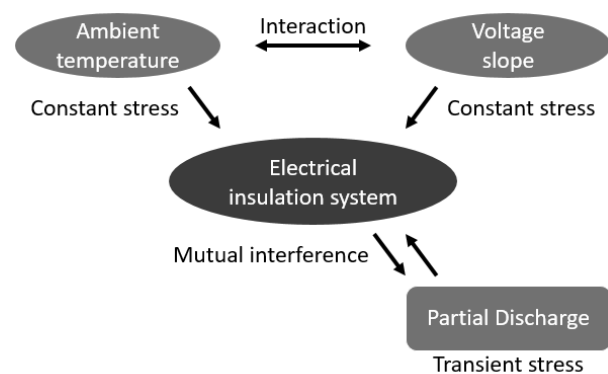


Figure 1: Examples of different stresses and their interactions.

A breakdown in the insulation system often occurs due to different combinations of the individual stresses or the interactions of these stresses.

As an example for the interaction of stresses figure 1 shows the interaction between the temperature and the voltage slope. Insulating materials heat up due to dielectric losses. The dielectric power loss as well as the dielectric loss factor are again frequency dependent. Power electronic signals with high voltage slope and high frequency harmonics can therefore influence the temperature of the insulation system.

An interaction can also become effective by changing the properties of the insulation system. Partial discharges can damage the insulation system and lead to air-filled cavities. Due to the low permittivity of air and the significantly lower dielectric strength of air compared to insulation materials, increased field strengths in these cavities can intensify partial discharges and the aging of the insulation system. In [3] an interaction due to changed properties of the insulation system is called indirect interaction.

2 Modeling methodologies

The modelling approaches can be divided into two different categories. The physical and the statistical approaches.

2.1 Physical modeling approaches

The goal of physical modelling approaches is the mathematical description of the aging effects. A holistic model, in which the different aging mechanisms from different physical disciplines are integrated, is very complex. In addition, at certain activation energies additional phenomena occur, become more dominant or the damage effect has to be described mathematically completely different.

The advantage compared to statistical models is that no lifetime tests are necessary for the parameterization of the model. Only the measurement of single physical quantities is necessary.

One of the first physical models to describe aging effects was developed by Crine and was originally used for extruded dielectric cables. Crine takes up Artbauer's theory and assumes that the dielectric strength of amorphous insulation materials is essentially determined by the presence of vacancies induced by electromechanical deformation of molecular chains in the insulation material. In these vacancies, the free electrons find more favourable conditions to absorb the energy necessary for impact ionization due to the applied electric field [4]. Crine assumes that above a critical field strength, the number of these

vacancies increases and they combine to form larger submicrocavities. When submicrocavities have formed, electrons or ions are strongly accelerated under the influence of the electric field and can absorb enough kinetic energy to break weak Van der Waals bonds. As the submicrocavities expand, the electrons absorb even more energy and break more molecular bonds until the insulating material finally collapses and electrical breakdown occurs [5].

Crine describes the probability of breaking Van der Waals bonds depending on temperature T and electric field strength E as follows:

$$p^+(T, E) \cong \frac{kT}{h} \exp\left(-\frac{\Delta G - e\lambda E}{kT}\right) \quad (1)$$

Here k is the Boltzmann constant, h the Planck's constant, ΔG the critical field strength, e the charge of an electron and δ the length of the free path. Additionally Crine considers the probability of the backward process as

$$p^-(T, E) \cong \frac{kT}{h} \exp\left(-\frac{\Delta G + e\lambda E}{kT}\right) \quad (2)$$

The net destruction rate is the subtraction of these probabilities

$$p = p^+ - p^- \cong \frac{2kT}{h} \exp\left(-\frac{\Delta G}{kT}\right) \sinh\left(\frac{e\delta E}{kT}\right) \quad (3)$$

In Crine's physical model, the lifetime of the insulation system is given as the reciprocal of the destruction rate

$$L \cong \frac{h}{2kT} \exp\left(\frac{\Delta G}{kT}\right) \operatorname{csch}\left(\frac{e\lambda E}{kT}\right) \quad (4)$$

At high fields, equation (4) reduces to

$$L \cong \frac{h}{2kT} \exp\left(\frac{\Delta G - e\lambda E}{kT}\right) \quad (5)$$

After Crine, Lewis developed a new physical model that is also based on the formation of microcavities and the breaking of molecular bonds. Just like Crine, Lewis also takes into account the temperature T and the electric field strength E . The destruction rate K_b and the formation rate K_r of the bonds are determined as follows:

$$K_b(T, E) = \frac{kT}{h} \exp\left(-\frac{U_b - \gamma_b \varepsilon E^2}{kT}\right) \quad (6)$$

$$K_r(T, E) = \frac{kT}{h} \exp\left(-\frac{U_r + \gamma_r \varepsilon E^2}{kT}\right) \quad (7)$$

U_b and U_r are the critical energies at which the bonds break or form again. ε is the dielectric permittivity and γ_b or γ_r are fitting parameters with the dimensions of a volume. The aging of the insulation system is equated with the propagation of cracks and voids in the insulation system according to the Griffith criterion [6]

$$L = \int_0^{b_c} \frac{1}{K_b(1-b) - K_r b} db \quad (8)$$

b_c represents the critical number of broken molecular bonds above which insulation failure occurs.

2.2 Statistical modeling approaches

In contrast to physical models, statistical lifetime models are based on accelerated lifetime tests. The most popular statistical lifetime model was developed by Arrhenius and describes the quantitative dependence of the chemical reaction rate r on the temperature T

$$r = A \cdot e^{-\frac{E_A}{kT}} \quad (9)$$

where A is a constant and E_A is the activation Energy. Dakin used Arrhenius' equation to describe the processes taking place in the insulation material as a function of temperature. Except for the constants Dakin uses the same equation as Arrhenius [7]:

$$L = A_{Dakin} \cdot e^{\frac{B_{Dakin}}{T}} \quad (10)$$

Partial discharges are often cited in the literature as the cause of electrical aging of electrical insulation systems. Their frequency and effects on service life increase exponentially with increasing voltage V . Mathematically, this relationship can be described using the inverse power model

$$L = c \cdot V^{-n} \quad (11)$$

here $c > 0$ is a material constant and $n > 0$ is the power law constant.

3 Design of Experiments

As mentioned before, the parameterization of the models from chapter 2.2 is done by accelerated lifetime tests. Despite the increased conditions compared to real operation, these tests are complex and time consuming. Physical models, on the other hand, are mathematically very complex and often only consider the electric field strength and temperature. The physical modelling of additional aging effects and the modelling of the individual interactions leads to an exponential increase in complexity and to the failure of the purely physical modelling approach. With the methodology of Design of Experiments (DoE) it is possible to model many effects including their interactions and to reduce the number of necessary life tests to a minimum.

The basic idea is that the insulation system, or even

any deterministic system, can be described with a mathematical model. Different approaches can be used as mathematical models. Some were presented in chapter 2. The parameterization of the model is done after the complete evaluation of the experimental design. DoE is therefore one of the statistical model approaches. In the following different experimental designs are discussed.

3.1 Full factorial design

In an experimental design, the number of effects to be investigated, the number of measuring points and the number of measured values per measuring point are defined. The advantage of an experimental design compared to a one-factor-at-a-time-plan is that each test result can be used to calculate several effects. This considerably reduces the test effort.

If, for example, eight measured values per measuring point are necessary to determine the influence of a factor on the service life of the insulation system with acceptable accuracy, a "one factor at a time" plan requires 32 tests. In the first 8 tests the lifetime would be determined for the factors x_1 , x_2 and x_3 at a low level. Then eight tests would be carried out on each factor at an increased level to investigate the influence of a factor on the service life. This is shown in figure 2a. With a full factorial design, 16 tests would be sufficient to determine the effect of a factor with the same accuracy. Instead of examining the effect once by comparing eight value pairs, the effect is examined four times by comparing two value pairs. The number of value pairs used to calculate an effect and the statistical validation are identical. Figure 2b shows the full factorial design. Since the test specimens can only be used once in the accelerated lifetime tests, the full factorial design reduces costs and time. In addition, the information content obtained from the measurements is increased, because the effect of the factors can be analyzed at different levels.

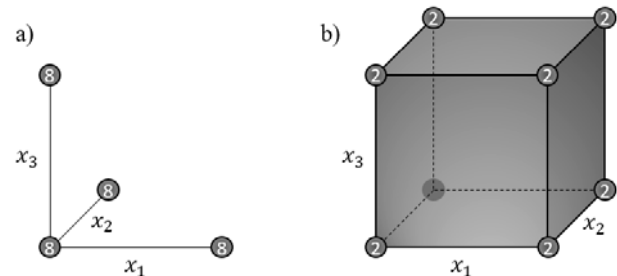


Figure 2: Comparison of a "one factor at a time" plan a) with a full factorial design b).

With each additional factor added to an experimental

design, the information content of the DoE increases. With increasing information content, the effort of the experimental design increases with F^m . F is the number of measuring points in one dimension and m the number of factors. The experimental design in figure 2b has $2^3 = 8$ measuring points. The number of measurements per measuring point determines the accuracy of the experimental design. The experimental design in figure 2b has a total of 16 measurement values with which the influence of the effects on the service life of the insulation system can be determined. If the number of tests per measurement point is increased from two to eight, 64 measurement values can be used. The accuracy increases, but also the effort. Information content, accuracy and effort of an experimental design are inseparably linked. The optimal experimental design must be adapted to the respective requirements. Figure 3 shows the combination of the three basic properties of an experimental design. If the point is further outside the triangle, an increase in information, an increase in accuracy or a reduction in effort is meant.

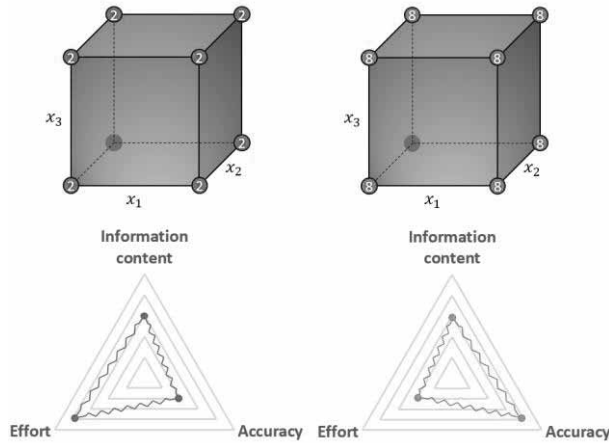


Figure 3: Variation of an experimental design by an increased number of tests.

With statistical models there is always the risk of errors. There are two possible types of errors that can occur. The influence of a factor on the aging of the insulation system could be assumed to be significant, although in reality it is not responsible for the aging or a significant effect on the aging of an insulation system could be overlooked. In a hypothesis test these errors are called first type errors and second type errors. In order to determine the number of test specimens required in a design for a given accuracy, such a hypothesis test should be performed.

The hypothesis is as follows: The factor has no effect

on the lifetime of the insulation system. In table 1 the four possible decisions of this hypothesis test are shown. The accuracy of the experimental design is determined by the errors of the first α and second type β and the change in lifetime ΔL to be detected. Table 2 shows the number of tests to achieve the desired accuracy.

	Hypothesis correct	Hypothesis wrong
Assume hypothesis	Non-significant effect detected	Type I error α non-significant effect assumed to be significant
Reject hypothesis	Type II error β Significant effect not recognized	Power $(1-\beta)$ Significant effect detected

Table 1: Possibilities of a hypothesis test.

	$\alpha = 10\%$				$\alpha = 5\%$				$\alpha = 1\%$				Type I error
$\Delta L/\sigma$	60%	70%	80%	90%	60%	70%	80%	90%	60%	70%	80%	90%	Power
0,5	60	78	102	140	82	102	128	172	132	158	192	242	
0,75	28	36	46	64	38	46	58	78	62	72	88	110	
1	16	22	28	36	22	28	34	46	36	42	52	64	
1,5	10	12	14	18	12	14	18	22	18	22	26	30	
2	6	8	8	12	8	10	12	14	12	14	16	20	

Table 2: Necessary number of tests for a certain accuracy when each factor has two levels.

If a significant effect is to be detected at $(1 - \beta) = 90\%$, if it changes the lifetime by at least one standard deviation σ on average, and a non-significant effect is to be falsely assumed to be significant only at $\alpha = 1\%$, 64 tests are required for this experimental design. This corresponds to the experimental design shown in figure 3 on the right. The experimental design in figure 3 left is less complex with 16 tests, but a significant effect is detected only at $(1 - \beta) = 60\%$ and the probability of erroneously assuming a non-significant effect as significant is $\alpha = 10\%$. Since the standard deviation σ influences the accuracy of the DoE, the test specimens should be manufactured as identically as possible and the accelerated lifetime tests should be performed under the same boundary conditions.

3.2 Fractional factorial design

Not only a high accuracy requires a high number of tests but also with increasing information content and the consideration of further influencing factors the effort of an experimental design increases with F^m . One possibility to efficiently investigate several effects despite the exponentially growing effort is the use of fractional factorial designs. The correlation between information content, accuracy and effort from figure 3 can not be avoided, but

in experimental designs that take many factors into account, some information is irrelevant or can be excluded in advance as a cause for the aging of the insulation system. This is especially true for the i -fold interactions between the factors m , which can be calculated as follows

$$\binom{m}{i} = \frac{m!}{i! \cdot (m-i)!} \quad (12)$$

The larger m the more interactions are predominantly investigated.

To investigate the effect of a factor on the lifetime of the insulation system is the primary goal of the lifetime model. Also the effect of the interaction of two factors on the lifetime is of interest. The influence of higher interactions is physically difficult to assess and often negligible compared to two-fold interactions or simple effects. Instead of these higher interactions additional single or double interactions could be investigated. In the case of fractional factorial designs not all measuring points are executed. Figure 4 shows the comparison of full factorial design and a fractional factorial design.

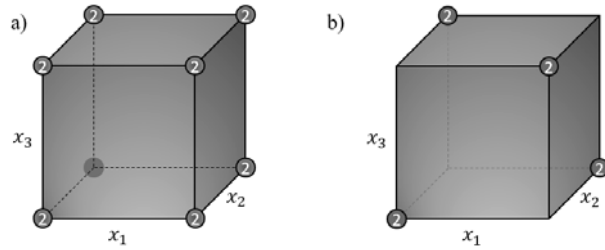


Figure 4: Comparison of a full factorial design a) and a fractional factorial design b).

This leads to a loss of information and to the fact that individual effects mix with each other. It is later not possible to distinguish which of these mixed effects is responsible for the aging of the insulation system. However, if simple factors and two-fold interactions are mixed exclusively with higher interactions whose effect on the lifetime is negligible, the question of how to distinguish between these effects is unnecessary. Table 3 lists and evaluates various fractional factorial designs.

4 Lifetime modeling

After different modelling methods were described in chapter 2 and the DoE was described in chapter 3, this chapter focuses on the development of the lifetime model based on the DoE.

4.1 Multiple regression

As a mathematical model to describe the insulation system, multiple regression is used. With this model it is possible to adapt the relationship between the influencing variables x_i and the lifetime as target variable y to the measured values. For the description a full factorial design with two measured values per measuring point is used, as shown in figure 4a. The formula for the calculation of the lifetime is given with the multiple regression as follows

$$y = z_0 + z_1x_1 + z_2x_2 + z_3x_3 + z_{12}x_{12} + z_{13}x_{13} + z_{23}x_{23} + z_{123}x_{123} \quad (13)$$

where z_i is the influence of factor x_i . The measured values required to parameterize equation (13) are designated c_i in table 4. In the experimental design in figure 4a, each factor has two levels, designated by the values -1 for the lower level and 1 for the higher level in table 4. For two measured values, the mean value could also be used instead of the expected value. If there are several measured values, the expected value should be used. The measured values in life cycle investigation are usually Weibull distributed.

Measuring points	Constant	x_1	x_2	x_3	x_{12}	x_{13}	x_{23}	x_{123}	Measured value 1	Measured value 2	Expected value
1	1	-1	-1	-1	1	1	1	-1	c_{11}	c_{12}	y_1
2	1	-1	-1	1	1	-1	-1	1	c_{21}	c_{22}	y_2
3	1	-1	1	-1	-1	1	-1	1	c_{31}	c_{32}	y_3
4	1	-1	1	1	-1	-1	1	-1	c_{41}	c_{42}	y_4
5	1	1	-1	-1	-1	-1	1	1	c_{51}	c_{52}	y_5
6	1	1	-1	1	-1	1	-1	-1	c_{61}	c_{62}	y_6
7	1	1	1	-1	1	-1	-1	-1	c_{71}	c_{72}	y_7
8	1	1	1	1	1	1	1	1	c_{81}	c_{82}	y_8

Table 4: Necessary number of tests for a certain accuracy when each factor has two levels.

To determine the influence of the factors z_i on the lifetime of the insulation system, equation (13) can be converted to matrix notation

$$\underline{Z} = \underline{X}^{-1} \cdot \underline{Y}_i \quad (14)$$

\underline{X} and \underline{Y}_i are shown in green and blue in table 4. The relation of the individual factors with the lifetime of the insulation system is not linear but can be described with the models from chapter two. These non-linear correlations must be transformed and used in the linear equation (13). In equation (15) this is represented for three factors which were described with the statistical approaches from chapter 2.2

Factors \ Measuring points	3	4	5	6	7	8	9	10	11	12	Resolution	Mix	Evaluation
4	2^{3-1} I										I	Factor and 2-fold	critical
8	2^3 complete	2^{4-1} II	2^{5-2} I	2^{6-3} I	2^{7-4} I						II	Factor and 3-fold 2-fold and 2-fold	critical
16		2^4 complete	2^{5-1} III	2^{6-2} II	2^{7-3} II	2^{8-4} II	2^{9-5} I	2^{10-6} I	2^{11-7} I	2^{12-8} I	III	Factor and 4-fold 2-fold and 3-fold	uncritical
32			2^5 complete	2^{6-1} IV	2^{7-2} II	2^{8-3} II	2^{9-4} II	2^{10-5} II	2^{11-6} II	2^{12-7} II	IV	higher resolution	uncritical
64				2^6 complete	2^{7-1} V	2^{8-2} III	2^{9-3} II	2^{10-4} II	2^{11-5} II	2^{12-6} II	V	higher resolution	uncritical
128					2^7 complete	2^{8-1} VI	2^{9-2} IV	2^{10-3} III	2^{11-4} III	2^{12-5} II	VI	higher resolution	uncritical

Table 3: Overview and evaluation of fractional factorial designs for the lifetime of the insulation system [8].

$$\begin{aligned}
\log y = & z_0 + z_1 \cdot e^{B_{Dakin} \cdot x_1} + z_2 \cdot \log x_2 + \\
& z_3 \cdot \log x_3 + z_{12} \cdot e^{B_{Dakin} \cdot x_1} \cdot \log x_2 + \\
& z_{13} \cdot e^{B_{Dakin} \cdot x_1} \cdot \log x_3 + \\
& z_{23} \cdot \log x_2 \cdot \log x_3 + \\
& z_{123} \cdot e^{B_{Dakin} \cdot x_1} \cdot \log x_2 \cdot \log x_3
\end{aligned} \quad (15)$$

The effect of the individual factors and the respective interaction, can be read off at the parameters z_i . A comparison of these parameters provides information about which effect contributes significantly to the aging of the insulation system and which effects are negligible with respect to service life. When designing robust insulation systems, special attention should be paid to the effects with a high contribution to aging.

4.2 Miner's rule

With the mathematical description of the insulation system by multiple regression and the efficient parameterization with the DoE, the lifetime of the insulation system at a certain load can be estimated. The lifetime model will now be extended to estimate the percentage of lifetime consumption due to a variable load spectrum. This is especially useful for applications with strong load fluctuations such as in the automotive industry.

Using the Miner's rule, the lifetime consumption of individual loads of a load spectrum can be calculated. It is assumed that the insulation system is loaded with a constant load for a short period of time. The duration of the constant load l_i is related to the total lifetime of the insulation system at the same load L_i . The sum of the individual loads result in the percentage lifetime consumption. The closer the sum of the individual loads approaches the value 1, the more lifetime is consumed and the more likely the insulation system is to fail. Miner's rule is given in equation (16)

$$\sum_{i=1}^n \frac{l_i}{L_i} = L_{\%} \quad (16)$$

where $L_{\%}$ is the percentage lifetime consumption.

5 Conclusion

In this paper a methodology for the lifetime calculation of the insulation system of electrical machines is presented. Besides the prediction of the lifetime, the effects of individual factors and the interactions considered in the modelling can be evaluated and compared. The model can be used in the design process of electrical machines or can be used as a virtual test bench, where the lifetime can be investigated at different loads. Figure 5 shows an overview of the entire methodology. The WLTP is shown as an example of the load spectrum, since the model can be used particularly well in the automotive industry.

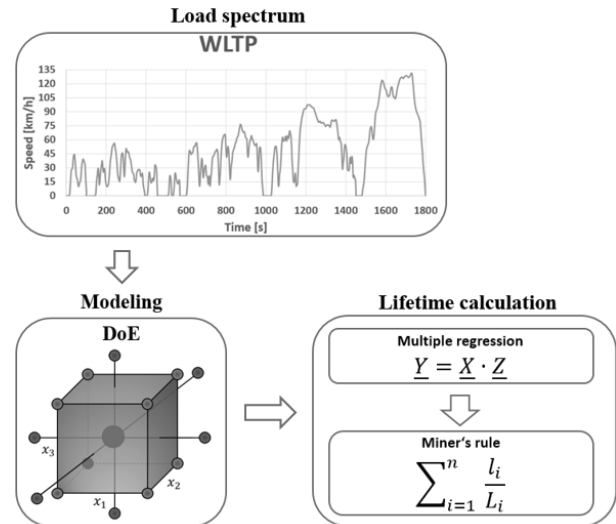


Figure 5: Overview of the modelling methodology.

Currently, durability tests are being carried out to apply the model to a real machine. Since the focus of this article

is on modelling, the results and aging effects will be published in a separate article.

References

- [1] A. H. Bonnett and C. Yung, "Increased Efficiency Versus Increased Reliability," in *IEEE Industry Applications Magazine*, vol. 14, no. 1, pp. 29-36, Jan.-Feb. 2008, doi: 10.1109/MIA.2007.909802.
- [2] R. Brüttsch, M. Tari, K. Fröhlich, T. Weiers and R. Vogelsang, "Insulation Failure Mechanisms of Power Generators [Feature Article]," in *IEEE Electrical Insulation Magazine*, vol. 24, no. 4, pp. 17-25, July-Aug. 2008, doi: 10.1109/MEI.2008.4581636.
- [3] A. C. Gjerde, "Multifactor ageing models - origin and similarities," in *IEEE Electrical Insulation Magazine*, vol. 13, no. 1, pp. 6-13, Jan.-Feb. 1997, doi: 10.1109/57.567392.
- [4] J. Artbauer, "Electric Strength of Polymers," *Journal of Physics D, Appl. Phys.*, Vol. 29, pp. 446-56, 1996.
- [5] J. -. Parpal, J. -. Crine and Chinh Dang, "Electrical aging of extruded dielectric cables. A physical model," in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 4, no. 2, pp. 197-209, April 1997, doi: 10.1109/94.595247.
- [6] G. Pietrini, D. Barater, F. Immovilli, A. Cavallini and G. Franceschini, "Multi-stress lifetime model of the winding insulation of electrical machines," *2017 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD)*, Nottingham, 2017, pp. 268-274, doi: 10.1109/WEMDCD.2017.7947758.
- [7] T. W. Dakin, "Electrical Insulation Deterioration Treated as a Chemical Rate Phenomenon," in *Transactions of the American Institute of Electrical Engineers*, vol. 67, no. 1, pp. 113-122, Jan. 1948, doi: 10.1109/T-AIEE.1948.5059649.
- [8] W. Kleppmann, "Versuchsplanung Produkte und Prozesse optimieren", 2013 Carl Hanser Verlag München Wien, ISBN, 978-3-446-43752-4.

Automatische Erstellung von digitalen Simulationszwillingen von Produktionssystemen

Walter Wincheringer, Tobias Sohny, Marec Kexel

Digitales Produktionslabor, Hochschule Koblenz, Konrad-Zuse-Straße 1, 56075 Koblenz, Germany;

dpl@hs-koblenz.de

Abstract. Die durch die Industrie 4.0 postulierte Mass Customization führt zu einer hohen Planungskomplexität einer Produktion, was den Einsatz von diskreten Simulationssystemen erfordert. Die Erstellung eines digitalen Simulationszwillinges ist jedoch zeitaufwändig, insbesondere, wenn sich in einer diskreten Werkstatt-, Matrixfertigung die Anforderungen häufig ändern. In diesem Beitrag wird ein Ansatz aufgezeigt, welcher die Generierung ablauffähiger Simulationsmodelle von Produktionssystemen, über eine VBA-Schnittstelle und ohne Programmierarbeit durch den Anwender, ermöglicht. Die Wahl verschiedener Prioritätsregeln zur Ablaufplanung, in Abhängigkeit typischer Produktionsziele, und deren Simulation wurden entwickelt. Somit sind Produktionsplaner in der Lage ihre Planung durch Simulation in kürzester Zeit abzusichern.

1. Einleitung

Durch die Globalisierung, kürzere Produktlebenszyklen sowie die Individualisierung der Kundenwünsche sind Unternehmen in ihrer Wettbewerbsfähigkeit gefordert. Um den Anforderungen gerecht zu werden, haben viele Unternehmen in den letzten Jahren ihr Produktionssystem nach Lean-Gesichtspunkten reorganisiert [1]. Mit Hilfe der Lean-Management-Methoden oder gemäß den Gestaltungsrichtlinien ganzheitlicher Produktionssysteme ist es gelungen die gestiegene Produktvielfalt zu beherrschen. Die Mass Customization schreitet jedoch weiter voran und bedingt, in Verbindung mit dynamischen Märkten, eine hohe Flexibilität und eine Wandelbarkeit des Produktionssystems. Dies führt zu einer zunehmenden Komplexität in der Produktionsplanung und -steuerung [2].

Das Konzept von Industrie 4.0 postuliert eine variantenreiche Produktion mit Losgröße Eins zu den Kosten einer Serienfertigung. Daher werden sich die klassischen Fertigungsprinzipien nach dem Flussprinzip, Reihen-

und Fließfertigungen, in Richtung einer Matrixproduktion wandeln. Hierzu werden flexible Produktionskapazitäten zukünftig nicht mehr mit Förderbändern statisch verknüpft, sondern über fahrerlose Transportsysteme dynamisch miteinander verbunden. Dadurch entstehen wandlungsfähige Produktionsabläufe mit unterschiedlichen Produktionstopologien die nach dem "plug and produce"-Konzept bedarfsgerecht, in Abhängigkeit der kundenspezifischen Auftragsstruktur, stets neu konfiguriert werden können. [2]

Auch im modernsten Produktionssystem müssen die klassischen Aufgaben der Produktionsplanung, der Kapazitätsauslastung sowie die Durchlaufterminierung, in Abhängigkeit vom jeweiligen Auftragsbestand, ausgeführt werden. Die Wandlungsfähigkeit der Produktion führt hierbei zu einer Planungskomplexität, die ohne den Einsatz digitaler Werkzeuge nicht beherrschbar ist.

Die ereignisdiskrete Simulation (discrete event simulation, DES) ist hierzu ein anerkanntes Werkzeug. Die Nachbildung existenter oder geplanter Systeme in ablauffähigen Simulationsmodellen, erlauben die Abbildung der bestehenden Dynamik [3].

Dieses Simulationsmodell, als digitaler Zwilling (Digital Twin, DT) der Produktionstopologie, muss mit der notwendigen Genauigkeit an das geplante oder bestehende Produktionssystem angepasst werden [4]. Dazu ist eine regelmäßige Anpassung des DT bzw. dessen Neuerstellung (bei Anpassung der Produktionsstruktur) erforderlich.

Der Aufwand zur Anpassung oder Neuerstellung von Simulationsmodellen ist jedoch zeitaufwendig, insbesondere, wenn verschiedene Produktionstopologien wöchentlich überprüft werden müssen [5]. Somit ist bei der Erstellung eines DT ein Ansatz zu wählen, welcher einen reduzierten Modellierungsaufwand begünstigt und eine Rekonfiguration des digitalen Abbilds zulässt.

2. Stand der Technik

Matrixfertigung. Mittels der Matrixfertigung lassen sich kundenindividuelle Produkte in der nötigen Flexibilität und Losgröße eins produzieren. Eine Matrixfertigung besteht u.a. aus einer Anzahl an Produktionsprozessen (Betriebsmittel), mit jeweils spezifischen Leistungsprofilen bzw. den Fähigkeiten, bestimmte Prozessschritte durchführen zu können. Auftragspezifische Arbeitsvorgänge können infolgedessen durch einen oder mehrere Produktionsprozesse bearbeitet werden. Der Transport zwischen den Ressourcen erfolgt auftragspezifisch und flexibel, unter Verwendung manueller oder autonomer Transportvorrichtungen, sog. fahrerlosen Transportsystemen (FTS). Der Maschinenbediener, als weitere wichtige Ressource eines Produktionssystems, wird bzgl. Qualität (Tätigkeitsprofil) und Quantität als ausreichend verfügbar angenommen [6].

Erste Beispiele sind bereits in der Roboterfertigung [7] oder auch in der PKW-Montage [8] realisiert oder befinden sich in der Planung [9].

FJSSP. Das sich daraus entwickelnde Produktionsablaufplanungsproblem, wird in der Literatur als „Flexible Job Shop Scheduling Problem“ bezeichnet [10]. Dies stellt, durch den komplexen Materialfluss, ein kombinatorisches Problem dar, dessen Optimum nicht durch analytische Berechnungen ermittelt werden kann. Die Anzahl der möglichen Reihenfolgen kann dabei, in Abhängigkeit der Ressourcenanzahl m , der Auftragsanzahl n und der Flexibilität, bis zu $(n!)^m$ mögliche Maschinenbelegungen ergeben. Dabei muss neben der Zielgröße *Termin-treue* auch die *Durchlaufzeit* (Bestände) und die *Betriebsmittelauslastung* berücksichtigt werden [11]. Die Generierung einer vermeintlich optimalen Maschinenbelegung führt innerhalb klassischer PPS Systeme zu einem hohen Planungs- und Rechenaufwand. Hierbei erfolgt die Terminierung der Ressourcennutzung häufig durch starre Verfahren bei fehlender Transparenz über die im Hintergrund laufende Prozesskoordination des PPS-Systems. Der Produktionsplaner bedient das PPS-System ohne die Möglichkeit einer Plausibilitätsüberprüfung der durch das System vorgeschlagenen optimalen Lösung, da hinterlegte Lösungsalgorithmen nicht bekannt sind oder deren Berechnung nicht beeinflusst werden kann [12].

Durch die hohe Relevanz wurden in den letzten Jahren hierzu mehrere Methoden entwickelt. Diese lassen

sich in exakte Bewertungsverfahren, welche aufgrund einer hohen Rechenleistung nicht weiter betrachtet werden, sowie heuristische Ansätze differenzieren (Abbildung 1) [13].

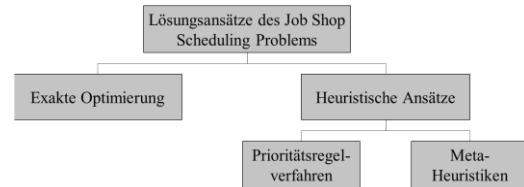


Abbildung 1: Lösungsansätze des Job Shop Scheduling Problems

Heuristische Ansätze stellen Näherungsverfahren dar, die zwar keine optimale Lösung garantieren, jedoch bei begrenzter Rechenleistung gute und zulässige Resultate ermöglichen. Hierzu zählen die sogenannten *Prioritätsregelverfahren* [14]. Die Meta-Heuristiken sind iterative Verfahren, welche ausgehend von einer Startsituation schrittweise Optimierungen berechnen, unter Berücksichtigung von Lerneffekten, was einen nicht unerheblichen Aufwand zur Programmierung bedingt [13].

Der Einsatz von Prioritätsregeln eignet sich somit aufgrund des geringen Rechenaufwands, einer Komplexitätsunabhängigkeit sowie einer einfachen Implementierung und schnellen Anpassung.

Zu den Prioritätsregeln zählen u.a.:

- FIFO (First in First Out)
- SPT (Shortest Processing Time)
- EDD (Earliest Due Date)
- LWKR (Least Work Remaining)

Im Kontext der PPS bedingt dies eine sinnvolle Zuordnung der Arbeitsgänge zu den entsprechenden Ressourcen, als auch eine sinnvolle Abarbeitungsreihenfolge, mit dem Ziel eine möglichst geringe Durchlaufzeit (DLZ) und eine hohe Auslastung der Betriebsmittel (BM), zu realisieren.

Mit Hilfe der Prioritätsregelverfahren kann eine Lösung, jedoch keine optimale Lösung, für das Planungsproblem erstellt werden. Hinzu kommt, dass unterschiedliche Prioritätsregeln und unterschiedliche Kombinationen zur Lösung eingesetzt werden können. Ob die Lösung jedoch die Ziele der Produktionsplanung (u.a. Termintreue, Kapazitätsauslastung) tatsächlich erfüllt, ist erst nach Ablauf der Produktion – im Nachhinein – fest-

stellbar. Darüber hinaus ist die Produktion durch unterschiedliche Ereignisse, wie z.B. Eilaufträge, Qualitätsprobleme, technische oder organisatorische Störungen, so dynamisch, dass es häufig zu erheblichen Abweichungen vom Planungsergebnis kommt [15]. Um die dynamischen Ereignisse über die Zeit in Produktionsprozessen zu berücksichtigen, hat sich die diskrete Event-Simulation bewährt [13].

Simulation. In Produktion und Logistik wird Simulation definiert als das "Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind; insbesondere werden die Prozesse über die Zeit entwickelt" [3]. Diese ausführbaren Modelle (Simulationsmodelle) ermöglichen eine Analyse des Systemverhaltens nach Parameter- oder Strukturänderungen.

Für die Erstellung von Simulationsmodellen ist ein genaues Verständnis des Systems sowie die Programmierung des Simulationsmodells erforderlich. Dies ist zeitaufwändig und ressourcenintensiv.

Bzgl. einer Matrixfertigung ist die vom Produktionsplaner getroffene Produktionstopologie als Simulationsmodell des Produktionssystems zu erstellen, um die, in Abhängigkeit der gewählten Prioritätsregeln, entstehende Ablaufplanung abzusichern. Bedingt dadurch, dass die Produktionstopologie häufig angepasst werden muss, ist eine stetige Neuerstellung eines Modells zu aufwendig und wirtschaftlich nicht tragbar.

Eine aufwandsarme Simulationsmodellerstellung, sowie die Wiederverwendung bereits erstellter Modelle, verspricht die Generierung. Dieser Ansatz gliedert sich in drei Aufgabenbereiche: die *Generierung*, die *Initialisierung* und die *Adaption* des Simulationsmodells [4].

Generierung. Die Generierung beschreibt die eigentliche Erzeugung des Simulationsmodells inklusive der Hinterlegung entsprechender Parameter (Bsp.: produktspezifische Zyklus-, Rüstzeiten, Steuerungsregeln, etc.) [4].

Initialisierung. Im Anschluss an die Generierung ist eine Initialisierung des Simulationsmodells nötig, um eine entsprechende Ausgangssituationen zum Start der Simulation festzulegen (Bsp.: Bearbeitungsstände, Betriebsmittelzustände, etc.) [16].

Adaption. Die Adaption beinhaltet das Ändern des generierten Simulationsmodells, ohne eine vollständige Neugenerierung des Simulationsmodells vorzunehmen (Bsp.: Veränderung der Betriebsmittelkapazitäten, der Losgrößen, etc.) [17].

Die genannten Aufgabenbereiche ermöglichen somit das automatische Erstellen und Anpassen eines Simulationsmodells für den Produktionsablauf, in Abhängigkeit der vom Produktionsplaner gewählten Produktionstopologie, inkl. einer produktspezifischen Parametrierung der Betriebsmittel, die Zuordnung von Logiken sowie die Initialisierung einer Ausgangssituation der Produktion. [4].

3. Umsetzung

Für die Realisierung eines Simulationsmodell-Generators, der eine automatische Erstellung eines Produktionssimulationsmodells ermöglicht, sind zunächst die Produktionsprozesse (Betriebsmittel), Materialpuffer, Transportvorrichtungen, FTS, etc. und die Prioritätsregeln als Module abzubilden.

Die einzelnen Module repräsentieren hierbei einzelne abgeschlossene und parametrierbare Teilsysteme, welche eine Mehrfachverwendung sowie deren Aufruf - in der Simulationsumgebung - ermöglichen. Zudem lassen sich die einzelnen Module an den erforderlichen Schnittstellen miteinander verknüpfen.

Die einzelnen Betriebsmittel (Ressourcen) sind in der Simulatorumgebung WITNESS bereits als einzelne Module vorhanden und können für den Ansatz der Generierung verwendet werden. Für eine hinreichend genaue Abbildung bedarf es der Angaben zur technischen und organisatorischen Verfügbarkeit, Qualitätsraten und Rüstzeiten, inklusive ihrer jeweils spezifischen Verteilungen. Diese erforderlichen Daten lassen sich mittels einer VBA-Schnittstelle durch eine vereinfachte Eingabeoberfläche hinterlegen (Abbildung 2).

Abbildung 2: Ausschnitt aus der Anwenderoberfläche

Daten über die Verfügbarkeit von Werkzeugen, Rohmaterial und Mitarbeiter werden im Simulationsmodell als vorhanden angenommen. Als Datenquelle können historische Produktions- und Auftragsdaten, z.B. aus einem MES, MDE, BDE, entnommen, aufbereitet und zur Anwendung kommen.

Die einzelnen Betriebsmittel lassen sich mittels dieser Schnittstelle parametrisieren. Dem Anwender wird deren Anordnung im Fabriklayout mittels eines Koordinatensystems ermöglicht, sodass ein visueller Bezug zum Realsystem besteht (Abbildung 3).

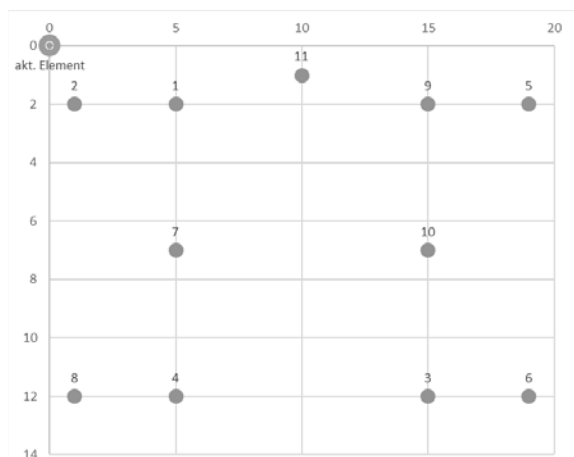


Abbildung 3: Vereinfachte Layout-Darstellung in der Anwenderoberfläche

Mittels einer Transportmatrix, welche die quellen- und senkenspezifische Distanz und die damit einhergehende Transportzeit inkl. der Be- und Entladezeit berücksichtigt, ist die Möglichkeit den Materialfluss mit realitätsnahen Prozesszeiten abzubilden, gegeben.

Zur Realisierung dieses Materialflusses lassen sich die simulatorspezifischen Regeln (Push/Pull) verwenden. Diese erlauben es einfache Flüsse, von Quellen zu Senken, inklusive der entsprechenden Steuerungslogik abzubilden. Zur Abbildung einer Matrixfertigung ist die Verknüpfung des Materialflusses allerdings von den Aufträgen, deren spezifischen Arbeitsvorgängen und den geplanten Abläufen, welche mit Hilfe der gewählten Prioritätsregeln geplant werden, abhängig. Hierzu sind alle Betriebsmittel-Module, inkl. Puffer-Module, mittels FTS automatisch untereinander verknüpft, die logische Reihenfolge wird mittels Prioritätsregel-Module realisiert.

Die jeweilige Zykluszeit der Produktionsprozesse ist spezifisch, je nachdem um welche Maschine oder Auftrag bzw. Arbeitsvorgang es sich handelt, zu hinterlegen.

Hierzu kann mittels der VBA-Schnittstelle die Anzahl der Betriebsmittel definiert werden. Darüber hinaus erlaubt diese den fertigungsspezifischen Ablauf eines Auftrags inkl. der Betriebsmittel zu hinterlegen (Tabelle 1)

Arbeitsplan Fallbeispiel								
j	M1	M2	M3	M4	M5	M6	M7	M8
1	1	1	3	3	2	4	2	5
2	1	4	3	5	2	2		6
3	1	4	3	5	2	7	2	6
4	1	1	3	3	2	4	2	4
5	3	1	3	1	2	4	2	4
j = Auftrag M = Maschine								

Tabelle 1: Exemplarischer Arbeitsplan

Dieser fertigungsspezifische Ablauf ist dem jeweiligen Auftrag j mittels Attribute zugeordnet, sodass alle Abläufe in der richtigen Reihenfolge berücksichtigt werden. Die Zuordnung zu den jeweiligen Ressourcen (M1-8) sowie der Anordnung in deren Warteschlangen ist abhängig von den gewählten Prioritätsregeln.

Die Prioritätsregeln wurden simulatorspezifisch in einzelne Logik-Module codiert. Dies bedingt, einen einmaligen Codieraufwand. Eine Kombination, Mehrfachverwendung sowie eine Erweiterung der Prioritätsregeln werden hierdurch ermöglicht. Die jeweilige Anzahl der Pufferplätze vor den jeweiligen Betriebsmitteln, lassen sich zudem individuell einstellen. Gleiches gilt für die Anzahl der FTS und ihrer Transportkapazität.

Die Abarbeitung der Transportaufträge erfolgt hierbei nach dem FIFO-Prinzip. Ein Transportauftrag wird beim Start des Fertigungsauftrags, für den erster Arbeitsvorgang, sowie nach Fertigstellung am jeweiligen Produktionsprozess generiert.

Alle Daten werden in einem für den Generator geeigneten Format gespeichert. Beim Start der Simulation, wird automatisch ein lauffähiges Simulationsmodell, anhand der hinterlegten Daten, generiert.

Unmittelbar nach der Generierung des Simulationsmodells, wird eine Initialisierungsdatei geladen. Die dort hinterlegten Informationen aus der VBA-Schnittstelle, erlauben es die aktuelle Situation der Produktion abzubilden. So lassen sich beispielweise die Bearbeitungsfortschritte von Restaufträgen, inkl. der vorhandenen Bestände in der Produktion (WIP), hinterlegen. Auch diese Daten können aus einem MES oder BDE-System übernommen werden. Nach der Initialisierung startet der Simulationslauf (Abbildung 4).

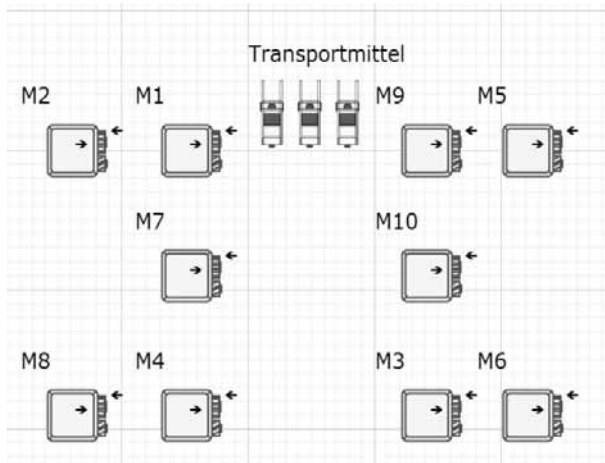


Abbildung 4: Generiertes und lauffähiges Simulationsmodell

Eine anschließende Adaption des generierten Simulationsmodells lässt sich effizient in der VBA-Schnittstelle einpflegen, wodurch Betriebsmittel hinzugefügt oder geändert werden können und die Realität – abhängig von der aktuellen Planungssituation – hinreichend genau abgebildet werden kann.

Für den Fall, dass die KPIs, z.B. Termintreue, Durchlaufzeit, nach einem Simulationslauf nicht den Anforderungen entsprechen, lassen sich unterschiedliche Prioritätsregeln per Anwahl austauschen. Dies ermöglicht eine erneue Ablaufplanung mit anschließender Überprüfung der Planungsergebnisse am Simulationsmodell. Zusätzlich lassen sich die unterschiedlichen Planungsergebnisse realitätsnah bewerten/vergleichen, ohne dass ein Programmieraufwand entsteht.

4. Evaluierung der Ergebnisse

Mittels der Simulation können die unterschiedlichen Prioritätsregelverfahren miteinander verglichen werden. Hierzu sind die Zielgrößen mittels einheitlicher Kennzahlen zu quantifizieren, um so Verbesserungsansätze zu vergleichen und die jeweilige Regel zu evaluieren.

Im Kontext der Produktionsplanung liegt der Schwerpunkt hierzu auf einer optimalen Termintreue. Die Termineinhaltung lässt sich u.a. durch nachstehende Zielgrößen messen: [18]

Termintreue (%)

- Pünktliche Aufträge / Gesamtanzahl an Aufträgen
- Ziel → Maximierung

Gesamtverspätung (Zeiteinheiten, ZE)

- Ziel → Minimierung

Durchschnittliche Verspätung pro Auftrag (ZE/Stk.)

- Gesamtverspätung / Anzahl verspäteter Aufträge
- Ziel → Minimierung

Maximale Terminabweichung je Auftrag (ZE/Stk.)

- Ziel → Minimierung

Für die Eruierung bestehender Wirkungszusammenhänge (z.B. Termintreue vs. Kapazitätsauslastung) sind zudem die Auslastung der Maschinen- und Transportkapazitäten über die Zeit (Abbildung 5) sowie die Durchlaufzeit der Aufträge zu betrachten. Eine absolute auf eine Periode bezogene Auslastung gibt keinen ausreichenden Aufschluss. Die Auslastungsanalysen wurden im Simulationssystem programmiert und werden anwendungsspezifisch mittels einer dynamischen Animation dem Produktionsplaner visualisiert.



Abbildung 5: Transportmittel-Auslastung über die Zeit

Bedingt durch die Vielzahl an Alternativen einer Matrixfertigung, ist die ausschließliche Anwendung eines PPS-Systems nicht ausreichend [15]. Darüber hinaus muss das Simulationsmodell als DT eines Produktionssystems mit jeder Planungsperiode an die veränderten Ausgangsbedingungen (Auftragsbestand, Terminwünsche, BM-Kapazität, etc.) anpasst werden. Eine aufwandsarme Erstellung des Simulationsmodells wird mittels der Generierung ermöglicht.

Die „Was-Wäre-Wenn-Szenarien“ erlauben hierbei die Abbildung und Untersuchung von Interdependenzen. So können die Anzahl an Betriebsmitteln, deren Anordnung, eine mengenmäßige Veränderung der Fördersysteme sowie deren Kapazitäten angepasst werden. Dies erlaubt die Darstellung der Zielgrößen in Abhängigkeit einer festzulegenden Prioritätsregel-Kombination und zeigt beispielsweise den Zusammenhang zwischen einer Transportkapazität und der Termintreue auf, welche sich ohne Simulation nicht abbilden lassen.

Wenn die Zielgrößen nicht erreicht wurden, kann eine Neuplanung der Produktion erfolgen. Hierzu können im ersten Schritt die Prioritätsregeln angepasst und ein neuer

Produktionsablauf, in Abhängigkeit der Produktionstopologie, geplant werden. Diese Planung kann erneut im Simulationsmodell abgesichert werden.

Darüber hinaus bietet das Simulationssystem die Möglichkeit mittels des sogenannten Experimentierers einen Planungsablauf zu untersuchen, bis die vorab festgelegten Zielgrößen mit geringster Abweichung erreicht werden (How-to-achieve). Hierbei werden alle Alternativen, welche sich durch das vorhandene Portfolio an hinterlegten Prioritätsregeln ergeben, automatisch durch das System abgebildet. Die benötigte Zeit für manuelle Simulationsversuche durch den Produktionsplaner entfallen hierbei.

Verschiedene Zielgrößen wie die DLZ, die Anzahl an verspäteten Aufträgen, der Zeitanteil der durchschnittlichen Verspätung sowie die min. und max. Verspätung werden betrachtet. Die Ergebnisse lassen sich via Schnittstelle (Excel) exportieren und innerhalb einer Graphik abbilden (Abbildung 6).

Dies bietet eine höchstmögliche Transparenz und erlaubt dem Produktionsplaner eine Evaluierung, ob zusätzliche BM-Kapazitäten oder FTS-Kapazitäten eine Verbesserung bringen.

Das erstellte System bietet somit eine schnelle Bestimmung der effizientesten Prioritätsregeln, in Abhängigkeit der jeweiligen Zielgrößen, für eine gegebene Auftragsituation.

5. Zusammenfassung und Ausblick

Die automatische Generierung von Simulationsmodellen einer Matrixproduktion erlauben es den Produktionsplaner situationsabhängig ihre Produktionsplanung aufwandsarm abzusichern.

Mittels vorgefertigter Module (Betriebsmittel, Puffer, FTS) sowie unterschiedlicher Prioritätsregeln lassen sich diese mittels einer VBA Schnittstelle parametrisieren, anordnen und erlauben anschließend die Generierung von ablauffähigen Simulationsmodellen ohne Programmierung.

An dem generierten und lauffähigen Simulationsmodell können die dynamischen Produktionseinflüsse untersucht und entsprechende Optimierungsalternativen, anhand von produktionsspezifischen KPIs, evaluiert werden. Darüber hinaus bietet das System die Option alle Ablaufalternativen, die durch unterschiedliche Prioritätsregeln möglich sind, automatisch in einem Simulationslauf zu betrachten und die Zielgrößen graphisch aufbereitet abzubilden. Dies ermöglicht eine effiziente Auswahl der geeignetsten Prioritätsregeln, für die aktuelle Situation in der Produktion.

In einer Weiterentwicklung wird die Verknüpfung mit einem MES-System angestrebt, sodass anlagen- und

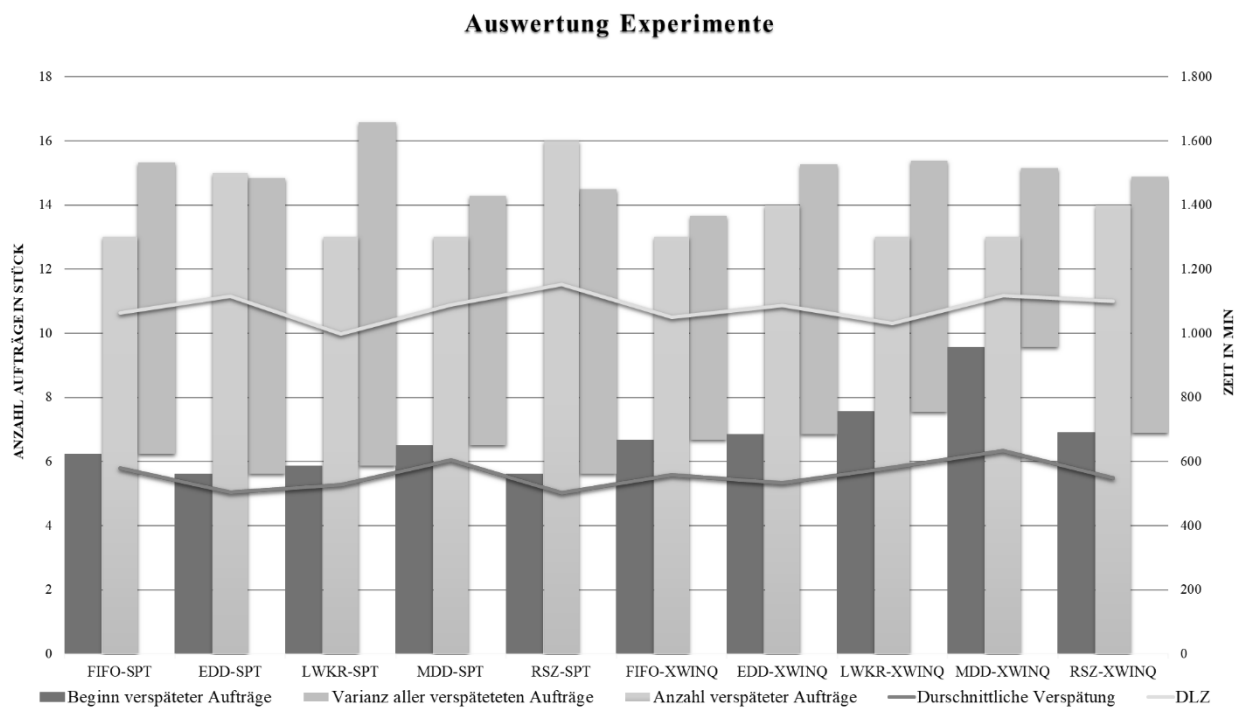


Abbildung 6: Ergebnissdarstellung in Excel

produktspezifische Kenngrößen bzgl. dem Ausfallverhalten, Rüstzeiten, Qualitätskennzahlen, etc., sowie die damit verbundenen Einflüsse, übernommen und realitätsnah abgebildet werden können. Eine manuelle Parametrierung via VBA-Schnittstelle kann somit weitgehend entfallen.

Darüber hinaus erfolgt eine Erprobung im industriellen Umfeld, um die Funktionalität abzusichern sowie die Überprüfung weiterer Prioritätsregeln zu ermöglichen.

References

- [1] Staufen AG, Institut PTW der technischen Universität Darmstadt. *25 Jahre Lean Management*. Köngen: Staufen AG; 2016.
- [2] Bauernhansl T, ten Hompel M, Vogel-Heuser B. *Industrie 4.0 in Produktion, Automatisierung und Logistik*. Wiesbaden: Springer; 2014.
- [3] VDI-Richtlinie 3633: *Simulation von Logistik-, Materialfluss- und Produktionssystemen, Grundlagen - Blatt 1*. Düsseldorf: Beuth; 2014.
- [4] Bergmann S, Starßburger S, Schulze T. *Automatische Generierung adaptiver Modelle zur Simulation von Produktionssystemen*. Ilmenau: Univ.-Verl. Ilmenau; 2013.
- [5] Baier J, Krieg R. Automatisierter Modellaufbau für Materialflusssimulation in der Nutzfahrzeugproduktion. In *Advances in Simulation for Production and Logistics Applications*, Stuttgart: Fraunhofer IRB Verlag; 2008. pp. 51-60.
- [6] Greschke P, Schönemann M, Thiede S, Herrmann C. Matrix Structures for High Volumes and Flexibility in Production Systems. In *Variety Management in Manufacturing. Proceedings of the 47th CIRP Conference on Manufacturing Systems*. Procedia CIRP; 2014. pp. 160-165.
- [7] KUKA AG. *Matrix-Produktion live im KUKA SmartProduction Center*. 21 Februar 2018. [Online]. Available: <https://www.kuka.com/de-de/presse/news/2018/02/smartproductioncenter>. [Zugriff am 11. September 2020].
- [8] AUDI AG. *Die Modulare Montage*, 17 November 2016. [Online]. Available: <https://www.audi-mediacycenter.com/de/audi-techday-smart-factory-7076/die-modulare-montage-7078>. [Zugriff am 11. September 2020].
- [9] Popp J, und Wehking K.H. Neuartige Produktionslogistik für eine wandelbare und flexible Automobilproduktion. *Logistics Journal: Proceedings*; 2016.
- [10] Zhang T, Xie S, Rose O. Flexible Job-shop Scheduling with Dynamic Stochastic Machine Sets. In *Simulation in Production and Logistics*. Dortmund: Fraunhofer Verlag; 2015. pp. 21-28.
- [11] Jaehn F, Pesch E. *Ablaufplanung. Einführung in Scheduling*. Berlin: Springer Verlag; 2014.
- [12] Hees A. F. *System zur Produktionsplanung für rekonfigurierbare Produktionssysteme*. München: Technische Universität München; 2017.
- [13] Gutenschwager, Rabe M, Spieckermann S, Wenzel S. *Simulation in Produktion und Logistik*. Berlin: Springer; 2017.
- [14] Briskorn D, Hartmann S. Anwendungen des Resource-Constrained Project Scheduling Problem in der Produktionsplanung. In *Produktionsplanung und -steuerung*. Berlin: Springer; 2015. pp. 109-129.
- [15] Niehues M. R. *Adaptive Produktionssteuerung für Werkstattfertigungssysteme durch fertigungsbegleitende Reihenfolgebildung*. München: Technische Universität München; 2016.
- [16] Hotz I. *Simulationsbasierte Frühwarnsysteme zur Unterstützung der operativen Produktionssteuerung und -planung in der Automobilindustrie*. Magdeburg: Universität Magdeburg; 2007.
- [17] Bergmann S. Automatische Generierung adaptiver und lernfähiger Modelle zur Simulation von Produktionssystemen. In *Tagungsband zum Doctoral Consortium der WI*. Bayreuth: Universität Bayreuth; 2011. pp. 9-16.
- [18] Kletti J, Schumacher J. *Die perfekte Produktion. Manufacturing Excellence durch Short Interval Technology (SIT)*. Berlin: Springer; 2014.

Ein simulationsbasiertes Optimierungssystem zur Priorisierung von Maschinenstillständen unter Einbeziehung eines Lookahead

Michael Hegemann^{1*}, Stefan Nickel²

¹Mercedes-Benz AG, Mercedesstraße 120, 70372 Stuttgart, Deutschland; *michael.hegemann@daimler.com

²Institut für Operations Research, Diskrete Optimierung und Logistik, Karlsruher Institut für Technologie (KIT), Karlsruhe, Deutschland

Abstract. This paper discusses the added value of prioritizing machine breakdowns, taking into account future planned downtime. For this purpose, an online-optimization problem with lookahead is formulated, whereby the information about planned downtimes is available as lookahead. To solve the problem a simulation-based optimization system is presented. By implementing this system in a simulation environment and performing a simulation study of a multidimensional flow production system it is shown that a prioritization with lookahead can reduce the occurrence of momentary bottlenecks due to planned downtimes, so that the overall performance of a production system can be increased.

1 Einleitung

Während der geplanten Betriebszeit eintretende Maschinenstillstände, resultierend beispielsweise aus technischen Störungen oder Wartungsmaßnahmen, führen bei hoch ausgelasteten Produktionssystemen zu hohen Produktionsverlusten [1]. Dabei besteht die Möglichkeit, dass zeitgleich Stillstände an mehreren Produktionsressourcen eines Produktionssystems anliegen. Da in der Praxis die Anzahl der Produktionsmitarbeiter, die zur Stillstandsbehebung zur Verfügung stehen, oftmals begrenzt ist, können durch eine objektive Priorisierung der Stillstände nach ihrer jeweiligen Auswirkung auf das Produktionssystem die Reaktionszeiten auf die schwerwiegendsten Stillstände reduziert und Produktionsverluste nachhaltig minimiert werden [2].

Neben den zu priorisierenden ungeplanten Stillständen können zum Zeitpunkt der Priorisierung auch Informationen über zukünftig eintretende geplante Stillstandszeiten vorliegen. Diese Stillstandszeiten resultieren beispielsweise aus anstehenden präventiven Instandhaltungsmaßnahmen. Zusätzliche Informationen über geplante Stillstandszeiten ergeben sich aus der Anwendung von Predictive Maintenance bzw. einer zustandsorientierten, vorausschauenden Instandhaltungsstrategie [3].

In diesem Kontext konnte bereits für verkettete Fließfertigungssysteme gezeigt werden, dass eine Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten zu einer anderen Priorisierungsreihenfolge führen kann [4]. In diesem Beitrag wird darauf aufbauend ein operatives Entscheidungsunterstützungssystem vorgestellt, welches den Produktionsmitarbeitern eine objektiv ermittelte Priorisierungsreihenfolge der zum Entscheidungszeitpunkt anliegenden Stillstände zur Verfügung stellt. Bei der Ermittlung der Priorisierungsreihenfolge wird dabei nicht nur die Auswirkung der in diesem Moment anliegenden Stillstände berücksichtigt, sondern es werden auch zukünftig eintretende geplante Stillstandszeiten in Form eines Lookahead als zusätzliche Information in die Entscheidungsfindung mit einbezogen. Damit soll erreicht werden, dass auch die Auswirkungen geplanter Stillstände auf die Produktionsleistung eines Fließfertigungssystems minimiert und so Produktionsverluste nachhaltig reduziert werden.

Nach einem kurzen Überblick über den aktuellen Stand der Technik wird eine simulationsbasierte Vorgehensweise vorgestellt, mit der die Priorisierung von Maschinenstillständen unter Berücksichtigung eines Lookahead möglich ist. Ferner wird die Implementierung dieser Vorgehensweise im Rahmen eines Entscheidungsunterstützungssystems in eine Simulationsumgebung skizziert. Abschließend werden die Ergebnisse einer durchgeführten Simulationsstudie zur Untersuchung des erzielbaren Mehrwerts einer Priorisierung mit Lookahead dargestellt.

2 Stand der Technik

In der Praxis wird für die Priorisierung ungeplanter Stillstände häufig ein subjektives Verfahren, basierend auf Expertenerfahrung, Expertenwissen oder Intuition, angewendet [5].

Bei wissenschaftlichen Ansätzen kommen hingegen für die Ermittlung einer Priorisierungsreihenfolge vor allem statische Verfahren, bei denen die Priorisierung anhand einer zuvor festgelegten Reihenfolge erfolgt, oder dynamische Verfahren, die überwiegend auf der Ermittlung eines Engpassrankings basieren, zur Anwendung. So werden in [6] und [7] zwei statische Bewertungsschemata vorgestellt, anhand derer Maschinen entsprechend ihrer Kritikalität für das Produktionssystem priorisiert werden können. Weitere statische Verfahren basieren auf der Bestimmung eines Engpassrankings. Die Priorisierung erfolgt dabei mittels eines bereits vor dem Entscheidungszeitpunkt feststehenden Rankings. Zur Engpassermittlung werden Methoden wie die „Active Period Method“ [8] [9], die „Shifting Bottleneck Detection Method“ [10], die „Arrow Based Method“ [11] sowie die „Turning Point Method“ [12] [13] angewendet.

Im Gegensatz zu statischen Verfahren wird bei dynamischen Verfahren die Priorisierungsreihenfolge jeweils zum Entscheidungszeitpunkt auf Grundlage des aktuellen Systemzustandes neu bestimmt. In [14] und [15] wurde dazu eine simulationsbasierte Priorisierungsmethode hergeleitet, mit der basierend auf Online-Produktionsinformationen eine Priorisierungsreihenfolge aktuell anliegender Stillstände zum jeweiligen Entscheidungszeitpunkt ermittelt werden kann. Im Gegensatz dazu wird in [16] eine analytische Engpassermittlungsmethode, die „Shifting Bottleneck Detection Method“, verwendet, um auf Basis von Echtzeitproduktionsdaten die momentanen Engpässe zu bestimmen und anhand derer eine Priorisierung von Maschinenstillständen vorzunehmen. In [2] wurde zudem eine Priorisierungsmethode hergeleitet, die auf der Bestimmung von Echtzeit-Engpässen und Engpässen in naher Zukunft beruht.

Die zur Bestimmung der Priorisierungsreihenfolge verwendeten Daten beziehen sich dabei häufig auf einen historischen Zeitraum oder aber auf den aktuellen Zustand des Produktionssystems. Einen Schritt weiter wird in [17] gegangen, indem mittels historischer Produktionsdaten die Engpässe der nächsten Zeitperiode prognostiziert werden, wobei hier bereits angemerkt wird, dass diese prognostizierten Engpässe letztendlich auch in die Priorisierungsentscheidung mit einfließen sollten.

Zusammenfassend zeigt die Analyse wissenschaftlicher Ansätze, dass bisher kein Ansatz existiert, der zum Entscheidungszeitpunkt verfügbare Informationen über zukünftig eintretende geplante Stillstandszeiten konkret in die Priorisierungsentscheidung ungeplanter Maschinenstillstände miteinbezieht.

3 Priorisierung mit Lookahead

Im Folgenden wird eine simulationsbasierte Vorgehensweise vorgestellt, mit der die Ermittlung einer Priorisierungsreihenfolge unter Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten möglich ist.

Der Ausgangspunkt ist ein Entscheidungszeitpunkt t_d , an dem ein Produktionsmitarbeiter entscheiden muss, welcher der zu diesem Zeitpunkt anliegenden Stillstände als Nächstes instandgesetzt wird. Damit eine Priorisierung überhaupt notwendig ist, muss dabei die Anzahl zu priorisierender Stillstände größer als eins sein. Ist die Anzahl eins, so liegt nur ein Stillstand an und der Mitarbeiter kann direkt zugeordnet werden.

Liegen mehrere Stillstände an, wird eine Priorisierungsreihenfolge bestimmt. Um die Priorisierungsreihenfolge ungeplanter Stillstände unter Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten zu ermitteln, können die zu priorisierenden Stillstände unter der Annahme, dass zeitlich parallel anliegende Stillstände nur sequentiell behoben werden, in ein Reihenfolgenoptimierungsproblem transformiert werden [14]. Die Formulierung dieses Optimierungsproblems ergibt sich mit der zu maximierenden Zielgröße Durchsatzmenge der Engpassprozessstufe wie folgt:

$$\text{Max} \rightarrow \text{Durchsatzmenge} = \max_{k=1}^L D(\text{Seq}_k) \quad (1)$$

Dabei sind L die Anzahl möglicher Priorisierungsreihenfolgen und $D(\text{Seq}_k)$ die mit der Reihenfolge k zu einem Vergleichszeitpunkt T_{Ende} erzielbare Durchsatzmenge der statischen Engpassprozessstufe. Da der Engpass die Ausbringung des gesamten Produktionssystems beeinflusst, wirken sich die unterschiedlichen Behebungsreihenfolgen ebenfalls auf die Durchsatzmenge des Engpasses aus. Als statische Engpassprozessstufe für das Abbild eines Fließfertigungssystems zum Zeitpunkt t_d wird hier die Prozessstufe mit der größten Taktzeit definiert. Diese limitiert die Ausbringung des gesamten Fließfertigungssystems, solange keine Stillstände anliegen, die zu einem Wartezustand der statischen Engpassprozessstufe aufgrund eines Werkstückmangels am Einlauf oder eines blockierten Auslaufs führen. Sollten mehrere Prozessstufen dieselbe Taktzeit aufweisen, so ist die am weitesten stromabwärtsgelegene Stufe die Engpassprozessstufe. Um die Engpassprozessstufe auch für mehrdimensionale Fließfertigungssysteme zu ermitteln, kann mit folgender Formel eine repräsentative Taktzeit für die insgesamt n redundant angeordneten Prozesse einer redundanten Prozessstufe, die in der Praxis häufig

auch als Parallelmaschine bezeichnet wird, berechnet werden [2]:

$$TZ_{rep} = \left[\sum_{i=1}^n \frac{1}{TZ_i} \right]^{-1} \quad (2)$$

Da sich die Zielfunktion für komplexe Fließfertigungssysteme meistens nicht in geschlossener Form darstellen lässt, wird die ereignisdiskrete Simulation zur Bestimmung des Zielfunktionswertes eingesetzt. Dieses simulationsbasierte Optimierungssystem, bei dem die dominierende Komponente die Optimierung ist, welche auf das Simulationsmodell als Zielfunktionswert zurückgreift und als Ergebnis die erzielbare Durchsatzmenge der untersuchten Behebungsreihenfolge zurückgegeben bekommt, ist in Abbildung 1 dargestellt.

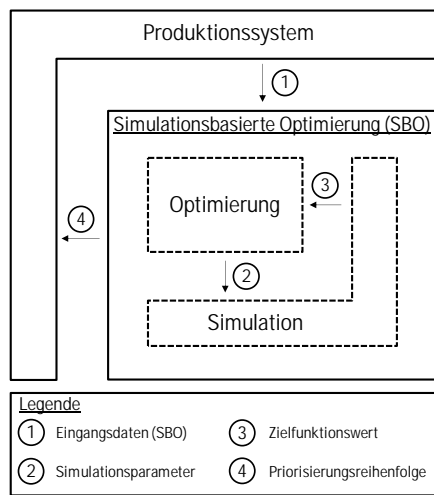


Abbildung 1: Schematische Darstellung des SBO-Systems

Um zukünftige geplante Stillstandszeiten bei der Priorisierung von Maschinenstillständen zu berücksichtigen, wird das vorliegende Optimierungsproblem als Online-Optimierungsproblem mit Lookahead aufgefasst. Im Gegensatz zur Offline-Optimierung, bei der zu Beginn alle Eingabedaten bekannt sind, werden bei der Online-Optimierung die Eingabedaten sequentiell bekannt gegeben, d. h. bei einer Online-Optimierung müssen Entscheidungen unmittelbar und ohne Wissen über zukünftige Ereignisse getroffen werden [18]. Bei einer Online-Optimierung mit Lookahead ist nun eine Teilmenge der zukünftigen Eingabedaten bekannt, sodass mehr Informationen für die Entscheidung bereitstehen. Somit kann die Online-Optimierung mit Lookahead zwischen den Extrema der klassischen Offline-Optimierung und der reinen Online-Optimierung eingeordnet werden [19]. In diesem Kontext können geplante Stillstandszeiten, die zum Zeitpunkt der Priorisierung bereits bekannt sind, als

Lookahead klassifiziert werden. Dadurch ist es möglich, die im Lookahead enthaltenen geplanten Stillstandszeiten in den Simulationsläufen zur Ermittlung der Zielfunktionswerte der potenziellen Priorisierungsreihenfolgen zu berücksichtigen und so letztendlich eine Priorisierung unter Einbeziehung zukünftig eintretender geplanter Stillstände zu ermöglichen.

Der Vergleichszeitpunkt, der im Folgenden als Simulationsendzeitpunkt T_{Ende} bezeichnet wird und zu dem die Ergebnisgrößen für die Bewertung der Behebungsreihenfolgen bestimmt werden, ist dadurch charakterisiert, dass sich die Auswirkungen der im Lookahead enthaltenen Stillstandszeiten vollständig in der Durchsatzmenge der statischen Engpassprozessstufe realisiert haben. Um den Simulationsendzeitpunkt bereits zum Entscheidungszeitpunkt t_d zu bestimmen, wird derjenige Stillstand aus dem Lookahead verwendet, dessen Summe aus Stillstandseintrittszeitpunkt T_{Start} und geplanter Stillstandsdauer TTR am weitesten in der Zukunft liegt. Der Simulationsendzeitpunkt ergibt sich dann mit der Nachwirkzeit $\Delta t_{Nachwirkzeit}$ sowie der Prozesszeit der statischen Engpassprozessstufe PZ_{Eng} wie folgt:

$$T_{Ende} = T_{Start} + TTR + \Delta t_{Nachwirkzeit} + PZ_{Eng} \quad (3)$$

Für einen Prozess der sich stromaufwärts der statischen Engpassprozessstufe befindet, ist die Nachwirkzeit die Zeitdauer, bis das erste Werkstück nach Stillstandsende am Einlauf der Engpassprozessstufe zur Verfügung steht und entspricht damit der Durchlaufzeit vom Einlauf des Prozesses bis zum Einlauf der Engpassprozessstufe. Befindet sich der Prozess hingegen stromabwärts der statischen Engpassprozessstufe, so ist die Nachwirkzeit die Zeitdauer, bis die Engpassprozessstufe das erste Werkstück nach Stillstandsbehebung an die nachfolgende Prozessstufe weitergeben und damit wieder entsprechend ihrer Taktzeit Werkstücke bearbeiten kann.

Nachdem die statische Engpassprozessstufe und der Simulationsendzeitpunkt bestimmt wurden, kann für jede Behebungsreihenfolge k ein Simulationslauf durchgeführt werden. Dazu wird der Zustand des Fließfertigungssystems zum Entscheidungszeitpunkt t_d als Startbedingung für das Simulationsmodell verwendet, sodass das Simulationsmodell ein möglichst genaues Abbild des realen Fließfertigungssystems ist. Auf die für die Initialisierung verwendeten Daten wird im Rahmen der Implementierung nochmals näher eingegangen.

Die zu priorisierenden Prozesse stehen zu Beginn eines Simulationslaufes still und werden im Simulationsverlauf entsprechend der zu bewertenden Behebungsreihenfolge sequentiell wieder produktiv geschaltet. Um die zukünftig eintretenden geplanten Stillstandszeiten zu berücksichtigen, werden die im Lookahead enthaltenden Stillstände für jeden Simulationslauf so parametrisiert, dass die betroffenen Prozesse entsprechend dem jeweiligen Eintrittszeitpunkt und der geplanten Stillstandsdauer stillstehen. Da neben den zu priorisierenden Stillständen und den zukünftig eintretenden geplanten Stillstandszeiten keine weiteren Stillstände und auch keine anderen stochastischen Einflüsse, wie beispielsweise Taktzeit-schwankungen, während des Simulationslaufs berücksichtigt werden, liegt hier folglich ein deterministisches Modell vor. Ein Simulationslauf endet, sobald der Simulationsendzeitpunkt T_{Ende} erreicht ist. Zum Zeitpunkt T_{Ende} werden die Durchsatzmenge und die Restprozesszeit der statischen Engpassprozessstufe ausgelesen und zwischengespeichert.

Sind schließlich alle L Behebungsreihenfolgen simuliert, kann die Priorisierungsreihenfolge ermittelt werden. Diese entspricht nach Formel 1 der Behebungsreihenfolge mit dem größten Wert für die Durchsatzmenge der Engpassprozessstufe. Tritt der Fall ein, dass mehrere Behebungsreihenfolgen den maximalen Wert aufweisen, so wird als zweites Entscheidungskriterium die Restprozesszeit zum Zeitpunkt T_{Ende} verwendet, wobei die Behebungsreihenfolge mit der geringsten Restprozesszeit die Priorisierungsreihenfolge darstellt. Sollte es weiterhin mehr als eine Reihenfolge geben, die die maximale Durchsatzmenge und die minimale Restprozesszeit aufweisen, sind diese Reihenfolgen im Sinne der Auswirkung auf die Ausbringung des Fließfertigungssystems identisch. Tritt dieser Fall ein, so kann eine definierte Auswahl mittels einer FCFS-Regel getroffen werden, wobei die Priorisierungsreihenfolge die Reihenfolge ist, deren am höchsten priorisierter Stillstand zeitlich am frühesten eingetreten ist.

4 Implementierung in Plant Simulation

Um den Mehrwert einer Priorisierung ungeplanter Stillstände unter Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten näher zu untersuchen, wurde die im vorherigen Kapitel beschriebene simulati-

onsbasierte Vorgehensweise zur Priorisierung von Stillständen mit Lookahead in Form eines Entscheidungsunterstützungssystems in eine Simulationsumgebung (Siemens Plant Simulation) umgesetzt. Das Entscheidungsunterstützungssystem besteht dabei aus zwei Instanzen, die über die Plant Simulation eigenen Schnittstellen miteinander kommunizieren. Unter einer Instanz wird hier eine Modelldatei der Simulationssoftware verstanden.

In einer ersten Instanz wird das ablauffähige Modell des untersuchten Fließfertigungssystems und die Resource Produktionspersonal abgebildet (Abbildung 2).

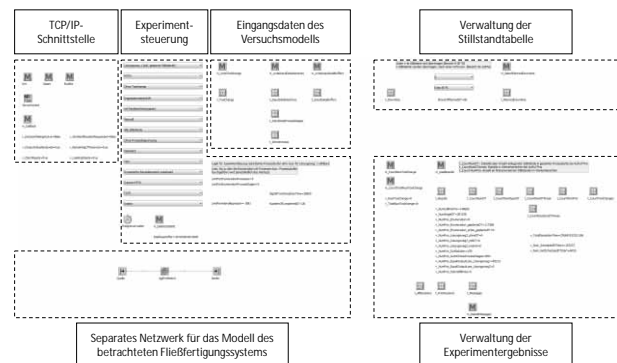
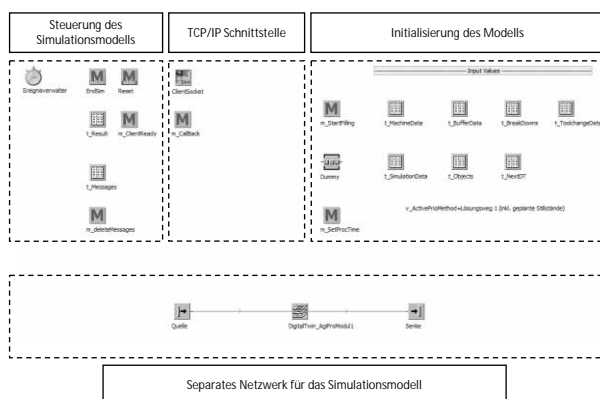


Abbildung 2: Erste Instanz des Entscheidungsunterstützungssystems (Screenshot)

Dieses Modell bildet das dynamische Verhalten des Fließfertigungssystems ab, wobei im Kontext einer Priorisierung von Stillständen die Aufprägung von Stillständen sowie eine entsprechende Steuerung der Stillstandsbehebung entscheidend sind. Dafür wurde anstelle des softwareeigenen Störgenerators eine manuelle Aufprägung mittels einer Stillstandstabelle, in der sämtliche Stillstandszeiten mit Eintrittszeitpunkt und Stillstandsdauer aufgelistet sind, und einem Baustein, der für jeden Produktionsprozess die Einplanung des nächsten Stillstandes in die Ereignisliste übernimmt, umgesetzt. Die Stillstandstabelle wurde in einer separaten Instanz, die ein identisches Simulationsmodell des Fließfertigungssystems beinhaltet, erstellt, indem sämtliche Stillstände mitgeschrieben wurden, die während eines Simulationslaufes aufgetreten sind. Die Stillstände wurden dabei durch den softwareeigenen Störgenerator mit Hilfe der für das Simulationsexperiment verwendeten Werte für die Verfügbarkeiten und die Reparaturdauern erzeugt. Da auf eine Abbildung des Produktionspersonals in diesem Modell verzichtet wird, werden die Stillstände hier umgehend und ohne Wartezeit auf verfügbares Produktionspersonal behoben. Diese manuelle Aufprägung der Still-

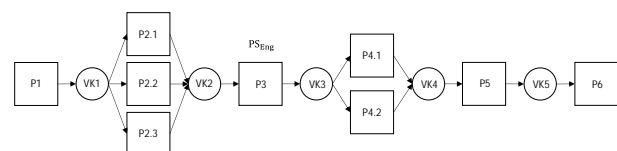
Tritt nun während eines Simulationslaufes der Fall ein, dass die Anzahl an Stillständen größer als die Anzahl zur Verfügung stehender Produktionsmitarbeiter ist, so wird ein Abbild des Produktionssystems durch die Initialisierung eines zweiten Modells (Forecast-Modell) in einer weiteren Instanz erzeugt, wobei in diesem Modell auf eine Abbildung des Produktionspersonals verzichtet wird (Abbildung 3).



Damit zum Entscheidungszeitpunkt der Zustand des Forecast-Modells dem aktuellen Zustand des Modells

Für die Übertragung der Eingabedaten zwischen den Instanzen wurde auf die Möglichkeit zurückgegriffen, Objektdateien zu erstellen und in einer anderen Instanz wieder einzulesen. Die Kommunikation zwischen den Instanzen wurde durch eine TCP/IP-Schnittstelle realisiert. Dadurch ist es möglich, die Befehle zum Initialisieren, Starten und Zurücksetzen des Forecast-Modells zwischen den Instanzen auszutauschen sowie die Übermittlung der Ergebnisgrößen sicherzustellen.

Im letzten Kapitel dieses Beitrages werden die Ergebnisse einer durchgeführten Simulationsstudie zur Untersuchung des erzielbaren Mehrwerts einer Priorisierung mit Lookahead vorgestellt. Betrachtet wird eine mehrstufige und mehrdimensionale Prozesskette, die aus vier einfachen sowie zwei redundanten Prozessstufen besteht, wobei alle Prozesse jeweils nur ein Werkstück parallel bearbeiten können (Abbildung 4). Die Prozessstufen sind durch pufferfähige Verkettungselemente verbunden.



In diesem System ist ein Mitarbeiter ausschließlich für die Behebung von Stillständen zuständig. Die zur Parametrisierung des Modells verwendeten Werte können den Tabellen 1 und 2 entnommen werden.

Tabelle 1: Eingabedaten der Prozesse

Prozess	Taktzeit in [s]	Verfügbarkeit in [%]	MTTR in [s]
P1	58	93	890
P2.1	184	96	640
P2.2	186	95	620
P2.3	180	97	630
P3	70	95	560
P4.1	126	94	580
P4.2	130	93	610
P5	65	92	890
P6	59	91	710

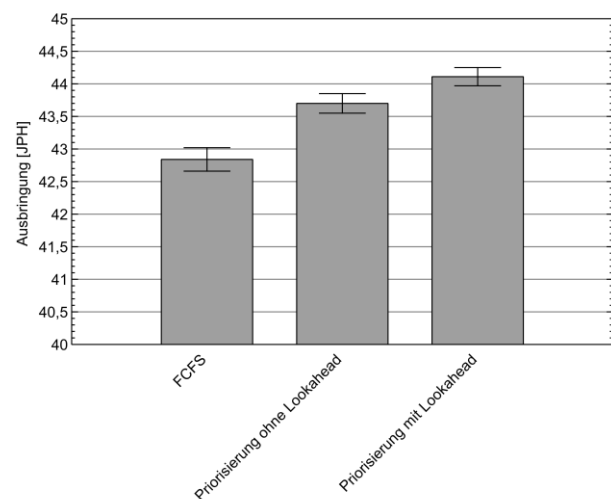
Tabelle 2: Eingabedaten der Verkettungselemente

Verkettungselement	Kapazität in [Stück]	Verweildauer in [s]
VK1	23	115
VK2	15	75
VK3	19	95
VK4	28	140
VK5	14	70

Die Simulationszeit für einen Simulationslauf wurde auf 150 Tage gesetzt, wobei die Einschwingphase zehn Tage beträgt und pro Simulationsexperiment zehn Replikationen simuliert wurden. Während eines Simulationslaufes wird nach jeweils 24 Stunden der erzielte Wert für die Ausbringungsmenge des Fließfertigungssystems innerhalb dieses Zeitraumes ermittelt. Aus diesen Werten wurde dann eine durchschnittliche Ausbringung ermittelt. Folglich beträgt der Stichprobenumfang für die Versuche 1400.

In Abbildung 5 ist die erzielbare Ausbringung in Abhängigkeit der gewählten Vorgehensweise zur Priorisierung ungeplanter Stillstände dargestellt. Neben der in Kapitel 4 vorgestellten simulationsbasierten Vorgehensweise zu Priorisierung mit Lookahead wurde auch eine FCFS-Strategie umgesetzt. Bei dieser Strategie werden die Stillstände entsprechend ihres Eintrittszeitpunktes priorisiert, wobei zeitlich am frühesten eingetretene Stillstände als Erstes behoben werden. Darüber hinaus wurde

eine weitere simulationsbasierte Vorgehensweise implementiert, mit der eine Priorisierung ohne Lookahead möglich ist. Die Vorgehensweise ist dabei überwiegend identisch mit der Vorgehensweise mit Lookahead, wobei sich zwei Änderungen ergeben. Da bei der Vorgehensweise ohne Lookahead keine zukünftig eintretenden geplanten Stillstandszeiten betrachtet werden, müssen diese bei der Initialisierung des Forecast-Modells auch nicht parametrisiert werden. Dadurch ergibt sich ebenfalls ein geändertes Vorgehen zur Ermittlung des Simulationsendzeitpunktes T_{Ende} . Bei dieser Vorgehensweise wird der Simulationsendzeitpunkt statisch auf acht Stunden gesetzt, sodass für jede Behebungsreihenfolge nach acht Stunden Simulationszeit des Forecast-Modells die Durchsatzmenge sowie die Restprozesszeit der Engpassprozessstufe ermittelt wird.

**Abbildung 5:** Durchschnittlich erzielbare Ausbringung in Abhängigkeit der Priorisierungsmethode

Die Versuchsergebnisse zeigen, dass bereits mit einer Priorisierung ohne Lookahead eine signifikant höhere Ausbringung im Vergleich zu einer FCFS-Strategie erzielt werden kann. Dabei liegt die durchschnittliche Ausbringung einer Priorisierung ohne Lookahead bei $43,7 \pm 0,15$ JPH (95 % Konfidenzintervall), was einem Mehrwert im Vergleich zur FCFS-Strategie von 2,01 % entspricht. Durch Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten in Form eines Lookahead kann darüber hinaus mit $44,1 \pm 0,14$ JPH (95 % Konfidenzintervall) eine signifikant höhere Ausbringung generiert werden. Der Mehrwert gegenüber der FCFS-Strategie beträgt dabei 2,96 %, wobei die Ermittlung einer Priorisierungsreihenfolge nur in 24 % aller Entscheidungen, bei denen der Produktionsmitarbeiter zu einem oder

mehreren Stillständen zugeteilt werden musste, angewendet wurde. Somit liegt bei 76 % der Entscheidungen nur ein Stillstand an, wobei in diesen Fällen der Mitarbeiter direkt zugeordnet wird. Der direkte Vergleich einer Priorisierung mit Lookahead zu einer Priorisierung ohne Lookahead zeigt schließlich, dass eine Priorisierung mit Lookahead zu einem signifikanten Mehrwert von 0,92 % führt. Dieser Mehrwert resultiert aus der Möglichkeit, durch die Wahl der Priorisierungsreihenfolge die Auswirkung geplanter Stillstandszeiten proaktiv zu beeinflussen. Da die Stillstandsdauer der statischen Engpassprozessstufe oder auch einer Prozessstufe, für die zukünftig ein Stillstand eingeplant ist, abhängig von der Priorisierungsreihenfolge ist, kann der Durchsatz dieser Prozessstufe und somit auch die Anzahl an Werkstücken, die sich zum Eintrittszeitpunkt eines geplanten Stillstandes zwischen der betroffenen Prozessstufe und der statischen Engpassprozessstufe befinden, durch die Priorisierung beeinflusst werden. Die freie Pufferkapazität in diesem Bereich entscheidet letztendlich über die Dauer der latenten Phase, d. h. der Zeitspanne, in der sich die Auswirkung eines Stillstandes noch nicht auf die Leistung des gesamten Fließfertigungssystems auswirkt, sodass sich folglich mit der Priorisierungsreihenfolge auch die Auswirkung geplanter Stillstände beeinflussen lässt.

Der Lookahead wurde hier als Time-Lookahead definiert. Dabei sind sämtliche Stillstandszeiten innerhalb der nächsten acht Stunden Simulationszeit, bezogen auf den jeweiligen Entscheidungszeitpunkt t_d , bekannt und wurden dementsprechend bei der Ermittlung der Priorisierungsreihenfolge berücksichtigt. Damit stellt der hier ermittelte Wert für die mit einer Priorisierung mit Lookahead erzielbare Ausbringung eine obere Grenze dar, da alle Stillstandszeiten während dieses Zeitraumes bekannt sind. In der Praxis kann allerdings nicht davon ausgegangen werden, dass sämtliche zukünftig eintretende Stillstandszeiten zum jeweiligen Entscheidungszeitpunkt bekannt sind. Deswegen wurde in einem weiteren Experiment untersucht, wie der Mehrwert einer Priorisierung mit Lookahead beeinflusst wird, wenn neben den geplanten und im Lookahead enthaltenen Stillständen auch ungeplante Stillstände, d. h. zum Entscheidungszeitpunkt nicht bekannte und damit auch nicht in die Priorisierungsentscheidung mit einbezogene Stillstände, eintreten. In Abbildung 6 ist der Mehrwert einer Priorisierung mit Lookahead in Abhängigkeit des Anteils geplanter Stillstände an der Gesamtanzahl eintretender Stillstände dargestellt.

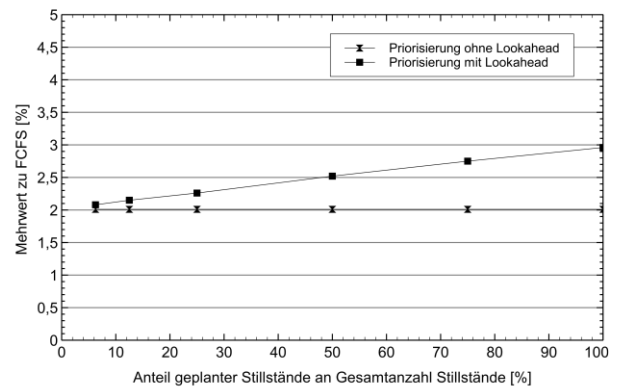


Abbildung 6: Einfluss des Anteils geplanter Stillstände auf den Mehrwert einer Priorisierung im Vergleich zur FCFS-Strategie

Es zeigt sich, dass der Mehrwert einer Priorisierung mit Lookahead mit sinkendem Anteil geplanter Stillstände abnimmt, wobei der Mehrwert stets größer als der Mehrwert einer Priorisierung ohne Lookahead bleibt. Weitere Untersuchungen einzelner Priorisierungsentscheidungen haben hier gezeigt, dass mit abnehmendem Anteil geplanter Stillstände eine gezielte Beeinflussung der verfügbaren Pufferkompensation zum Eintrittszeitpunkt eines geplanten Stillstandes nicht mehr möglich ist. Die verfügbare Pufferkompensation wird vielmehr durch das Auftreten der ungeplanten und nicht im Lookahead enthaltenen Stillstände beeinflusst, sodass eine Priorisierung mit Lookahead bei einem geringen Anteil geplanter Stillstände zu keinem signifikanten Mehrwert gegenüber einer Priorisierung ohne Lookahead führt.

6 Zusammenfassung und Ausblick

In diesem Beitrag wurde der Mehrwert einer Priorisierung von Maschinenstillständen unter Berücksichtigung zukünftig eintretender geplanter Stillstandszeiten untersucht. Dazu wurde die Priorisierung als Online-Optimierungsproblem mit Lookahead aufgefasst und eine simulationsbasierte Vorgehensweise vorgestellt, mit der geplante Stillstandszeiten in die Ermittlung einer Priorisierungsreihenfolge mit einbezogen werden können. Durch die Implementierung dieser Vorgehensweise als Entscheidungsunterstützungssystem in eine Simulationsumgebung, konnte schließlich am Beispiel eines mehrdimensionalen Fließfertigungssystems ein signifikanter Mehrwert einer Priorisierung mit Lookahead im Vergleich zu einer Priorisierung ohne Lookahead aufgezeigt

werden, wobei der erzielbare Mehrwert von dem Anteil geplanter und damit im Lookahead berücksichtigter Stillstände abhängig ist.

Im weiteren Verlauf der Forschungsarbeit sollen weitere Einflussfaktoren auf den Mehrwert einer Priorisierung mit Lookahead identifiziert werden. Darauf aufbauend wird untersucht, unter welchen Randbedingungen eine Priorisierung mit Lookahead zu einem signifikanten Mehrwert führt. Zudem soll die simulationsbasierte Vorgehensweise zur Priorisierung mit Lookahead auch in einer praxisnahen Umgebung validiert werden. Dazu wird der erzielbare Mehrwert in Kooperation mit der Mercedes-Benz AG an einer komplexen Kurbelgehäusefertigungsline überprüft.

Literatur

- [1] Kröning, S.; Denkena, B.: Dynamic scheduling of maintenance measures in complex production systems. *Journal of Manufacturing Science and Technology* 6 (2013) 4, S. 292-300, S. 292
- [2] Wedel, M.: Effektive Priorisierung bei reaktiven Instandhaltungsmaßnahmen zur Steigerung der Ausbringung von komplexen Transferstraßen am Beispiel der Automobilindustrie. Aachen: Shaker 2016, S. 2
- [3] Zhai, S.; Reinhart, G.: Predictive Maintenance als Wegbereiter für die instandhaltungsgerechte Produktionssteuerung. *Zeitschrift für wirtschaftlichen Fabrikbetrieb (ZWF)* 113 (2018) 5, S. 298-301, S. 299
- [4] Hegemann, M.; Nickel, S.: Einfluss von zuverlässig prognostizierten Stillstandzeiten auf die simulationsbasierte Priorisierung von Maschinenstillständen in komplexen Produktionssystemen. In: ASIM Fachtagung Simulation in Produktion und Logistik 2019. Chemnitz: 18.-20.09.2019
- [5] Guo, W.; Jin, J.; Hu, S. J.: Allocation of maintenance resources in mixed model assembly systems. In: *Journal of Manufacturing Systems* 32 (2013), S. 473-479
- [6] Bengtsson, M.: Classification of Machine Equipment. In: *Conference on Maintenance Performance Measurement and Management*. Lulea (SWE), 2011, S. 1-5
- [7] Gupta, S.; Bhattacharya, J.: Cost-effective importance measure: A new approach for resource prioritization in a production plant. In: *International Journal of Quality and Reliability Management* 30 (2013) 4, S. 379-386
- [8] Gopalakrishnan, M.; Skoogh, A.; Laroque, C.: Simulation-based Planning of Maintenance Activities in the Automotive Industry. In: *Proceedings of the 2013 Winter Simulation Conference*. Washington D. C. (USA), 2013, S. 2610-2621
- [9] Guner, H. U.; Chinnam, R.; Murat, A.: Simulation platform for anticipative plant-level maintenance decision support system. In: *International Journal of Production Research* (2015), S. 1-19
- [10] Gopalakrishnan, M.; Skoogh, A.; Laroque, C.: Simulation-based Planning of Maintenance Activities by a Shifting Priority Method. In: *Proceedings of the 2014 Winter Simulation Conference*. Savannah Georgia. (USA), 2014, S. 2168-2179
- [11] Langer, R.; Li, J.; Biller, S.; Chang, Q.; Huang, N.; Xiao, G.: Simulation study of a bottleneck-based dispatching policy for a maintenance workforce. In: *International Journal of Production Research* 48 (2010) 6, S. 1745-1763
- [12] Li, L.; Chang, Q.; Ni, J.; Biller, S.: Real time production improvement through bottleneck control. In: *International Journal of Production Research*. 47 (2009a) 21, S. 6145-6158
- [13] Li, L.; Ni, J.: Short-term decision support system for maintenance task prioritization. In: *International Journal of Production Economics* 121 (2009), S. 195-202
- [14] Yang, Z.; Chang, Q.; Djurdjanovic, D.; Ni, J.; Lee, J.: Maintenance Priority Assignment Utilizing On-line Production Information. In: *International Journal of Manufacturing Science and Engineering* 129 (2007), S. 435-446
- [15] Ni, J.; Jin, X.: Decision support systems for effective maintenance operations. In: *CIRP Annals - Manufacturing Technology* 61 (2012), S. 411-414
- [16] Subramaniyan, M.; Skoogh, A.; Gopalakrishnan, M.; Salomonsson, H.; Hanna, A.; Lämckull, D.: An algorithm for data-driven shifting bottleneck detection. In: *Cogent Engineering* 3 (2016), S. 1-19
- [17] Subramaniyan, M.; Skoogh, A.; Salomonsson, H.; Bangalore, P.: A data-driven algorithm to predict throughput bottlenecks in a production system based on active periods of machines. In: *Computers and Industrial Engineering* 125 (2018), S. 533-544
- [18] Dunke, F.; Necil, J.; Nickel, S.: Online-Optimierung und Simulation in der Logistik. In: Lübbecke, M.; Weiler, A.; Werners, B.: *Zukunftsperspektiven des Operations Research*. Wiesbaden. Springer Gabler: 2014, S. 33-47
- [19] Dunke, F.; Nickel, S.: A general modeling approach to online optimization with lookahead. In: *Omega* 63 (2016), S. 134-153
- [20] Hanisch, A.; Schulze, T.: Initialization of Online Simulation Models. In: *Proceedings of the 2005 Winter Simulation Conference*. Orlando (USA), 2005, S. 1795-1803

Simulationsgestützte Optimierung des Materialflusses in einem Aluminium-Gussbetrieb

Johannes Dettelbacher*, Wolfgang Schlüter

Hochschule Ansbach, Residenzstraße 8, 91522 Ansbach, *johannes.dettelbacher@hs-ansbach.de

Abstract. Aluminium-Druckgussbetriebe haben aufgrund ihres hohen Energieverbrauchs ein besonders großes Energieeinsparpotential und bieten sich für Optimierungsmaßnahmen an. In vorherigen Arbeiten wurde bereits gezeigt, dass die innerbetriebliche Verteilung des Flüssigaluminiums die Energieeffizienz beeinflusst. Ausgehend davon wird die Entwicklung eines Softwaretools beschrieben, welches Handlungsempfehlungen über die innerbetrieblichen Prozessabläufe ausgibt. In der Software wird ein Optimierungsmodell mit einer Simulation des Gussbetriebes gekoppelt. Das Ziel ist die Optimierung des innerbetrieblichen Materialflusses hinsichtlich der Energieeffizienz und der Auslastung der Produktionsmaschinen. Die Empfehlungen werden anschließend anhand einer Simulation auf ihre Aussagekraft beurteilt. In Rahmen der Arbeit wird das Optimierungsmodell hinsichtlich seiner Eignung bewertet. Dabei werden die Vorteile der Kopplung von Simulation und Optimierung und die Einsatzmöglichkeiten in der Aluminium-Gussbranche herausgearbeitet und insbesondere der Mehrwert zu dem Einsatz einer einfachen Simulation quantifiziert.

Einleitung

Aufgrund der Energiewende und der steigenden Konkurrenz durch die Globalisierung hat die Energieeffizienz insbesondere in Deutschland stark an Bedeutung zugenommen. Ein besonders hohes Energie- und Kosteneinsparungspotenzial ist dabei in energieintensiven Branchen, wie z. B. der Nichteisen (NE)-Schmelz- und Druckgussindustrie, zu finden. Bei der Aluminium-Gussindustrie zeigt sich die Energiekostenbelastung, welche 25 % der Bruttowertschöpfung übersteigen kann [1]. Um Einsparpotentiale und Energieeffizienzmaßnahmen besser zu beurteilen, wurde eine Material- und Energieflusssimulation entwickelt, mit welcher sich beliebige Aluminium-Schmelz- und Gussbetriebe abbilden lassen. Anhand der Simulation konnte in vorherigen Arbeiten bereits gezeigt werden, dass die Steuerung von innerbetrieblichen Abläufen die Energie- und Anlageneffizienz erheblich beeinflusst [2]. Ein großes Potential wird der Verteilung des Aluminiums durch Stapler zugeschrieben. Ausgehend hiervon wird ein Optimierungsmodell entwickelt, wel-

ches die Generierung von Stapleraufträgen für die Druckgussmaschinen (DGM) unterstützt. Um das Optimierungsmodell und dessen Ergebnisse auf die Realisierbarkeit sowie das Potential der Energieeinsparung zu bewerten, werden diese an die Simulation übergeben und in der implementierten Prozesssteuerung berücksichtigt. Im Rahmen der Arbeit werden die verwendeten Modelle und die Schnittstelle beschrieben. Des Weiteren werden die Ergebnisse des Optimierers sowie deren Implementierung in die Simulation bewertet und der Mehrwert der Software herausgearbeitet.

1 Aluminium-Gussbetrieb

Im Rahmen dieser Arbeit wird ein Aluminium-Druckgussbetrieb, wie er in Abbildung 1 dargestellt wird, untersucht. Der Betriebsablauf umfasst sowohl kontinuierliche (z. B. Schmelzen) als auch ereignisdiskrete (z. B. Staplertransport) Prozessschritte.

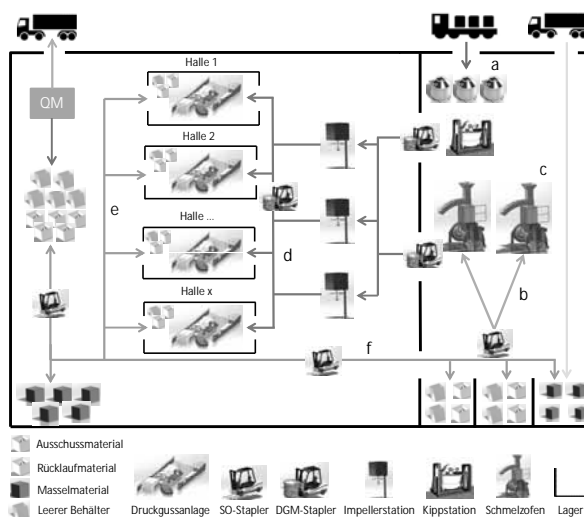


Abbildung 1: Schema eines Aluminium-Druckgussbetriebs mit Prozessschritten

Die zugrundeliegenden Prozesse sind:

- Anlieferung von flüssigem Aluminium (Abb. 1, a),
- Beschickung der gasbetriebenen Schachtschmelzöfen über Stapler mit Masseln (Metallbarren), Rücklauf- oder Ausschussmaterial (Abb. 1, b)
- Erwärmen, Schmelzen und Überhitzen bzw. Warmhalten des Metalls (Abb. 1, c)
- Verteilung des flüssigen Aluminiums mit Staplern auf die Dosieröfen der Druckgussmaschinen (Abb. 1, d)
- Produktion von Gussteilen in den Druckgussanlagen und Qualitätsprüfung (Abb. 1, e)
- Transport von Materialbehältern aus dem Druckgussbetrieb oder von Masselpaketen aus dem Lager zum Schmelzbetrieb (Abb. 1, f)

Im Fokus dieser Untersuchung steht die Verteilung des Flüssigaluminiums zu den einzelnen Druckgussmaschinen. Daher wird die Verteilung im Folgenden genauer beschrieben. Das flüssige Aluminium wird entweder aus der Kippstation oder aus den Ofenwannen entnommen und mit den DGM-Staplern zu den Druckgussmaschinen geliefert. Dazwischen erfolgt bei einem Zwischenstopp an einer Impellerstation ein Reinigungsprozess des Aluminiums. Eine Belieferung mit einem DGM-Stapler umfasst die Befüllung von bis zu zwei Druckgussmaschinen, welche mit der selben Legierung produzieren. An den Druckgussmaschinen wird das Aluminium in den elektrisch beheizten Dosieröfen warmgehalten, bis es zu Bauteilen gegossen wird. Der Materialverbrauch der Druckgussmaschinen ist hierbei abhängig von dem Bruttogewicht und der Taktzeit der gegossenen Bauteile. Die Druckgussmaschinen können je nach Produktionsplan auch Bauteile unterschiedlicher Legierung produzieren. Des Weiteren beeinflussen Maschinenstörungen und Ausfälle den tatsächlichen Verbrauch des Flüssigaluminiums. Beim Füllstand eines Dosierofens zeigen sich konkurrierende Ziele. Zum einen sorgt ein hoher Füllstand für eine hohe Versorgungssicherheit und vermeidet Ausfälle aufgrund von Aluminiummangel. Zum anderen führt eine kürzere Warmhaltedauer des Aluminiums dazu, die Materialeigenschaften und den Energieverbrauch zu optimieren. Insbesondere der Energieverbrauch in den elektrisch beheizten Dosieröfen verursacht einen Großteil der Energiekosten.

2 Optimierungmodell

Das Ziel des Optimierungsmodells ist, die Verteilung des Flüssigaluminiums auf die Druckgussmaschinen zu optimieren. Im Modell wird bestimmt, wann welche Druckgussmaschine optimal zu beliefern ist. Hierfür wird eine dynamische Optimierung verwendet, welche alle be-

trachteten Zeitpunkte in einem Gleichungssystem berücksichtigt. Um die Rechenzeit und Komplexität des Modells gering zu halten, werden als Zeitpunkte jeweils einmündige Zeitintervalle verwendet. Auch werden die Materialquellen wie z. B. Ofenwannen und die Impellerstationen als Zwischenstationen in diesem Modell nicht explizit betrachtet. Als Variablen im Optimierungsmodell werden die Belieferungen der Dosieröfen zu den gegebenen Zeitpunkten definiert. Somit ergibt sich für die Variablen eine Matrix aus der Anzahl der Druckgussmaschinen und der Anzahl der betrachteten Zeitpunkte. Die Variablen können für jeden Zeitpunkt sowie jeder Druckgussmaschine jeweils Zustand 1 (Befüllen) oder Zustand 0 (Nicht Befüllen) annehmen. Die Belieferungsmenge wird hierbei bereits vorgegeben und hängt maschinen-spezifisch vom Fassungsvermögen des jeweiligen Dosierofens ab. Es ergibt sich ein ganzzahliges lineares Optimierungsproblem, welches im Rahmen des Modells beschrieben und gelöst wird. Für die Optimierung werden Eingangsgrößen, wie die Materialverbräuche der Druckgussmaschinen, die aktuellen Füllstände und das Fassungsvermögen der Dosieröfen sowie die Anzahl und die Fahrtzeiten der DGM-Stapler aus der Anlagen- und Betriebskonfiguration eingelesen. Ausgehend von diesen Daten werden die Zielfunktion und die Nebenbedingungen definiert. Im Rahmen dieser Arbeit werden mit dem Optimierungsmodell die Belieferungszeiten für einen Zeitraum von 5 Stunden bzw. 300 Minuten bestimmt. Implementiert wurde das Optimierungsmodell in Matlab. Im Folgenden werden die verwendete Zielfunktion sowie die Nebenbedingung der Optimierung beschrieben.

2.1 Zielfunktion

Die Zielfunktion umfasst die gesamten Füllstände $y_{DGM,t}$ aller Dosieröfen über den gesamten Betrachtungsraum und wird im Rahmen der Optimierung minimiert.

$$\min \sum_{DGM=1}^k \sum_{t=t,Start}^{t,Ende} y_{DGM,t} \quad (1)$$

Damit wird bezweckt, dass stets so wenig Aluminium wie möglich in den Dosieröfen warmgehalten wird. Die Füllstände der Dosieröfen sind über Nebenbedingungen definiert und ergeben sich jeweils aus dem vorherigen Füllstand der Maschine und die mögliche Füllstandsänderung durch Belieferung und Materialverbrauch.

2.2 Nebenbedingungen

Weiterhin werden die Bedingungen vorgegeben, dass der Füllstand weder das maximale Fassungsvermögen übersteigen sowie einen minimalen Füllstand unterschreiten darf. Der minimale Füllstand, welcher für eine Produktionssicherheit sorgt, wird maschinenspezifisch über den Verbrauch ermittelt. Dazu wird vorgegeben, dass jede

Maschine zu jedem Zeitpunkt über mindestens 30 Minuten Restlaufzeit verfügt. Damit wird für eine ausreichende Versorgungssicherheit der Produktionsmaschinen gesorgt. Weiterhin wird vorgegeben, dass nur die Anzahl an Belieferungen in einem bestimmten Zeitintervall möglich sind, welche die Anzahl der DGM-Stapler bei einer angenommenen Belieferungszeit realisieren können.

3 Simulationsmodell

Das Simulationsmodell, welches die Optimierungsergebnisse auf ihre Aussagekraft beurteilen soll, bildet den gesamten Aluminium-Druckgussbetrieb mit den dazugehörigen Prozessschritten ab. Das Modell lässt sich, wie in Abbildung 2 dargestellt, in die Teilmodelle Energieflussmodell, Materialflussmodell und Prozesssteuerung untergliedern.

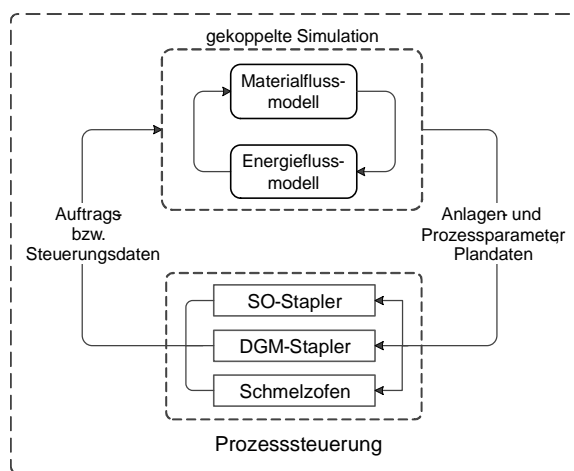


Abbildung 2: Bestandteile der hochdetaillierten Betriebs-simulation

Während im Energieflussmodell die thermodynamischen Vorgänge in den Aluminiumschmelzöfen betrachtet werden, erfasst das Materialflussmodell den Materialfluss des festen und flüssigen Aluminiums innerhalb des Betriebes. Zwischen den Modellen liegt eine bidirektionale Kopplung vor, da die Schmelzleistung sowohl von Materialfluss als auch vom Energiefluss abhängt. In der Prozesssteuerung wird der Betriebsablauf gesteuert. Ein besonderer Fokus liegt hierbei auf die Steuerung der SO-Stapler und der DGM-Stapler. Eine besondere Relevanz in dieser Untersuchung hat die Steuerung der DGM-Stapler, die im Folgenden genauer beschrieben wird. Eine ausführlichere Beschreibung der einzelnen Teilmodelle wurde von Schlüter und Buswell vorgenommen [2, 3, 4].

3.1 DGM-Stapler-Steuerung

Eine stetig gesicherte Versorgung der Druckgussmaschinen mit Aluminium ist für den Produktionsbetrieb entscheidend. Die Prozesssteuerung wird zu jedem Zeitpunkt innerhalb des Simulationsmodells ausgeführt und erzeugt je nach Versorgungssituation und Verfügbarkeit der Stapler die Stapleraufträge. Über einen Steuerungsalgorithmus werden die Materialquelle (Ofenwanne/Kippstation), die zu befüllenden Materialsinken (Druckgussmaschinen), die verwendete Impellerstation und die resultierende Entnahmemenge bestimmt. Bei der Bestimmung der zu befüllenden Druckgussmaschinen können unterschiedliche Kriterien ausgewählt werden. Bei dem Ampelverfahren, welches auch in Industriebetrieben verwendet wird, wird einem bestimmten Füllstandbereich eine definierte Signalfarbe (rot – gelb – grün) zugewiesen, welche die Priorisierung der Druckgussmaschinen bestimmt. Verbesserte Varianten berücksichtigen den genauen Füllstand der Maschinen und berechnen die Restlaufzeiten der einzelnen Druckgussmaschinen. Je nach Versorgungssituation und ausgewählter Steuerungsstrategie wird die Belieferung von bis zu zwei Druckgussmaschinen zu einem Auftrag zusammengefasst. Der Auftrag wird anschließend von der Prozesssteuerung generiert und an die gekoppelte Simulation übergeben und ausgeführt.

3.2 Umsetzung der Simulation

Für den Aufbau der Simulation wird ein objektorientierter Ansatz verfolgt. Als Simulationsumgebung für das Simulationsmodell wird Matlab, Simulink und Stateflow verwendet. Während Simulink für die Simulation der kontinuierlichen Prozesse und Stateflow für die ereignisdiskreten Prozesse genutzt wird, dient Matlab hauptsächlich für die Simulationssteuerung und der Objekterzeugung und -verwaltung.

3.3 Validierung der Simulation

Die gekoppelte Simulation konnte in vorherigen Arbeiten durch Betriebsdaten zwei realer Betriebe validiert werden. Für das Materialflussmodell ergeben sich Abweichungen in der Anzahl der produzierten Aluminiumgussteile und der verbrauchten Aluminiummenge von 1,4 bzw. 0,9 %. Bei dem Energieflussmodell weichen geschmolzene Aluminiummasse und der Gasverbrauch um 1,5 bzw. 0,5 % von den tatsächlichen Werten ab.

4 Kopplung

Im Rahmen dieser Arbeit wird die entwickelte Optimierung mit der Simulation gekoppelt. Dieser Ansatz kombiniert die Vorteile beider Werkzeuge. Während die Optimierung die beste Parameterauswahl findet, wird die Si-

mulation für die Bewertung und Überprüfung der Optimierungsergebnisse verwendet [5]. Die Optimierung startet zuerst und leitet die Ergebnisse an die Simulation weiter. Der Ablauf, wie er in Abbildung 3 dargestellt ist, wird im Folgenden genauer beschrieben.

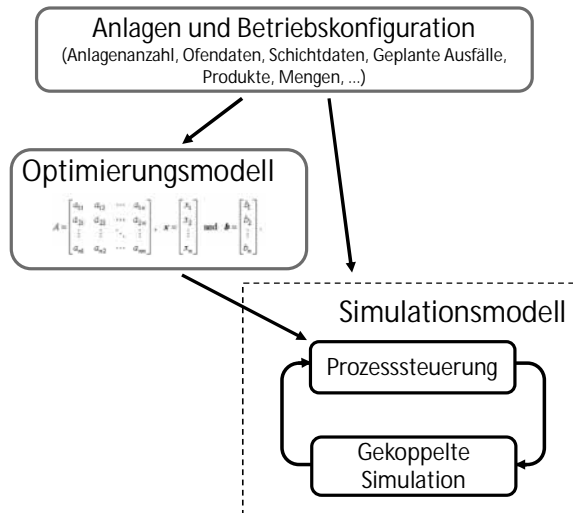


Abbildung 3: Kopplung von Optimierung und Simulation

Nachdem vom Optimierungsmodell die Belieferungszeiten der Druckgussmaschinen für eine optimale Zielerreichung bestimmt wurden, werden diese in ein Matlab-File gespeichert. Dieses File wird anschließend in der Simulation geladen und die Werte beim Start der Simulation der Prozesssteuerung übergeben. Bei der Generierung der neuer DGM-Stapleraufträge wird eine neue Steuerungsstrategie implementiert, welche die von der Optimierung bestimmten Druckgussmaschinen mit der jeweiligen Belieferungsmenge einkalkuliert. Bei der Erzeugung neuer DGM-Stapleraufträge im Simulationsmodell laufen unterschiedliche Funktionen nacheinander ab. Zuerst wird geprüft, ob eine vorgegebene Belieferung aus der Vergangenheit noch aussteht. Ist dies der Fall, wird die am weitesten zurückliegende Belieferung für den neuen Auftrag übernommen. Anschließend wird in einem Intervallbereich von 15 Minuten um der übernommenen Belieferung nach weiteren Belieferungen gesucht. Wenn eine weitere gefunden wird, wird diese ebenfalls in diesen Auftrag zusammengefasst. Falls keine weitere Belieferung in diesem Intervallbereich vorliegt, wird lediglich die eine zu beliefernde Druckgussmaschine im Auftrag hinterlegt. Für den Fall, dass keine Belieferung aus der Vergangenheit aussteht, wird nach Belieferungen in naher Zukunft gesucht. Es wird ein Zeitraum von maximal 10 Minuten betrachtet. Liegt eine Belieferung vor, wird gleichermaßen in einem Intervallbereich von 15 Minuten nach weiteren Belieferungen gesucht. Die zu beliefernden Druckgussmaschinen werden in einem Auftrag gespeichert. Während das Optimierungsmodell nur vorgibt,

welche Druckgussmaschine zu welchem Zeitpunkt beliefert werden muss, muss die Bestimmung der Materialquelle und die Auswahl der Impellerstation weiterhin unabhängig von der Steuerung innerhalb der Simulation erfolgen.

5 Ergebnisse

Im Rahmen der Untersuchung werden zwei Betriebe mit den Modellen untersucht:

- Betrieb 1 mit 5 Schmelzöfen, 24 Druckgussmaschinen und ohne zusätzlicher Anlieferung von Flüssigaluminium
- Betrieb 2 mit 4 Schmelzöfen, 31 Druckgussmaschinen und zusätzlicher Anlieferung von Flüssigaluminium

Zum einen wird betrachtet, welche Ergebnisse der Optimierung liefert und wie diese in der Simulation umgesetzt werden. Zum anderen wird untersucht, welchen Einfluss die Optimierung auf die Produktionssicherheit und Energieeffizienz des Betriebes hat.

5.1 Optimierungsergebnisse

Die Ergebnisse des Optimierungsmodells geben an, zu welchem Zeitpunkt welche Druckgussmaschine befüllt werden soll. Um diese Resultate zu visualisieren, wird aus den Befüllungen und dem Materialverbrauch der Maschinen der zeitliche Verlauf der vorraussichtlichen Füllstände berechnet. Ein solcher zeitlicher Verlauf ist in Abbildung 4 für Betrieb 2 dargestellt. Es zeigt sich, dass für jede Druckgussmaschine stets ausreichend Aluminium für die Produktion zur Verfügung steht. Auch wird deutlich, dass Maschinen mit einem höheren Materialverbrauch einen höheren Mindestfüllstand aufweisen, um eine Restlaufzeit von 30 Minuten zu gewährleisten.

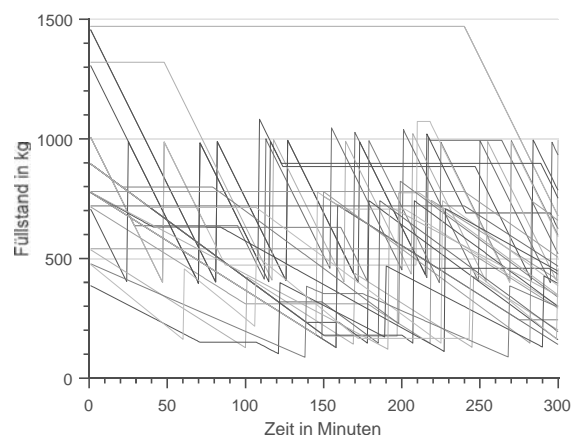


Abbildung 4: Füllstände der Dosieröfen im Optimierungsergebnis

5.2 Implementierung in die Simulation

Bei der Implementierung der Optimierungsergebnisse in der Simulation wird untersucht, wie die geplanten Befüllungen der Optimierung mit den tatsächlich umgesetzten Befüllungen in der Simulation übereinstimmen. Damit wird überprüft, ob die vereinfachte Betrachtung im Optimierungsmodell ausreicht, um umsetzbare Ergebnisse zu liefern. Hierfür wird beispielhaft der Materialverlauf im Dosierofen der Druckgussmaschine 1 in Abbildung 5 dargestellt. Es wird ersichtlich, dass die vom Optimierer erzeugten Belieferungen in der Simulation umgesetzt wurden. Lediglich der Zeitpunkt der Umsetzung kann sich in Simulation durch die Verfügbarkeit der Stapler und der Dauer der Staplerfahrt unterscheiden. Bei einer hohen Auslastung der Stapler konnte eine Verzögerung der Belieferung von bis zu 13 Minuten auftreten. Da jedoch durch die Optimierung ein Mindestfüllstand für eine Restlaufzeit von 30 Minuten vorgegeben wird, ist zu keinem Zeitpunkt die Produktionssicherheit gefährdet.

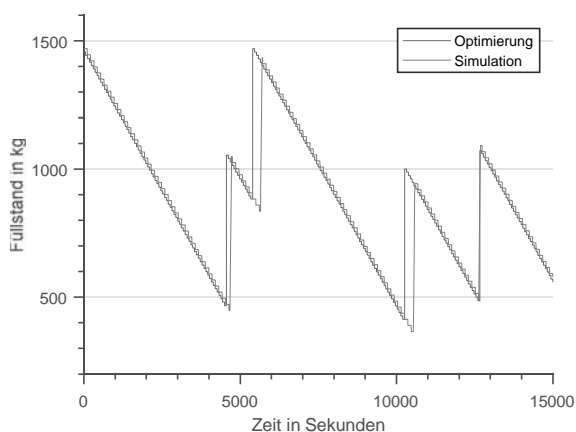


Abbildung 5: Druckgussfüllstand in Optimierung und Simulation

5.3 Bewertung der Optimierungsergebnisse

Ziel des Optimierungsalgorithmus ist die Bestimmung der optimalen Belieferungszeiten für eine optimale Zielerreichung. Als Ziel wurde eine möglichst geringe Aluminiummenge in den Dosieröfen bei kontinuierlich gegebener Produktionssicherheit definiert. Um die Vorteile der Optimierungsergebnisse zu anderen Steuerungsalgorithmen zu untersuchen, wird der mittlere Füllstand in den Dosieröfen bei verschiedenen Steuerungen verglichen. Die Ergebnisse hierzu sind in Tabelle 1 dargestellt. Es kann gezeigt werden, dass mit der Steuerung über eine Optimierung eine deutliche Reduzierung der Warmhalte-masse erreicht werden kann. Dieser Effekt konnte anhand beider Betriebe verdeutlicht werden. Die durchschnittliche Aluminiummasse in den Dosieröfen reduziert sich in

beiden Fällen etwa um 40 %. Die Reduzierung hängt jedoch auch maßgeblich von Faktoren wie Auslastung und Störfällen an den Druckgussmaschinen ab.

	Betrieb 1	Betrieb 2
Durchschnittliche Masse in den Dosieröfen bei der Steuerung über relative Füllstand [kg]	13467	32102
Durchschnittliche Masse in den Dosieröfen bei der Steuerung über mathematische Optimierung [kg]	8031	19039
Einsparung der Warmhalte-masse [%]	40,4	40,7

Tabelle 1: Dosierofenmasse bei unterschiedlichen DGM-Staplersteuerungen

Weiterhin sind bei beiden Steuerungen keine Ausfälle aufgrund von Aluminiummangel aufgetreten. Eine genaue Kalkulation der möglichen Energieeinsparung durch die Optimierung ist im Simulationsmodell noch nicht möglich, da das vorhandene Energiemodell lediglich die Schmelzöfen und Ofenwannen abbildet. Hierfür bietet sich an, die Energiebetrachtung in weiteren Untersuchungen auch auf die Druckgussmaschinen auszuweiten.

6 Zusammenfassung und Ausblick

Im Rahmen dieser Arbeit wurde ein Optimierungsmodell für die Verteilung des Flüssigaluminiums zu den Druckgussmaschinen aufgebaut. Dieses Modell dient zur optimalen Parameterwahl und wird mit einer Simulation, welche die Umsetzung und Bewertung der Parameter erzeugt, gekoppelt. Die einzelnen Modelle sowie die Koppelung der beiden Modelle wurden im Rahmen dieser Arbeit beschrieben. Anhand von zwei Betrieben wurden Optimierungsergebnisse und die Umsetzung in der Simulation präsentiert. Es zeigte sich, dass die optimierten Ergebnisse zu einer Reduzierung der warmzuhaltenden Aluminiummasse in den Dosieröfen führt ohne die Ausfälle der Druckgussmaschinen zu erhöhen.

Während mit der Optimierung die Verteilung des Flüssigaluminiums verbessert wurde, zeigte sich ein weiteres Potential bei der Verteilung des festen Aluminiums auf die Schmelzöfen. Hier zeigt sich deutlich eine Abhängigkeit des Schmelzofenfüllstandes zum spezifischen Energieverbrauch. In weiteren Untersuchungen ist geplant, auch diesen Bereich mit einem Optimierungsmodell zu verbessern.

Literatur

- [1] Schimansky C.: Energiepolitik.
<http://www.bdguss.de/themen/energie/#.WLAU5PJCMQM>. Letzter Zugriff am 15.09.2020
- [2] Schlüter, W.; Henninger, M.; Buswell, A.; Schmidt, J.: Schwachstellenanalyse und Prozess-verbesserung in Nichteisen-Schmelz-und Druckgussbetrieben durch bidirektionale Kopplung eines Materialflussmodells mit einem Energiemodell. Herausgeber: S. Wenzel und T. Peter, kassel university press, Kassel, S. 19-28, 2017.
- [3] Buswell, A.; Schlüter, W.: E|Melt: Erweiterung einer unternehmensspezifischen Materialfluss- und Energiesimulation zur Abbildung variable Betriebsstrukturen der Nichteisen- Schmelz und Druckgussindustrie. In: Tobias Loose (Hg.): Tagungsband Workshop 2018 ASIM/GI-Fachgruppen. Heilbronn 2018.
- [4] Buswell, A; Schlüter, W.: E|Melt: A flexible material flow and energy simulation in the context of Industry 4.0. ASIM 2018 – 24. Symposium Simulationstechnik. Hamburg, 2018.
- [5] März, L; Krug, W; Rose, O.; Weigert, G.: Simulation und Optimierung in Produktion und Logistik, Praxisorientierter Leitfaden mit Fallbeispielen. Berlin, 2011.

Potenziale der Ablaufsimulation für die Entwicklung von einer Pilot- zur Volumenfertigung am Beispiel der Heliatek GmbH

Felix Diener^{1*}, Samuel Horler², Pierre Grzona², Philipp Wilsky²

¹Heliatek GmbH, Treidlerstraße 3, 01139 Dresden, *felix.diener@heliatek.com

²Institut für Betriebswissenschaften und Fabrikssysteme, Professur Fabrikplanung und Fabrikbetrieb, Technische Universität Chemnitz, Erfenschlager Str. 73, 09125 Chemnitz, Deutschland

Abstract

In this paper the potential of an agile simulation proceeding in start-ups is described. The authors from the Chemnitz University of Technology worked together with Heliatek GmbH on the vision of a large scale organic solar film production. To accomplish the project goals, it was necessary to implement an agile simulation process, so that the right information could be gathered and integrated in the simulation environment. Due to this a high knowledge gain in the company and by the project participants could be reached, but also deficits of actual use of simulation tools in SME became visible. At the end, the knowledge gain could be transferred in specific guidance points for SME.

Einleitung

Die Simulation in Produktion und Logistik als Werkzeug der digitalen Fabrik hat sich in vielen Anwendungskontexten bewährt [1]. Gerade für Start-ups sowie kleine und mittlere Unternehmen (KMU) ergeben sich neben den vielfach versprochenen Chancen jedoch auch Hürden für die erfolgreiche Anwendung der Simulation in der Produktion. Insbesondere der Aufwand und das benötigte Wissen zur Erstellung einer Simulationsstudie in kleinen und mittleren Unternehmen bzw. Start-ups stellt eine große Hürde dar. Spezifisches Prozesswissen ist vorhanden, doch fehlt es meist an methodischem Wissen zum Einsatz und Grenzen von Simulationswerkzeugen für die Fabrikplanung.

Ergänzend benennt eine Studie des Bundesministeriums für Wirtschaft und Energie wirtschaftliche Potenziale von Simulation beim Einsatz im Mittelstand. Zusätzlich

werden aber auch Hemmnisse bei einmaligen und laufenden Kosten von Simulationsumgebungen und deren Komplexität, Modularität und Nutzerfreundlichkeit gesehen. [2]

Simulation in Start-Ups und KMU

Beispiele für Einsatzmöglichkeiten von Simulation in KMU sind die durchgehende Planungsbegleitung und operative Produktionsplanung [3, 4]. Die Hürden konnten seit diesen Beiträgen im Vergleich zur aktuelleren Studie des BMWi gesenkt werden, insbesondere durch die angemahte stärkere Einbindung in die Lehre [5]. Ein weiteres aktuelles Beispiel aus dem Jahr 2019 ist die Lern-App PSIMA, die sich mit der Gamification beim Erlernen des Umganges mit der Simulationsumgebung Tecnomatix Plant Simulation beschäftigt [6].

Ein weiterer Baustein ist der Ansatz über webbasierte Simulationsangebote die Investitionshürde für KMU zu senken, beispielsweise über das Projekt simKMU [7]. Eine Untersuchung aus dem Jahr 2014 kam zu dem Ergebnis, dass kein Werkzeug am Markt die Anforderungen in Form eines KMU-gerechten Bausteinkasten erfüllt [8].

Aus der Literatur wird das enorme Potenzial von Simulation deutlich, aber auch, dass noch große Lücken bei der Durchdringung bei KMU und dementsprechend bei Start-ups als Teil dieser Gruppe vorherrschen. Die Erstellung von Simulationsstudien im Rahmen von Studienabschlussarbeiten ist hierbei ein probates Mittel, um den Aufwand für KMU zu reduzieren und strategisch Simulationswissen im Unternehmen aufzubauen.

Ausgangssituation bei der Heliatek GmbH

Die Heliatek GmbH ist ein Hersteller von innovativen organischen Dünnschicht-Solarlösungen. Zu den Produkten zählen flexible, gebrauchsfertige Solarmodule für die Bauindustrie. Die weltweit erste qualitativ hochwertige Vakuum-Roll-to-Roll-Produktion garantiert die versprochenen Moduleigenschaften, stellt jedoch besondere Anforderungen an die Produktionsgestaltung.

Derzeit werden umfangreiche Pläne für die Erweiterung des Produktspektrums und hin zu einer Skalierung der Produktion umgesetzt. Die in der Start-up-Phase des Unternehmens realisierte Pilot-Fertigung soll deswegen um eine Volumenproduktion ergänzt werden. Die Planung und Umsetzung dieses neuen Produktionsbereiches sind aufgrund des Neuheitsgrades der Produkte und damit einhergehender Anlagenkomplexität sowie des starken Wachstums innerhalb des Unternehmens herausfordernd. Da keine vergleichbaren Best Practices existieren, können wichtige Erfahrungen häufig erst während der Planung gemacht werden. Um die Materialflüsse im neuen Produktionsbereich vorzudenken und zu optimieren, wurde eine Simulation angestrebt. Dabei wurden folgende Ziele verfolgt:

- Analyse der Transportmittel- und Anlagenauslastungen
- Aufzeigen von möglichen Engpässen
- Dimensionierung von Transportmitteln
- Visualisierung der Prozessabläufe

Entsprechend dieser Ziele, wurde gemeinsam mit der TU Chemnitz seit 2018 eine simulative Betrachtung durchgeführt und gleichzeitig untersucht, welche ergänzenden Potenziale mit dem Einsatz des Werkzeuges in einem stark wachsenden Unternehmensumfeld gehoben werden können.

So sollte neben der Erreichung der definierten Simulationsziele überprüft werden, welchen Nutzen, welche Chancen aber auch welche etwaigen Grenzen die Simulation speziell für Start-ups und KMU besitzt. Zu diesem Zweck wurde entschieden, über eine Befragung von Angestellten der Heliatek GmbH Aussagen zu diesen Fragestellungen zu erhalten. Im Nachgang wurden die Erwartungen der Beteiligten vor der Studie mit den tatsächlich generierten Ergebnissen abgeglichen.

Hierzu wurde ein Fragebogen entworfen und von ausgewählten Angestellten ausgefüllt. Die zentralen Fragestellungen adressierten folgende Punkte:

- Anforderungen/Wünsche/Erwartungen/Ziele an die Studie zur Simulation des Materialflusses
- konkrete Fragestellungen zu denen ein Erkenntnisgewinn erwartet wird
- Grenzen der Simulation bei der Heliatek GmbH
- Abstraktion vom konkreten Anwendungsfall der Heliatek GmbH auf allgemeine Chancen und Probleme/Grenzen bei Start-Ups

Im Folgenden werden die Ergebnisse der Befragung zusammengefasst. Generell sollte ein Wissensaufbau zu den erwarteten Materialflüssen durch deren Analyse stattfinden, und es sollten greifbare Ergebnisse aus dem dynamischen Modell generiert werden. Die Anforderungen und Erwartungen an die Simulation decken sich zum großen Teil mit den Zielen und werden detailliert im Rahmen der Phase 1 der gewählten Vorgehensweise erläutert.

Im Anschluss an die Fragestellungen sollten die Probanden einschätzen, wo aus ihrer Sicht Grenzen der Simulationsstudie für die Heliatek GmbH sind. Hier wurde der Fakt herausgestellt, dass es sich um eine in dieser Größenordnung komplett neue Produktionsstrecke handelt und keine Datenbasis oder Erfahrungswerte zur Verfügung standen und so viele Inputdaten auf Annahmen basierten und eventuell nicht die Realität abbilden. Als weiterer Punkt wurde hier der begrenzte Detaillierungsgrad benannt, durch den die hochkomplexen Prozesse sowie die einzelnen Entstehungsstufen des Produkts nur schwer abzubilden waren.

Ausgehend von den getätigten Aussagen bezüglich der konkreten Simulationsstudie für das Unternehmen, wurden die Befragten anschließend gebeten zu abstrahieren, welche Möglichkeiten und auch Probleme sie generell beim Einsatz von Simulation in Start-ups und KMU sehen. Bezüglich der Möglichkeiten wurde die bessere Planbarkeit von Logistik und Produktion sowie die Beseitigung von Unsicherheiten im Vorfeld des Aufbaus einer neuen Produktionsstrecke erwähnt. Außerdem wurde hier die Erleichterung bei der Erarbeitung von Materialflusskonzepten sowie die Möglichkeit einer umfassenden Prozessanalyse genannt.

Bei der Einschätzung von Grenzen und Problemen für Start-ups und KMU wurden die hohen Kosten für den Erwerb der Simulationssoftware und die erforderlichen Lizenzen genannt. Des Weiteren haben die Befragten den hohen zeitlichen Aufwand und das eventuell nicht vorhandene Know-how zur Erstellung von Simulationsmodellen angeführt. Zuletzt wurde noch das fehlende Verständnis zum Nutzen von Simulationsanwendungen für KMU angemerkt.

Vorgehensweise bei der Simulationsstudie bei der Heliatek GmbH

Nachdem der Einsatz der Ablaufsimulation in KMU ausgehend von Wissenschaft und Praxis am Beispiel der Heliatek GmbH beleuchtet wurde, kann resümiert werden, dass deren Einsatz unter gegebenen Bedingungen erschwert ist. Aus diesem Grund wurde für die Simulationsstudie ein eigens konzipiertes Vorgehenskonzept ausgearbeitet und zur Anwendung gebracht.

Der nachfolgenden Ausführungen beschreiben das Vorgehen, die Herausforderungen und Ergebnisse der damit durchgeführten Simulationsstudie. Dabei werden insbesondere die aus der vorherrschenden Situation resultierenden Hürden beschrieben sowie transparent dargelegt, welchen Beitrag die Simulation im Planungsprozess des neuen Fertigungsbereiches leisten konnte. Der Beitrag adressiert also zudem den Nutzen der Simulationsstudie, welcher über die unmittelbaren Simulationsergebnisse hinausgeht und arbeitet weiterhin Handlungsempfehlungen für vergleichbare Problemstellungen heraus.

Das Vorgehen wurde in Anlehnung an das bewährte Simulationsvorgehensmodell nach VDI 3633 gewählt [9]. Basierend auf diesem Rahmen, wurden besonders die Schritte der *Zielbeschreibung*, *Systemanalyse* und *Experimente* um Formate ergänzt, die ein systematisches Einbeziehen der relevanten Beteiligten bei Heliatek sicherstellte. Im Speziellen wurden mehrere interaktive Ziel- und Konzept-Workshops durchgeführt, deren Nutzen nicht nur für die eigentliche Simulationsstudie nachgewiesen werden konnte. Das Konzeptmodell wurde in einer an SysML angelehnten Darstellung realisiert – die Umsetzung des ausführbaren Modells erfolgte in Tecnomatix Plant Simulation.

Abbildung 1 stellt das Zusammenarbeitsmodell der Beteiligten bzgl. deren Mitwirkung in den entscheidenden

den Phasen dar. Dieses Vorgehen wurde in Anlehnung an den partizipativen Planungsprozess nach Schenk et. al. gestaltet [10]. Im Kern zielt es darauf ab, möglichst frühzeitig die relevanten Beteiligten in partizipativen Formaten an der Ideenfindung und Lösungspräzisierung teilhaben zu lassen. Als Resultat lassen sich die Planungszeit reduzieren und Fehler vermeiden. In der Simulationsstudie hat sich der gezeigte Dreiklang aus Fachexperten, Simulationsexperten sowie aus wissenschaftlicher Unterstützung bewährt.

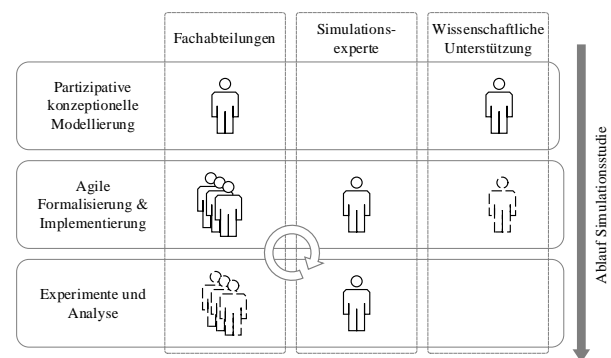


Figure 1: Partizipatives Simulationsvorgehen in Start-ups

Nachfolgend werden die zwei wesentlichen Phasen des Simulationsvorgehens beschrieben.

PHASE 1: Partizipative top-down Konzeptmodellerstellung

Diese Phase startete mit der Aufgaben- und Zieldefinition der Simulationsstudie. Kern dieser waren interaktive Workshops, an welchem relevante Wissens- und Entscheidungsträger des Unternehmens eingebunden wurden. Dieser Tages-Workshop diente dem Zweck der Zielfindung und Aufgabendefinition für die Konzeptmodellerstellung.

Folgende inhaltliche Fragestellungen waren Bestandteil dieses ersten Schrittes der partizipativen Erarbeitung:

- Einführung in die Fertigungs- und Logistiksimulation sowie Schaffung eines simulativen Grundverständnisses unter Aufzeigen von Einsatzmöglichkeiten und Grenzen
- Gemeinsame Erarbeitung von Zielstellungen, die mit der Simulation verfolgt werden
- Einstieg in die Top-Down-Systemanalyse: Definition von Systemgrenzen und Systemelementen

Mittels einer offenen und später moderierten Frage-
runde in Abstimmung aller Beteiligten wurden potenzi-
elle Zielstellungen gesammelt, priorisiert und kategori-
siert. Als grundsätzliches Ergebnisziel konnte die An-
ordnung und Dimensionierung von Logistikmitteln,
Lagerflächen sowie Personal im neu zu gestaltenden
Fertigungsbereich (FAB2) identifiziert werden. Die
weiteren Leistungsziele, welche für das Erreichen der
Zielstellung nötig sind, wurden darin unterschieden, ob
das Werkzeug der Simulation unmittelbar einen Mehr-
wert für diese Problemstellung bieten kann. Abbildung
2 stellt die im Zielworkshop erarbeiteten Ergebnis- und
Leistungsziele der Simulationsstudie zusammen.

Validierung Planungsstand	Validierung Output/Leistungs- fähigkeit
Dimensionierung Transportmittel	Transparenz Anlagenauslastung
Dimensionierung Lagerkapazitäten	Dimensionierung Puffer
Dimensionierung Eingangslager/ Ausgangslager	Lokalisierung & Dimensionierung Shaft Puller
Dimensionierung Mitarbeiterkapazität	Optimierung der Transportwege (Materialfluss)
Skalierung, Leistungssteigerung 2019, 2020, 2021	Transparenz über Materialfluss
Identifikation bottlenecks	Transparenz über Entsorgung/ Müllaufkommen

Figure 2: Ergebnis-/Leistungsziele der Studie

Ergänzend wurden weitere ‚Randthemen‘ aufge-
nommen, welche durch das Unternehmen in den Pla-
nungsaktivitäten außerhalb dieses Projektes bearbeitet
werden. Zusammenfassend wurde eine Entscheidungs-
unterstützung in folgenden Kategorien angestrebt:

- Transparenz der Fabrikabläufe auf Grundlage
des aktuellen Planungsstandes
- Dimensionierung der Fertigungs- und Logis-
tikmittel
- Anordnung ortsfester Fertigungsmittel

Aus den kollaborativ erarbeiteten Zielen leiteten sich
die für die Erstellung des Konzeptmodells nötigen Ein-
gangsdaten und Parameter ab. Weiterhin wurden die für
die später durchzuführende Interpretation wichtigen

Messgrößen identifiziert. Auch hier zeigte sich, dass das
frühzeitige Einbinden der Entscheidungsebene Missver-
ständnisse und falsch eingesetzte Ressourcen vorbeugt.

Die mit der Leitungsebene im Rahmen des Ziel-
workshops abgestimmten Ergebnisse, wurden im nach-
folgenden Detailworkshop mittels einer Systemanalyse
vertieft. Der inhaltliche Input erfolgte durch Fachexper-
ten aus den Bereichen Supply Chain Management,
Ramp-Up, Produktionsplanung, Logistik sowie Fabri-
kintegration. Methodisch wurde der Workshop durch
die Durchführenden der TU Chemnitz geleitet. Für die
sich aus den Zielstellungen sowie den Vorarbeiten des
Unternehmens ergebenden Objektklassen (Simulations-
elemente) wurden jeweils in Form von Expertengesprä-
chen sequenziell folgende Inhalte erarbeitet

- Relevanz des Fabrikobjektes für die Simulation
- Festlegung des nötigen Abstraktionsgrades
- Definition nötiger Simulationsparameter sowie
Inputs und Outputs
- Punktuelle Detaillierung des Ablaufverhaltens

In Theorie und Praxis hat es sich bewährt, das Kon-
zeptmodell anhand folgender vier Schritte nach Mög-
lichkeit sequenziell erarbeitet – auch der Detail-
workshop lehnte sich an diese Schritte an:

- Hierarchie und Abgrenzung gegen die Sys-
temumgebung
- Funktionale Analyse der Systemelemente
- Strukturelle Analyse der Systemelemente
- Analyse von Strategien und Prozessregeln

Aufgrund des nötigen Detaillierungsgrades bei
gleichzeitig hoher Flexibilität wurde sich entschieden,
die konzeptionelle und formale Modellierung in einem
Modell zusammenzuführen. Für die Darstellung kam
die grafische Modellierungssprache SysML zum Ein-
satz. Die darin bereitgestellten Diagramme eignen sich
besonders für die Visualisierung komplexer Systeme.
Damit kann die Aufbau- und auch die Ablaufstruktur
modelliert und Zusammenhänge zwischen verschiede-
nen Darstellungen hergestellt werden. Für den ersten
Teil des Zielworkshops wurden im Sinne der Schritte 1
und 2 sogenannte ‚Blöcke‘ für feste und bewegliche
Fabrikobjekte vorbereitet (Blockdiagramm). Mithilfe
dieser wurden die relevanten Elemente der Simulation

abgegrenzt sowie deren Funktion im Simulationsmodell detailliert in Abstimmung mit den Teilnehmern vertieft.

Die Ablaufstruktur stellt die Materialflussprozesse zwischen den Elementen eines Systems dar. Im Fall der Heliatek GmbH wurden die örtlich getrennten Fabrikräume und die Wege sowie Lager als Swimlanes abgebildet. Innerhalb dieser Swimlanes wurden die Lager und Betriebsmittel als Blöcke, die Materialflüsse, also die Logistikkittel und transportierten Elemente, als Pfeile dargestellt. Abbildung 3 zeigt einen Ausschnitt aus dem Aktivitätendiagramm.

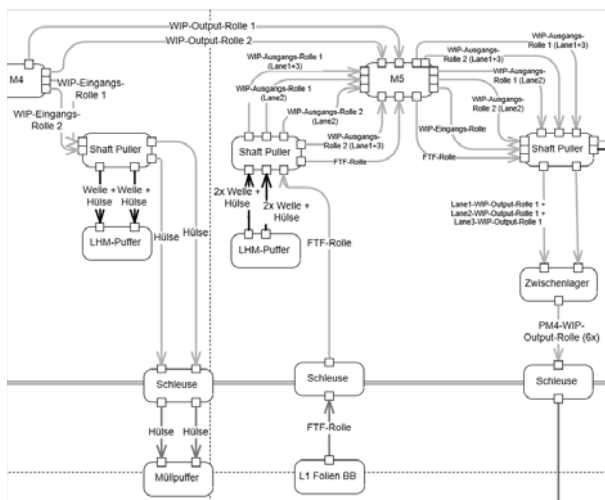


Figure 3: Ablaufstruktur modelliert als Aktivitätendiagramm

In der sich anschließenden Aufarbeitungsphase wurden die zusammengetragenen Informationen seitens der TU Chemnitz auf den Simulationsnutzen geprüft und auf dieser Basis das Konzeptmodell erarbeitet.

PHASE 2: Agile Modellerstellung und Simulation

Nach der Erarbeitung des Konzeptmodells sollte die in Planung befindliche Produktionslinie simulativ untersucht werden. Im Rahmen einer studentischen Abschlussarbeit wurde eine Materialflussanalyse durchgeführt.

Neben den in Phase 1 gezeigten Zielen, sollte weiterhin überprüft werden, inwieweit die Erwartungen und Zielstellungen der vorherigen Befragung im Unternehmen erfüllt wurden und welchen Mehrwert die Simulation im konkreten Fall bieten konnte.

Die Materialflusssimulation erfolgte vor der Fertigstel-

lung der neuen Produktionsstrecke, um das Verhalten des Systems kennenzulernen und Rückschlüsse ziehen zu können, bevor es aufgebaut ist. Dies erfolgte in Vorbereitung des bevorstehenden Volumen-Ramp-up.

An die baulichen Bedingungen und Restriktionen am Standort in Dresden angepasst sowie aufgrund der vorherrschenden Produktstrukturen der einzelnen Produkte, wurde die Produktion auf zwei separate Hallenbereiche aufgeteilt. Sechs der insgesamt neun Produktionsmaschinen befinden sich im ersten Hallenteil. Hier findet die Basisfertigung statt. Die restlichen drei Maschinen stehen im nachgelagerten Hallenbereich, in welchem das Finishing der Rollen umgesetzt wird. Gefertigt werden Solarmodule im Rolle-zu-Rolle-Verfahren welche in Breite und Länge variabel sind. Es können Breiten zwischen 0,3m und 1,2m sowie Längen zwischen 2m und 12m gefertigt werden.

Anknüpfend an das Konzeptmodells erfolgte anschließend die quantitative Datenbeschaffung nach Vorlage der im Konzeptmodell erarbeiteten Systemelemente. Die agile Datenbeschaffung gestaltete sich dermaßen, dass mit den jeweiligen Beteiligten die spezifischen Szenarien der Parameter erarbeitet und geplant wurden in Form von Einzelgesprächen. Zur einfachen Daten-Dokumentation erfolgte die Erhebung in Form von Excel-Sheets, um diese anschließend als Parameter in die Simulation zu überführen.

Im Konzeptmodell war unter anderem die Gesamtheit der Materialflüsse über die komplette Produktionsstrecke sowie den Lagerbereich erfasst. Dies diente als Grundlage zur Abbildung der Flüsse im Simulationsmodell. Zusätzlich wurden anhand von internen Planungs-dokumenten die Rüst-, Bearbeitungs- sowie Wartungszeiten ermittelt und für das Modell aufbereitet. Weiterhin wurden Verfügbarkeiten pro Maschine in Abstimmung mit den Lieferanten ermittelt, da zum Zeitpunkt der Studie noch keine eigenen Erfahrungswerte dahingehend vorlagen.

Die Modellierung erfolgte anhand der zuvor generierten Daten und des Konzeptmodells.

Nach Fertigstellung des Modells wurden verschiedene Szenarien simuliert. Der Fokus lag hierbei auf der Untersuchung verschiedener Produkte und der Abhängigkeit von Maschinenauslastungen, Ausbringung und Durchlaufzeiten von den jeweiligen Produkten. Nach Rücksprache mit dem Unternehmen wurde sich im Rahmen der Simulationsversuche auf den ersten Hallen-

teil fokussiert. Das komplette Modell wurde abschließend in einem Testszenario auf seine Funktionalität verifiziert. Im Weiteren werden nun die wichtigsten Ergebnisse der Simulationsversuche zum ersten Hallenteil in Abgleich mit den Erkenntnissen der vorangegangenen Befragungen erläutert.

Eine grundlegende Forderung war der allgemeine Erkenntnisgewinn über die Zusammenhänge des späteren Produktionssystems und die Generierung greifbarer Ergebnisse aus dem dynamischen Modell. Der Erkenntnisgewinn für das Unternehmen begann hierbei nicht erst bei der Durchführung und Auswertung der Versuche, sondern schon bei der agilen Datenaufnahme und deren statistischen Auswertung. Dies brachte bereits im Vorlauf der Versuche verwertbare Ergebnisse. So wurden für diverse Produkte die benötigten Bearbeitungszeiten ermittelt und erste Erkenntnisse über den Einfluss der Modullängen auf diese gewonnen. Die Maschine 2 ist mit dem Verdampfen der Organikmaterialien die maßgebende Anlage im Produktionsablauf. Ziel der Produktionsplanung ist es daher, diese Maschine möglichst auszulasten. Die Berechnungen haben ergeben, dass beim Fertigen kürzerer Module klare Tendenzen bestehen, dass andere Maschinen im Bottleneck liegen. Diese Feststellung wurde durch die Simulationsversuche untermauert. Ein Beispiel einer solchen Auswertung wird in der folgenden Abbildung gezeigt.

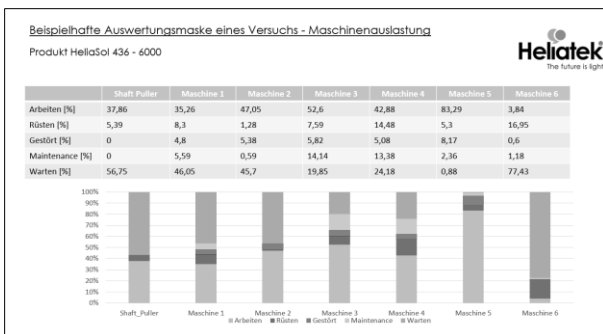


Figure 4: Produktspezifische Ergebnisauswertung

Für dieses Szenario ist die Maschine 5 die Anlage im Engpass und somit von besonderer Bedeutung bei der Planung und Steuerung der Produktionsprozesse. Durch die Kombination der statischen Berechnungen und die nachfolgenden Simulationsversuche konnten die jeweiligen Maschinen im Bottleneck sowie die allgemeinen Auslastungen der Anlagen ermittelt werden.

Außerdem wurden die zu verarbeitenden Rollenlängen und Stückzahl je Produkt errechnet, welche nötig sind, um die jeweils gewünschte Maschine maximal auszulasten. Anschließend konnten diese Erkenntnisse durch Überprüfung mittels der Simulation bestätigt werden. So konnte die Simulation auch bei der Fragestellung nach der Identifizierung von Bottlenecks Antworten liefern. Dieses Vorgehen erfolgte in agilen Iterations Schleifen.

Von Interesse war außerdem die hinreichende Auslegung der Logistikmittel. So wurde deren relative Auslastung für die Versuche ausgewertet. Darüber hinaus wurden die Gabelstapler und der Shaft-Puller gesonderten Auswertungen unterzogen, um zu überprüfen, ob die derzeitige Planung hinsichtlich der Logistikmittel ausreichend ist. Der Shaft-Puller ist ein Hilfsmittel zum Handling der Hülsen, Wellen und kompletten Rollen und dient zum Vor- und Nachbereiten der Rollen.

Es hat sich gezeigt, dass die Stapler und der Shaft-Puller Tendenzen aufweisen, bei bestimmtem Systemkonstellationen zum Engpass werden. Hier bedarf es weiterer Untersuchungen, ob durch gezielte technische oder organisatorische Maßnahmen eine Beseitigung bzw. Verringerung dieser Tendenzen erreicht werden kann.

Neben den Anforderungen an die Studie wurden auch erwartete Grenzen und Probleme im Rahmen der Befragung aufgenommen. Hier wurde die Abbildung der komplexen Produktionsprozesse als mögliche Problemquelle beschrieben. In der Tat war die Realisierung der Prozesse mit den Mitteln in *Plant Simulation* eine der maßgeblichsten Herausforderungen beim Erstellen des Modells. Die Erstellung eines solchen Modells beansprucht einen nicht zu vernachlässigenden Zeitaufwand und ist für Start-ups und KMU ein zu berücksichtigender Faktor bei der Evaluierung, ob Simulation als Unterstützung infrage kommt. Eine weitere Problematik bei der vorliegenden Simulationsstudie lag in der fehlenden Datenbasis für die Inputangaben des Modells. Der Aufbau der Produktionsstrecke erfolgt in dieser Größenordnung erstmalig. Die installierten Anlagen sind allesamt Sonderanlagen, und so bestehen keine Erfahrungswerte hinsichtlich der Anlagentechnik und der Fertigungsprozesse. Viele Inputdaten beruhen daher auf Annahmen und sind im weiteren Verlauf stetig auf Richtigkeit zu überprüfen und zu aktualisieren. Hierbei haben die Szenariotechnik sowie das agile Vorgehen geholfen.

Handlungsempfehlungen für Start-Ups und KMU

Zusammenfassend wird festgehalten, dass bezüglich der Chancen und Grenzen bei dem Einsatz von Simulation für Start-ups sowie kleine und mittlere Unternehmen noch weiterer Forschungsarbeit bedarf. Die Fachliteratur gibt nur begrenzt Aufschluss darüber, was Simulation konkret für diese Art von Unternehmen leisten kann und unter welchen Rahmenbedingungen ein Einsatz sinnvoll ist. Die größten Hindernisse, die einem breiten Einsatz von Simulationsanwendungen in den Unternehmen im Weg stehen, sind mit den hohen Anschaffungs- und Lizenzkosten sowie den zu komplexen Software-Tools und deren benötigtem Expertenwissen identifiziert worden. Gelingt es, die Rahmenbedingungen für einen Einsatz von Simulation für solche Unternehmen zu verbessern und Simulationsanwendungen zu entwickeln, welche auf die Bedürfnisse dieser ausgerichtet sind, kann Simulation eine entscheidende Unterstützung bei der Planung und Auslegung von Produktionsstätten für Start-ups und KMU sein.

Neben den allgemeinen Erkenntnissen kann für KMU zukünftige Handlungsempfehlungen abgeleitet werden:

- Die Gültigkeit der VDI 3633 kann auch in Start-ups/KMU bestätigt werden.
- Ein Top-Down-Vorgehen mit der Einbindung aller Entscheidungsebenen wird empfohlen.
- Aufgrund der oft heterogenen und dynamischen Datenlage, sind alle relevanten Wissensträger im Unternehmen möglichst früh in den Prozess einzubinden - dafür haben sich partizipative Methoden bewährt.
- Der Zweck sowie die Möglichkeiten und Grenzen der Simulation müssen häufig erst noch vermittelt werden - überambitionierte Zielstellungen sind zu vermeiden.
- Während der Modellierung ist mit vermehrten Änderungen in den Anforderungen und folglich im Modellaufbau zu rechnen. Aus diesem Grund bietet sich die agile, modulare Modellierung im ausführbaren Modell an.
- Das Simulationsmodell sollte als mitwachsender dauerhafter Begleiter agieren, mit dem ad hoc Simulation durchgeführt werden können (Erweiterung zum digitalen Zwilling sollte geprüft werden).

- In der Zusammenarbeit mit Hochschulen kann im Unternehmen systematisch Simulation-Knowhow aufgebaut werden, was auch nach Beendigung der Studie Nutzen stiftet.

Literaturverzeichnis

- [1] BRACHT, Uwe ; GECKLER, Dieter ; WENZEL, Sigrid: *Digitale Fabrik : Methoden und Praxisbeispiele*. 2., aktualisierte und erweiterte Auflage. Berlin : Springer Vieweg, 2018 (VDI-Buch)
- [2] BISCHOFF, Jürgen ; TAPHORN, Christoph ; WOLTER, Denise ; BRAUN, Nomo ; FELLBAUM, Manfred ; GOLOVEROV, Alexander ; LUDWIG, S. ; HEGMANN, T. ; PRASSE, C. ; HENKE, M. ; OTHERS: *Erschließen der Potenziale der Anwendung von Industrie 4.0 im Mittelstand*. In: Berlin: Studie BMWi (2015)
- [3] SPIECKERMANN, Sven: *Durchgängige Planungsbegleitung mit Simulation im Mittelstand*. In: ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb 103 (2008), 1-2, S. 83–85
- [4] SCHULZ, Andreas ; ZÜFLE, Edgar ; SOMMER, Lutz ; HAUG, Manuel: *Simulation in der operativen Produktionsplanung – Erfolgsfaktoren für KMU*. In: ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb 102 (2007), 1-2, S. 32–36
- [5] SOMMER, Lutz ; PLANKENHORN, Andreas: *Simulation – Verborgene Chancen für den Mittelstand*. In: ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb 99 (2004), Nr. 6, S. 303–305
- [6] SCHUMACHER, Bastian Clemens: *Beitrag zur Einarbeitung in ereignisdiskrete Simulation zur Neu- und Umplanung von Materialflusssystemen*. Berlin, Technische Universität Berlin, Fakultät V - Verkehrs- und Maschinensysteme. Dissertation. 2019-12-12. URL <https://dnb.info/1205804277/34>
- [7] WIMPF, Daniel-Percy ; SEITZ, Stefan ; ERGIN, Tamer ; ROMMEL, Steve ; HERMANN, Marco: *simKMU - Internet- und webbasierte Simulationsdienste : Forschungsergebnisse im Teilvorhaben »Grundlagen, Konzeption und Intralogistik« des Verbundprojekts simKMU*. Abschlussbericht. Stuttgart, 2011
- [8] RUDEL, Steffi: *Prozess-Simulation in kleinen und mittleren Unternehmen mittels des Baukastensystems KMUSimMetall*. In: WENZEL, Sigrid ; PETER, Tim (Hrsg.): *Simulation in Produktion und Logis-*

tik 2017 : kassel university press GmbH, 2017

- [9] Technische Regel VDI 3633 Blatt 1:2014-12.
Simulation von Logistik-, Materialfluss- und Produktionssystemen - Grundlagen
- [10] SCHENK, Michael ; WIRTH, Siegfried ; MÜLLER, Egon: *Fabrikplanung und Fabrikbetrieb : Methoden für die wandlungsfähige, vernetzte und ressourceneffiziente Fabrik*. 2., vollständig überarbeitete und erweiterte Auflage. Berlin, Heidelberg : Springer-Vieweg, 2014 (VDI-Buch)

Simulation-based Analysis of Dispatching Methods on Seaport Container Terminals

Anne Schwientek^{1*}, Ann-Kathrin Lange¹, Carlos Jahn¹

¹ Institute of Maritime Logistics, Hamburg University of Technology, Am Schwarzenberg-Campus 4, 21073 Hamburg, Germany; *a.schwientek@tuhh.de

Abstract. Horizontal transport on a container terminal is the interface between quayside and yard. Efficient transport operations between these two areas are an important issue for terminal operators to achieve a high terminal performance. Thereby, an essential task is to dispatch the respective vehicles to pending container transport orders. Despite the large amount of literature in this field, there is no systematic approach investigating the effects of terminal parameters on the performance of dispatching methods. This discrete event simulation study aims to close that research gap for the case of different dynamic dispatching methods. It shows that the performance of a dispatching method depends highly on the terminal's objectives (e.g. maximizing crane productivity, minimizing driven distances). In addition, it reveals that some terminal parameters influence the performance ranking of the dispatching methods. This implies that there is no dominant dispatching method; the choice should rather depend on the specific terminal objectives and parameters.

Introduction

Seaborne containerized trade increases constantly (except in 2009). In 2018, 152 million TEU (Twenty-foot Equivalent Unit) were shipped around the world in contrast to 110 million TEU in 2010 [1]. Within the maritime supply chain, container terminals are interfaces between sea transport and the hinterland, where the handling operations between modes take place and goods are stored. Therefore, they have a high importance for the overall success of maritime transports. The increasing container volumes demand a constant improvement of the terminal parameters, especially of storage space and handling processes.

Besides the increasing cargo volumes, there is another challenging trend for container terminals: vessel sizes grow without cease. Larger vessels make higher demands on ports and terminals in various kinds [2]. On the one hand, physical enlargements of the existing superstructure are necessary to suffice the new vessel dimensions. The port basin and access need to show enough water depth for the vessels' draught and enough width to

enable vessels to turn or to pass each other. The terminals' ship-to-shore cranes (STS) need a sufficiently large boom. On the other hand, the handling processes on the terminal need to be updated to fit the changed requirements. As the container carriers mainly expect similar handling times for the Ultra Large Container Ships as for the smaller ones, the terminals need to ensure fast operations at the quayside to stay competitive. Therefore, the terminal equipment has to be able to move a large amount of discharged containers fast enough from the quayside to the yard (or equivalently from the yard to the quayside) and the storage area has to be able to cope with high peak situations. Furthermore, the strategies in the yard have to be optimized to ensure short container handling times with the minimal amount of container rehandling. These challenges lead to the necessity to optimise terminal operations regarding speed and efficiency.

1 Container Terminal Operations

To organize container handling processes, container terminals comprise different main functional areas [3, 4]: Quayside, horizontal transport, storage area and landside.

The quayside includes vessels, berths and quay wall. Processes at the quayside are mainly vessel loading and discharging. The term horizontal transport refers to the transport operations between quayside and storage area. In the storage area, either empty or full containers are stored in blocks for a short time period until they are loaded on a truck, train or vessel. The landside consists of the gate, where external trucks enter the terminal area, and the truck and train loading area together with the corresponding processes. In each of these areas, different types of equipment can be used, depending especially on the size, the location and the required productivity of the terminal.

On the quayside, the typical equipment type are STS [5]. Some terminals employ mobile harbour cranes (MHC). While STS productivity ranges usually between

25 and 35 moves/hour, MHC productivity is only between 15 and 20 moves/hour [3].

Horizontal transport equipment is differentiated between active and passive equipment [4]. Active equipment such as Straddle Carriers (SC) is able to lift a container independently. In contrast, passive equipment like Tractor-Trailer-Units (TT) or Automated Guided Vehicles (AGV) needs another equipment type (e.g. a gantry crane or a STS) to be loaded or unloaded. This implies that more equipment is needed to achieve the same STS productivity if passive equipment is used compared to active equipment as the next vehicle should be always available for the STS to avoid waiting times of the most expensive equipment type [6].

The equipment type used in the storage area depends on various factors such as e.g. the terminal volumes or the required yard capacity. Smaller terminals usually employ Reach Stackers (RS) as they are flexible and have low investment costs. SC are typically employed in medium-sized terminals. They are flexible and productive, but they usually allow only a maximum stacking height of three containers leading to a comparatively high demand for ground space. Other typical equipment types are Rubber-Tyred Gantry Cranes (RTG) and Rail-Mounted Gantry Cranes (RMG). RMG are more productive, have larger span widths and higher stacking heights. Contrarily, RTG are less expensive and more flexible as they are not rail-mounted and, therefore, can change yard blocks whenever necessary [3].

Regarding the landside, external trucks often enter the terminal and bring or collect one or two container(s) directly to/from the storage area. Therefore, for truck (un)loading, the respective storage equipment is used. For train (un)loading, RMG, RS, or SC are employed.

Depending on the employed equipment, the terminal system is defined. Thereby, the terminal system is understood as the combination of technical equipment used for container handling operations [7]. The most common terminal systems are the RTG/TT-system (more than 50 % worldwide) and the SC-system (20-25 %) [8]. Therefore, these two systems are chosen for the following investigation.

2 Dispatching in Horizontal Transport

As discussed above, the term horizontal transport refers to the interface between quayside and yard as one of the

main functional areas of a container terminal. It is an essential process to allow for an efficient service to vessels and, therefore, enable fast and reliable quayside operations. Regarding horizontal transport, one main decision problems is the dispatching problem.

Dispatching aims to find an efficient assignment (and sometimes sequence) of transport orders and vehicles (e.g. TT or SC) for a defined time to fulfil the given objective. Typical objectives are to serve the STS continuously, to minimize the driven distances of vehicles or to reduce the amount of used vehicles. Thereby, dynamic and static dispatching can be distinguished [9]. The dynamic approach triggers the assignment at certain events, e.g. if a transport order is completed and a vehicle is available or at the beginning of a shift. This approach is very flexible but also myopic. On the contrary, the static approach (often also called scheduling) generates a long-term plan based on estimates for arrival and operation times. The plan can be optimized in advance using mathematical methods, but is normally not altered during its execution. However, it is highly dependent on the quality of the time estimates [10]. There are also hybrid forms using a static approach with a short rolling horizon. Container terminal processes are quite stochastic. For example, the terminal workload fluctuates depending on the arrival rates of vessels, trucks and trains. Furthermore, important equipment can fail. Therefore, this study focuses on dynamic dispatching methods.

There are several literature reviews on dispatching and scheduling on container terminals. Kizilay and Eliiyi [11] provide a recent overview on quay, yard and integrated scheduling problems. Huang et al. [12] focus on resource allocation problems on container terminals. A detailed analysis of the dispatching literature on container terminals is provided in Schwientek et al. [13].

The respective publications usually develop an own dispatching method (e.g. [14–16]) or compare a few methods for a specific terminal (e.g. [17–20]). Zeng et al. [21] and Liu and Ioannou [19] show that the number of available vehicles influences the dispatching method performance ranking. Garro et al. [22] discuss that dispatching methods developed for AGV are not suitable for SC due to different processes. Schwientek et al. [23] investigate five terminal parameters and three dispatching methods.

The interrelations between terminal parameters and dispatching method performance are so far not investigated systematically. Thus, it is possible that a dispatching method chosen under certain terminal conditions is

no longer the best solution if the terminal conditions change due to the rather unpredictable environment. Therefore, the research question is, whether certain terminal parameters affect the performance of a dispatching method and if yes, which ones are most important parameters. A simulation study is conducted to approach that question.

3 Simulation Study

The simulation study bases on a reference terminal that is implemented in Tecnomatix Plant Simulation using modified data of a real terminal. Both terminal parameters and dispatching methods are varied to investigate the effects of the respective combinations on the efficiency of the horizontal transport.

The reference container terminal reflects a typical RTG/TT-terminal. It focuses on the horizontal transport area framed by the quayside and the yard area. The simulated terminal has a capacity of 1,600,000 TEU, the quay length is 800 m. Eight STS are installed. The yard comprises 20 blocks (6x28 TEU) parallel to the quay wall. For every yard block, there is a RTG available. There are 40 TT (five per STS) on the terminal. Three different vessel types arrive at the terminal: Feeder vessel and two types of deep-sea vessels. Feeder vessels discharge and load on average 500 containers, smaller deep-sea vessels discharge and load on average 1,250 containers and larger deep-sea vessels discharge and load on average 3,500 containers.

The choice of dispatching methods, terminal parameters and terminal objectives bases on a preceding literature analysis [13]. The terminal objectives typically relate to an object such as a vessel, STS, or vehicle. Examples for objectives are: to minimize the makespan of a vessel, to minimize the departure delay, to maximize STS productivity or to minimize STS wait time, to maximize vehicle productivity or to minimize vehicle wait time, travel distance or fleet size. The majority of dispatching studies focuses on vessel- or vehicle-related objectives [13]. Thus, output parameters of this study are exemplarily STS productivity and distances driven by the vehicles.

3.1 Investigated Dispatching Methods

Five different dynamic dispatching methods are investigated: fixed assignment of vehicles to a STS, distance-based, inventory-based, time-based, and a hybrid method. The chosen dispatching methods showed a good

performance in comparison to others in earlier studies as described in the following.

The fixed method is the most used method in practice. Thereby, a fixed number of vehicles is assigned to a particular STS serving only this specific STS. This is a very simple method which is easy to implement and comfortable to the terminal personnel. However, the fixed method is highly inefficient as roughly half of the trips between quay and yard are empty trips (assuming single-cycle operation of the STS, i.e. discharging and loading are separate processes).

Studies by Kim and Bae [24], Koster et al. [20], and Meer [25] recommend a distance-based dispatching method. The implemented distance-based method aims to minimize the driven distances of the vehicle by always assigning the transport order with the nearest starting location.

The inventory-based method is a dispatching method referring to Briskorn et al. [16]. It was evaluated positively in several other publications. Thereby, an available vehicle is assigned to a transport order that belongs to the STS with the smallest inventory. The inventory of a STS is understood as the number of transport orders that is already assigned to this specific STS.

Cao et al. [26] state that a time-based method performs higher than a genetic algorithm. The implemented time-based dispatching method chooses the transport order for a vehicle that waits longest for assignment. It focuses like the inventory-based method on the STS as the waiting time of a transport order potentially leads to waiting time for the STS.

A hybrid dispatching method is positively evaluated by Meer [25], Angeloudis and Bell [27] and Song and Huang [28]. The implemented hybrid method combines the distance-based, the inventory-based and the time-based method. It calculates a score that comprises separate scores for shortest distance, lowest inventory and longest waiting time.

3.2 Varied Terminal Parameters

As with the dispatching methods, terminal parameters are chosen that were identified to influence the dispatching method performance ranking or that appear to be possibly relevant. Analyzing the parameters being varied in the sensitivity analyses of relevant studies, the most considered parameter is the number of vehicles, followed by the number of STS and yard cranes. The number of STS and yard cranes is typically motivated by modifying the terminal size. Furthermore, problem parameters are varied

like the time between jobs, time between vessels, STS load/discharge combination or others. In addition, layout-related parameters are of interest such as distance between quay and yard, number of blocks or the stacking height in the yard. Scarcely investigated parameters are the degree of stochasticity, vehicle capacity or vehicle speed [13]. Eleven terminal parameters are evaluated in this study regarding their influence on the performance of dispatching methods:

1. Number of vehicles
2. Speed of vehicles
3. Yard block assignment to containers
4. Utilization of seaside capacity
5. Vessel sizes
6. Terminal size
7. Equipment type
8. Range of handling times
9. STS handling rate
10. Quay layout
11. Share of landside traffic

The number of vehicles on the terminal for the different simulation runs ranges between 24 and 48. It reflects values of 3, 4, 5 or 6 TT per STS. The number of vehicles is analyzed in many respective sensitivity analyses. Zeng et al. [21] and Liu and Ioannou [19] indicate that this parameter influences the performance ranking of dispatching methods. The speed of vehicles is rarely investigated. Behera et al. [29] find a certain influence of TT speed on handling volumes. The varied values are 5.6 m/s, 8.4 m/s, and 11 m/s for TT.

The assignment of containers to yard blocks influences the dispatching substantially as it determines the start respectively the destination of the transport orders. Within the simulation study, the investigated values are *random*, three blocks *close* to the respective STS, *three* defined blocks distributed over the yard area and *mixed*, which is a combination of *close* for export containers and *three* for import containers.

The utilization of the seaside capacity is an essential container terminal parameter. A high capacity utilization leads to fewer options to reduce waiting times and distances [25]. This might influence the dispatching method ranking. Varied values are 50 %, 63 %, 75 %, 88 %, and 100 %. The size of vessels arriving at the terminal affects the structure of the workload for the horizontal transport. This directly affects the direction of the transport orders and therefore the dispatching. Varied values are only small vessels (average 500 containers), typical size distribution, and only large vessels (average 1250 and 3500

containers). The terminal size refers to the maximum container handling capacity of the quayside. It influences the distances the vehicles have to drive on the terminal. Investigated values are 0.8, 1.2, 1.6, and 3.2 million TEU/a. These numbers are representative for typical container terminals.

As described earlier, different equipment types are employed for the horizontal transport on container terminals. The most common types are TT and SC. While TT represent passive equipment, SC are active equipment. This influences the container handling processes. TT and STS respectively RTG have to wait for each other to transfer a container. SC handle a container independently from the cranes. This fact might also affect the dispatching results. Garro et al. [22] state that a dispatching method developed for AGV is not appropriate for SC.

The range of handling times is a value for the simulation model. The handling times are assumed to follow a triangular distribution. The range reflects the difference between the minimum and the maximum of the cranes' handling times. Kim and Bae [24] and Meersmans and Wagelmans [30] state that the range of handling times does not affect the ranking of dispatching methods. The varied values are *deterministic* handling times, *normal*, and *large*.

The average handling rate of STS are also investigated. The handling speed depends on several factors as e.g. the crane driver capabilities, the wind conditions, or the vessel type. Varied values are 24, 30, and 36 moves/hour.

The quay layout of a container terminal is often just straight. However, depending on the terminal location, many other layouts are possible. For the simulation study, besides the *straight* layout two other layouts are chosen: *L-shaped* and *rectangular* (see Figure 1).

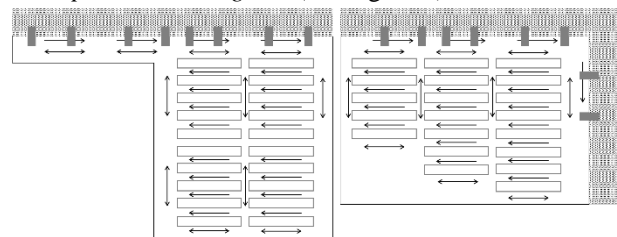


Figure 1: Alternative Quay Layouts (left L-shaped, right rectangular)

On a typical RTG/TT-terminal, external trucks enter the terminal area and drive directly into the yard area for loading respectively unloading. This landside process disturbs the RTG handling process and it might delay

horizontal transport operations. To investigate the influence of the landside traffic, typical variations of import/export containers are assumed (values: 0 %, 15 %, 30 %, 45 %).

3.3 Design of Experiment

To determine the influence of each terminal characteristic, a reference terminal is defined based on the terminal specifications analyzed before. The reference terminal uses TT for horizontal transport. 40 TT are available driving with a speed of 8.4 m/s. The yard block assignment of containers is *random* and the utilization of the seaside capacity is 75 %. Vessels of all sizes arrive at the terminal. The terminal size is 1.6 million TEU/a. The range of handling times is *normal*, the handling times of STS is 30 moves/hour. The quay layout is *straight*; the share of landside traffic is 0 %.

Based on this reference terminal, every terminal characteristic is modified *ceteris paribus* to analyze the effects of a parameter variation. Each parameter variation is simulated in combination with the five different dispatching methods. This leads to 29 parameter variations and combined with five different dispatching methods to a total of 145 experiments. For each experiment, 50 simulation runs are conducted.

4 Simulation Results

The fixed assignment of TT to a STS is – both in terms of STS productivity and distances driven – the worst dispatching method. The time-based method leads on average to the highest STS productivity. It is 4 % higher compared to the STS productivity of the fixed method. The inventory-based and the hybrid method perform similarly to the time-based method. The second lowest STS productivity is achieved with the distance-based method.

Regarding the driven distances per container, the distance-based method performs best. It reduces the mean distance by 8 % compared to the fixed method. The hybrid method performs similar. Thus, the hybrid method leads to a high STS productivity as well as to low driven distances per container. The time- and the inventory-based method lead to similar results as the fixed method.

These results are plausible as the distance-based method focuses on minimizing the driven distances of the TT, and the inventory- and the time-based method aim to avoid waiting times for the STS. The hybrid method combines the strengths of all three methods.

Two of the eleven evaluated terminal parameters influence the performance ranking of the dispatching methods, namely the yard block assignment to containers and the equipment type. This means that if the value of one of these parameters changes, it would be beneficial to apply a different dispatching method. The variation of all other evaluated terminal parameters led only to minor or no changes of the performance ranking of the dispatching methods. Their impact on the STS performance and driven distances per container are explained afterwards.

4.1 Effects of Yard Block Assignment to Containers

As Figure 2 shows, the assignment of containers to yard blocks affects the dispatching method performance ranking regarding the terminal objective to minimize the driven distances per container. If the assignment is *random* or the containers are transported to *three* defined blocks distributed over the yard area, the distance-based and the hybrid method are the best choices. If the containers are assigned to yard block *close* to the STS, the fixed method (which is usually the least successful one) performs best.

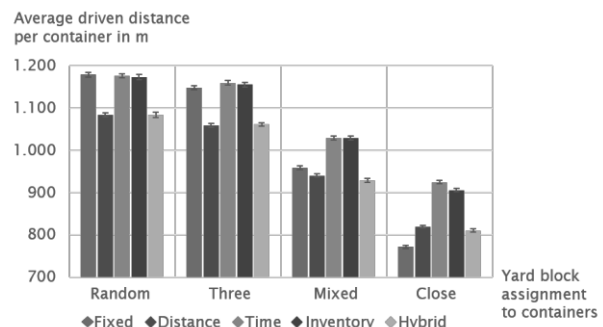


Figure 2: Average Driven Distances per Container Depending on the Yard Block Assignment

An explanation might be the short driving distances for the TT between quayside and yard. Using the fixed method, the TT always have short driving distances as the containers are located very close. All other methods allow the TT to drive a longer distance to pick up or bring a container from / to a yard block further away when no other transport order is available. However, this yard block assignment might lead to longer driving distances and possible congestion for external trucks on the container terminal, as they are required to bring and pick up the containers near the quayside.

The STS productivity is highest in case of a *random* assignment of containers to yard blocks (see Figure 3). This is due to the fact, that the TT approach more yard

blocks and the risk to wait for a busy RTG is lower. If the TT approach only three yard blocks, the probability is higher that they have to wait for the RTG and let the STS wait. The time-, inventory-based and hybrid method are similarly suited for all three assignments. The second lowest STS productivity is achieved with the distance-based method and the lowest with the fixed method.

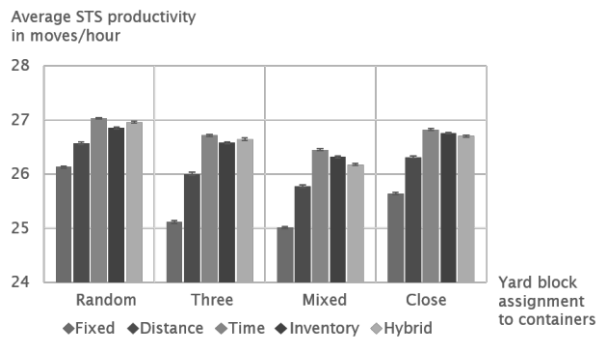


Figure 3: Average STS Productivity Depending on the Yard Block Assignment

4.2 Effects of Equipment Type

The chosen type of equipment affects the dispatching method performance ranking regarding the driven distances per container (see Figure 4). If TT are employed, the distance-based and the hybrid method perform both well, reducing distances by approximately 9 % compared to the fixed method. The time- and the inventory-based method are on a similar level as the fixed method. If SC are used for horizontal transport and yard operations, the fixed assignment of SC to STS leads to the shortest driven distances per container. The hybrid method is the next best followed by the distance-, inventory- and time-based method.

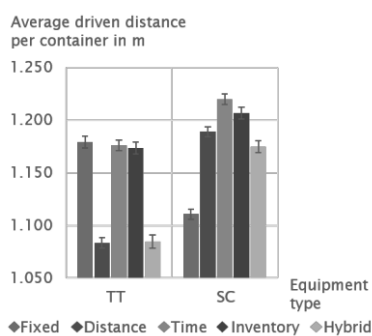


Figure 4: Average Driven Distances per Container Depending on the Equipment Type

The STS productivity is equally high for all evaluated dispatching methods for the SC case and it is higher than for the TT case (see Figure 5). For TT, the time-based methods generates the best results followed shortly by all other methods.

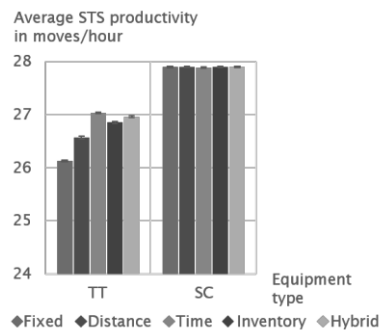


Figure 5: Average STS Productivity Depending on the Equipment Type

4.3 Effects of Other Terminal Parameters

All other evaluated terminal parameters do not affect the dispatching method ranking, i.e. one dispatching method is always the best choice to achieve a certain terminal objective when the respective terminal parameter is varied. However, terminal parameters do often affect the STS productivity and the driven distances of the vehicles. Therefore, a short description of their impact is given hereafter.

The number of vehicles used for horizontal transport influences both STS productivity and driven distances. If the number is too low, the STS productivity drops significantly below 24 moves/hour for the time-, inventory-based and hybrid method and down to 19 moves/hour for the fixed assignment method. This means that there are many delays in horizontal transport operations and waiting times for the STS. If 48 TT are available, all dispatching methods lead to a similarly high STS productivity. However, this also implies longer distances for the distance-based and the hybrid method. This is caused by the fact that on average, there are less transport orders to choose from for a TT and, therefore, the potential to reduce distances is lower.

A variation of the vehicle speed does not affect the driven distances. Likewise, an increase from 8.4 m/s to 11 m/s does not improve the STS productivity. Only a decrease to 5.6 m/s reduces the STS productivity slightly.

A variation of the utilization of the seaside capacity does not affect the dispatching method performance ranking. If the capacity usage decreases, the STS productivity improves slightly because there is the same amount of vehicles available for less transport orders leading to reduced waiting times for a transport order to be executed. At the same time, the driven distances rise because a free vehicle finds a smaller number of transport orders

to choose from reducing the possibilities to minimize distances.

Changing the vessel size and thereby the structure of the workload for the horizontal transport influences the STS productivity. If the workload structure is relatively balanced, the STS productivity increases by approximately 0.5 moves/hour.

The terminal size correlates with the driven distances. Interestingly, the gap between the dispatching methods is relatively small for smaller terminals (4 %), and increases with the terminal size. For the largest terminal in this study, the gap between distance-based and hybrid on the one hand and inventory-, time-based and fixed assignment is approximately 18 %. This implies that the choice of a dispatching method is more important for larger container terminals than for smaller terminals. Regarding STS productivity, the fixed assignment method is almost as good as the other dispatching methods for the case of the small terminal.

The range of handling times has no impact on the driven distances. However, a large range reduces the STS productivity by more than 1 move/hour compared to a normal range. This is plausible as a high range of handling times especially in the yard increases the risk that a TT arrives late at the STS causing waiting times for the STS.

Shorter handling rates of the STS reduce the STS productivity as expected. If the number of available TT stays constant while the handling times improve, the beneficial effect on the STS productivity lessens. A variation of the handling times does not affect the driven distances.

A variation of the quay layout influences the STS productivity not significantly. Only for the case of the L-shaped layout, the STS productivity decreases slightly. However, the quay layout affects the driven distances. The rectangular layout leads to distances between 1000 and 1100 m (depending on the dispatching method), the straight layout to distances between 1080 and 1180 m and the L-shaped layout leads to distances of more than 1300 m.

The landside traffic affects the STS productivity negatively if the percentage rises above 30 %. This means that there are more occasions when a RTG serves an external truck while an urgent order for an internal transport should be handled. There is no influence of the landside traffic on the driven distances.

5 Conclusion

This simulation study shows a first start to evaluate the interrelations between container terminal parameters and dispatching method performance depending on different terminal objectives. Especially the choice of the terminal objectives influences the choice of the dispatching method. Five different dispatching methods are examined on their effect on STS productivity and driven distance per container of horizontal transport vehicle. These methods are the fixed assignment of vehicles to STS, a distance-, a time-, an inventory-based and a hybrid method. Two terminal parameters out of eleven have been found out to have an impact on the performance ranking of dispatching methods and, therefore, on the choice of the best dispatching method for a specific terminal to achieve a specific objective. These parameters are the assignment of containers to yards block and the equipment type used for the horizontal transport. All other evaluated terminal parameters are analyzed on their impact on STS productivity and on the driven distance per container.

Future research aims to systematically extend this study by further parameters, dispatching methods and terminal objectives. Especially the hybrid method performed well for both analyzed performance indicators. It combines the benefits of the distance-, time- and inventory-based methods while still being easy to implement. In future research, different weightings of the scores of the three singular methods can be evaluated to highlight potential for further improvement. Furthermore, it could be reasonable to change the weightings of the hybrid dispatching method during operation. For example, if a large deep-sea vessel is moored at the quay side or if less vehicles are available, the focus should be on STS productivity, while in case of a lower utilization of the seaside capacity, a focus on reducing the driven distances would be beneficial.

Furthermore, future research may focus on the evaluation of using meta-heuristics, e.g. genetic algorithms, for the dispatching process. Here, especially the computational times and the accuracy are important to ensure a sufficient transferability to real life container terminals. The use of a rolling horizon might prove a possible solution. Another promising field of interest is the integration of landside processes, e.g. the implementation of a truck appointment system, and their impact on the dispatching strategies in the horizontal transport, especially for terminals with a high share of import and export containers.

References

- [1] UNCTAD. Review of Maritime Transport 2019. New York, Geneva, United Nations, 2019.
- [2] Gharehgozli AH, Roy D, Koster RBM de. Sea container terminals: New technologies and OR models. *Marit Econ Logist* 2016;18:103–140.
- [3] Brinkmann B. Seehäfen - Planung und Entwurf. Berlin, Springer, 2005.
- [4] Steenken D, Voß S, Stahlbock R. Container terminal operation and operations research - a classification and literature review. *OR Spectrum* 2004;26:3–49.
- [5] Kemme N. Design and Operation of Automated Container Storage Systems. Heidelberg, Physica-Verlag HD, 2013.
- [6] Bae HY, Choe R, Park T, Ryu KR. Comparison of operations of AGVs and ALVs in an automated container terminal. *J Intell Manuf* 2011;22:413–426.
- [7] Brinkmann B. Operations Systems of Container Terminals: A Compendious Overview. Edited by Böse JW. New York, Springer, 2011, pp. 25–39.
- [8] Wiese J, Kliwer N, Suhl L. A Survey of Container Terminal Characteristics and Equipment Types, 2009.
- [9] Bian Z, Yang Y, Mi W, Mi C. Dispatching Electric AGVs in Automated Container Terminals with Long Travelling Distance. *Journal of Coastal Research* 2015;73:75–81.
- [10] Grunow M, Günther H-O, Lehmann M. Strategies for dispatching AGVs at automated seaport container terminals. *OR Spectrum* 2006;28:587–610.
- [11] Kizilay D, Eliyi DT. A comprehensive review of quay crane scheduling, yard operations and integrations thereof in container terminals. *Flexible Serv Manuf J* 2020.
- [12] Huang J, Wang F, Shi N. Resource allocation problems in port operations: A literature review. Edited by Zhu Q, Wang S, Chai J, Yu L, Institute of Electrical and Electronics Engineers Inc, 2014, pp. 154–158.
- [13] Schwientek AK, Lange A-K, Jahn C. Literature classification on dispatching of container terminal vehicles. Edited by Jahn C, Kersten W, Ringle CM. Berlin, epubli, 2017, pp. 3–36.
- [14] Lee LH, Chew EP, Tan KC, Wang Y. Vehicle dispatching algorithms for container transshipment hubs. *OR Spectrum* 2010;32:663–685.
- [15] Bian Z, Zhang Y, Zhang X, Xiao Y, Chai J, Mi W. Simulation-based AGV dispatching in automated container terminal, Institute of Electrical and Electronics Engineers Inc, 2019, pp. 414–420.
- [16] Briskorn D, Drexel A, Hartmann S. Inventory-based dispatching of automated guided vehicles on container terminals. *OR Spect.* 2006;28:611–630.
- [17] Nguyen VD, Kim KH. Heuristic algorithms for constructing transporter pools in container terminals. *IEEE Trans. Intell. Transp. Syst.* 2013;14:517–526.
- [18] Grunow M, Günther H-O, Lehmann M. Dispatching multi-load AGVs in highly automated seaport container terminals. *OR Spectrum* 2004;26:211–235.
- [19] Liu C-I, Ioannou PA. A comparison of different AGV dispatching rules in an automated container terminal. Edited by Lee D.-H, Srinivasan D, Cheu R.L, Institute of Electrical and Electronics Engineers Inc, 2002, pp. 880–885.
- [20] Koster RBM de, Le-Anh T, Meer JR van der. Testing and classifying vehicle dispatching rules in three real-world settings. *Journal of Operations Management* 2004;22:369–386.
- [21] Zeng Q, Yang Z, Lai L. Models and algorithms for multi-crane oriented scheduling method in container terminals. *Transp. Policy* 2009;16:271–278.
- [22] Garro A, Monaco MF, Russo W, Sammarra M, Sorrentino G. Agent-based simulation for the evaluation of a new dispatching model for the straddle carrier pooling problem. *SIMULATION* 2015;91:181–202.
- [23] Schwientek AK, Lange A-K, Jahn C. Simulation-Based Analysis of Dispatching Methods on Seaport Container Terminals. Edited by Freitag M, Kotzab H, Pannek J, Springer International Publishing, 2018, pp. 167–171.
- [24] Kim KH, Bae JW. A Look-Ahead Dispatching Method for Automated Guided Vehicles in Automated Port Container Terminals. *Transportation Science* 2004;38:224–234.
- [25] Meer JR van der. Operational control of internal transport. [Rotterdam], [ERIM], 2000.
- [26] Cao J, Shi Q, Lee D-H. A decision support method for truck scheduling and storage allocation problem at container. *Tinshhua Sci. Technol.* 2008;13:211–216.
- [27] Angeloudis P, Bell MGH. An uncertainty-aware AGV assignment algorithm for automated container terminals. *Transportation Research Part E: Logistics and Transportation Review* 2010;46:354–366.
- [28] Song LQ, Huang SY. A hybrid metaheuristic method for dispatching automated guided vehicles in container terminals, 2013, pp. 52–59.
- [29] Behera JM, Diamond NT, Bhuta CJ, Thorpe GR. The impact of job assignment rules for straddle carriers on the throughput of container terminals. *ATR* 2000;34:415–444.
- [30] Meersmans PJM, Wagelmans APM. Effective Algorithms for Integrated Scheduling of Handling Equipment at Automated Container Terminals. Rotterdam, 2001.

Simulation als Bestandteil eines BIM-basierten Vorgehens zur Planung der Baustellenlogistik im Großanlagenbau

Jana Stolipin^{1*}, Ulrich Jessen¹, Jan Weber², Sigrid Wenzel¹, Markus König²

¹Institut für Produktionstechnik und Logistik, Fachgebiet Produktionsorganisation und Fabrikplanung, Universität Kassel, Kurt-Wolters-Straße 3, 34125 Kassel; *jana.stolipin@uni-kassel.de.

²Fakultät für Bau- und Umweltingenieurwissenschaften, Lehrstuhl für Informatik im Bauwesen, Ruhr-Universität Bochum, Universitätsstraße 150, Gebäude IC, 44780 Bochum.

Kurzfassung. Für die Konstruktions- und Montageplanung von Großanlagen hat sich eine digitale modellbasierte Arbeitsweise, auch als Building Information Modeling (BIM) bezeichnet, durchgesetzt. Dieser Beitrag stellt für diesen Anwendungskontext ein Vorgehensmodell zur integrierten Planung und Gestaltung von logistischen Prozessen auf einer Baustelle vor. Ausgehend von einem BIM-basierten digitalen Modell und mit Unterstützung einer fachspezifischen Ontologie und Beschreibungen von Standardprozessen wird die Planung der Baustellenlogistik durchgeführt und mittels Simulation überprüft. Anhand eines Beispielmodells werden die einzelnen Schritte des Vorgehensmodells vorgestellt, das Planungsergebnis simulativ überprüft und die Ergebnisse abschließend diskutiert.

Einleitung

Effiziente und wirtschaftliche Produktions- und Logistikprozesse sind für den Unternehmenserfolg essentiell [1]. Ihrer Planung und Gestaltung wird damit eine hohe Relevanz zugesprochen; die geforderte hohe Planungsqualität bei gleichzeitig hoher Planungssicherheit auch bei komplexen Systemen kann durch die Anwendung digitaler Planungsmethoden und -modelle erreicht werden [2]. Auch für die logistischen Prozesse auf einer Baustelle im Großanlagenbau ist eine zuverlässige Planung entscheidend, da nur auf diese Weise eine abgesicherte Terminierung aller Bauprozesse auf dem Baustellengelände und ein konfliktfreier Ablauf umgesetzt werden können.

Heute werden im Anlagenbau digitale Modelle insbesondere zur Entwurfsplanung eingesetzt, jedoch können diese nicht direkt zur Planung und Steuerung der Baustellenlogistik genutzt werden, da beispielsweise die logistikrelevanten Informationen einzelner Anlagekomponen-

ten fehlen oder die vorhandenen Informationen nicht einheitlich strukturiert und somit nicht automatisiert auswertbar sind. Erst wenn zu jedem Bauelement die logistikrelevanten Informationen (z. B. Transport-, Lager- und Montagehinweise) vorliegen, kann eine Planung der Logistikprozesse auf der Baustelle realisiert werden. Das digitale Logistikmodell kann dann analysiert und im Hinblick auf ausgewählte Zielgrößen (z. B. Lagerbelegung oder Ressourcenauslastung) überprüft und ggf. verbessert werden. Ein derartiges digitales Planungsmodell kann u. a. auch als Grundlage für die Steuerung der Baustellenlogistik eingesetzt werden und die Entscheidungen zur operativen Durchführung der Logistikprozesse auf der Baustelle unterstützen. Dazu müssen allerdings die aktuellen Baufortschrittsinformationen regelmäßig gepflegt werden [3].

Der Beitrag stellt ein BIM-basiertes Vorgehen zur digitalen Planung der Baustellenlogistik im Großanlagenbau vor. Das Ziel dieses Vorgehens ist, ausführenden Unternehmen Hilfsmittel zur effizienten Organisation und zur Sicherstellung der Bauprozesse der Anlage vor Ort anzubieten; dabei wird zur Absicherung der BIM-basierten Planung die Simulation eingesetzt. In diesem Beitrag wird – ausgehend vom Stand der Forschung zur Baustellenplanung im ersten Abschnitt – im zweiten Abschnitt das BIM-basierte Vorgehen vorgestellt. Neben der Darstellung der einzelnen Elemente des Vorgehens wird auch die Rolle der Simulation zur Validierung der für eine Baustelle geplanten Logistikprozesse im Großanlagenbau erläutert. Der dritte Abschnitt konzentriert sich auf die Beschreibung der Umsetzung des Vorgehens anhand eines Anwendungsbeispiels mit besonderem Fokus auf die Validierung der dazugehörigen Planung mittels eines mit dem Simulationswerkzeug AnyLogic erstellten

Simulationsmodells. Nach der Darstellung der Simulationsergebnisse werden im abschließenden vierten Abschnitt weitere Einsatzmöglichkeiten der Simulation im Rahmen der Methodik diskutiert und potenzielle Forschungsfragen abgeleitet.

1 Baustellenlogistikplanung

Die Bauwirtschaft ist durch einen sehr hohen Kostendruck getrieben und gilt als eine eher konservative Branche [4]. Im Großanlagenbau (z. B. Kraftwerke und Chemieanlagen) ähneln die Rahmenbedingungen in vielen Aspekten denen des klassischen Baubetriebs (z. B. Hoch- und Tiefbau), jedoch mit dem Unterschied, dass die Montage in Bezug auf die Logistik eine zentrale Stellung einnimmt. Logistik und Montage im Großanlagenbau stellen ein komplexes und ineinander verzahntes System dar [5]. Die Baulogistik lässt sich in die drei Bereiche Versorgungs- oder Beschaffungslogistik, Baustellen- oder Produktionslogistik und Entsorgungslogistik unterteilen, wobei die Baustellenlogistik alle logistischen Aufgaben auf dem Baustellengelände umfasst [2]. Die Baustellenlogistik im Großanlagenbau hängt u. a. von der geplanten Anlage, der Lage des Bauplatzes und den verfügbaren Ressourcen ab und stellt einen Schlüsselfaktor für den Projekterfolg dar [6].

1.1 Praxis der Logistikplanung

Planung und Steuerung der Prozesse auf der Baustelle basieren vorwiegend auf dem Wissen der erfahrenen Mitarbeitenden [7]. Dies führt häufig zu nicht standardisiertem Planungsvorgehen und erschwert die Koordination der Kommunikation aller Projektbeteiligten [8]. Bei der Projektplanung im Bauwesen ist meist nur wenig Zeit für die Arbeitsvorbereitung vorgesehen, um die verschiedenen Möglichkeiten des Bauablaufs, die Ressourcenplanung und die Baustelleneinrichtung zu prüfen und zu bewerten [9]. Außerdem erschweren die heterogenen Strukturen und die Vielzahl kleiner Unternehmen in der Bauwirtschaft die systematische Logistikplanung auf Baustellen und gleichzeitig die detaillierte Vorbereitung und sinnvolle Umsetzung der Baustellenlogistik [10].

Zwar ist mittlerweile die digitale BIM-basierte Planung von Gebäuden und Großanlagen weit verbreitet, jedoch finden Aspekte der Baustellenlogistik kaum Berücksichtigung, obwohl bereits einige wichtige Informationen zur Planung und Steuerung der Baustellenlogistik in aktuellen digitalen Modellen vorhanden sind [11].

1.2 Stand der Forschung

In der Bauindustrie setzt sich zurzeit das Konzept der Digitalen Baustelle zur Planung immer stärker durch. Zu den Bestandteilen der Digitalen Baustelle gehören die dreidimensionale Bauwerksmodellierung, die Einbindung logistischer Prozesse, eine zentrale Datenverwaltung sowie die Animation und Simulation von Vorgängen auf der Baustelle [12]. So werden für die Konstruktions- und Montageplanung von Großanlagen digitale Methoden eingesetzt, um eine BIM-basierte Arbeitsweise zu realisieren. Informationen zu Bauwerken, Baumaterialien, Terminen und Kosten können über Kommunikationsplattformen im gesamten Bauprozess allen Beteiligten zur Verfügung gestellt werden [2].

Hierzu werden digitale 3-D Modelle nach dem BIM-Prinzip erstellt, die von Fachabteilungen im Rahmen von Planungsaufgaben eingesetzt werden können. Neben geometrischen Informationen der Anlage enthält ein BIM-Modell daher weitere Informationen wie Termine, Kosten oder auch Eigenschaften zu einzelnen Anlagenelementen [13]. Charakterisiert wird ein BIM-Modell durch die Anzahl der Informationstypen, die als Dimensionen des Modells bezeichnet werden. [14]. Die Erweiterung von dreidimensionalen CAD-Modellen mit der Projektterminplanung stellt einen Teil der sogenannten n-D-Modellierung dar. Die Grundidee ist die Erweiterbarkeit eines 3-D-Modells um weitere Dimensionen. Mittels 4-D BIM werden die Modelle um eine Zeitdimension in Form eines Terminplans erweitert, wodurch eine Koordination der Logistikprozesse mit der dynamisch sich verändernden Baustellenumgebung ermöglicht wird [15].

BIM-Modelle für die Baulogistik werden heute im Hoch-, Tief- und Tunnelbau eingesetzt. BIM-Modelle in der Bauwirtschaft werden in der Literatur für die Baustelleneinrichtungsplanung [16], Materialflusslogistik [17], Bauprojektplanung [6, 18] und Baustellensimulation [19] angewandt. Da die ereignisdiskrete Simulation zur methodischen Absicherung der Planung, Steuerung und Überwachung der Material-, Personen-, Energie- und Informationsflüsse eingesetzt wird [21], kann sie auch die Planung der Baustellenlogistik sinnvoll unterstützen.

Für die Planung der Baustellenlogistik im Großanlagenbau ist ein Vorgehen notwendig, das die spezifischen Eigenschaften der Baubranche berücksichtigt. Vor dem Hintergrund der zuvor beschriebenen Problematik wird in diesem Beitrag ein Vorgehen zur BIM-basierten Logistikplanung im Großanlagenbau und zu ihrer simulativen Absicherung vorgestellt.

2 BIM-basiertes Vorgehen zur Baustellenplanung

Bei der Planung der Baustellenlogistik müssen nicht nur der Ablauf des Bauprojekts, die Bauprozesse und die logistischen Aktivitäten, sondern auch die verfügbaren Ressourcen, die spezifischen Transport- und Lagerungsbedingungen für die Bauelemente sowie der Materialfluss und die Lagerkapazitäten auf der Baustelle berücksichtigt werden. Eine Herausforderung besteht darin, die BIM-basierten Modelle, um Logistikbeschreibungen zu erweitern und unter den jeweiligen Umgebungsbedingungen in konkrete Logistikmodelle zu überführen.

Die Vorgehensweise umfasst neben der Formalisierung von produktindividuellen Logistikanforderungen auf Basis projektspezifischer Randbedingungen und Methoden zur semi-automatischen Generierung von Logistikmodellen für die Analyse und Planung der Baustellenlogistik auch Visualisierungs- und Adaptioniskonzepte zur Verwendung der digitalen Modelle auf der Baustelle. Das Zusammenwirken der einzelnen Bausteine der Vorgehensweise einer BIM-basierten Planung der Baustellenlogistik im Großanlagenbau ist in Abbildung 1 dargestellt. Ausgehend von einem detaillierten BIM-Modell der Anlage (Anwendungsbeispiel) mit allen dazugehörigen Komponenten, zuvor definierten Montageprozessen und Montagereihenfolgen werden im ersten Schritt Anforderungen bezüglich Baustellenlogistik und der eigentlichen Durchführung der Montage mit entsprechenden Ressourcen über eine Ontologie verknüpft. Dabei werden auch das individuelle Baustellenlayout und die Ressourcen für das Anwendungsbeispiel berücksichtigt. Um auf Basis der Anforderungen und dazugehörigen Informationen eine semi-automatische Generierung eines Logistikmodells (inkl. Bauanleitung) zu realisieren, werden Methoden für die Erstellung des digitalen Modells mit logistikrelevanten Informationen vorgeschlagen.

Im nächsten Schritt werden die Gültigkeit des digitalen Logistikmodells und der auf seiner Basis erstellten Bauanleitung, die zusammen die Basis für die Logistikplanung repräsentieren, simulativ überprüft. Im Laufe des Projektfortschritts auf der Baustelle können die Randbedingungen oder Prozesse verändert werden, daher müssen die jeweils neuen Informationen ebenfalls bewertet werden. Hierzu werden diese in das bestehende Logistikmodell integriert und wiederum simulationsgestützt geprüft, um Probleme oder einen Anpassungsbedarf im Projektverlauf frühzeitig zu erkennen.

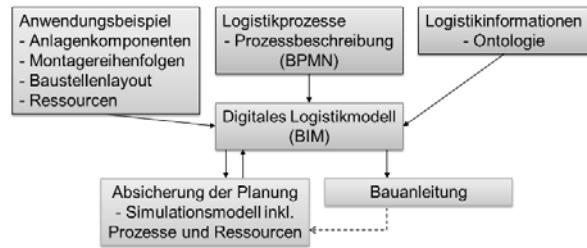


Abbildung 1: Zusammenwirken der Bausteine einer BIM-basierten Planung der Baustellenlogistik im Großanlagenbau

Nachfolgend werden die einzelnen Bestandteile (erarbeitete Ontologie mit logistikrelevanten Informationen, all-gemeingültig beschriebene Prozesse auf der Baustelle, Planungsszenarien, Bauanleitung und Simulation) sowie ihre Rolle im Rahmen des entwickelten Vorgehens kurz vorgestellt.

2.1 Rolle der Ontologie

Für die Realisierung der digitalen Baustellenlogistik im Großanlagenbau ist eine für die Vorgehensweise fachspezifische Ontologie entwickelt worden, um Informationen aus einem BIM-Modell, bestehend aus einer Anlage mit seinen Komponenten, Montageprozessen und Montagereihenfolgen, mit den Informationen der Baustellenlogistik sowie Informationen zur Durchführung der Montage mit den dazu notwendigen logistischen Ressourcen zu verknüpfen. Die Ontologie [22] beschreibt einerseits formal die semantischen Zusammenhänge zwischen den Anforderungen an die Baustellenlogistik sowie zwischen den Anforderungen an Montageprozesse, an den Projektgegenstand (z. B. Bauelemente einer Großanlage) und an weitere relevante Objekte auf der Baustelle (z. B. Lager, Anlieferungsfläche, Ressourcen usw.); sie enthält auch Montage- und Logistikinformationen für die Gestaltung der Abläufe auf der Baustelle in den Planungsszenarien. Andererseits wird die Ontologie als ein Informationsmodell verwendet, aus dem basierend auf einem detaillierten BIM-Modell einer Großanlage logistische Anforderungen abgeleitet und sogenannte Bauanleitungen generiert werden können.

2.2 Beschreibung der Logistikprozesse

Für die Unterstützung der Logistikplanung auf der Baustelle und ausgehend von den Anforderungen im Großanlagenbau werden Prozessabläufe für die Baustellenlogistikplanung mittels der Modellierungssprache BPMN

(Business Process Modeling Language) als Referenzprozesse dargestellt, mit denen die Beispielabläufe auf Baustellen (in den Planungsszenarios) abgebildet werden können. Die definierten BPMN-Referenzprozesse betreffen die Baustellenbereiche: Anlieferung, Baustellenlager, Zwischenlager, Montage und Entsorgungslager. Die einzelnen Referenzprozesse beinhalten logistikbezogene Tätigkeiten auf der Baustelle, wie das Transportieren, Lagern, Puffern, Entpacken, Umschlagen, Prüfen und Kommissionieren von Baumaterialien und -elementen. Auf Basis der dokumentierten BPMN-Referenzprozesse wird die Erstellung des digitalen Logistikmodells sowie des Simulationsmodells umgesetzt und somit eine detailliertere Planung von Prozessen auf der Baustelle ermöglicht.

2.3 Rolle des BIM-basierten Logistikmodells

Das BIM-basierte Logistikmodell wird unter Verwendung eines konkreten BIM-Anwendungsbeispiels (inkl. Anlagekomponenten, Montagereihenfolgen, Baustellenlayout und geplanten Ressourcen) semi-automatisch erstellt. Das Logistikmodell beinhaltet neben den projektspezifischen Informationen und Geometriedaten der Anlage auch logistikrelevante Informationen aus der fachspezifischen Ontologie. In diesem Logistikmodell können nicht nur Termine für die einzelnen Bau- und Logistikprozesse gesichert werden, sondern es kann auch für die Erstellung einer Bauanleitung verwendet werden. Das Logistikmodell (inkl. Bauanleitung) stellt die Grundlage für die Erstellung von Planungsszenarien dar, die im Rahmen der BIM-basierten Planung der Baustellenlogistik im Großanlagenbau simulativ abgesichert werden.

2.4 Rolle der Bauanleitung

In 4-D BIM-Modellen ist zumeist nur eine grobe zeitliche Abfolge der einzelnen Bauabschnitte abgebildet, bei der beispielsweise alle Bauelemente eines Gewerks in einer Etage innerhalb einer festgelegten Woche eingebaut werden. Für eine genaue Planung der Logistikprozesse wird jedoch eine präzisere Zeitplanung benötigt. Da der Aufwand für die manuelle Erzeugung einer Schritt-für-Schritt-Bauplanung auf individueller Bauteilebene nicht wirtschaftlich ist, wird dieser Prozess automatisiert. Hierzu werden Regeln zur Montage der Bauelemente definiert. Diese Regeln beschreiben die Abhängigkeit der Bauelemente voneinander und ermöglichen – in Kombination mit Informationen über die Lage der Elemente relativ zueinander – die semi-automatische Generierung einer Schritt-für-Schritt-Bauanleitung. Mit dieser lässt sich

schließlich die genaue Abfolge der korrespondierenden Logistikprozesse festlegen. Durch die Betrachtung und Einreihung jedes einzelnen Bauelements wird so bei Bedarf eine Terminplanung für jedes einzelne Bauelement ermöglicht.

2.5 Erstellung der Planungsszenarien

Auf Basis der definierten BPMN-Referenzprozesse, eines BIM-basierten Logistikmodells, einer Bauanleitung und dazugehörigen Informationen aus der Ontologie kann eine Planung der Prozesse auf der Baustelle durchgeführt werden. Hierzu werden Planungsszenarien definiert. Ein Planungsszenario enthält die terminliche Anordnung der Montage- und Logistikprozesse sowie Anlieferungen von Baumaterialien unter Berücksichtigung der für ein Anwendungsbeispiel geplanten Ressourcen und dessen Layouts. Dabei werden verschiedene Szenarien betrachtet und somit auch Zustände einer Baustelle definiert, sodass relevante Eigenschaften sowohl der Bau- und Logistikaktivitäten als auch der Bauelemente und -materialien des Bauprojekts berücksichtigt werden können. Beim Erstellen der Szenarien werden insbesondere die von der Bauanleitung vorgeschlagenen Reihenfolgen der Montage und die Zeitpunkte der Anlieferungen von Baumaterialien berücksichtigt und aufeinander abgestimmt. Zur Ausgestaltung der Prozesse auf der Baustelle werden die Informationen aus der Ontologie (z. B. Anforderungen der Baumaterialien für die Lagerung und den Transport auf der Baustelle) und die BPMN-Referenzprozesse verwendet. Die für ein Projekt geplanten Ressourcen, das Personal und das entsprechende Layout sowie die definierten Planungsszenarien werden dann mit Hilfe eines mit dem Simulationswerkzeug AnyLogic entwickelten Simulationsmodells hinsichtlich ihrer Gültigkeit überprüft.

2.6 Rolle der Simulation

Zur Sicherstellung der Gültigkeit des Logistikmodells werden die im BIM-Modell aufgeführten Reihenfolgen, Restriktionen und verwendeten Ressourcen überprüft. Hierzu wird ein Materialflusssimulationsmodell auf Basis der Planungsdaten erstellt, das die in den Planungsszenarien definierten Prozesse (wie Reihenfolge der Montage, Transporte und Lagerungen) sowie Nutzung der verfügbaren Ressourcen (wie Lagerflächen und Transportmittel) modelliert und die ausgewählten Planungsszenarien experimentell untersucht. Nach der simulationsgestützten Überprüfung liegt ein konsistentes

und valides Logistikmodell vor, das auch zur Steuerung der Baustellenlogistik verwendet werden kann. Darüber hinaus liefert das Simulationsmodell bereits vor Beginn der eigentlichen Bautätigkeiten Leistungsdaten für eine mit entsprechender Materialfluss- und Lagertechnik vorgegebenen Ausstattung der Baustelle. Mit Hilfe der Simulation können somit die Anforderungen an die Dimensionierung der Lagerflächen auf der Baustelle überprüft und der Einsatz des Personals und der Ressourcen für die Realisierung des Bauprojekts quantifiziert werden. Außerdem wird es möglich, die auf der Baustelle geplanten logistischen Ressourcen über einen definierten Zeitraum zu überprüfen (u. a. Auslastung von Transportmitteln, z. B. von Kränen und Gabelstaplern, Belegung von Lagerflächen sowie Personaleinsatz).

Im Rahmen der BIM-basierten Planung der Baustellenlogistik wird das jeweils anwendungsspezifische Simulationsmodell zur Absicherung der Planung basierend auf den Daten der Referenzprozesse, Ontologie und BIM-Logistikmodelle standardisiert aufgebaut. Dieses Vorgehen ist auch als Basis für zukünftige Bauprojekte wiederzuverwenden.

3 Anwendungsbeispiel

In diesem Abschnitt wird die Umsetzung des BIM-basierten Vorgehens auszugsweise an einem Anwendungsbeispiel vorgestellt. In diesem Zusammenhang wird zunächst das Anwendungsbeispiel vorgestellt und anschließend wird auf die Simulation zur Absicherung der durchgeführten Baustellenlogistikplanung eingegangen. Abschließend werden die Simulationsergebnisse diskutiert.

3.1 BIM-Modell

Zur nachvollziehbaren Umsetzung wird ein einfaches BIM-Modell verwendet. Da Modelle aus der Praxis in der Regel sehr komplex sind, wird ein kleines Stahlbau-Modell (vgl. Abbildung 2) konstruiert. Dieses besteht aus zwei Lagerflächen und 78 Bauelementen, die trotz ihrer geringen Komplexität einem realen Stahlbau-Modell entsprechen. Die Konstruktion setzt sich aus einem Fundament und einer darauf befindlichen Stahlbaukonstruktion zusammen. Die Stahlkonstruktion besteht aus vier Stützen, fünf Trägern, darauf liegend zwei Betonplatten, 60 Bolzenverbindungen in 30 Gruppen, 16 Stahlwinkeln und 16 Ankerstäben. Darüber hinaus werden in der Vorfertigung Fußplatten mit den Stützen und beidseitig Stirnplattenanschlüsse an einen der Träger geschweißt.

Die im Modell verwendeten Bolzen haben den gleichen Durchmesser, um die Lagerhaltung zu vereinfachen [23].

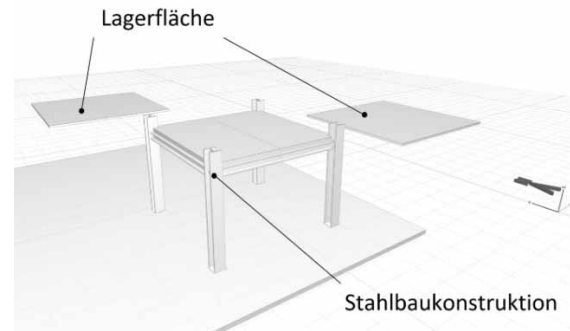


Abbildung 2: Layout des Anwendungsbeispiels

3.2 Durchführung der Logistikplanung

Für die Planung der zugehörigen Prozesse auf der Baustelle wird zunächst auf Basis des BIM-Logistikmodells zum Anwendungsbeispiel eine Bauanleitung semi-automatisch erstellt. Mittels dieser Bauanleitung werden sechs aufeinanderfolgende Bauprozesse (Errichtung von vier Stützen, Einbau von Trägern und Geschossdecke (bestehend aus zwei Betonplatten)) definiert. Dabei wird der Aufbau des Fundaments nicht berücksichtigt, da bei dem Anwendungsbeispiel ausschließlich Bauprozesse der Stahlkonstruktion betrachtet werden. Um die Logistikplanung durchführen zu können, werden Annahmen bezüglich der Anlieferungen der Bauteile getroffen. Insgesamt werden drei Anlieferungen geplant (zwei Anlieferungen für die Stahlträger und Stahlstützen sowie eine Anlieferung der Geschossdecke), die Anlieferungen sollen in Abhängigkeit von definierten Szenarien vor dem Beginn der Montageprozesse erfolgen. Für die Definition der Planungsszenarien werden auch Annahmen bezüglich der auf der Baustelle zur Verfügung stehenden Ressourcen und Baustellenmitarbeiter getroffen. Somit werden für das Anwendungsbeispiel zwei Planungsszenarien ausgewählt. Im ersten beispielhaften Planungsszenario sind ein Kran und zwei Baustellenmitarbeiter (ein Monteur und ein Kranführer) eingeplant. Die sechs Montageprozesse werden hier gemäß der Bauanleitung nacheinander ausgeführt, die drei Anlieferungen der Bauteile erfolgen durch Lieferanten; alle Bauteile werden vor ihrer Montage zwischengelagert und mit dem Kran zum Montageort transportiert und montiert. Die Anforderungen an die Transporte und Lagerungen der einzelnen Bauelemente werden gemäß der Ontologie definiert. Im zweiten Planungsszenario sollen die Montageprozesse der Stahlstützen parallelisiert werden (d. h., es werden jeweils zwei Stützen gleichzeitig montiert); anschließend

werden, wie im ersten Szenario, Stahlträger und Geschossdecke nacheinander aufgebaut. Die Anlieferungen der Bauelemente erfolgen auch hier vor der Montage der Bauelemente zur Anlieferungsfläche, anschließend werden die Bauelemente je nach Bedarf gelagert. Um das zweite Planungsszenario zu realisieren, wird zusätzlich zu den eingeplanten Ressourcen aus dem ersten Planungsszenario ein zweiter Kran eingeplant.

Aus dem BIM-Modell und dem dazugehörigen Layout werden die einzelnen Positionen der Objekte, Lagerflächen und Montageorte auf der Baustelle und die Längen der Transportwege zwischen diesen Positionen bestimmt. Für das Anwendungsbeispiel wird neben einer Anlieferungsfläche auch ein offenes Baustellenlager eingeplant. Auf dieser Basis können die Zeiten der Transporte auf der Baustelle berechnet werden. Unter Berücksichtigung der projektrelevanten Termine, der Montager Reihenfolge und der logistischen Abläufe werden die zwei ausgewählten Szenarien in Form von Gantt-Diagrammen dargestellt. In den Planungsszenarien werden die einzelnen Termine der geplanten Abläufe auf der Baustelle (Transporte, Lagerungen und Montagen) und die Liefertermine koordiniert. Somit kann mittels der Planungsszenarien die Projektdauer bis zur Fertigstellung der Großanlage näherungsweise bestimmt werden. Für die Gestaltung der Transporte und Lagerungen werden u. a. Informationen zu Material- und Ressourcenanforderungen für jeden Prozess (z. B. Lagerbedingungen für Stahlträger) aus der fachspezifischen Ontologie verwendet.

3.3 Durchführung der Simulation

In der Simulation werden die in den Szenarien geplanten Prozesse (Logistik- und Montageprozesse) und der Einsatz verschiedener Ressourcen auf der Baustelle überprüft. Das Simulationsmodell berücksichtigt hierfür Personal, Ausführungszeiten, Material und Ressourcen. Für seine Erstellung werden als Eingangsdaten der Termin- und Ablaufplan (Startpunkt, Dauer und Reihenfolge der Prozesse), die Anlieferungstermine, die Material- und Ressourcenanforderungen für jeden Prozess sowie die Angaben für die Gestaltung der Bauprozesse, Warenannahme und Transportvorgänge verwendet. Ein gemäß dem Terminplan generierter Auftrag durchläuft während der Simulation einen in einem Planungsszenario definierten Prozess (z. B. Warenannahme, Lagerungen, Transporte und Montage). Die Aufträge enthalten die notwen-

digen Informationen, um jeden Prozess richtig und vollständig abzubilden (z. B. Ressourcenanforderungen, Lademenge, Stückliste der Bauprozesse, Startzeitpunkt und Dauer).

Die Prozessabläufe (Krantransporte, Anlieferung auf der Baustelle sowie Montage und Vormontage) werden als Referenzprozesse im Simulationsmodell nachgebildet und laufen während der Simulation entsprechend der Vorgaben in dem definierten Planungsszenario ab. Die modellierten Referenzprozesse entsprechen einer allgemeingültigen Prozessbeschreibung auf der Baustelle (BPMN-Referenzprozesse). Sie können somit auch bei neuen Bauprojekten und Planungsszenarien erweitert, angepasst und wiederverwendet werden.

Im Simulationsmodell (vgl. Abbildung 3) werden auch das Layout, die Positionen der sechs Montageorte der einzelnen Bauelemente und die relevanten Prozesse (als Warteschlangen für die Montageprozesse im Bereich „Processes“ und für die Lagerungen im Bereich „Storages“) nachgebildet. Zudem werden als Ressourcen im ersten Planungsszenario ein Kran, ein Kranführer und ein Arbeiter hinterlegt, die bei Bedarf (bspw. im zweiten Planungsszenario) um weitere Ressourcen (bspw. um einen zweiten Kran) ergänzt werden können. Die Planungsdaten, wie die Reihenfolge der Lieferungen, der Montage, der Lager- und Transportprozesse, die Stücklisten für jeden Bauprozess, die bei den Prozessen verwendeten Ressourcen, die geplanten Ankunftszeiten der Lieferungen auf der Baustelle sowie die geplanten Startzeitpunkte der Bauprozesse, werden in Form von definierten Tabellen in dem Simulationsmodell implementiert. Nach Durchführung der Simulationsläufe können die Planungszeiten (Start- und Endzeitpunkte der Bauprozesse in der Projektplanung) und die simulierten Zeiten der Montageprozesse verglichen werden. Mit Hilfe der Simulation wird sichergestellt, dass die Komponenten nur dann geliefert werden, wenn sie bedarfsgerecht gelagert, transportiert oder direkt eingebaut werden können. Somit wird während der Simulation geprüft, ob die für ein Bauprojekt eingeplanten Ressourcen für die Fertigstellung der Anlage ausreichen.

Im Fall eines Fehlers im Planungsszenario wird während der Simulation eine Fehlermeldung ausgegeben (z. B., dass ein Montageprozess nicht ausgeführt werden kann, da die notwendige Ressource nicht frei ist).

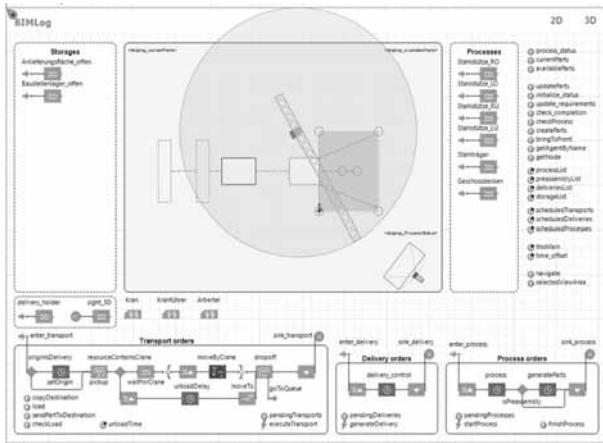


Abbildung 3: Simulationsmodell in AnyLogic

Die simulierten Zeiten der Montageprozesse werden während der Simulation in einer Tabelle eingetragen. Somit können die Dauer der einzelnen Bauprozesse (in Abstimmung mit den simulierten Transport- und Lagerprozessen) und die gesamte Projektdauer näherungsweise bestimmt werden. Auf diese Weise wird simulativ die Bauanleitung überprüft. Die Animation der Prozesse verdeutlicht den simulierten Ablauf der Anlieferungen der Bauelemente, der Montage- und Logistikprozesse (inkl. Transportvorgänge und Lagerungen der Bauelemente) auf der Baustelle. Zur Bewertung der Planungsszenarien (z. B. 1. Planungsszenario mit einem Kran und 2. Planungsszenario mit zwei Kränen) werden neben den simulierten Prozesszeiten auch Leistungskennzahlen wie Auslastung der Ressourcen (in Prozent) und Auslastung der Lagerflächen (in Lagereinheiten) über die gesamte Dauer des Projekts protokolliert (siehe Abbildung 4 für das erste Planungsszenario).

3.4 Simulationsergebnisse

Durch die Simulation der beiden oben beschriebenen Planungsszenarien werden nicht nur die geplanten Termine und Annahmen überprüft, sondern auch die Abstimmung der Montagereihenfolge mit den logistischen Prozessen validiert. Dabei werden die tatsächlichen Start- und Endzeitpunkte der Montageprozesse in Tabellen festgehalten (für das erste Planungsszenario mit einem Kran beträgt die simulierte Projektdauer 118,3 Stunden, für das zweite Planungsszenario mit zwei Kränen beträgt die simulierte Projektdauer insgesamt 93,3 Stunden) und die Lager- und Ressourcenauslastungen ausgewertet. Auf dieser Basis können die geplanten Planungsszenarien angepasst und ggf. verbessert werden.

Die Diagramme zur Auslastung der Ressourcen und

Arbeiter (Abbildung 4) verdeutlichen, dass beim ersten simulierten Planungsszenario der Kran und der Kranführer vor allem zu Projektbeginn fast immer vollständig ausgelastet sind. Bei der Auslastung der Lagerflächen ist erkennbar, dass am Anfang der Bauprozesse die Anlieferungsfläche eine hohe Auslastung hat und dass der Bestand im Lager über die betrachtete Projektdauer sehr gering ist (maximal drei Ladeeinheiten).

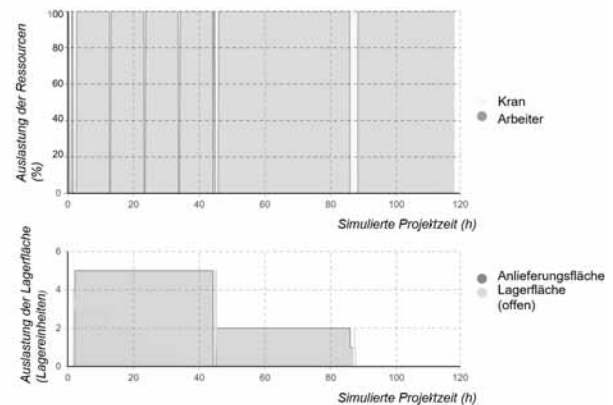


Abbildung 4: Ausgewählte Simulationsergebnisse

Bei dem zweiten Planungsszenario wird durch den Einsatz von zwei Kränen auf der Baustelle die Auslastung der einzelnen Kräne über die Projektdauer reduziert. Allerdings ist die Kollisionsprüfung der beiden Kräne bei einer Logistiksimulation nicht Gegenstand der Betrachtung; dazu müssen entsprechende 3-D Kollisionsuntersuchungen durchgeführt werden. Die Auslastung der Lagerfläche ist ähnlich niedrig wie im ersten Szenario, sodass die Dimensionierung der Lagerflächen bei beiden Szenarien gleichbleibt; allerdings kann die Projektdauer im zweiten Planungsszenario gegenüber dem ersten Szenario reduziert werden.

4 Zusammenfassung und Ausblick

Der Beitrag stellt eine BIM-basierte Vorgehensweise vor, mit der eine digitale Planung der Baustellenlogistik realisiert werden kann. Bei der Planung wird insbesondere der Zusammenhang zwischen der Montage und Logistik der einzelnen Bauelemente und -materialien berücksichtigt. Dazu wird das logistikrelevante Planungswissen in einer Ontologie modelliert. Mit den Informationen aus dieser Ontologie können BIM-Modelle erweitert werden. Die für diese BIM-Modelle erstellten Bauanleitungen bilden die Basis für die Erstellung von Planungsszenarien. Mit der simulationsgestützten Überprüfung dieser Planungsszenarien liegt ein konsistentes und

valides BIM-basiertes Logistikmodell vor; dieses Modell kann auch als Basis für die Steuerung der Logistik auf der Baustelle verwendet werden.

In den obigen Untersuchungen zum Anwendungsbeispiel wird die Simulation nur zur Absicherung von Planungsszenarien mit festgelegten Transportaufträgen verwendet. Die Evaluation des BIM-basierten Vorgehens zur Baustellenplanung hat jedoch gezeigt, dass es sinnvoll sein kann, das Simulationsmodell so zu erweitern, dass es auch für eine umfangreichere Planung der Logistikprozesse auf der Baustelle einsetzbar wird.

Zukünftig werden auf Grundlage der erzielten Ergebnisse weitere Forschungen zum Thema digitale Planung und Steuerung der Baustellenlogistik durchgeführt. Ein wichtiges Forschungsziel ist in diesem Zusammenhang die automatische Erstellung von Logistikmodellen für die Baustelle und ihre semi-automatische Validierung mittels Simulation.

Danksagung. Dieser Beitrag entstand im Rahmen des Forschungsprojekts „BIMLog - BIM-basierte Logistikplanung und -steuerung im Großanlagenbau“, das unter der IGF-Vorhaben-Nummer 19720 N der Bundesvereinigung Logistik (BVL) geführt und über die Allianz industrieller Forschung (AiF) im Rahmen des Programms zur Förderung der Industriellen Gemeinschaftsforschung (IGF) vom Bundesministerium für Wirtschaft und Energie (BMWi) aufgrund eines Beschlusses des Deutschen Bundestages gefördert wird.

Literatur

- [1] Schuh G, Hering N, Brunner A. Einführung in das Logistikmanagement. In Schuh G., Stich V., editors. *Logistikmanagement. Handbuch Produktion und Management*. 2 Auflage. Berlin: Springer; 2013. S. 1–33.
- [2] Schach R, Schubert N. Logistik im Bauwesen. *Wissenschaftliche Zeitschrift der Technischen Universität Dresden*. 2009; 58(1-2): 59–63.
- [3] Wenzel W, Gliem D, Laroque C, Kusturica W. Sichere Prognose der Dauer logistischer Prozesse. *Industrie 4.0 Management*. 2018; 34(5): 43–46.
- [4] Kügler M. CAD-integrierte Modellierung von agentenbasierten Simulationsmodellen für die Bauablaufsimulation im Hochbau. Kassel: Kassel University Press. 2012.
- [5] Bernd F. Die Entwicklung projekt- und fertigungsspezifischer Baulogistikprozesse. In Volkhard F., editor. *Simulation von Unikatprozessen – Neue Anwendungen aus Forschung und Praxis*. Kassel: Kassel University Press; 2011. S. 45–62.
- [6] Liu H, Al-Hussein M, Lu M. BIM-based integrated approach for detailed construction scheduling under resource constraints. *Automation in Construction*. 2015; 53: 29–43.
- [7] Horenburg T. Simulationsgestützte Ablaufplanung unter Berücksichtigung aktueller Baufortschrittsinformationen. München: Technische Universität München. 2014.
- [8] Kalusche W. Projektmanagement für Bauherren und Planer. München: Oldenbourg Wissenschaftsverlag. 2012.
- [9] Hofstadler C. Bauablaufplanung und Logistik im Baubetrieb. Berlin: Springer. 2007.
- [10] Weber J. Simulation von Logistikprozessen auf Baustellen auf Basis von 3D-CAD Daten. Dortmund: Universitätsbibliothek Technische Universität Dortmund. 2007.
- [11] Whitlock K, Abanda FH, Manjia MB, Pettang C, Nkeng GE. BIM for Construction Site Logistics Management. *Journal of Engineering, Project & Production Management*. 2018; 8(1): 47–55.
- [12] Baumgärtel T, Borrmann A, Günthner WA, Juli R, Klauert C, Lederhofer E, Mack J, Willberg U. Bauen heute und morgen. In Günthner W., Borrmann A., editors. *Digitale Baustelle- innovativer Planen, effizienter Ausführen. Werkzeuge und Methoden für das Bauen im 21. Jahrhundert*. Berlin: Springer; 2011. S. 1–21.
- [13] Borrmann, A., König, M., Koch, C. & Beetz, J. Building Information Modeling. Wiesbaden: Springer. 2015.
- [14] Nävy J. Facility Management. Grundlagen, Informationstechnologie, Systemimplementierung, Anwendungsbeispiele. 5 Auflage. Berlin: Springer. 2018.
- [15] Smith P. BIM & the 5D Project Cost Manager. *Procedia - Social and Behavioral Sciences*. 2014; 119: 475–484 [https://doi.org/10.1016/j.sbspro.2014.03.053]
- [16] Astour H. Entwicklung eines BIM-basierten Systems zur Entscheidungsunterstützung mittels Simulation für die Baustelleneinrichtungsplanung. Kassel: Kassel University Press. 2015.
- [17] Cheng JCP, Kumar S. A BIM-Based Framework for Material Logistics Planning. In Seppänen O., González V.A., Arroyo P., editors. *23rd Annual Conference of the International Group for Lean Construction*; 2015. Perth, Australia. S. 33–42.
- [18] Kim H, Anderson K, Lee S, Hildreth J. Generating construction schedules through automatic data extraction using open BIM (building information modeling) technology. *Automation in Construction*. 2013; 35: 285–295.
- [19] Song S, Yang J, Kim N. Development of a BIM-based structural framework optimization and simulation system for building construction. *Construction Innovation*. 2012; 63(9): 895–912.
- [20] Schober K.-S., Hoff P. Think act. Beyond Mainstream: Digitalisierung in der Bauwirtschaft.: Der europäische Weg zur Digitalisierung; 2016. URL: https://www.rolandberger.com/publications/publication_pdf/roland_berger_digitalisierung_bauwirtschaft_final.pdf.
- [21] Wenzel S. Simulation logistischer Systeme. In Tempelmeier H., editor. *Modellierung logistischer Systeme*. Berlin, Heidelberg: Springer; 2018. S. 1–34.
- [22] Wenzel S, Stolipin J, Weber J, König M. Digitale Planung der Baustellenlogistik im Großanlagenbau Ontologie zur Nutzung digitaler Modelle für die Logistikplanung auf der Baustelle. *Industrie 4.0 Management*. 2019; (3): 55–59.
- [23] Lohse W, Laumann J, Wolf C. Stahlbau 1: Bemessung von Stahlbauten nach Eurocode mit zahlreichen Beispielen. Springer-Verlag. 2016.

Referenzmodell basierend auf Wertstromsimulation zur Bewertung von Produktionssystemen in der Angebotsphase

Markus Rabe¹, Walter Wincheringer², Tobias Sohny²

¹Fachgebiet IT in Produktion und Logistik, Technische Universität Dortmund, Leonhard-Euler-Str. 5, 44221 Dortmund, Germany; markus.rabe@tu-dortmund.de

²Digitales Produktionslabor, Hochschule Koblenz, Konrad-Zuse-Straße 1, 56075 Koblenz, Germany; wincheringer@hs-koblenz.de; sohny@hs-koblenz.de

Abstract. Zur Absicherung eines Angebotes von kundenindividuellen Produktionssystemen mit komplexen Materialflüssen bedarf es einer Simulation in der Angebotsphase. Basierend auf der bekannten Wertstrommethode stellt dieser Beitrag ein geeignetes Modellierungskonzept für ein Simulationsreferenzmodell vor. Es wird analysiert, welche Systemelemente für eine wertstrombasierte Simulation in der Angebotsphase unabdingbar sind. Darüber hinaus wird der Detaillierungsgrad dieser Elemente betrachtet, der (i) den Anforderungen eines begrenzten Aggregations- und Planungsaufwandes in dieser Phase und (ii) den Anforderungen, die sich aus dem dynamischen Charakter der Produktionssysteme ergeben, entspricht.

Einleitung

Anbieter von kundenindividuellen Produktionssystemen (PS) mit komplexen Materialflüssen garantieren mit der Abgabe ihres Angebots einen Mindestdurchsatz [1]. Wenn ein realisiertes PS mit komplexem Materialfluss den versprochenen Durchsatz nicht erfüllt, können hohe Kosten für den Anbieter, beispielsweise durch eine Nachbesserung, entstehen. Eine Überdimensionierung des PS schränkt die Wettbewerbsfähigkeit ein. Daher sind die Anbieter bestrebt, ihre Planung hinsichtlich des garantierten Durchsatzes bereits in der Angebotsphase abzusichern.

Zu diesem Zweck müssen die Produktionsprozesse und der Materialfluss des PS auf Auslegungsfehler analysiert werden. Die Wertstrommethode (WSM) ermöglicht hierzu eine übersichtliche Darstellung sowie deren Bewertung [2]. Allerdings können mit dieser Methode die Auswirkungen von zufällig auftretenden Ereignissen,

wie z. B. stochastische Betriebsmittelausfälle, nicht abgebildet werden [3]. Die Bewertung dynamischer Interdependenzen innerhalb eines PS mit komplexem Materialfluss erfordert daher den Einsatz der zeitdiskreten Simulation [1]. Diese ermöglicht es, dynamische Systeme über die Zeit in ausführbaren Modellen zu betrachten [4].

Die Simulation ist jedoch zeitaufwendig, kostspielig und daher für die Angebotsphase nur bedingt geeignet. Eine detaillierte Planung mit Simulation beginnt daher oft erst, wenn der Auftrag bereits erteilt ist [1].

1 Stand der Technik

Wertstrommethode. Die Wertstrommethode (WSM) ermöglicht eine transparente Darstellung aller produktspezifischen Produktionsprozesse mit den zugehörigen Material- und Informationsflüssen in der diskreten Fertigung [2]. Für die Bewertung dynamischer Aspekte hat sich die ereignisdiskrete Simulation (discrete event simulation (DES)) als geeigneter Ansatz erwiesen [1].

Simulation. In Produktion und Logistik wird die Simulation als das „Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell [...]“ [5] definiert. Ihr primäres Defizit ist die zeitaufwändige Modellierung, welche mittels Referenzmodellen (RM) auf ein vertretbares Niveau reduziert werden kann [1].

Referenzmodell. Im Bereich der Simulation dienen Referenzmodelle als Konstruktionsschemata für den Entwurf von aufgabenbezogenen Simulationsmodellen

[6]. Ein Referenzmodell befindet sich dabei auf der gleichen semantischen Stufe wie das Modell, das mit ihm modelliert wird. Es fokussiert sich auf die Semantik und verallgemeinert die Syntax [7]. Darüber hinaus berücksichtigt ein Referenzmodell nicht die Systemarchitektur eines Simulationssystems [7]. Ein Modulelement-Baukasten eines Simulationssystems ist folglich kein Referenzmodell, sondern die simulatorspezifische Implementierung eines Referenzmodells [6].

Dynamische Wertstrommethode. Die Kombination der statischen WSM mit der DES ist daher naheliegend, jedoch keine neue Idee. Den existierenden Ansätzen ist gemein, dass die Wertstrommodelle in Simulationssysteme übertragen oder um diese ergänzt wurden. Das Ziel war jedoch, eine dynamische Lean-Methode zu erhalten [8]. Für die Sicherung der Planungsqualität in der Angebotsphase mit Wertstromsimulation ist jedoch eine höhere Granularität der Eingangsdaten erforderlich [9].

2 Struktur des Referenzmodells

Für die Kombination der WSM mit der DES ist es erforderlich, eine geeignete Granularität der Datenbasis und der Prozessbeschreibungen zu bestimmen. Aufbauend auf der Modellierung der WSM müssen hierzu deren Grundelemente *Produktionsprozess*, *Material-* und *Informationsfluss* analysiert und erweitert werden.

Produktionsprozess. Der Produktionsprozess wird in der WSM durch einen Datenkasten mit produktionspezifischen Kennzahlen abgebildet [2]. Bestimmte vorhandene Parameter werden um die dynamischen Betrachtungen für eine Simulation ergänzt [9].

Sowohl die *Zykluszeit* (ZZ), als kapazitives Angebot eines Produktionsprozesses, als auch die *Prozesszeit* (PZ), zur Abbildung von Chargenprozessen [2], müssen produkt- und prozessspezifisch betrachtet werden.

Zur Darstellung von Haupt- und Nebenmaterialflüssen ist die *Prozessmenge* (PM), zur Abbildung der in einer Charge integrierten Einheiten [2], ebenfalls produkt- und prozessspezifisch zu berücksichtigen.

Auftragsbezogene Ereignisse wie Losgrößen, die eine Rüstaktion bedingen, sowie stochastische Perioden der *Rüstzeit* müssen bedacht werden [4].

Ein Prozentsatz zur Abbildung der *Verfügbarkeit* ist

für eine Simulation nicht ausreichend. Zu diesem Zweck muss der Parameter um die Zeitpunkte des Ausfalls (Mean Time Between Failure – MTBF) in Form von Verteilungen wie der Exponentialverteilung [4], der Anzahl der Aktionen bis zum Auftreten eines Ausfalls oder der Ereignisse, die eine Störung verursachen, erweitert werden. Darüber hinaus muss die Reparaturdauer (Mean Time To Repair – MTTR) in Verbindung mit einer Verteilung, wie der Erlang-Verteilung, angepasst werden [4].

Zuletzt muss die produkt- und rohstoffabhängige *Produktionsausbeute* eines Produktionsprozesses sowie Ereignisse, welche Schlechteile bedingen, mit einbezogen werden [4].

Abbildung 1 zeigt den statischen Prozessdatenkasten (PDK) der WSM, erweitert um die notwendigen Parameter für eine Simulation.

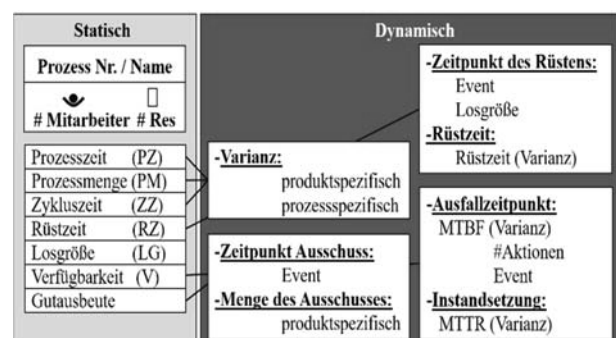


Abbildung 1: PDK erweitert für die DES [9].

Materialfluss. „Der Materialfluss ist die Verkettung aller Vorgänge beim Gewinnen, Be- und Verarbeiten sowie bei der Verteilung von Gütern innerhalb festgelegter Bereiche“ [10]. Mittels der WSM wird dieser lediglich durch die Fließrichtung (Pfeilsymbol) und Materialbestände in Form von Reichweiten (Dreiecke) zwischen Produktionsprozessen dargestellt [2]. Zur Abbildung der logistischen Funktionen zur geographischen oder zeitlichen Transformation von Gütern wird ein erweiterter Datenkasten mit Kennzahlen (Key Performance Indicator – KPI) für den Transport benötigt [11].

Unter *Transport* ist die Ortsveränderung von Gütern mit technischen Mitteln zu verstehen. Ist diese Ortsveränderung räumlich begrenzt, beispielsweise innerhalb einer Fabrik (Intralogistik), wird dies mit dem Begriff *Fördern* spezifiziert und die technischen Transportmittel als *Fördermittel* deklariert [12].

In Bezug auf den diskreten Materialfluss von Gütern erzeugen Fördermittel einen kontinuierlichen oder diskontinuierlichen Fluss des geförderten Materials. Daher

wird zwischen stetiger und unstetiger Förderung differenziert [12].

Analog zu den Produktionsprozessen müssen für den Materialtransport im wertstrombasierten Referenzmodell dynamische Kenngrößen erarbeitet werden, welche die Förderprozesse unabhängig davon, ob es sich um einen Stetig- oder Unstetigförderer handelt, in geeigneter Granularität abbilden. Hierzu ist eine Analyse der Stetig- und Unstetigförderer erforderlich.

Stetigförderer. Stetigförderer (z.B. ein Förderband) sind mit festen Führungsschienen ausgestattet. Die Fördergüter überholen sich dabei nicht und werden nach dem FIFO-Prinzip gefördert [12]. Die Beladung und die Entladung können während des Transportprozesses erfolgen. Abhängig vom Typ dieser Förderer ist eine Stauung des Förderguts möglich [12].

Hierzu sind, in Abhängigkeit der Modellierungsaufgabe, spezifische KPIs notwendig. Abgängig von den Relationen der Zykluszeiten der Produktions- und Förderprozesse können drei Fälle definiert werden:

Für den Fall, dass die ZZ der Produktionsprozesse einen wesentlich größeren Zeitanteil als die Förderzeit aufweist ($ZZ_{\text{Produktionsprozess}} \gg ZZ_{\text{Förderer}}$), muss der Förderprozess nicht abgebildet werden, da er keinen Engpass im Materialfluss bildet. Beispielsweise ist hier ein Glühofen ($PZ = 2h$) zu nennen. Die Förderzeit zum Folgeprozess ist hier mit einer PZ von vier Minuten vernachlässigbar klein und bedarf keiner weiteren Betrachtung. Grundsätzlich ist die Pufferkapazität zu betrachten, welche im Weiteren analysiert wird.

Für den Fall, dass die Zykluszeit eines Förderers wesentlich größer als die Zykluszeit des Produktionsprozesses ist ($ZZ_{\text{Produktionsprozess}} \ll ZZ_{\text{Förderer}}$), liegt ein Auslegungsfehler des Systems vor. Dieser Engpass kann durch eine zusätzliche Fördereinheit oder durch Erhöhung der Fördergeschwindigkeit beseitigt werden. Folglich wird dieser Fall nicht weiter untersucht. Auch hier ist die Pufferkapazität zu berücksichtigen.

Für den Fall, dass die Zykluszeit des Förderers nicht allzu sehr von der Zykluszeit des Produktionsprozesses abweicht, bedarf es einer detaillierten Betrachtung. Neben der Überbrückung von räumlichen Distanzen haben Förderer die Aufgabe zur Entkopplung von Produktionsprozessen [12]. Aus Materialflusssicht ist hierbei die Pufferkapazität entscheidend, welche u. a. einen temporären Taktausgleich der vor- und nachgelagerten Produktionsprozesse erlaubt. Der Puffer ist hierbei ein Anlagen-

bereich (Entkopplungsmodul), welcher eine vorübergehende Pufferung von Fördergütern, um zwei Anlagenbereiche zu entkoppeln, ermöglicht [12].

Aus Wertstromgesichtspunkten handelt es sich hierbei stets um eine FIFO-Verkopplung [2]. Taktzeitdifferenzen oder Rüstzeitpuffer der zu verknüpfenden Produktionsprozesse können hiermit ausgeglichen werden [2]. Folglich kann die Anzahl an Pufferplätzen mittels der vorliegenden Taktzeitdifferenz in Form einer Zykluszeit statisch abgebildet werden (Abbildung 2).

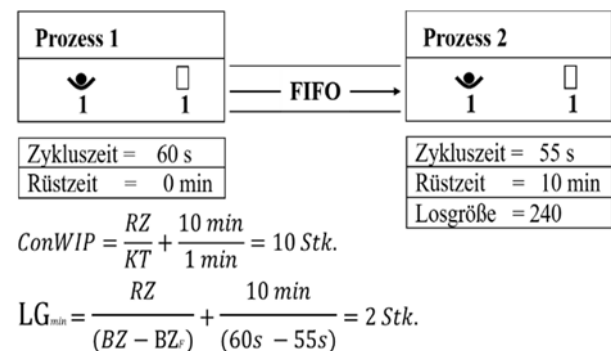


Abbildung 2: Beispiel einer FIFO-Verkopplung (nach [2])

Für einen optimalen Fluss ist somit u. a. die optimale Auslegung des Puffers eines Förderers notwendig. Stochastische Ausfälle der vor- und nachgelagerten Prozesse eines Förderers sowie eine produktspezifische ZZ erschweren diese Auslegung. Hierzu bedarf es einer Analyse der Einflussgrößen. Für den Fall, dass große Distanzen durch Förderer überbrückt werden sollen, existieren im Kontext der Pufferbetrachtung drei Szenarien:

Ist der Förderer leer (der Puffer also nicht genutzt), so fährt der Förderer mit einer Geschwindigkeit von v_{max} und die Fördergüter werden ohne Verzögerung oder Beschleunigung dem nachgelagerten System zugeführt. Entscheidend ist, wann die Fördergüter dieses nachgelagerte System erreichen. Hier bestimmt sich der Zeitbedarf zur Distanzüberbrückung durch die Förderzeit in Abhängigkeit der konstanten Fördererlänge (s_{konst}) und der maximalen Geschwindigkeit (v_{max}):

$$t_{min} = \frac{s_{konst}}{v_{max}} \quad (1) [13]$$

Sobald der Förderer teilbefüllt ist, ergibt sich aus der Materialflusssicht die Frage, wann der nächste Artikel für den Folgeprozess zur Verfügung steht. Sobald der Folgeprozess ausgelastet ist ($PM = 1$) wird maximal ein Teil

weitergeleitet. Auf dem Fördersystem ergibt sich somit eine Stausituation. Das Fördergut erfährt hierbei eine *Verzögerung* ($a_v < 0$) bei einem Stopp sowie eine *Beschleunigung* ($a_b > 0$) bei einer erneuten Anfahrt. Daraus ergibt sich eine vereinfachte lineare Geschwindigkeits-Darstellung, die auch als Trapezfahrt bezeichnet wird (Abbildung 3) [13].

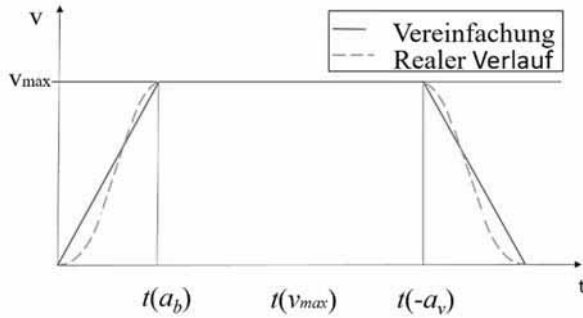


Abbildung 3: Geschwindigkeits-Zeit-Diagramm (nach [13]).

Die Förderzeit untergliedert sich dabei in drei Teilspele: die Beschleunigungsphase $t(a_b)$, die konstante Fahrzeit $t(v_{max})$ und die Verzögerungsphase $t(-a_v)$.

Das reale Geschwindigkeitsprofil $v(t)$ weicht in der Beschleunigungs- und Bremsphase von dem vereinfachten trapezförmigen Geschwindigkeitsprofil ab (S-Kurve). Das reale Systemverhalten kann mit dem trapezförmigen Geschwindigkeitsprofil allerdings ausreichend genau angenähert werden [14]. Die gesamte Spielzeit ergibt sich somit aus der Summe der Teilspele [13].

$$t_{ges} = t(a_b) + t(v_{max}) + t(-a_v)$$

$$= \frac{s_{konst}}{v_{max}} + \frac{v_{max}}{a} \quad (2) [13]$$

Folglich fallen die Beschleunigung und Verzögerung additiv bei einem Stoppvorgang mit anschließender Anfahrt des Förderers an. Für die Ermittlung des Zeitanteils bei einer Beschleunigung und Verzögerung ist die zu überbrückende Distanz von Bedeutung. Ist die Wegstrecke zu kurz, kann die Nenngeschwindigkeit v_{max} (Trapezfahrt) nicht erreicht werden. Die Beschleunigungsphase geht direkt in die Verzögerungsphase über und weist eine dreieckige v - t -Darstellung auf (Dreiecksfahrt, Abbildung 4) [15].

Der Zeitbedarf einer Beschleunigung oder Verzögerung ist somit unabhängig von der Fördererlänge zu berücksichtigen. Entscheidend sind die durch den Folgeprozess verursachten Anzahlen an Stoppvorgängen n .

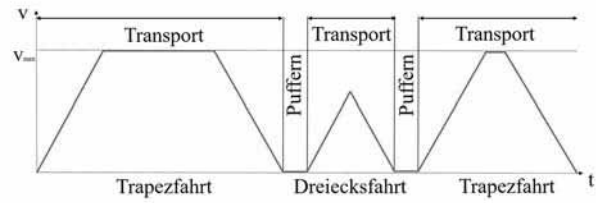


Abbildung 4: v - t -Darstellung eines Stetigförderers

$$t_{ges} = \frac{s_{konst}}{v_{max}} + n * \frac{v_{max}}{a} \text{ für } s \geq \frac{v_{max}^2}{a} \text{ Trapezfahrt}$$

$$\text{bzw. } 2 * t(a) = 2 * \sqrt{\frac{s}{a}} \text{ für } s < \frac{v_{max}^2}{a} \text{ Dreiecksfahrt}$$

mit $\{n \in \mathbb{N}\}$ (3) [15]

Es sei angemerkt, dass Trägheitsmomente sowie ein entstehender Schlupf bei einer Anfahrt nicht berücksichtigt werden. Darüber hinaus wird $a_b = -a_v$ angenommen. Eine durch einen Stopper verursachte Verzögerung mit $-a_v > a_b$ wird nicht betrachtet.

Die zu überbrückende Distanz bei einem erneuten Anlauf ergibt sich aus der *Länge* des jeweiligen Förderelements plus seinem *Abstand* zu den folgenden Fördergütern (Abbildung 5).

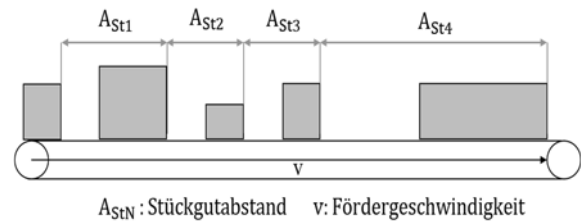


Abbildung 5: Stückgutabstand [12]

Diese ergeben den *Stückgutabstand* und ermöglichen die Differenzierung zwischen einer Dreiecks- und einer Trapezfahrt. Die jeweilige Fördergutlänge einschließlich des Abstandes ist produktspezifisch und muss dem jeweiligen *Produkt* zugeordnet werden.

Die vorangegangenen Analysen bedingen, dass der Einfluss des Stückgutabstands sowie die Geschwindigkeit und der in Abhängigkeit einer Stausituation relevante Faktor $2t(a)$ bei einer Zuführung berücksichtigt werden müssen. Dies gilt insbesondere, wenn der vorgelagerte Prozess des Förderers eine schnellere PZ aufweist als die Fördergüter weggefördert werden. Das Fördergut kann dem System erst zugeführt werden, sobald die erste Position (abhängig vom Fördergutabstand) auf dem Förderer frei ist (Abbildung 6).

Dieser entstehende zeitliche Verlust einer Teilzykluszeit führt u. U. zu einem Materialflussabriss und ist dementsprechend zu berücksichtigen. Darüber hinaus ist die

zusätzliche Zeitspanne, bis das Produkt das Fördersystem verlässt, zu beachten. Diese muss, in Abhängigkeit der *Stückgutlänge*, zur Förderzeit addiert werden. Bei einer diskreten Fertigung in der Praxis ist diese zusätzliche Zeit in der Regel vernachlässigbar klein.

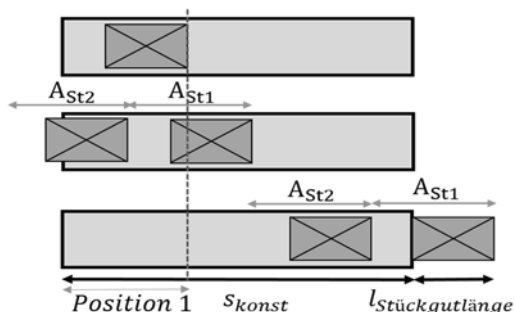


Abbildung 6: Zuführung und Abfuhr eines Förderelements

$$t_{\min+\text{Ausschleusung}} = \frac{s_{\text{konst}} + l_{\text{Stückgutlänge}}}{v_{\max}} \quad (4)$$

Aus Szenario 2 ergeben sich folgende Kenngrößen:

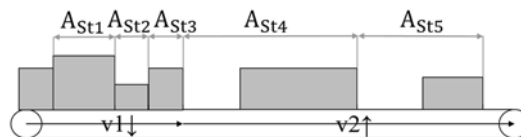
- Fördererlänge
- Geschwindigkeit
- Beschleunigung/Verzögerung (Anzahl n)
- Artikellänge
- Stückgutabstand
- Zufuhr/Abfuhr

Sobald die Fördergüter – z. B. bedingt durch einen Ausfall des Folgeprozesses – sich bis zu dem zuführenden Prozess aufstauen, liegt eine Blockade vor. Der so entstehende Stau beschreibt die zeitliche Verzögerung eines Materialflussobjektes, bedingt durch das Warten auf die Abfertigung eines oder mehrerer sich vor ihm befindlichen Objekte [16]. Entscheidend hierbei ist die Kapazität des Förderers. Diese entspricht dem maximalen physischen Aufnahmevermögen des Förderers [13].

Die Kapazität ergibt sich aus dem Verhältnis des produktspezifischen *Stückgutabstandes* zu der *Fördererlänge*. Bedingt durch den schwankenden Stückgutabstand ergibt sich eine variable Kapazität. Diese soll folglich durch das System ausgewertet werden. Bei nicht stauenden Förderern ist diese geringer, da der Förderer stoppt sobald der Folgeprozess ausgelastet ist, was den vorgelagerten Prozess blockiert. Erst wenn der Folgeprozess frei wird, ist eine Wiederaufnahme der Förderung möglich.

Der Fördergutstau kann mittels eines Gruppen- oder sequentiellen Abzuges aufgelöst werden [12]:

Bei einem Gruppenabzug werden die Fördergüter in einem Block weitergefördert [12]. Bei einem sequentiellen Abzug werden die aufgestauten Güter unter Einhaltung des Stückgutabstandes nacheinander weitergefördert (Abbildung 7) [12].



A_{StN} : Stückgutabstand v : Fördergeschwindigkeit

Abbildung 7: Auflösung des Materialstaus [12]

Aus Szenario 3 ergeben sich folgende Kenngrößen:

- Kapazität
- Stauauflösung (sequenzieller oder Gruppenabzug)

Unabhängig von der betrachteten Kenngröße ist die *Verfügbarkeit* der Förderer, wie bei den Produktionsprozessen, mittels Verteilungen abzubilden.

Ein weiterer Aspekt, welcher bei der Analyse von Stetigförderern zu berücksichtigen ist, betrifft die *Verknüpfung* von Förderern. Bei einem Übergang von unterschiedlichen Stetig-Fördersystemen können verschiedene Geschwindigkeiten vorliegen. Hierzu ergeben sich drei Fälle (Abbildung 8):

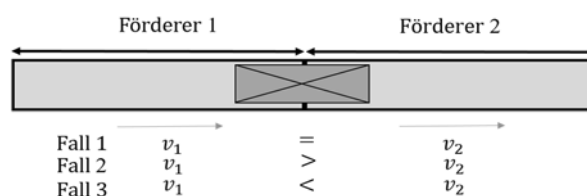


Abbildung 8: Übergang von Förderern

Der Massenschwerpunkt, Reibungskräfte sowie ein möglicher Schlupf werden nicht betrachtet. Sobald das Fördergut sich auf dem folgenden Förderer befindet, sind dessen Geschwindigkeit und Beschleunigung maßgebend.

Wenn die Geschwindigkeiten identisch sind ($v_1 = v_2$), können die Förderer als eine Fördereinheit kombiniert werden.

Ist $v_1 > v_2$, so ist v_2 beim Übergang bestimmend und verursacht einen Materialstau. Vorhandene Stückgutabstände werden bei dem Übergang reduziert. Der

zusätzliche Zeitbedarf des spezifischen Stückgutabstands wird beim Verlassen von Förderer 2 berücksichtigt.

In dem Fall, dass $v_1 < v_2$ ist, wird v_2 bestimmend für die Zuführung und das Fördergut wird beim Übergang schneller gefördert. Vorhandene Stückgutabstände werden am Übergang vergrößert. Der zusätzliche Zeitbedarf des spezifischen Stückgutabstands wird beim Verlassen von Förderer 2 berücksichtigt.

Abbildung 9 fasst die Erkenntnisse der Analyse in einem Datenkasten für Stetigförderer zusammen. Die definierten *Eingabeparameter* sowie die *produktspezifischen Attribute* werden bei der Simulation berücksichtigt, sodass die sich daraus ergebende *Förderzeit* und *Kapazität* bestimmt werden können. Zudem werden die produktspezifische Ablauflogik bei einer *Anfahrt* oder *Stoppaktion* sowie die *Systemzufuhr* und *-abfuhr* automatisch verarbeitet. Darüber hinaus bedarf es der Auswahl, ob es sich um einen *stauenden* oder *nicht stauenden* Förderer handelt sowie der Definition der *Stauauflösung*. Mittels des Datenkastens soll neben der Abbildung der Stetigförderer mit einer Simulation die optimale Pufferkapazität eines Förderers durch den Anwender ermittelt werden (Was-wäre-wenn-Szenarien).

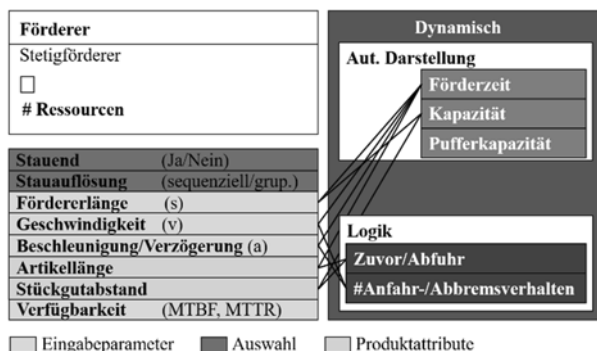


Abbildung 9: Datenkasten für Stetigförderer

Unstetigförderer. Im Vergleich zu den Stetigförderern können sich die Unstetigförderer (z. B. Fahrerlose Transportsysteme) frei bewegen und haben daher einen höheren Grad an Flexibilität. Der Be- und Entladevorgang findet statt, sobald der Unstetigförderer stoppt [12]. Der Be- und Entladevorgang kann hierbei mit dem zuvor definierten Datenkasten für Produktionsprozesse dargestellt werden [9]. Die Darstellung von Unstetigförderern wird in weiteren Forschungsansätzen analysiert. Neben seiner Kapazität und Verfügbarkeit spielt die Förderzeit eine entscheidende Rolle. Basierend auf der Geschwindigkeit und Beschleunigung ist diese abhängig

von der Quellen- und Senkenposition und dem sich daraus ergebenden Pfad. Zur Darstellung ist die Definition einer in der Praxis üblichen Transportmatrix, welche die Abbildung aller Fahrdistanzen ermöglicht, denkbar [17]. Potentielle Verzögerungen wie das Stoppverhalten durch Mitarbeiter auf der Fahrbahn oder Vorfahrtsregeln an Kreuzungen lassen sich in einer frühen Angebotsphase durch Erfahrungsgrößen berücksichtigen [18]. Aus der Materialflusssicht ist allerdings die Anzahl an FTS relevant, sodass der Fluss nicht abreißt. Ein einzelnes Fahrzeug ist bei einer solchen frühen Planung nicht ausschlaggebend. Entsprechende Kenngrößen für die ausreichend genaue Abbildung von Unstetigförderern sind zu erarbeiten und in Form eines Datenkastens darzustellen.

Informationsfluss. Für eine Kombination der Produktions- und Transportprozesse bedarf es einer logistischen Verknüpfung. Innerhalb der WSM werden hierzu die Informationsflüsse visuell abgebildet. Für eine DES betrachten die WSM und die bestehenden Ansätze der Wertstromsimulation die Materialflusslogik zu oberflächlich [19]. Parallel ablaufende Prozesse, stochastische Einflüsse und Montageprozesse mit verschiedenen Produkten werden nicht beachtet. Hierzu bedarf es einer Berücksichtigung der Materialflusslogik, sodass Produktionsprozesse untereinander oder mit diversen Förderern verknüpft werden können. Hierzu werden die Grundelemente der WSM um die *Logik*, welche genau dies ermöglichen soll, erweitert (Abbildung 10).

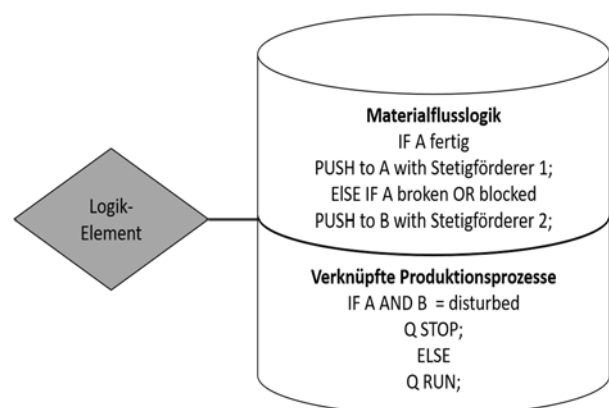


Abbildung 10: Beispiel zur Abbildung der Materialflusslogik

Beispielsweise kann nach dem Montageprozess (Maschine Q) das Produkt auf Prozess A oder B geprüft werden. Der reguläre Materialfluss erfolgt über eine direkte Verbindung von Q nach A durch den Stetigförderer

1 (Förderband). Wenn Prozess A ausfällt oder blockiert ist, wird das Produkt durch Stetigförderer 2 (Förderband) nach B transportiert. Sind Prozess A und B ausgefallen, stoppt Prozess Q.

Produkt. Die WSM analysiert den Wertstrom eines diskreten Produkts oder Produkttyps [2]. Für eine realistische Wertstromsimulation müssen verschiedene Produkte mit unterschiedlichen Zykluszeiten und Verhaltensweisen berücksichtigt werden. Dies ermöglicht eine Darstellung von Wertströmen und Montageprozessen mit unterschiedlichen Einheiten in Abhängigkeit von verschiedenen Produktionsaufträgen. Realisiert wird dies durch die Abbildung des *Produktes* mit spezifischen Attributen, z. B. der Prozessablauffliste einschließlich der Informationen über Quellen und Senken sowie des spezifischen Stückgutabstands. Das produktspezifische Attribut wird hierbei am jeweiligen Prozess abgefragt, was zu unterschiedlichen Ereignissen führt (Abbildung 11).

Produktname
Produktfamilie
Losgröße
Prozessablauf:
• Prozess Q
• Prozess A v B
Stückgutabstand

Abbildung 11: Darstellung eines Produktes

3 Referenzmodell basierend auf der WSM

Die Vorteile des WSM, der DES und des RM sollen miteinander kombiniert werden. Dazu gehören die Transparenz des WSM, die Dynamik der DES und der reduzierte Modellierungsaufwand durch ein RM.

Darüber hinaus sollten für jeden Prozess Daten wie Zyklus und Transportzeit berücksichtigt werden. Ein Zeitdiagramm der Zyklus- und Transportzeiten ermöglicht die Erkennung von Engpässen. Abbildung 12 zeigt – basierend auf dem vorherigen Beispiel – wie ein Simulationsmodell, welches mit dem auf der Wertstromsimulation basierenden Referenzmodell modelliert wurde, aussehen würde.

Zusätzlich zur Analyse der WSM-Elemente (Materialflusslogik und Produkt) für das Referenzmodell ist eine Kosten-Nutzen-Bewertung erforderlich, um den Aufwand für die Modellierung mit dem Referenzmodell

auf der Grundlage der Wertstromsimulation zu bestimmen. Diese Evaluierung und die Aspekte der Abbildung von Beständen und Puffern, die Integration des Zeitdiagramms und die Definition der notwendigen Ergebnisse (wie Durchsatz und Vorlaufzeit) werden in der weiteren Forschung betrachtet werden.

4 Zusammenfassung und Ausblick

Die Etablierung der Simulation in der Angebotsphase bietet eine Verbesserung für Unternehmen, die Produktionssysteme mit komplexen Materialflusssystemen planen. Mögliche Planungsfehler können früher erkannt werden, was die Wettbewerbsfähigkeit erhöht. Dies bildet die Grundlage für eine sichere und korrekte Planung sowie für eine exakte Angebotserstellung.

Eine wichtige Voraussetzung ist, dass der Modellierungsaufwand und die damit verbundene Vorlaufzeit der Angebotserstellung gering bleiben müssen. Die WSM ermöglicht hierzu eine transparente und übersichtliche Darstellung aller für die Herstellung eines Produktes erforderlichen Produktionsprozesse in Verbindung mit den entsprechenden Material- und Informationsflüssen [2]. Die Simulation ermöglicht eine exakte Abbildung dynamischer Systeme im Wertstrom [4]. Dies motivierte die Idee, VSM und Simulation zu kombinieren. Die Nachteile der Simulation sind der hohe Modellierungsaufwand sowie der intensive Ressourcenbedarf, der mit einem wertstromspezifischen Referenzmodell auf ein akzeptables Maß reduziert werden soll. Dieses Referenzmodell, das auf der Wertstromsimulation basiert, soll sich in den Hauptmerkmalen seiner Anwendung nicht von der herkömmlichen statischen WSM unterscheiden. Für den Aufbau des Referenzmodells werden die Grundelemente des WSM analysiert und um dynamische Aspekte erweitert.

Der Detaillierungsgrad des Referenzmodells – insbesondere zur Abbildung der Materialflusslogik – und seine Kosten-Nutzen-Bewertung werden in weiteren Untersuchungen erarbeitet.

Danksagung

Das Forschungsprojekt wurde von der Europäischen Union aus dem Europäischen Fonds für regionale Entwicklung sowie vom Land Rheinland-Pfalz gefördert, in Kooperation mit dem Industriepartner AtlanticC GmbH in Bernkastel-Kues.

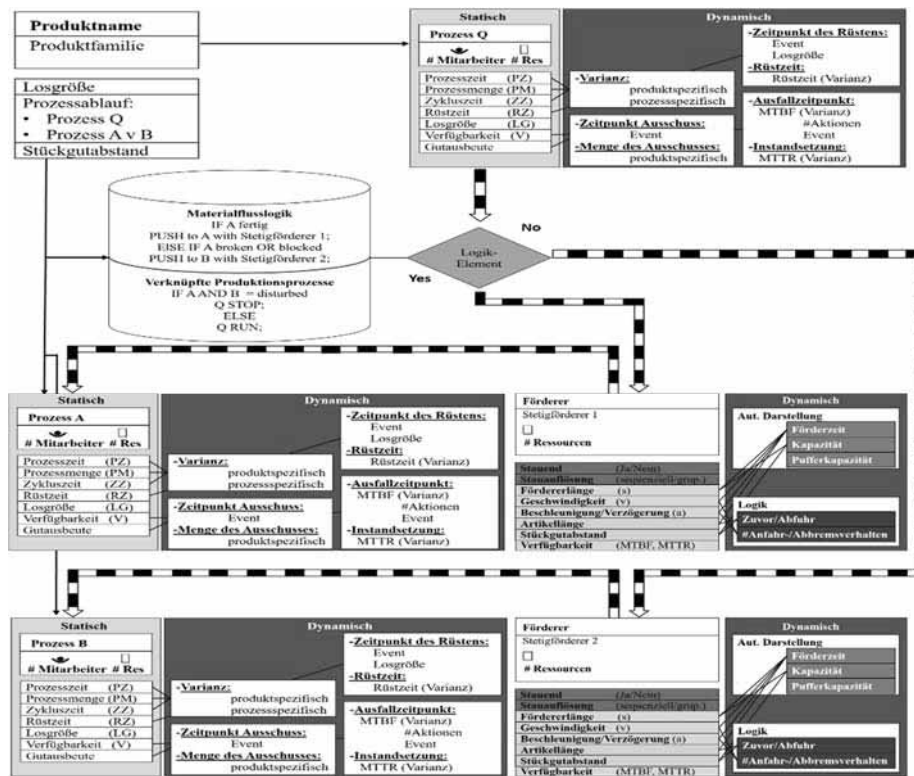


Abbildung 12: Simulation mit dem RM basierend auf Wertstromsimulation

References

- [1] Friedland R, Kühling M. Referenzmodelle für Fertigungssysteme. In Wenzel S., ed. *Referenzmodelle für die Simulation in Produktion und Logistik*. Erlangen: SCS; 2000.
- [2] Erlach K. *Wertstromdesign: Der Weg zur schlanken Fabrik*. 2nd ed. Berlin: Springer; 2010.
- [3] Luger A, Winkler H. Von der Wertstromanalyse zum Wertstrommanagement. *ZWF*. 2017; 112 (4), 261-265.
- [4] Gutenschwager K, Rabe M, Spieckermann S, Wenzel S. *Simulation in Produktion und Logistik: Grundlagen und Anwendung*. Berlin: Springer; 2017.
- [5] VDI-Richtlinie 3633: *Simulation von Logistik-, Materialfluss und Produktionssystemen – Blatt 1*. Berlin: Beuth, 2014.
- [6] Klinger A, Wenzel S. Referenzmodelle – Begriffsbestimmung und Klassifizierung. In Wenzel S., ed. *Referenzmodelle für die Simulation in Produktion und Logistik*. Erlangen: SCS; 2000.
- [7] Schütte R. *Grundsätze ordnungsmäßiger Referenzmodellierung: Konstruktion konfigurations- und anpassungsorientierter Modelle*. Wiesbaden: Springer; 1998.
- [8] Drees J. Neue Perspektiven für die WSM: Wertstrom Management 4.0. *ZWF*. 2018; 113 (9), 605-609.
- [9] Rabe M, Wincheringer W, Sohny T. Reference model based on value stream simulation for the evaluation of production systems in the bidding phase. *RIRL-SCM*; 2020, to be published.
- [10] VDI-Richtlinie 2689: *Leitfaden für Materialflussuntersuchungen*. 2019.
- [11] Knössl T. Logistikorientierte Wertstromanalyse. In Günther W, Boppert J., editors. *Lean Logistics: Methodisches Vorgehen und praktische Anwendung in der Automobilindustrie*. Berlin: Springer; 2013.
- [12] Ten Hompe M, Schmidt T, Dregge J. *Materialflusssysteme: Förder- und Lagertechnik*. Berlin: Springer; 2018.
- [13] VDI-Richtlinie 3978: *Durchsatz und Spielzeitberechnungen in Stückgut-Fördersystemen*. 2018.
- [14] Gudehus, T. *Logistik 2: Netzwerke, Systeme und Lieferketten*. Berlin, Heidelberg: Springer; 2012.
- [15] Lippolt CR. *Spielzeiten in Hochregallagern mit doppeltiefer Lagerung*. Dissertation TH Karlsruhe. 2003.
- [16] Jung E, ten Hompe M. Analytische Stauprognose in Stetigfördersystemen im Rahmen der Systemplanung. *Logistics Journal: Proceedings – ISSN 2192-9084*; 2013.
- [17] Lienert T, Fottner J. Entwicklung einer generischen Simulationsmethode für das zeitensterbasierte Routing Fahrerloser Transportfahrzeuge. *Logistics Journal: Proceedings – ISSN 2192-9084*; 2013.
- [18] Ullrich G. *Fahrerlose Transportsysteme Eine Fibel – mit Praxisanwendungen – zur Technik – für die Planung*. 2nd ed. Wiesbaden: Springer; 2014.
- [19] Türk S, Weimer A, Schubert L, Drees J. Dynamische Simulation von Wertströmen. *ZWF*. 2014; 109 (11), 839-842.

A Simulation Study on the Performance of Wafer Fabs with Hot Lots Under WIP Balance and Due Date Control Policies

Zhugen Zhou^{1*}, Oliver Rose¹

¹Chair of Modeling and Simulation, Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany; *zhugen.zhou@unibw.de

Abstract. In semiconductor wafer fabs, hot lots refer to a group of products which have high priority for various reasons, e.g. pilot products or due date commitments to customers. Hot lots are given the highest priority with the purpose of reducing their cycle time. However, hot lots often cause irregular WIP flow that has great impact on cycle time and throughput of regular lots. In this paper, we present a simulation study including two cases on handling degraded performance of regular lots under the control of minimum inventory variability scheduling (MIVS) and operation due date (ODD). In the first case, when hot lots represent 10% of total release, we propose to improve the pace of lot movement for MIVS and break dominance of due date control ODD. In the second case, as the percentage of hot lots increases to 30%, we apply a hierarchical dispatching scheme in which target cycle time is set higher for hot lots and regular lots. The simulation results show that for the first case, the compensatory methods are able to improve the performance of regular lots by overcoming deficiencies of MIVS and ODD; for the second case, it is important to establish a target cycle time if a trade-off is needed between hot lots and regular lots.

Introduction

Semiconductor wafer fabrication facilities (wafer fabs) contain hundreds of production equipment and dozens of kinds of wafer products. Each kind of product has a unique technologic process flow which includes hundreds of processing steps. There are many characteristics of wafer fabs, such as hot lots, re-entrant processing flows, batch tools, sequence dependent setups, unpredictable equipment failures and so on, which differentiate wafer fabs from other job shops and flow shops. In general, release rules and dispatching rules are two major ways that are applied to control the wafer fabs for the purpose of decreasing cycle time, cycle time variance and achieving on time delivery [1].

For traditional wafer fabs operating mass production, work-in-process (WIP) is the main concern in shop floor control since it has a major influence on overall manufacturing costs. To reduce the inventory level, WIP oriented

dispatching rules are applied in these wafer fabs. For that purpose, minimum inventory variability scheduling (MIVS) [2] is the representative rule. MIVS considers both upstream and downstream operations. It gives the highest priority to an operation which has a high WIP and its downstream operation has a low WIP, in order to avoid starvation at downstream operations. In contrast, it gives the lowest priority to an operation which has a low WIP and its downstream operation has a high WIP. The MIVS intends to keep the WIP of each operation close to the average target WIP level. As many wafer fabs change from mass production to mass customization to satisfy customers, due dates become another critical factor. Due date oriented dispatching rules [3], e.g., earliest due date and operation due date (ODD) are applied to achieve on-time delivery in these wafer fabs. The ODD rule breaks up the slack time into as many segments as the number of operations of a lot, which means it considers due dates for all intermediate operations. The ODD value of operation i is defined as: $ODD = ReleaseTime + RPT(i) * DDFF$, where $RPT(i)$ denotes the raw processing time for a sequence of processing steps or operations from operation 1 to operation i (including operation i) and $DDFF$ denotes target due date flow factor which is the ratio of the target cycle time and the raw processing time of a lot.

Although different kinds of dispatching rules are already available for wafer fabs, shop floor control is no trivial task due to production variations, e.g., hot lots. Hot lots refer to a group of products that have the highest priority for various reasons, e.g., pilot products, process testing and due date commitments to customers. Hot lots are given high priority to reduce their cycle time. However, hot lots often cause an irregular WIP flow that has great impact on the cycle time and throughput of regular lots. A number of researchers have examined the impact of hot lots on the performance of wafer fabs. [4] considered hot lots as one of the production variabilities that affect overall cycle time and should be well managed. [5] carried out

a simulation study to understand the impact of hot lots on the cycle time of regular lots. They demonstrated that as the ratio of hot lots increases, average cycle times and standard deviation of cycle times of regular lots increase drastically. [6] showed that hot lots cause production capacity loss, especially for the batch processing. [7] applied simulation to analyze the impact of different percentages of hot lots on the cycle time of two different products. To reduce the impact of hot lots on the overall cycle time of products, they developed rules-of-thumb to release appropriate amounts of hot lots to the wafer fabs. [8] developed an analytical method based on mean value analysis to predict the performance of wafer fabs in the presence of hot lots. The simulation results also demonstrate that hot lots have a significant impact on the mean cycle time, variance of cycle time and throughput rate of regular lots. [9] carried out a simulation study on the scheduling policies to minimize the cycle time of hot lots in batch processes.

In conclusion, hot lots have a significant impact on the performance of wafer fabs, in particular, the cycle time and throughput are degraded tremendously for regular lots. The researchers above focus on either small manufacturing lines or applying the first in first out (FIFO) dispatching policy. It is of much importance to examine hot lots in a complete wafer fabs with different shop floor dispatching policies like MIVS or ODD. Furthermore, as the ratio of hot lots increases, it is of practical interest to find a solution to improve the degraded performance of regular lots. We attempt to address these two issues in this study.

In this paper, we study two cases that are defined by the percentage of hot lots in wafer fabs. The first case, where wafer fabs contain 10% hot lots of the total release, is considered as low ratio of hot lots. Hot lots cause irregular WIP flow for regular lots since hot lots have priority at all stages of processing. Both MIVS and ODD rules can not successfully handle it. For MIVS, we focus on improving the pace of regular lots by introducing a flow factor (FF) rule. For ODD, we suggest to break the dominance of due date control to speed up lot movement. The second case, when the hot lots ratio is increased to 30%, the wafer fabs are running in an extreme manner. As the hot lots enter to the wafer fabs continuously, all resources are occupied by them. The consequence is that the regular lots can not be processed on time which causes serious congestion in front of tool groups. In this situation, we try to find a tradeoff between hot lots and regular lots to answer the question how the performance of the regular lots

can be improved at the cost of performance of the hot lots.

This paper is organized as follows. In Section 1, we introduce the wafer fabs model used in this study. In Section 2, we present two study cases by describing the problem and solutions in detail. The conclusions can be found in Section 3.

1 Simulation Model

The whole wafer fabs dataset MIMAC6 from the Measurement and Improvement of MANufacturing Capacities (MIMAC) project is used for this study. We refer the interested reader to [10] for details. The MIMAC6 is a typical complex wafer fabs model including:

- 9 products, 9 process flows, a maximum of 355 process steps. (Table 1 lists the basic information of the products)
- 24 wafers in a lot. 2777 lots are released per year under a fab loading of 100%.
- 104 tool groups (work-centers), 228 tools (machines). 46 single processing tool groups, 58 batching processing tool groups.
- Sequence dependent setups, rework, MTTR (mean time to repair), and MTBF (mean time between failures) of tool groups.

The simulation experiments are carried out by Factory eXplorer from WWK. The wafer fabs loading is set to 95%. The simulation length is 48 weeks with 3 replications, and the first 12 weeks are considered as warm-up periods.

Products	Raw Processing Time (days)	Time until next Release (hours)
B5C	17.6	30.4
B6HF	16.6	92.9
C4PH	10.9	43.9
C5F	15.1	36.4
C5P	11.8	10.9
C5PA	13.5	17.2
C6N3	14.9	47.6
C6N2	13.2	41.1
OX2	12.8	35.2

Table 1: Basic information of products in MIMAC6 model.

2 Simulation Cases

2.1 Low Ratio of Hot Lots

Before we introduce hot lots into the MIMAC6 wafer fabs, the MIVS and ODD rules are utilized for shop floor control, respectively. They are represented as ‘MIVS(1)’ in Table 2 and ‘ODD(1)’ in Table 3. For the first study case, according to Table 1 product ‘B5C’ comprises approximately 10% of total release. Thus, product ‘B5C’ is considered as hot lot, and other products are considered as regular lots. Thus, the dispatching policies are changed to hierarchical priorities which are shown as ‘MIVS(2)’ in Table 2 and ‘ODD(2)’ in Table 3. The hot lots have priority 1, and if two hot lots have the same priority, FIFO is used for tie-breaking. The regular lots have priority 2, and if two regular lots have the same priority, the MIVS and ODD rules are used for tie-breaking, respectively.

When the hot lots obtain higher priority over the regular lots at all stages of processing, a large number of regular lots pile up in front of tool groups. The fact is that the shortcomings of MIVS and ODD rules are magnified to certain extent.

Problem and Solution for MIVS.

The MIVS rule focuses on balancing WIP to reduce average cycle time. Nevertheless, it ignores the importance of good pace of lot movement. It would rather push a lot with less queue time to balance downstream tool-groups than push a lot with a long queue time, which leads to a degraded performance of cycle time variance. In a tool-group, after processing hot lots the MIVS rule faces a huge challenge as a large number of regular lots wait in queue. In order to overcome the drawback, the flow factor (FF) rule is applied to improve the WIP flow of regular lots.

The FF rule is an extension from a performance indicator called flow factor. It is a dynamic dispatching rule based on the ratio between accumulated cycle time and accumulate raw processing time [11], $FF = \text{AccumulatedCycleTime} / \text{AccumulatedRawProcessingTime}$. Obviously, a small flow factor is desirable as it indicates a low cycle time. The FF rule is expected to improve the pace of lot movement as it attempts to keep lots going through the wafer fabs with the same flow factor. The FF rule is incorporated into the hierarchical dispatching to better distinguish lots that obtain the same priority from the MIVS rule, which is depicted as ‘MIVS(3)’ in Table 2. When two lots obtain the same priority from MIVS, the

one with a higher FF value is preferred.

Hierarchical dispatching policies based on MIVS (10% ratio of hot lots)		
MIVS(1)	MIVS(2): Hot lots + MIVS	MIVS(3): Hot lots + (MIVS + FF)
All lots: -> MIVS	Priority 1: Hot lots: -> FIFO Priority 2: Regular lots: -> MIVS	Priority 1: Hot lots: -> FIFO Priority 2: Regular lots: -> MIVS -> FF

Table 2: Three hierarchical dispatching policies based on MIVS.

Problem and Solution for ODD.

In contrast to the MIVS rule, even though the hot lots disturb the WIP flow of the regular lots, the ODD rule still manages the difficulty to achieve good pace of movement, which brings an excellent performance of cycle time variance. The fact is although regular lots have loose target due dates, some of them are already close to their due dates or even late after hot lots finish processing. The ODD rule overemphasizes the pace of movement. Thus, the fresh regular lots have to wait for the late regular lots. This procedure leads to slow movement. As a result, the ODD rule produces high cycle time for regular lots. We realize that breaking the dominance of ODD rule is the way to accomplish fast movement for regular lots.

The modified operation due date rule (MOD) is applied in this case. The MOD rule is a combination of ODD and shortest processing time (SPT). It is expressed as follows: $MOD = \text{Max}(ODD, \text{now} + PT)$, where ODD is the operation due date of a lot, now is current time and PT is the processing time of a lot. A smaller MOD value indicates a higher priority. The MOD rule has the potential to solve the problem of ODD because it performs like the SPT rule when the due date becomes tight, and the SPT rule aims at achieving low cycle times. The dispatching policy with the MOD rule is listed as ‘ODD(3)’ in Table 3.

Hierarchical dispatching policies based on ODD (10% ratio of hot lots)		
ODD(1)	ODD(2): Hot lots + ODD	ODD(3): Hot lots + (ODD + SPT)
All lots: -> ODD	Priority 1: Hot lots: -> FIFO Priority 2: Regular lots: -> ODD	Priority 1: Hot lots: -> FIFO Priority 2: Regular lots: -> ODD+SPT

Table 3: Three hierarchical dispatching policies based on ODD.

Simulation Results.

The simulation results are presented in Tables 4 and 5. Cycle time, cycle time variance, cycle time upper percentile 95% and throughput are considered as performance measure.

Table 4 shows results from the dispatching policies in Table 2. The 'MIVS(1)' policy produces the results without hot lots in the wafer fabs. For the 'MIVS(2)', when product 'B5C' is introduced as hot lot, the performance

of regular products is degraded. However, the overall performance of the wafer fabs is similar to the case of 'MIVS(1)'. The problem is that the 'MIVS(2)' policy produces particularly high cycle time variance and cycle time upper percentile 95% for products 'C4PH' and 'C6N2'. The 'MIVS(3)' policy utilizing the FF rule successfully solves the problem and outperforms 'MIVS(2)' for all performance measures. After processing hot lots in tool groups, it is crucial that the combination of MIVS and FF selects the lot with high cycle time (high flow factor) to balance WIP. On one hand it has ability to lower cycle time, on the other hand it manages to improve cycle time variance. In addition, the 'MIVS(3)' is able to increase throughput compared to the 'MIVS(2)'.

Similarly, Table 5 demonstrates the results from the dispatching policies in Table 3. After the introduction of hot lots, 'ODD(2)' is still able to achieve good performance of cycle time variance. Whereas, the cycle time performance of regular products is significantly affected. In this case, it is necessary to break the dominance of due date control. By introducing the SPT rule, the 'ODD(3)' policy manages to reduce the cycle time of regular products, although the cycle time variance is slightly degraded. It achieves better throughput performance as well.

Prod- ucts	Avg. Cycle Time (days)			Cycle Time Variance (days ²)			Cycle Time Upper Pct. 95% (days)			Throughput (lots)		
	MIVS(1)	MIVS(2)	MIVS(3)	MIVS(1)	MIVS(2)	MIVS(3)	MIVS(1)	MIVS(2)	MIVS(3)	MIVS(1)	MIVS(2)	MIVS(3)
B5C	31.3	22.6	22.5	2.5	0.4	0.2	34.7	24.0	23.7	227	234	236
B6HF	30.1	32.1	30.3	1.7	1.7	1.5	34.7	35.3	32.0	73	74	74
C4PH	24.7	25.6	23.2	2.9	4.5	1.8	28.3	29.9	27.3	159	161	162
C5F	29.1	29.9	30.6	2.5	2.1	1.0	33.3	34.0	33.3	191	191	191
C5P	23.9	25.5	23.9	1.6	1.2	0.7	27.3	27.7	26.0	649	645	647
C5PA	26.0	26.6	26.6	1.8	1.4	1.4	29.3	29.7	29.3	408	406	409
C6N3	29.3	29.9	28.8	1.8	1.4	0.9	32.7	34.0	31.0	148	145	147
C6N2	26.5	28.3	25.9	1.6	3.2	1.6	30.0	31.8	28.0	172	172	172
OX2	25.8	27.0	25.7	2.4	1.6	1.0	29.7	30.7	28.0	200	199	200
Sum- mary	27.4	27.5	26.3				32.7	32.0	30.8	2227	2227	2238

The wafer fabs loading is 95%;

B5C: Hot lot; MIVS(1): There are no hot lots in wafer fabs; MIVS(2): Hot lots + MIVS; MIVS(3): Hot lots + (MIVS + FF).

Table 4: Four performance measure comparison among MIVS(1), MIVS(2) and MIVS(3).

Prod- ucts	Avg. Cycle Time (days)			Cycle Time Variance (days ²)			Cycle Time Upper Pct. 95% (days)			Throughput (lots)		
	ODD(1)	ODD(2)	ODD(3)	ODD(1)	ODD(2)	ODD(3)	ODD(1)	ODD(2)	ODD(3)	ODD(1)	ODD(2)	ODD(3)
B5C	35.9	22.5	22.5	0.3	0.4	0.3	37.3	24.0	23.7	224	234	234
B6HF	34.2	36.2	31.3	0.9	0.7	2.3	36.0	37.3	36.0	74	74	74
C4PH	21.6	23.6	24.5	0.3	0.4	1.9	23.0	24.0	23.3	163	163	162
C5F	32.0	33.9	31.5	0.5	0.6	2.0	33.3	34.7	34.3	190	189	192
C5P	24.0	25.8	23.8	0.2	0.4	0.7	25.3	26.3	25.3	650	648	648
C5PA	26.7	28.7	27.7	0.5	0.4	1.1	28.0	29.7	30.3	408	406	409
C6N3	28.8	29.8	29.9	0.9	1.0	1.9	30.7	32.0	31.0	147	146	146
C6N2	25.1	27.3	26.9	0.8	0.6	1.6	27.0	28.6	28.7	173	171	172
OX2	25.3	27.3	25.5	0.2	0.6	1.6	26.3	28.7	27.3	200	200	200
Sum- mary	28.1	28.3	27.0				36.7	37.7	32.7	2229	2231	2237

The wafer fabs loading is 95%;

B5C: Hot lot; ODD(1): There are no hot lots in wafer fabs; ODD(2): Hot lots + ODD; ODD(3): Hot lots + (ODD + SPT);

Target due date flow factor for regular products: 2.5.

Table 5: Four performance measure comparison among ODD(1), ODD(2) and ODD(3).

2.2 High Ratio of Hot Lots

Product ‘C5P’ represents approximately 30% of the total release. The second case is to study the performance of the wafer fabs when ‘C5P’ is considered as hot lot. As a great many hot lots are processed everywhere, the wafer fabs are operated in an extreme manner. All resources including tools and operators are occupied by the hot lots. In addition, the hot lots enter to the wafer fabs continually. As a result, some regular lots have no chance to go through the wafer fabs, which causes serious congestion in front of tool groups. The wafer fabs are running at an extremely high WIP level.

We notice that some low volume products which share critical tool groups with the hot lots are affected the most. In the MIMAC6 wafer fabs, ‘C4PH’ and ‘OX2’ are two low volume products which have the most degraded performance. Even ‘MIVS(3)’ and ‘ODD(3)’, which were capable of improving performance for regular lots in the first case, can not sufficiently handle this issue. Since the amount of hot lots is huge, on one hand, the only way to break their monopoly is to introduce high priority for competing lots; On the other hand, the performance of hot lots is still the major concern. Thus, firstly the affected low volume products are defined as ‘urgent lots’ which are assigned priority 2, and the other regular lots have priority 3. Furthermore, the class of priority 1 is divided into 3 sub-levels by means of target cycle time comparison.

- Sub-priority 1 is assigned to the hot lots which are late for their target cycle time of the current step. The target cycle time for each step is defined as follows: $TargetCT = ReleaseTime + RPT(i) * FF$, where $RPT(i)$ denotes the raw processing time for a sequence of processing steps or operations from operation 1 to operation i (including operation i) and FF denotes the target flow factor.
- Some lots from urgent lots class obtain sub-priority 2 if they are late for their target cycle time.
- If the hot lots are on schedule compared to their target cycle times, they are assigned sub-priority 3. The hierarchical dispatching policies are represented as ‘MIVS(4)’ and ‘ODD(4)’ in Table 6.

Sub-priority 1 ensures that the late hot lots receive needed resources to catch up with their target cycle time. The purpose of sub-priority 2 is to make a trade-off between hot lots and urgent lots. In fact, whether the performance of urgent lots can be improved depends on the amount of hot lots with sub-priority 1. In other words, if the target cycle times are tight, most of the hot lots are late, then the urgent lots have to wait till the hot lots finish processing. On the contrary, if the target cycle times are loose, the urgent lots are able to compete with the hot lots for resources since some hot lots have sub-priority 3. In the following experiment, the target cycle times of hot lots are defined from tight to loose. We intend to find out if the performance of urgent lots can be improved at the

cost of good performance of the hot lots.

Hierarchical dispatching policies based on MIVS and ODD (30% ratio of hot lots)	
MIVS(4): Hot lots + Urgent lots + (MIVS + FF)	ODD(4): Hot lots + Urgent lots + (ODD + SPT)
Priority 1: Priority 1.1: Hot lots: -> Accu.CT>=TargetCT -> FIFO Priority 1.2: Urgent lots: -> Accu.CT>=TargetCT -> FIFO Priority 1.3: Hot lots: -> Accu.CT<TargetCT -> FIFO Priority 2: Urgent lots: -> MIVS -> FF Priority 3: Regular lots: -> MIVS -> FF	Priority 1: Priority 1.1: Hot lots: -> Accu.CT>=TargetCT -> FIFO Priority 1.2: Urgent lots: -> Accu.CT>=TargetCT -> FIFO Priority 1.3: Hot lots: -> Accu.CT<TargetCT -> FIFO Priority 2: Urgent lots: -> ODD+SPT Priority 3: Regular lots: -> ODD+SPT
Accu.CT is accumulated cycle time for step, TargetCT is target cycle time for step.	

Table 6: Hierarchical dispatching policies by introduction of urgent lots and target cycle time.

Simulation Results.

At first we examine the simulation results in Table 7. The ‘MIVS(3)’ policy, which achieves good performance with the 10% ratio of hot lots, shows opposite behavior with a 30% ratio of hot lots. We focus on products ‘C4PH’ and ‘OX2’ which are affected the most by hot lots. A large number of lots from ‘C4PH’ and ‘OX2’ are not able to go through the wafer fabs. Thus, ‘MIVS(3)’ produces tremendous cycle times and low throughput for them. We notice that ‘C4PH’ and ‘OX2’ are not able to be processed unless they obtain high priority to complete with the hot lots. ‘MIVS(4)’ is developed for this purpose. According to the average cycle time and cycle time upper percentile 95% of ‘C5P’ (hot lot), its actual cycle time flow factors are calculated between 1.4 to 1.6. To make sure the hot lots provide their priorities to the urgent lots, the target cycle time flow factors of ‘C5P’ are set from 1.5 (tight) to 1.9 (loose) with an increment of 0.2. The target cycle time flow factor of ‘C4PH’ and ‘OX2’ is set 2.0. We expect more and more urgent lots can finish processing as the target cycle time flow factors of ‘C5P’ change from 1.5 to 1.9.

Apparently, the throughput performance is improved. The ‘MIVS(4)FF:1.9’ manages to finish 112 lots for ‘C4PH’ and 143 lots for ‘OX2’, which is significantly improved compared to ‘MIVS(3)’. ‘C4PH’ and ‘OX2’ struggle to obtain high priority. However, due to a large amount of hot lots, as long as the hot lots meet their target cycle times, ‘C4PH’ and ‘OX2’ lose chances to be processed. As a consequence, the cycle time and variance performance can not be improved greatly. In Table 8, ‘ODD(4)’ policy shows a similar behavior as ‘MIVS(4)’.

	Avg. Cycle Time (days)				Cycle Time Variance (days^2)				Cycle Time Upper Pct. 95% (days)				Throughput (lots)			
Prod-ucts	MIVS(3)	MIVS(4)FF:1.5	MIVS(4)FF:1.7	MIVS(4)FF:1.9	MIVS(3)	MIVS(4)FF:1.5	MIVS(4)FF:1.7	MIVS(4)FF:1.9	MIVS(3)	MIVS(4)FF:1.5	MIVS(4)FF:1.7	MIVS(4)FF:1.9	MIVS(3)	MIVS(4)FF:1.5	MIVS(4)FF:1.7	MIVS(4)FF:1.9
B5C	31.8	35.4	35.8	36.4	2.7	1.1	1.5	1.2	36.0	38.7	38.7	38.7	229	225	225	224
B6HF	31.0	34.1	34.5	34.1	2.8	1.0	0.8	1.3	36.0	37.3	37.3	38.0	76	75	74	74
C4PH	83.5	78.0	77.3	72.1	5.4	6.4	5.1	6.5	178.7	154.7	138.7	126.7	74	92	105	112
C5F	29.5	31.1	31.5	32.0	2.6	1.3	1.8	1.3	33.3	34.7	34.0	34.7	192	192	190	190
C5P	16.6	16.9	16.9	17.1	0.5	0.4	0.5	0.6	18.3	18.3	18.3	18.7	668	666	666	664
C5PA	26.1	26.4	26.6	27.2	2.2	1.3	1.4	1.3	29.3	29.3	29.3	30.0	407	410	409	405
C6N3	29.8	28.9	29.2	29.8	2.7	1.5	1.3	1.3	34.0	32.3	32.0	32.7	147	148	146	146
C6N2	26.9	25.4	25.6	26.1	2.7	1.0	1.2	1.2	31.0	28.3	28.3	28.7	173	173	172	172
OX2	73.9	73.5	72.3	70.6	6.0	5.0	4.0	6.6	133.3	144.0	133.3	124.0	93	116	129	143
Summary	38.8	38.8	38.8	38.4					58.7	53.3	82.7	84.0	2059	2097	2116	2130

The wafer fabs loading is 95%;
C5P: Hot lot; C4PH and OX2: Urgent lots;
MIVS(3): There are hot lots and regular lots (no urgent lots) in wafer fabs, Hot lots + (MIVS + FF);
MIVS(4): Introduction of urgent lots, Hot lots + Urgent lots + (MIVS + FF);
FF: Target cycle time flow factor.

Table 7: Four performance measure comparison among MIVS(3), MIVS(4)FF:1.5, MIVS(4)FF:1.7 and MIVS(4)FF:1.9.

	Avg. Cycle Time (days)				Cycle Time Variance (days^2)				Cycle Time Upper Pct. 95% (days)				Throughput (lots)			
Prod- ucts	ODD(3)	ODD(4) FF:1.5	ODD(4) FF:1.7	ODD(4) FF:1.9	ODD(3)	ODD(4) FF:1.5	ODD(4) FF:1.7	ODD(4) FF:1.9	ODD(3)	ODD(4) FF:1.5	ODD(4) FF:1.7	ODD(4) FF:1.9	ODD(3)	ODD(4) FF:1.5	ODD(4) FF:1.7	ODD(4) FF:1.9
B5C	33.9	35.3	35.3	35.9	0.8	0.9	1.0	1.2	36.7	38.0	38.7	40.0	225	226	226	225
B6HF	32.0	33.7	33.6	34.1	1.8	0.9	1.5	2.0	35.3	36.7	36.7	38.0	75	75	74	75
C4PH	120.0	90.8	73.6	76.8	4.2	5.2	6.2	5.4	197.3	152.0	152.0	114.7	60	84	94	92
C5F	29.8	31.2	31.0	31.6	1.0	1.3	1.0	1.6	32.7	34.0	34.7	36.0	192	191	191	191
C5P	17.1	17.0	17.0	17.2	0.8	0.4	0.4	0.5	19.7	19.9	20.1	20.9	664	666	666	666
C5PA	25.0	26.2	26.2	26.8	0.9	1.1	1.0	1.3	27.7	28.7	29.3	30.7	409	410	409	409
C6N3	27.2	28.4	28.3	28.9	1.3	1.3	1.6	1.7	30.3	30.7	31.3	32.7	148	147	148	148
C6N2	23.7	24.9	24.8	25.4	0.9	0.9	0.9	1.0	26.3	27.0	27.7	29.0	173	173	172	173
OX2	122.7	91.8	68.7	75.7	5.1	10.9	13.9	6.0	208.0	146.7	154.7	138.7	83	116	118	122
Sum- mary	47.9	42.1	37.6	39.1					72.0	77.3	93.3	96.0	2029	2088	2098	2101

The wafer fabs loading is 95%;
C5P: Hot lot; C4PH and OX2: Urgent lots;
ODD(3): There are hot lots and regular lots (no urgent lots) in wafer fabs, Hot lots + (ODD + SPT);
ODD(4): Introduction of urgent lots, Hot lots + Urgent lots + (ODD + SPT);
FF: Target cycle time flow factor; Target due date flow factor for regular products: 2.5.

Table 8: Four performance measure comparison among ODD(3), ODD(4)FF:1.5, ODD(4)FF:1.7 and ODD(4)FF:1.9.

3 Conclusions

In this paper, we present a simulation study on handling the degraded performance of regular lots caused by hot lots. Even though the wafer fabs are operated by well-known WIP balance and due date control policies, i.e., MIVS and ODD, the hot lots still have great impact on cycle time and throughput of the regular lots. Thus, we present two cases to discuss the problem of different ratios of hot lots and intend to find out corresponding solutions.

The first case, when the hot lots comprise 10% of total release, is considered as low ratio case. In order to overcome the deficiencies of the MIVS and ODD rules, we propose to apply a flow factor rule to improve the pace of movement for the MIVS rule, and a shortest processing time rule to break the dominance of due date control for the ODD rule. The simulation results indicate that the proposed methods are able to improve the performance of regular lots.

As the percentage of hot lots are increased to 30%, the

second case is considered as high ratio case of hot lots. Because the hot lots occupy resources everywhere in the wafer fabs, the performance of the regular lots degrades severely. In particular, some low volume products which share critical resources with the hot lots are affected the most. Under this circumstance, the methods proposed in the first case are not able to tackle the problem. Therefore, we propose a hierarchical dispatching scheme in which 1) high priorities are assigned to the most affected products; 2) target cycle times are established for the hot lots and the most affected products. The simulation results tell us that when the wafer fabs are running with large amount of hot lots, a trade-off between hot lots and the most affected products is necessary. It is important to set up target cycle times so that we can determine if the improvement of the most affected products is achieved at the cost of a good performance of the hot lots.

References

- [1] Lu SCH, Ramaswamy D, Kumar PR. Efficient scheduling policies to reduce mean and variance of cycle-time in sem-

- iconductor manufacturing plants. *IEEE Transactions semiconductor manufacturing*. 1994. 374–388. doi: 10.1109/66.311341.
- [2] Li S, Tang T, Collins DW. Minimum inventory variability schedule with applications in semiconductor fabrication. *IEEE Transactions on Semiconductor Manufacturing*. 1996. 9:1-5. doi: 10.1109/66.484296.
 - [3] Keskinocak P, Tayur S. Due Date Management Policies. In: Simchi-Levi D., Wu S.D., Shen ZJ. (eds) *Handbook of Quantitative Supply Chain Analysis*. International Series in Operations Research & Management Science. Springer, Boston, MA; 2004. p 485-554.
 - [4] Robinson JK. Understanding and improving wafer fab cycle times. FabTime Inc; 2002.
 - [5] Ehteshami B, Petrakian RG, Shabe PM. Trade-offs in cycle time management: hot lots. *IEEE Transactions on Semiconductor Manufacturing*. 1992. 5:101-105. doi: 10.1109/66.136270.
 - [6] Atherton LF, Atherton RW. *Wafer fabrication: Factory performance and analysis*. Boston: Kluwer Academic Publishers. 1995.
 - [7] Fronckowiak D, Peikert A, Nishinohara K. Using discrete event simulation to analyze the impact of job priorities on cycle time in semiconductor manufacturing. *IEEE/SEMI 1996 Advanced Semiconductor Manufacturing Conference and Workshop. Theme-Innovative Approaches to Growth in the Semiconductor Industry. ASMC 96 Proceedings*; 1996; Cambridge, MA, USA. 151-155, doi: 10.1109/ASMC.1996.557987.
 - [8] Narahari Y, Khan LM. Modeling the effect of hot lots in semiconductor manufacturing systems. *IEEE Transactions of Semiconductor Manufacturing*. 1997. 10:185–188. doi: 10.1109/66.554507.
 - [9] Gupta AK, Ganesan VK, Sivakumar AI. Hot lot management: minimizing cycle time in batch processes. *2004 IEEE International Engineering Management Conference*; 2004; Singapore. 1217-1221. doi: 10.1109/IEMC.2004.1408887.
 - [10] Fowler J, Robinson J. *Measurement and improvement of manufacturing capacities (mimac): Final report*. Technical Report 95062861A-TR, SEMATECH, Austin, TX.
 - [11] Zhou ZG, Rose O. Cycle time variance minimization for WIP balance approaches in wafer fabs. In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill and M. E. Kuhl; 2013; Washington D.C. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc. 3777-3788. doi: 10.1109/WSC.2013.6721737.

Generisches Simulationsmodell für automatische Hochregallagersysteme

Walter Wincheringer, Marko Sekulic, Marec Kexel

Digitales Produktionslabor, Hochschule Koblenz, Konrad-Zuse-Str. 1, 56075 Koblenz, Germany;
dpl@hs-koblenz.de; wincheringer@hs-koblenz.de; sekulic@hs-koblenz.de; mkexel@hs-koblenz.de;

Zusammenfassung. Der Einsatz von automatischen Hochregallagern (aHRL) hat sich seit Jahrzehnten in der Praxis bewährt. Eine Auslegung bzgl. der Leistung und des Durchsatzes erfolgt meist mit Hilfe von empirischen Näherungslösungen gemäß der FEM 9.851 [1] oder der VDI 3561 [2]. Für eine realistische Leistungsbetrachtung und für Systeme, die nicht durch diese Richtlinien abgedeckt sind, bedarf es jedoch einer zeitdiskreten Simulation (discrete event simulation, DES) des Lagersystems. Der Aufwand zur Durchführung einer Simulationsstudie, inkl. dem Aufbau eines Simulationsmodells, wird mit 1,5 – 6 Monaten bemessen [3][4]. Eine Weiterentwicklung von spezifischen Referenzmodellen hin zu einem parametrierbaren generischen Simulationsmodell ist ein vielversprechender Lösungsansatz und reduziert den Aufwand erheblich. Die Entwicklung und Anwendung eines generischen Simulationsmodells für aHRL, sowie die damit verbundenen Vorteile, werden im Beitrag erläutert.

1 Ausgangssituation

Automatische Lagersysteme, wozu automatische Hochregallagersysteme (aHRL) gehören, sind in der Praxis stark verbreitet und übernehmen bei der Lagerhaltung in Industrie und Handel eine wichtige Rolle. Der Markt für automatische Lagersysteme in Europa wächst jährlich um ca. 9,5 Prozent und umfasst ca. 1,9 Mrd. Euro [5].

Daher gibt es ein breites Spektrum an aHRL-Herstellern, die unterschiedliche Lösungskonzepte im Markt anbieten. Zur Auslegung und Leistungsberechnung wird häufig auf Erfahrungswerte und Normen zurückgegriffen, wie die FEM 9.851, VDI 3561 sowie VDI 4480 [6]. Insbesondere die techn. Auslegung des Trägerfahrzeuges (z.B. Regalbediengerät, RBG), dessen Beschleunigungen und Geschwindigkeiten in der Horizontalen und Vertikalen, die Lagergeometrie (Gesamtlänge, -höhe, Lagerfachhöhe und Lagerplatzabstände), sowie die Lagerstrategie (Art und Umfang des Arbeitsspiels,

Ein-, Auslagerspiel, Lagerplatzvergabe und Zugriff auf Ladeeinheiten), bestimmen die Ein- und Auslagerungsleistung eines aHRL. Die maximal mögliche Durchsatzleistung (Summe der Ein- und Auslagerungen pro Zeiteinheit) wird hierbei häufig als zentrale Kenngröße definiert. Bei der Auslegung basiert diese auf statischen Betrachtungen und mittleren Spielzeiten für Ein- und Auslagerungen auf Basis von Referenzlagerplätzen.

Die Richtlinien haben nur eine eingeschränkte Tauglichkeit, da sie auf Annahmen basieren, wie z.B. chaotischer Einlagerstrategie, und nur für bestimmte Lagerkonfigurationen (i.d.R. einfachtiefe Lagerung) eine Gültigkeit besitzen. Sie sollten nur für eine näherungsweise Bestimmung der Leistung herangezogen werden, da die tatsächliche Leistung in der Praxis davon z.T. erheblich abweicht [7]. Die Anzahl der Doppelspiele, welche von der zeitlichen Abfolge der Ein- und Auslagerungsaufträgen sowie der Bewegungsstrategie abhängt, kann ggf. nicht mit der Annahme aus der Planungsphase übereinstimmen oder die auszulagernden Ladeeinheiten wurden an ungünstigen Positionen im aHRL eingelagert (z.B. bei einem hohen Füllungsgrad des Lagers). Die negativen Auswirkungen können zu erheblichen Produktivitätsverlusten (Rückstau der Fertigung in die Produktion, Produktionsstillstände, mangelnde Versand-, Verladeleistungen) in unterschiedlichen Bereichen führen.

Um diese Probleme zu vermeiden, ist die Durchführung einer ereignisdiskreten Simulation des aHRL notwendig [7]. Eine derartige Simulationsstudie, nach VDI 3633, ist jedoch mit einem erheblichen Aufwand für die Erstellung eines validierten Simulationsmodells verbunden und erfordert die Verfügbarkeit eines Simulationsexperten [8]. Um diesen Aufwand zu reduzieren, haben einige Hersteller in der Vergangenheit sogenannte Referenzmodelle erstellt, die eine spezifische Abbildung ihrer aHRL-Systeme ermöglichen [9]. Die in den Refer-

enzmodellen nicht abgebildeten Funktionen bzw. Bausteine müssen durch eine manuelle Programmierung aufwendig implementiert werden.

Die Simulation von aHRL ist jedoch nicht nur in der Planungsphase sinnvoll, sondern auch in der Betriebsphase, z.B. bei veränderten Anforderungen durch den Betreiber. Diese können unter anderem die Betriebszeiten (Ein-, Auslagerzeiträume), Anzahl an lagerhaltigen Artikeln (SKU, meist steigend), als auch die Art der Lagerplatzvergabe und den Zugriff auf Ladeinheiten betreffen. Daher müssen ebenfalls die unterschiedlichen Lagerstrategien bei der Simulation berücksichtigt werden.

2 Lösungsansatz

Mit der zunehmenden Digitalisierung der Geschäftsprozesse, der Produkte und der Produktion (Smart Factory) ist die Nutzung von Digitalen Zwillingen (digital twin, DT) stärker in den Fokus gerückt. Sie ermöglichen eine Überprüfung von Konzepten und Ideen in der virtuellen Welt, ohne Einfluss auf das reale System zu nehmen [10]. Übertragen auf aHRL besteht die Zielsetzung darin, einen DT als Simulationsmodell zu erzeugen, um Veränderungen der Anforderungen oder eine Neuauslegung als mögliche „Was-wäre-wenn-Szenarien“ zu simulieren und zu evaluieren.

Der DT soll jedoch nicht in einer Simulationssoftware individuell programmiert werden, sondern soll als Weiterentwicklung von Referenzmodellen, in Form eines parametrierbaren generischen Simulationsmodells, dem Anwender zur Verfügung gestellt werden. Um dies zu realisieren ist es erforderlich, die verschiedenen Lagerkonfigurationen einer aHRL-Auslegung, als auch die jeweiligen Lagerszenarien (Initialbestände, Ein- und Auslagerungsaufträge) über eine Benutzerschnittstelle in die Simulation zu integrieren. Die anwendungsspezifische Lagerkonfiguration und das –szenario dienen als Eingangsdaten für das generische Simulationsmodell (siehe Abbildung 1).

Aus den Daten und Parametern der Lagerkonfiguration, die über eine Schnittstelle eingegeben (Excel-Template) oder ggf. automatisch von einem ERP, PLM oder CAD-System übernommen wurden, erfolgt die automatische Generierung des Simulationsmodells in Sekundenschnelle. Dadurch ist die Nachbildung einer Vielzahl an unterschiedlichen aHRL-Modellen möglich (siehe Kapitel 3.1).

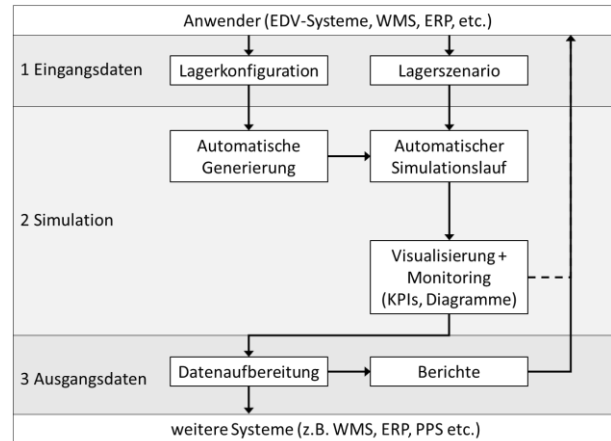


Abbildung 1: Funktionsweise des Simulationsmodells

Lauffähig wird das Simulationsmodell über eine weitere Schnittstelle. Diese lädt die Initialbestände für den Simulationsstart (anfängliche Lagerbelegung, bestehender Auftragsbestand, etc.) des Lagersystems und die Auftragsdaten. Die Auftragsdaten beschreiben das geplante Ein- und Auslagerszenario des aHRL und können von einem Warehouse Management Systems (WMS) bzw. ERP-System importiert bzw. extrapoliert oder von einem Auftragsgenerator erstellt werden [11]. Letzteres ist notwendig, wenn das zu betrachtende Lagerszenario in der Zukunft liegt und auf prognostizierte Lagerbestände und veränderte Auftragsdaten aufbaut (siehe Kapitel 3.2). Anschließend kann der Simulationslauf gestartet und die Aufträge im Lagersimulationssystem durchgespielt werden.

Über eine Darstellung der Ergebnisse, parallel zum Simulationslauf, mittels geeigneter Diagramme und Key-Performance-Indicators (KPI) können die Simulationsergebnisse dynamisch visualisiert und vom Anwender überprüft werden. Die Ergebnisse können für die Erstellung von Berichten und weiterer Verwendungen exportiert werden.

Da das generische Simulationsmodell für aHRL-Systeme bereits verifiziert und validiert ist, bedarf es keiner Überprüfung des Simulationsmodells. Zur Eingabe der Daten und zur Generierung des Simulationsmodells bedarf es ebenso keiner simulationsspezifischen Programmierarbeit. Somit ist kein Simulationsexperte erforderlich. Jedoch sollte die Anwendung des Tools durch einen Lagersystemfachmann erfolgen, da er die jeweiligen Parameter schnell und einfach zuordnen kann.

Für die Optimierung bestehender oder die Auslegung neugeplanter Lagersysteme ist eine Variation der Ein-

gangsparemeter (Simulationsexperimente) empfehlenswert, um "Was-wäre-wenn-Szenarien" zu generieren, damit ein optimales Lager nach den Vorstellungen und Restriktionen des Nutzers gefunden werden kann. Zusätzlich sollte durch Simulationsexperimente der Einfluss des Zufalls (in Bezug auf techn. Störungen des Trägerfahrzeuges oder bei einer chaotischen Einlagerung) auf die Ergebnisse bestimmt werden, über eine Variation der Zufallszahlenreihen, um Fehlentscheidungen bzgl. der Neuauslegung zu vermeiden.

3 Eingangsdaten

3.1 Lagerkonfiguration

Als Datengrundlage dient die Lagerkonfiguration, die das Lagersystem (bau-)technisch und ablauforganisatorisch definieren. Folgende Größen sind variabel über eine Schnittstelle (z.B. Excel-Template) einstellbar:

Tabelle 1: Eingabeparemeter zur Lagerkonfiguration

Lagerkonfiguration	Größen
Technik	Regalabmessungen, Fächeranzahl, -größe und -tiefe Anzahl und Position der Bereitstellplätze
Trägerfahrzeug	Typ (RBG, Shuttle) Lastaufnahmemittel, Maximale Geschwindigkeiten in x-, y-, (z)-Richtung mit/ohne Last, Beschl. und Verzögerungen in x-, y-, (z)-Richtung mit/ohne Last, Konstantzeiten (Transfer, Mastaus-schwingung, Feinpositionierung), Ausfallverhalten, etc.
Strategien	Bewegungs-, Einlagerungs-, Auslage-rungs-, Ruhepositions-, Umlagerungs-strategie, etc.

Die Konfiguration der Regale beinhaltet die Angabe der Anzahl der Regalfächer mit der Fachteilung, -ebenen und -tiefe samt den Abmessungen. Daraus werden die einzelnen Lagerplatzpositionen berechnet, sofern eine Homogenität der Regalfächer bzw. -wände gegeben ist [12]. Alternativ können die Lagerplatzpositionen eingegeben werden.

Die Anzahl der Ein- und Auslagerungsbereitstellplätze sowie deren Position können frei variiert werden. Bereitstellplätze können auch innerhalb einer Regalwand

platziert werden, wie in der folgenden Abbildung 2 rechts ersichtlich. Hierbei sind die Einlagerungsbereitstellplätze grün, die Auslagerungsbereitstellplätze rot dargestellt.

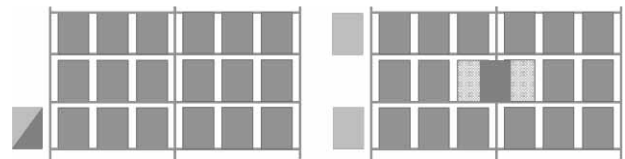


Abbildung 2: Skizze unterschiedlicher Anzahl und Anordnungen der Bereitstellplätze (Seitenansicht)

Neben Regalbediengeräten, welche zeitlich parallel horizontal und vertikal fahren, sind auch Verschiebewagen bzw. Shuttles einstellbar. Diese Fahrzeuge fahren ausschließlich horizontal. Die Vertikalebewegung der Fahrzeuge bzw. der Ladeeinheiten findet hierbei über Lifte statt.

Für die Fahrzeitberechnung der einzelnen Arbeitsspiele sind die Fahrparameter: Maximale Geschwindigkeit, Beschleunigungen und Verzögerungen relevant. Für die Genauigkeit der Fahrzeitberechnung werden die Größen hinsichtlich horizontal bzw. vertikal und beladen bzw. unbeladen unterschieden. Bei kurzen Fahrten kann die maximale Geschwindigkeit ggf. nicht erreicht werden, dabei kommt es zu einem dreieckigen Geschwindigkeitsverlauf (auch Dreiecks- bzw. Spitzfahrten genannt) [13].

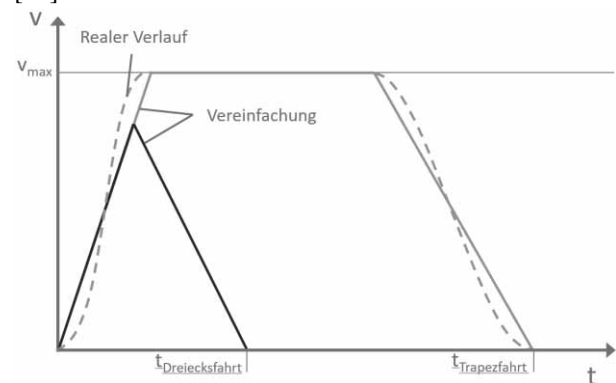


Abbildung 3: v-t-Diagramm von Fahrbewegungen

Bei Regalbediengeräten setzt sich die Diagonalebewegung aus beiden parallelen Bewegungen des Fahr- und Hubwerks zusammen. Somit beträgt die Fahrzeit für die Diagonalfahrt:

$$t_{x,y} = \max(t_x, t_y)$$

Für die Berechnung der Arbeitsspiele werden zusätzlich Verweilzeiten hinzugegerechnet. Diese bestehen aus den

Fahrtbewegungen des Lastaufnahmemittels in z-Richtung (Gabelspielzeit bzw. Transfer der Ladeeinheit) mit den Zeiten für die Feinpositionierung bzw. der Mastaus-schwingung und evtl. anfallenden Tot- bzw. Schaltzeiten.

Die realitätsgetreue Abbildung des Lagersystems in einem Simulationsmodell bedingt zusätzlich die stochas-tisch auftretenden Störungen des Trägerfahrzeugs. Hier-bei werden Mean Time Between Failures (MTBF) und Mean Time To Repair (MTTR) verwendet [14]. Diese können nach vorgegebenen Verteilungsfunktionen schwanken. Bei unbekannten Stördaten eignet sich die negative Exponentialverteilung für die Initiierung der Störzeitpunkte und die Erlang-k-Verteilung mit $k = 2$ für die Reparaturdauer [15]. Bei bekannten Ausfallverhalten können benutzerdefinierte Verteilungsfunktionen ange-geben werden.

Die Positionen, die bei den Arbeitsspielen angefahren werden, werden anhand der gewählten Lagerbewirt-schaftungsstrategie generiert. Die Bewegungsstrategie entscheidet, ob Einzelsspiele (eine Ein- oder Auslager-ung pro Fahrauftrag), Doppelspiele (eine Einlagerung und Auslagerung pro Fahrauftrag) oder Mehrfachspiele (nur bei einem Mehrfach-Lastaufnahmemittel möglich) ge-fahren werden können.

Wo konkret die Ladeeinheiten im Regal eingelagert werden, bestimmt die Einlagerungsstrategie, auch Bele-gungsstrategie genannt. Hierbei können verschiedene Strategien innerhalb folgender Kategorien eingestellt werden:

- Frei (z.B. chaotisch oder kürzester Fahrweg),
- Fest (durch Vorgabe durch den WMS)
- Zoniert (z.B. ABC-Zonierung)
- Artikel- bzw. chargenrein bei Mehrfachttiefe, etc.

Die Auslagerungsstrategien bestimmen dagegen den Ort bzw. die Ladeeinheit, welche zu einem bestimmten Zeit-punkt ausgelagert wird. Hierbei kann FIFO (bzw. schwaches FIFO bei Mehrfachttiefe [16]) und manuell (Vor-gabe durch WMS) unterschieden werden.

Neben diesen Strategien können weitere Strategien, wie z.B. Nichtbeschäftigungs- (Umlagerung von Lade-einheiten in die Nähe des Auslagerungsbereitstellplatzes oder Mischkanäle verringern) bzw. Umlagerstrategien zur Anwendung kommen.

3.2 Lagerszenario

Nachdem das Lagersystem für das Simulationsmodell technisch und ablauforganisatorisch definiert wurde, muss der Initialbestand sowie die Auftragslast für den Simulationslauf bestimmt werden. Hierunter sind unter

anderem folgende Daten zu verstehen:

- Zeitpunkt der Auftragsentstehung (Ein- und Auslage-rungsauftrag)
- Artikelbezeichnung und Attribute:
 - Klassifizierung für Zonierung
 - Chargennummer
 - Ladehilfsmittel, etc.
- Ort der Ein- bzw. Ausschleusung (Zuweisung zu den Bereitstellplätzen)
- Lagerplatz (bei der festen Lagerplatzzuordnung bzw. manuellen Auslagerung)

Die Initialdaten sind ähnlich aufgebaut. Nicht benötigt werden die Angaben zum Zeitpunkt der Auftragsentste-hung sowie der Ort der Ein- bzw. Ausschleusung, da die Ladeeinheiten nicht eingelagert werden, sondern initial zum Simulationszeitpunkt Null im Lager vorhanden sind.

Die Daten für das Lagerszenario können bei einem bestehenden Lagersystem aus der WMS-Auftragshistorie (z.B. Auftragsdaten einer gewählten Periode) erstellt werden, wenn für den simulierten Zeitraum eine ähnliche Auftragslast angenommen werden kann. Wenn das nicht der Fall ist, z.B. bei einer veränderten Einlagerung bzw. Auslagerungsstruktur oder einer Neuplanung bei fehlen-der Auftragshistorie, gibt es zwei Möglichkeiten diese Daten zu erhalten:

- Extrapolation vorhandener WMS-Daten,
- Auftragsdatengenerator.

Eine Extrapolation vorhandener WMS-Daten kommt in-frage, wenn zum einen die WMS-Daten verfügbar sind und zum anderen die Auftragsstruktur nicht grundlegend zum Simulationszeitraum verschieden sind. Wenn z.B. neue Artikel hinzukommen und/oder das Einlagerungs- und Abrufverhalten stark abweicht, empfiehlt sich ein Auftragsdatengenerator.

4 Simulation

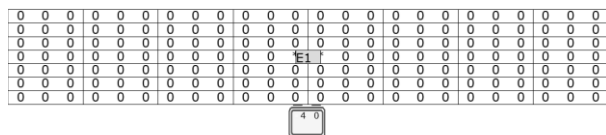
4.1 Automatische Generierung

Auf Basis der Eingangsdaten (3.1) wird im nächsten Schritt das Simulationsmodell des aHRL automatisch generiert. Es wird eine aufgeklappte Gasse des Regalsys-tems dargestellt.

In Abbildung 4 ist ein einfachtiefes aHRL generiert worden, welches fünf Ebenen hoch und 7 Regalfächer à 3 Lagerplätze lang ist. Es sind die Bereitstellplätze (grün: Einlagerung; rot: Auslagerung) sowie das Trägerfahr-zeug zu sehen. Die Lagerplätze werden durch die Zahlen repräsentiert. In diesem Zusammenhang steht ein

[illegible]

Zum Vergleich ist in Abbildung 5 ein weiteres aHRL mit anderen Abmessungen (7 Ebenen, 8 Regalfächer) und unterschiedlichen Positionen der Bereitstellplätze abgebildet. Ersichtlich wird, dass Bereitstellplätze in der Regalwandmitte entsprechende Lagerplätze blockieren (vgl. Abbildung 2).



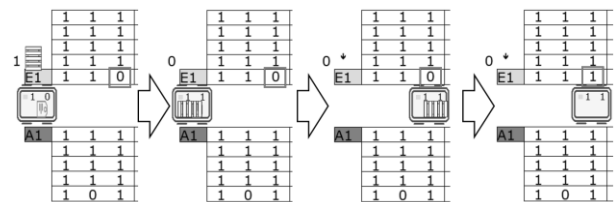
4.2 Automatischer Simulationslauf

[illegible]

Der Simulationszeitraum wird dabei anhand der Daten des Lagerszenarios bestimmt. Wenn entsprechende Auftragsdaten für 3 Monate vorliegen, so beträgt auch

Ein Beispiel für ein Einlagerungsspiel ist in der Abbildung 7 ersichtlich und läuft wie folgt ab:

1. Ankunft einer Ladeeinheit am Bereitstellplatz und Auftragsannahme durch das Trägerfahrzeug.
2. Aufnahme der Ladeeinheit.
3. Fahrt zum Lagerplatz, gemäß der Einlagerungsstrategie.
4. Abgabe der Ladeeinheit an den Lagerplatz.



Die Auftragsstruktur ist mitentscheidend für die tatsächliche Leistung eines Lagersystems, da anhand der Aufträge im Simulationsmodell entschieden wird, wie die Fahrten des Trägerfahrzeugs ausgeführt werden, z.B. ob Einzel-, Doppel- oder Mehrfachspiele gefahren werden. Diese Einflussgröße wird, wie in Kapitel 1 beschrieben, nicht in den mathematisch-analytischen Berechnungsmethoden z.B. der FEM 9.851 oder VDI 3561 betrachtet. Es wird lediglich die mittlere Spielzeit für ein Einzelspiel bzw. für ein Doppelspiel bestimmt. Wieviele Doppelspiele tatsächlich pro Zeiteinheit gefahren werden können, ist nur durch die Simulation zu bestimmen.

Parallel zum Simulationlauf werden die unterschiedlichsten Diagramme dynamisch dargestellt und KPIs aufgezeichnet. So werden unter anderem die Auslastung des Trägerfahrzeugs in variablen Zeitintervallen aufgezeichnet (siehe Abbildung 8) sowie die Lagerauslastung über die Zeit (siehe Abbildung 9).

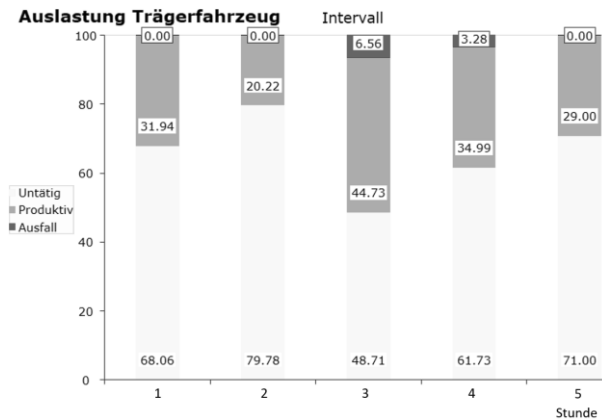


Abbildung 8: Beispielhafte Auslastung des Trägerfahrzeugs im Stundenintervall

Bei einem zonierten Lager lassen sich ebenfalls die Auslastung der unterschiedlichen Zonen im Lagerauslastungsdiagramm darstellen. Bei einem mehrfachtiefen Lager lassen sich die voll-, teilbelegten und leeren Kanäle anzeigen.



Abbildung 9: Beispielhafte Auslastung des Lagers über die Zeit [min]

Die Diagramme ermöglichen es Lastspitzen über die komplette Simulationsdauer zu identifizieren. Desweiteren werden weitere KPIs wie:

- Durchsatz,
- mittlere Spielzeit,
- Anzahl der Einzel-, Doppel- und Mehrfachspiele,
- Grenzdurchsatz, etc.

über den Simulationslauf hinweg berechnet und dargestellt. Wenn es die Lagerkonfiguration (z.B. Lagergeometrie) zulässt, werden parallel dazu die Ergebnisse der Simulation mit den Ergebnissen der FEM-Berechnung gegenübergestellt, um so Abweichungen der statischen Berechnung auf einen Blick zu erkennen.

Die Vielzahl und Darstellungsweise der KPIs ermöglichen es dem Lagerplaner und -betreiber Engpässe und Optimierungspotenziale auf einen Blick zu erkennen.

Dabei können die Simulationsergebnisse durch die entsprechenden Schnittstellen exportiert und nach der Datenaufbereitung für Reports, Big-Data-Analysen oder für weitere Systeme, wie z.B. WMS, ERP, PPS, als Eingangsdaten weiterverwendet werden.

Lagerkapazität		Anzahl der Spiele	
Anzahl der Lagerplätze	210	Anzahl der Spiele	86
Belegte Lagerplätze	209	Anzahl der Einlagerungsspiele	30
Blockierte Plätze	0	Anzahl der Auslagerungsspiele	20
Lagerauslastung	0.995	Anzahl der Doppelspiele	36.5
Anzahl der Lagerkanäle	210	Anzahl der Zyklen	124
Leere Kanäle	1	Anteil DS / ES	
Volle Kanäle	209	Anteil an Doppelspielen	0.414
Teilbelegte Kanäle	0	Anteil an Einzelspielen	0.586
Blockierte Kanäle	0	Anteil an Einlagerungsspielen	0.356
		Anteil an Auslagerungsspielen	0.230
Spielzeiten [sec]		Durchsatz	
Summe der Spielzeiten	5064	Gesamt-Durchsatz	123
Mittlere Spielzeit	58.88	Durchsatz pro Std.	29.25
Summe der Doppelspielzeiten	2807	Anzahl an Einlagerungen	67
Mittlere Doppelspielzeit	77.15	Anzahl an Auslagerungen	56
Summe der Einlagerungsspielzeit	1347	Einlagerungen pro Std.	15.93
Summe der Auslagerungsspielzeit	910	Auslagerungen pro Std.	13.62
Mittlere Einlagerungsspielzeit	44.90	Grenzdurchsatz	
Mittlere Auslagerungsspielzeit	45.51	Max. Doppelspiele pro Std.	45.53
		Max. Einlagerungsspiele pro Std.	80.17
		Max. Auslagerungsspiele pro Std.	79.11

Abbildung 10: Ausschnitt aus dem KPI-Dashboard

5 Erkenntnisse

Das generische Simulationsmodell für automatische Hochregallager wurde in einer zweijährigen Zusammenarbeit mit einem Industriepartner entwickelt und auf Basis des Softwaretools WITNESS programmiert. Als Schnittstelle wird ein Excel-Template verwendet. Eine ausführliche Validierung mit Real-Daten wurde ausgeführt und das Tool ist seit 2019 im praktischen Einsatz. Es werden sowohl Auslegungen von aHRL-Neuplanungen, als auch Optimierungen von bestehenden Systemen, mit Hilfe des Simulationstools ausgeführt.

Durch die Verwendung des generischen Simulationsmodells entfallen Programmierarbeiten. Somit sind auch Nicht-Simulationsexperten, wie beispielsweise Mitarbeiter im technischen Vertrieb, in der Lage das Simulationstool zu nutzen. Dadurch kann die Simulation bereits in der frühen Angebotsphase eingesetzt werden. Dies führt dazu, dass Sicherheitszuschläge und Leistungsreserven reduziert werden können. Zusätzlich lässt sich der aHRL-Abnahmeprozess vereinfachen, da bereits eine Absicherung der Leistungsdarstellung durch die Simulation erfolgt ist. Weiterhin lassen sich die optimale Lagerkonfiguration und -strategie durch den Einsatz des generischen Simulationsmodells frühzeitig bestimmen und absichern. Die Vorteile werden nachfolgend zusammengefasst:

- Erhebliche Verringerung des Aufwandes der Simulation (keine Formalisierung und Implementierung bzw. Programmierung, gemäß der VDI 3633, notwendig).
- Durch die Schnittstelle können die benötigten Daten zielgerichtet und schnell erfasst werden.
- Kein Simulationsexperte oder Programmierer erforderlich.
- Nutzung der Simulation in einer frühen Angebotsphase, z.B. durch den technischen Vertrieb.
- Sicherheitszuschläge und Leistungsreserven können reduziert werden.
- Abgesicherte Leistungsdarstellung vereinfacht den aHRL-Abnahmeprozess.
- Frühzeitige Bestimmung und Absicherung einer optimalen Lagerkonfiguration samt –strategie, gemäß dem gewählten Szenario.
- Simulationsexperimente können direkt ausgeführt und –ergebnisse präsentiert werden.

Das generische Simulationsmodell für aHRL wird aktuell für weitere Lagersystemkonfigurationen (u.a. zusätzliche Lagerbewirtschaftungsstrategien) weiterentwickelt.

References

- [1] Fédération Européenne de la Manutention, *FEM 9.851: Leistungsnachweis für Regalbediengeräte – Spielzeiten*. 2003.
- [2] Verein Deutscher Ingenieure, *VDI-Richtlinie 3561: Testspiele zum Leistungsvergleich und zur Abnahme von Regalförderzeugen*. Düsseldorf: VDI-Verlag; 1973.
- [3] Wortmann D. Webinar zum Tag der Logistik SimPlan: *Wie die Simulation bei der Planung und Optimierung von Distributionszentren und Lagersystemen hilft*. 16.04.2020
- [4] Vialog Logistik Beratung. *Wann lohnt sich eine Lager-Simulation?* In: <https://vialog-logistik.com/2015/10/wann-lohnt-sich-eine-lager-simulation/>. Zugriff am: 14.09.2020. 2015.
- [5] MarketsandMarkets. *Automated Storage and Retrieval System (ASRS) Market with COVID-19 Impact Analysis by Type (Unit Load, Mini Load, VLM, Carousel, Mid Load), Function (Storage, Order Picking, Assembly, Distribution, Kitting), Industry, and Region- Global Forecast to 2025*. 2020.
- [6] Verein Deutscher Ingenieure, *VDI 4480: Durchsatz von automatischen Lagern mit gassengebundenen Regalbediensystemen, Blatt 1*. Berlin: Beuth Verlag; 1998.
- [7] Ten Hompel M, Schmidt T, Dregge J. *Materialflusssysteme: Förder- und Lagertechnik*. 4. Auflage. Berlin: Springer; 2018. S.204.
- [8] Verein Deutscher Ingenieure, *VDI-3633: Simulation von Logistik-, Materialfluss- und Produktionssystemen – Grundlagen, Blatt 1*. Berlin: Beuth Verlag; 2014.
- [9] Klinger A, Wenzel S. Referenzmodelle – Begriffsbestimmung und Klassifikation. In: Wenzel, S. *Referenzmodelle für die Simulation in Produktion und Logistik*, Erlangen: SCS; 2000. S. 13 – 29.
- [10] Fraunhofer-Institut für Produktionsanlagen und Konstruktionstechnik (IPK). *Smarte Fabrik 4.0 – Digitaler Zwilling*. Berlin; 2018.
- [11] Rabe M, Spieckermann S, Wenzel S. *Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodell und Techniken*. Berlin, Heidelberg: Springer; 2008. S. 89 ff.
- [12] Atz T. *Eine algorithmenbasierte Methode zur ganzheitlichen Systemplanung automatischer Hochregallager*. Dissertation, Technische Universität München (TUM); 2016. S.59 f.
- [13] Lippolt CR. *Spielzeiten in Hochregallagern mit doppelt tiefer Lagerung*. Dissertation Karlsruher Institut für Technologie (KIT); 2003. S. 58.
- [14] Verein Deutscher Ingenieure, *VDI 3973: Durchsatz und Spielzeitberechnungen in Stückgut-Fördersystemen*. Berlin: Beuth Verlag; 2018.
- [15] Gutenschwager K, Rabe M, Spieckermann S, Wenzel S. *Simulation Produktion und Logistik: Grundlagen und Anwendungen*. Berlin: Springer; 2017. S. 107 f.
- [16] Gudehus T. *Logistik 2: Netzwerke, Systeme und Lieferketten*. 4. Auflage. Berlin, Heidelberg: Springer; 2012. S. 648.

Einsatzmöglichkeiten der Rückwärtssimulation zur Produktionsplanung in der Halbleiterfertigung

Christoph Laroque^{1*}, Christoph Löffler¹, Wolfgang Scholl², Germar Schneider²

¹Institut für Management und Information, Westsächsische Hochschule Zwickau, Kornmarkt 1, 08012 Zwickau, Deutschland; * christoph.laroque@fh-zwickau.de

²Infineon Technologies Dresden GmbH & Co. KG, Königsbrücker Straße 180, 01099 Dresden, Germany

Abstract Manufacturing is in general characterized by a growing number of customer-specific products that have to be manufactured and delivered in given lead times, according to concrete delivery dates. Thus, highly relevant questions like "When to start a production order at latest, in order to stay within my lead time?" are answered by more or less primitive, backward-oriented planning approaches and without taking into consideration uncertainty or alternatives. It gets more complex, if different products are to be produced and the more complex the underlying manufacturing system is (e.g. semiconductor with re-entry cycles). These questions could be answered more specifically, more detailed and more robust, if discrete, event-based simulation (DES) would be applied in a backward-oriented manner. Research results show, that the backward-oriented simulation approach can be in principle applied successfully for the scheduling of customer-specific orders.

1 Motivation

Mit der fortschreitenden digitalen Transformation und der stetigen Entwicklung hin zu der Vision von Industrie 4.0 und dem Konzept einer „Smarten Fabrik“ ändern sich auch die Anforderungen an die Informationssysteme für die Arbeitsvorbereitung und operative Produktionsplanung. War hier in der Vergangenheit oft die wirtschaftlich optimale Auslastung der Produktionsketten zur Senkung der Kosten wesentliches Ziel aller Optimierung, so hat sich dies in den letzten Jahren zunehmend hin zu einer stärker kundenorientierten Fertigung verschoben, bei dem das Hauptaugenmerk auf der Einhaltung zugesagter Liefertermine durch das jeweilige Produktionsunternehmen ist. Diese müssen natürlich

dennoch wirtschaftlich produziert und in möglichst kurzen Durchlaufzeiten ausgeliefert werden. Insbesondere die Einführung neuer Produkte ist bei stetig verkürzten Produktlebenszyklen eine sehr grosse Herausforderung.

Die Produktionsprozesse der im hier beschriebenen Vorhaben adressierten Halbleiterfertigung gelten auch im Vergleich zu anderen Branchen als wesentlich komplexer, weil die Technologien im Mikro- und Nanometerbereich sehr sensitiv im Bezug auf die Prozessstabilität sind und sehr viele Produktionsschritte für die einzelnen Produktionslose benötigt werden (bis zu 1000 Produktionsschritte und teilweise bereits mehr). Des Weiteren müssen viele Produkte innerhalb des Produktmixes mehrfach mit hohem Automationsgrad und unter Reinraumbedingungen über spezielle Anlagen und Transportrouten prozessiert werden (Re-Entry-Cycles). Die gesamte Produktion findet dabei bereits teilweise über verschiedene Standorte hinweg statt, sodass an den vielen unterschiedlichen Prozessschritten Ausschuss von angefertigten Produkten in relevanter Größenordnung entstehen kann, der kurzfristig durch zusätzliche Einschleusungen neuer Produktionslose ausgeglichen werden muss. In der Produktionsfeinplanung ergeben sich aus dieser Kombination vielfältige Fragestellungen, die mit den existierenden Werkzeugen zur Generierung von Ablaufplänen derzeit nicht oder nicht hinreichend gelöst werden können.

Im Rahmen des EU-ECSEL-Projektes iDEV40¹ wird die Erarbeitung von Einsatzmöglichkeiten der innovativen Methode der Rückwärtssimulation zur Planung und Steuerung von Entwicklungs- und Fertigungslosen auf Basis der diskret, ereignisgesteuerten Materialflusssimulation erarbeitet. Die prinzipielle Machbarkeit der Methode auf Fabrikebene (vgl. [1], [2]) muss hinsichtlich

¹ Vgl. Darstellung des Gesamtprojektes unter www.idev40.eu

der Spezifika der Branche übertragen und angepasst werden. Der Beitrag beschreibt Teile der bisher erzielten Ergebnisse und damit Anwendungsmöglichkeiten der Methode sowie die nächsten Schritte zur Realisierung einer praxisnahen Anwendung zur operativen Entscheidungsunterstützung. Nach einer kurzen Darstellung des prinzipiellen Lösungsansatzes wird beispielhaft eines der erzeugten Testmodelle detaillierter beschrieben und die Ergebnisse der entsprechenden Simulationsergebnisse dargestellt. Eine Zusammenfassung beschreibt abschließend die nächsten Schritte im Projekt.

2 Lösungsansatz Rückwärtssimulation

Heutige Ziele der Produktionsplanung und -steuerung (PPS) sind wettbewerbsfähige Kosten, kurze Durchlaufzeiten, hohe Termintreue und die Erfüllung der Qualitätsanforderungen bei möglichst niedrigen Beständen. Flexibilität als Optimierungsfaktor erlebt ebenfalls eine steigende Bedeutung, wobei die jeweiligen Gewichtungen von Einsatzzweck zu Einsatzzweck variieren. Die Maximierung der Kapazitätsauslastung hat ihre große Bedeutung im heutigen Käufermarkt eingebüßt [6]. Zur Lösung dieser Planungsaufgaben kommen herkömmlich Methoden der gemischt-ganzzahlige Optimierung, unterschiedliche Heuristiken wie bspw. Tabu Search oder Simulated Annealing oder einfache Voraus- oder Rückwärtsplanung (mit oder ohne Kapazitätsbeschränkungen) zur Erzeugung erster gültiger Lösungen zum Einsatz. Abhängig vom konkreten Planungsziel wird bei diesen Verfahren (abgesehen von der Optimierung) der Planungszeitraum zeitlich vorwärts oder rückwärts betrachtet. Für die Maximierung des Durchsatzes wird der Planungszeitraum vorwärts betrachtet und es wird versucht, beginnend bei einem Startzeitpunkt, die gegebenen Aufträge möglichst schnell fertigzustellen. Bei der Terminplanung sollen bestimmte Liefertermine eingehalten werden, deshalb wird der Planungszeitraum hier häufig rückwärts geplant. Jeder Auftrag wird beginnend bei seinem Liefertermin auf der letzten Ressource eingeplant und dann wird bestimmt, wann die vorgelagerten Ressourcen von diesem Auftrag belegt werden (vgl. bspw. [3]). Ziel dieser Untersuchung ist es, für jeden Auftrag den Zeitpunkt zu erfassen, an dem ein Auftrag spätestens eingesteuert werden muss, um noch zum vereinbarten Liefertermin fertiggestellt zu werden.

Sehr viel komplexere Modelle können mit der diskreten ereignisorientierten Simulation (DES) bearbeitet werden. DES wird meist auf strategischer oder taktischer Ebene eingesetzt, beispielsweise im Rahmen der Fabrik- oder Logistikplanung. Im operativen Produktionsbereich ist der Einsatz von DES zur Absicherung von konkreten Produktionsplänen oder Anlagen(um)planungen bekannt. Voraussetzung ist hier ein konkretes Produktionsprogramm, das als Eingabe für ein spezifisches Simulationsmodell dient. Ergebnis solcher Simulationen ist eine Aussage, ob das Produktionsprogramm realisierbar ist und ob alle vorher festgelegten Liefertermine eingehalten werden. Ist dies nicht der Fall, muss durch systematische Suche und Auswahl von Planvariationen (ggf. in anderen Werkzeugen oder mit anderen Algorithmen) ein durchführbarer Produktionsplan gefunden werden, dessen Validierung durch die Simulation positiv ausfällt. Modelle zur DES können aber eine sehr genaue Abbildung der Realität darstellen, sind gut zu parametrisieren und berücksichtigen die Variabilität der Realität, indem zufällige Einflüsse der Realität über stochastische Bestandteile der Modelle integriert werden können[7]. Zusätzlich können verschachtelte Ressourcenbeziehungen, Wartungsvorgänge und spezifische Ablauf-, Prioritäts-, Batch- oder Rüstregeln modelliert werden. Aus diesen Gründen sind diese Systeme eher für die Planung in der Halbleiterfertigung geeignet und werden im Rahmen dieses Vorhabens detaillierter betrachtet.

Im Allgemeinen wird die Simulation aber sowohl im Einzeleinsatz, als auch in der Kombination mit Heuristiken im Rahmen der simulationsgestützten Optimierung nur zur Untersuchung von zeitlich vorwärts gerichteten Planungsproblemen benutzt. Der Einsatz in zeitlich rückwärtsgerichteten Planungsproblemen (im Folgenden: Rückwärtssimulation) ist selten und hat sich bisher nicht in der wissenschaftlichen und industriellen Anwendung etablieren können, obwohl die Vorteile der Simulation auch in der Anwendung der rückwärtsorientierten Planung zum Tragen kommen [4][9]. Erste Anwendungsstudien, bei denen Aufträge mit Hilfe einer Rückwärtssimulation zeitlich rückwärtsgerichtet eingeplant werden, sind bereits seit mehr als 15 Jahren verfügbar. Watson et al. [11], [12], Ying und Clark [13] und Jain und Chan [5] nutzen solche Verfahren, um die Freigabezeitpunkte von Aufträgen oder Losen auch unter stochastischen Charakteristika der Modelle berechnen zu können. In Rückwärtsmodellen treten Aufträge an den Stellen in das System ein, an denen sie dieses im Vorwärtsmodell verlassen

(Umkehrung der Quelle-Senke-Beziehung). Sie verlasen das Rückwärtsmodell an den Eintrittsstellen des Vorwärtsmodells („From product to raw material“). Die mit der Rückwärtssimulation gesuchten Freigabezeiten sind dann die Zeitpunkte, die sich ergeben, wenn die Durchlaufzeiten mit den Eintrittszeitpunkten in Beziehung gesetzt werden. Dabei ist die Rückwärtssimulation keine reine „Umkehrfunktion“ der Vorwärtssimulation. Vorwärts- und Rückwärtssimulation müssen zur selben berechneten Simulationszeit nicht denselben Zustand aufweisen [13]. Speziell die in den Modellen verwendeten Steuerungsregeln lassen sich nicht 1:1 in das entsprechende Rückwärts-Pendant übertragen. Analog zur Rückwärtsterminierung in PPS-Systemen wird auch die Rückwärtssimulation in Kombination mit vorwärts gerichteten Simulationsläufen durchgeführt, um die resultierenden Pläne nochmals abzusichern.

In der Domäne der Halbleiterfertigung konnten in den vergangenen Jahren erste Beispiele durch die Autoren erfolgreich realisiert und publiziert werden (vgl. [14]). Die dabei untersuchten Modelle hatten aber einen relativ einfachen Charakter und orientierten sich im wesentlichen an einem linearen Produktionsablauf. Spezielle Eigenschaften der Halbleiterfertigung (s.o.) wurden nicht mit abgebildet.

3 Bisherige Ergebnisse

Das Modell *Lead Acid Battery Production* der Anylogic Beispiellibliothek² bildet eine kleine Produktionslinie von Bleiakkumulatoren ab und stellte die Grundlage für die ersten Untersuchungen der Rückwärtsmodellierung dar. Innerhalb dieses Modells kommen neben Förderbändern und Industriekränen auch fahrerlose Transportsysteme zum Einsatz, welche sich bei dem Transport von Elektrodenchargen und Batterien sowohl entlang von Führungslinien als auch im freien Raum bewegen können. Neben der Modellierung der Prozesskette umfasst das Modell eine zweidimensionale als auch eine dreidimensionale Darstellung. Das Modell wurde im Rahmen des Vorhabens mehrfach hinsichtlich verschiedener Charakteristika der Halbleiterfertigung erweitert, so dass neben parallelen Bearbeitungsschritten, den oben beschriebenen Re-Entry-Cycles auch Batch-Prozesse in dem Modell abgebildet werden. Das Modell wurde zur

Erhöhung der Komplexität noch gespiegelt und besitzt in der letzten Ausbaustufe ein gemeinsames Auslieferungslager in der räumlichen Mitte der Fertigung. Das abgebildete Auftragsprogramm im Modell wurde sukzessive um verschiedenen Prioritätsstufen der Aufträge und verschiedene Losgrößen für die Aufträge ergänzt. Abbildung 1 zeigt die grafische Darstellung der finalen Ausbaustufe (oben) sowie die resultierende Materialflussstruktur des Modells (unten).

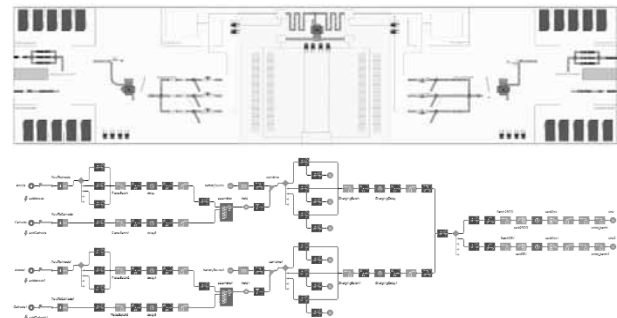


Abbildung 1: Modellstruktur des erweiterten Beispiels zur Batterieherstellung

Sowohl die Vorwärtsmodellierung als auch die Rückwärtsmodellierung wurden in experimentellen Versuchen hinsichtlich der Auswirkungen sich verändernder Parameter betrachtet (Sensitivitätsanalysen). Für das Vorwärtsmodell konnte somit in einem ersten Schritt eine vernünftige Einstellung der Parameter erreicht werden: Mit 150 Elektroden pro Los in der Produktion konnten 668 Stunden Simulationszeit insgesamt 388 Batterien für die im Modell hinterlegten Aufträge produziert werden. Bei einer Losgröße von 50 Elektroden wäre dieselbe Anzahl Batterien erst nach 672 Stunden erfüllt worden. Für die Rückwärtsmodellierung konnten weitestgehend die gleichen Parameter angesetzt werden, wobei entsprechende Parameter zur Rohstoffverfügbarkeit („filling quantity“ als Füllmenge des flüssigen Metalls, welches in der Vorwärtsmodellierung als Quelle dient) durch korrespondierende Parameter (hier „plates batch“ als Batchgröße des entsprechenden Prozesses) ersetzt wurden. Auch die entsprechenden Steuerungsregeln des Vorwärtsmodells mussten im Rahmen der Modellierung des Rückwärtssimulationsmodells entsprechende umgekehrt werden. Abbildung 2 zeigt exemplarisch die Auswertung eines Simulationslaufs, wobei die einzelnen Ausliefe-

² Das ursprüngliche Modell ist frei unter <https://cloud.anylogic.com/> verfügbar

rungstermine für Produktionslose (PROD) und Entwicklungslose (DEV) separat ausgewiesen werden.

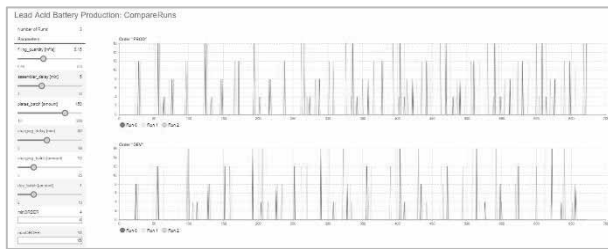


Abbildung 2: Grafische Auswertung des Vorwärtssimulationsmodells für ein spezifisches Parameter-Setting

Nach der Modellierung der beiden Einzelmodelle für Vorwärts- und Rückwärtssimulation (Abbildung 3 zeigt den resultierenden Materialfluss für das entsprechende Rückwärtsmodell) sollten die beiden Modelle in einem weiteren Schritt gegeneinander validiert werden.

Im Fall des hier dargestellten Beispiels wurde dazu auf Basis der Ergebnisse der Sensitivitätsanalyse des Vorwärtsmodells ein Auftragsprogramm selektiert, dass eine relativ hohe Auslastung der Fabrik zur Folge hat. Für dieses Auftragsprogramm wurden zunächst mittels einer ersten Vorwärtssimulation (Simulation 0) Fertigstellungstermine erzeugt. Diese wurden als Eingabeparameter für dasselbe Auftragsprogramm in der korrespondierenden Rückwärtssimulation als Liefertermine übernommen. Als Ergebnis der Simulationsläufe der Rückwärtssimulation (R-Simulation 1) entstehen dann für dieses Auftragsprogramm Einschleustermine als Ergebnis des Planungslaufes. Für die meisten Aufträge unterscheiden sich diese Termine allerdings von ihrer ursprünglichen Einsteuerung im Vorwärtslauf 0.

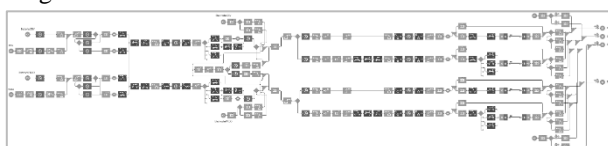


Abbildung 3: Materialflussstruktur des Rückwärtssimulationsmodells in der finalen Ausbaustufe

Als eine erste Validierung des Rückwärtsmodells wurden daher die Einschleustermine der Rückwärtssimulation erneut in das Vorwärtsmodell übernommen, simuliert und die erzielten Fertigstellungstermine (Simulation 2) mit den ursprünglichen Lieferterminen (aus Simulation 0) verglichen. Die erzielten Ergebnisse zeigen, dass die über die Rückwärtssimulation erzeugten Einschleustermine in >95% aller Aufträge auch die pünktliche Einhaltung der Liefertermine zur Folge haben. Die beiden Modelle verhalten sich somit konsistent zueinander und

konnten im Anschluss um stochastische Einflüsse in beiden Modellen ergänzt werden. Nachfolgend wurden die entsprechenden Experimente mit den stochastischen Modellen wiederholt, wobei hier natürlich auf eine entsprechende stochastische Breite der einzelnen Simulationsexperimente Rücksicht genommen wurde. Die jeweils erzielten Ergebnisse der Simulationsläufe (vgl. bspw. Abbildung 4) wurden zusammengetragen und ausgewertet. Insgesamt ist dazu festzuhalten, dass die Konstanz zwischen den beiden jeweiligen Simulationsmodellen erhalten blieb, wenn die Ergebnisse auch einen etwas schlechteren Erfüllungsgrad bei der Validierung der Einschleustermine aus der Rückwärtssimulation durch die Simulation 2 zur Folge hatten. Eine Liefertermintreue > 85% konnte jedoch in allen Simulationsexperimenten nachgewiesen werden.

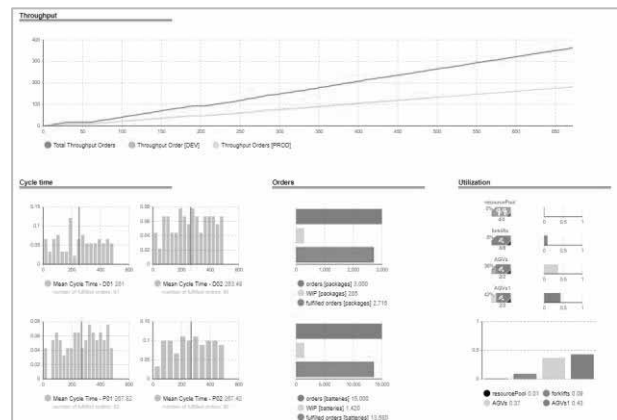


Abbildung 4: Ergebnisübersicht der Rückwärtssimulation mit stochastischen Einflüssen

Logisch gut nachzuvollziehen ist überdies, dass die Integration von Maschinenausfällen und anderen stochastischen Einflüssen wie Bearbeitungszeiten einzelner Maschinen einen merklichen Einfluss auf Gesamtdurchsatz (*Total Throughput*) beider Modelle aufgezeigt hat. Die erzielten Ergebnisse haben aber in einem ersten Schritt im Rahmen des Vorhabens die prinzipielle Anwendbarkeit der Methode auch mit komplexeren Materialflussstrukturen nachgewiesen und sind Basis für weitere Entwicklungen im Projekt, die im nächsten Abschnitt kurz beschrieben werden.

4 Zusammenfassung und Ausblick

Die mit dem erweiterten Beispiel erzeugten Ergebnisse zeigen, dass der methodische Ansatz zur Generierung eines Produktionsplans durch Rückwärtssimulation

prinzipiell auch unter den Spezifika der Halbleiterfertigung und unter Berücksichtigung stochastischer Einflüsse funktioniert und vielversprechende Ergebnisse liefern kann. Die hier vorgenommene Validierung diene nur in einem ersten Schritt dazu, eine prinzipielle Konsistenz der beiden Modelle nachzuweisen.

Im eigentlichen Vorhaben sind weitere Schritte zur Evaluation der Methode und ihrer Einsatzgrenzen im beschriebenen Einsatzgebiet geplant. In einem ersten Schritt wird für ein weiteres, komplexeres Simulationsmodell des Projektpartners das korrespondierende Rückwärtssimulationsmodell erzeugt und die Experimente wiederholt. Im besten Fall zeigt sich hier eine Bestätigung der Ergebnisse. Das Modell entspricht einem realen Produktionsprozess und schlägt damit die Brücke von einem Testmodell mit entsprechenden Branchencharakteristika zu einem möglichen realen Einsatz der Methode für „real-world-problems“.

Hierfür ist in einem weiteren Schritt aber zunächst die Lösungsgüte der aus der Rückwärtssimulation ermittelten Einschleusplanung nicht nur hinsichtlich der prinzipiellen Erfüllbarkeit des Auftragsprogramms zu überprüfen, sondern auch mit konkurrierenden Planungsverfahren der kapazitätsbeschränkten Rückwärtsterminierung etc. Die Autoren erwarten, dass der Einsatz der Simulationmethode einfachen Ansätzen und Heuristiken zur Planung überlegen sein sollte, weil viele der Kapazitätsbeschränkungen in dem Materialflussmodell schon berücksichtigt werden. Mathematisch optimierende Verfahren können sollten sich hinsichtlich der Lösungsgüte durchsetzen können, erzeugen ihrerseits aber einen erheblichen Aufwand zur Modellierung und Berechnung und können die stochastischen Einflüsse des realen Modells vermutlich auch nur unzureichend abbilden. Hier werden weitere Untersuchungen in der Zukunft hoffentlich genauere Vor- und Nachteile der Rückwärtssimulation benennen können.

Aknowledgements

This work has been partially funded by the European project iDEV40. The iDev40 project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 783163. The JU receives support from the European Union's Horizon 2020 research and innovation programme. It is co-funded by the consortium members, grants from Austria, Germany, Belgium, Italy, Spain, and Romania.

References

- [1] Arakawa, M.; Fuyuki, M.; Inoue, I.: A Simulation-based Production Scheduling Method for Minimizing the Due-date-deviation, *International Transactions in Operational Research*, Vol. 9 Issue 2, S. 153-167, 2002
- [2] Graupner, T. D.; Bornhäuser, M.; Sihn, W.: Backward simulation in food industry for facility planning and daily scheduling, 16th European Simulation Symposium [ESS 2004], SCS Press, 2004
- [3] Hopp, W. J.; Spearman, M. L.: *Factory Physics*, McGraw-Hill, New York, 2001
- [4] Huang, C.; Wang, H.: Backward Simulation with Multiple Objectives Control, *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 2009
- [5] Jain, S.; Chan, S.: Experiences with Backward Simulation Based Approach for Lot Release Planning, *Winter Simulation Conference*, S. 773-780, 1997
- [6] Kurbel, K.: *Produktionsplanung und -steuerung im Enterprise Resource Planning und Supply Chain Management*, Oldenbourg Verlag, 2005
- [7] Law, A.; Kelton, D.: *Simulation Modeling and Analysis*, McGraw Hill, 2000
- [8] Rossi, H.: Ein heuristisches Dekompositionsverfahren für mehrstufige Losgrößenprobleme, *Dissertation*, Freie Univ. Berlin, 2003
- [9] Schumacher, R.; Wenzel, S.: Der Modellbildungsprozess in der Simulation, In: Wenzel, S. (Hrsg.): *Referenzmodelle für die Simulation in Produktion und Logistik*, S. 5-11, SCS-Europe BVBA, Gent, Belgien, 2000
- [10] Tempelmeier, H.; Günther, H.-O.: *Produktion und Logistik*, 7. Auflage, Springer, Berlin, 2007
- [11] Watson, E. F.; Medeiros, D. J.; Sadowski, R. P.: Generating Component Release Plans with Backward Simulation, *Proceedings of the 1993 Winter Simulation Conference*, S. 930-938, 1993
- [12] Watson, E. F.; Medeiros, D. J.; Sadowski, R. P.: A simulation-based backward planning approach for order-release, *Proceedings of the 29th Conference on Winter Simulation*, Atlanta, Georgia, ACM Press, New York, NY, S. 765-77, 1997
- [13] Ying, C. C.; Clark, G. M.: Order release planning in a job shop using a bidirectional simulation algorithm, *Proceedings of the 26th Conference on Winter Simulation*, Orlando, Florida, Society for Computer Simulation International, S. 1008-1012, 1994
- [14] Scholl, Wolfgang; Laroque, Christoph; Weigert, Gerald: Evaluations on Scheduling in Semiconductor Manufacturing by Backward Simulation. In: *Proceedings of the 2014 Winter Simulation Conference*, Dec. 2014, Omnipress

Die Auswirkungen der Stornier- und Umbuchfunktion in Truck Appointment Systemen

Katharina Beck¹, Ann-Kathrin Lange^{1*}, Carlos Jahn¹

¹Institut für Maritime Logistik, Technische Universität Hamburg, Am Schwarzenberg-Campus 4 (D), 21073 Hamburg;

*ann-kathrin.lange@tuhh.de

Abstract. Größere Containerschiffe führen zu höheren Umschlagmengen je Anlauf und damit zu Peaks im Zu- und Ablauf der Containertransporte per Lkw. Um die Ankunftszeitverteilungskurve abzuflachen, haben viele Terminals in den vergangenen Jahren sogenannte Truck Appointment Systeme (TAS) eingeführt. In diesen Systemen buchen Lkw Zeitfenster für die Aufnahme oder Abgabe von Containern. Bei Abweichungen vom geplanten Transportablauf stehen teilweise unterschiedliche Flexibilitätsoptionen zur Zeitfensteranpassung zur Verfügung. Neben dem Tauschen von vorhandenen Slots können die Zeitfenster je nach Verfügbarkeit auch storniert und umbuchung werden. Die Studie untersucht die Auswirkungen der Nutzung der Flexibilitätsoptionen Tauschen, Umbuchen und Stornieren auf die KPI von Transportunternehmen unterschiedlicher Größe und bei verschiedenen Verkehrssituationen im Hafengebiet. Um die komplexen Prozesse realitätsnah abbilden zu können, wird ein diskretes ereignisorientiertes Simulationsmodell (DES) erstellt.

Einführung

Im Jahr 2018 wurden weltweit 784 Mio. TEU (Twenty-foot Equivalent Unit) umgeschlagen [1]. Immer größer werdende Schiffe und neu formierte Reederei-Allianzen, welche ihre Services mit geringerer Frequenz und dafür mit mehr Containern pro Anlauf planen, erhöhen die Komplexität der intermodalen Transportketten [2]. Die Zunahme der je Anlauf umgeschlagenen Container führen zu Peaks bei der Containeranlieferung und -abholung mittels Lkw und übersteigen die Gate- und Yardbearbeitungskapazitäten der Terminals [3]. Zur Verbesserung des Ankunftszeitmanagements von Lkw haben weltweit viele Terminals TAS eingeführt, deren Ziele u. a. eine bessere Koordinierung des Containerflusses und die Reduzierung von Staus und Wartezeiten sind [2]. Bei TAS werden die Öffnungszeiten der Terminals in Zeitslots unterteilt. Jedem Zeitslot wird eine Slotkapazität zugewiesen, die unter anderem abhängig vom zur Verfügung stehenden Handlingequipment ist. Die Lkw können die zu ihren Wünschen passenden Slots buchen, bis die voreingestellten Kapazitäten ausgeschöpft sind [4]. Das gebuchte Zeitfenster gibt den Zeitkorridor vor, in dem der

Lkw am Terminal ankommen muss [5]. Die implementierten TAS sind unterschiedlich ausgestaltet und bieten den Transportunternehmen teilweise verschiedene Flexibilitätsoptionen, um auf Störungen im Transportablauf reagieren zu können [6]. Die vorliegende Arbeit soll die Fragestellung klären, welche Auswirkungen die Möglichkeit des Stornierens und Umbuchens auch in Kombination mit der Möglichkeit des Tauschens von Zeitfenstern in TAS auf unterschiedliche Leistungsindikatoren wie Auftragserfüllungsquote, Wartezeit oder Pünktlichkeit eines Transportunternehmens besitzt.

1 Stand der Forschung

Bisher gibt es wenig Studien über die Auswirkungen von TAS auf die Produktivität von Transportunternehmen oder Evaluierungen von TAS hinsichtlich der Auswirkungen verschiedener Zeitfensterbreiten, Strafen oder Flexibilitätsoptionen auf das Gesamtsystem [7,8].

Das zu lösende Problem der vorliegenden Arbeit lässt sich wie folgt beschreiben: Im Hafenbereich muss eine bestimmte Anzahl an Containern von einem Fuhrunternehmen transportiert werden. Die identischen Lkw (homogener Fuhrpark) starten und enden an einem Depot und transportieren im Tagesverlauf Container zwischen Depots, Kunden und Servicestationen (z. B. Terminals oder andere intermodale Anlagen). Bei der Lösung des Problems sind die durch die Kunden und Servicestationen vorgegebenen Zeitfenster für die Abgabe und Aufnahme des Containers, die auf einen Container je Transportauftrag begrenzte Fahrzeugkapazität und die maximal zulässige Lenkzeit des Fahrers zu beachten [9,10]. Diese Tourenplanungsprobleme und Vehicle Routing Probleme sind NP-schwer und exakte Algorithmen sind nur bei kleinen Problemstellungen möglich. Die Anzahl der Schritte, um alle durchführbaren Touren zu ermitteln, steigt exponentiell mit der Problemgröße an und macht die exakte Lösung des Problems häufig unmöglich, weshalb auf Heuristiken und Metaheuristiken zurückgegriffen wird [11,8].

In der Literatur werden unterschiedlich ausgestaltete TAS betrachtet. Yi et al. [12] untersuchen ebenso wie

Chen et al. [13] ein TAS, bei welchem Slots in einem iterativen Prozess auf Grundlage der voraussichtlichen Wartezeit gebucht werden. Torkjazi et al. [14] untersuchen ein Centralized TAS, bei welchem die Slots von einer übergeordneten unabhängigen Plattform an die Lkw vergeben werden, die die angefragten Slotzeiten mit einbezieht. Ein kollaboratives TAS wird von Azab et al. [15] untersucht, bei welchem der Terminal die Abfertigungsprozesse simuliert, um die Durchlaufzeiten der angefragten Zeitfenster zu ermitteln. Die Fuhrunternehmen passen ihre Slotzeiten entsprechend an. Phan et al. [16] betrachten einen Verhandlungsprozess über Slots, um die Summe aller Wartezeiten, den Stau und die Kosten für Verspätungen zu reduzieren. Eine Kollaboration von Transportunternehmen in einem Hafen mit TAS wird von Schulte et al. [5] beleuchtet. Zehendner et al. [17] ermitteln die optimalen Kapazitäten der Zeitfenster in TAS in Terminals, die zusätzlich auch Schiffe und Bahnen abfertigen. Wie sich unterschiedliche Verteilungen der Zeitfensterkapazitäten und -breiten im TAS auf die Fuhrunternehmen auswirken, wird von Namboothiri et al. [18] untersucht. Shiri et al. [19] erweitern das Problem von [18] und beziehen zusätzliche Restriktionen wie Zeitfenster beim Kunden oder die Leercontainerallokation mit ein.

Huynh et al. [7] stellten fest, dass Studien, die TAS optimieren, meistens mathematische Programmierungen verwenden. Studien, die die Auswirkungen von TAS Parametern untersuchen nutzen eher Simulations- oder Warteschlangenmodelle. Dabei eignen sich speziell DES, um bspw. Lkw-Abläufe auf dem Terminalgelände sowie den Betrieb des Terminalequipments detailliert darzustellen und mit unterschiedlichen Szenarien zu testen [20].

So nutzen Azab et al. [15] DES für die Bestimmung der Durchlaufzeiten der Lkw, Do et al. [21] für die Vorhersage von Wartezeiten und Fahrstrecke des Handlingequipments, Huynh [22] und Huynh et al. [23] für die Auswirkungen der gewählten Zeitfensterkapazitäten auf die Terminalprozesse und Lkw, Ramírez-Nafarrate et al. [2] für den Gate Betrieb, van Asperen et al. [24] für die Auswirkungen von TAS auf die Performance der Containerstapelregeln und Speer [25] für die Optimierung von automatischen Lagerkransystemen. Schulte et al. [5] verwenden DES zur Simulation einer Kollaboration zwischen Fuhrunternehmen in einem Hafen mit TAS.

DES zur Untersuchung der Auswirkungen von TAS auf die Tourendurchführung von Transportunternehmen finden sich nach Kenntnis der Autoren nur in Lange et al. [26].

2 Aufbau der Simulationsstudie

Das DES wurde mit Tecnomatix Plant Simulation Version 15 erstellt und basiert auf Daten verschiedener Unternehmen des Hamburger Hafens. Dabei werden neben den vier großen Containerterminals 18 weitere Knoten wie Leercontainerdepots, Packbetriebe, sonstige Logistikknoten und das Fahrzeugdepot betrachtet.

Die Simulationsstudie soll untersuchen, ob sich die Möglichkeiten der flexiblen Stornierung und Umbuchung von Zeitfenstern in TAS positiv auf relevante Kennzahlen von Transportunternehmen, wie z. B. Anzahl der pro Tag durchgeführten Aufträge, Wartezeit vor den Knoten, Auftragsabbrüche oder Pünktlichkeit der Ankunft, auswirken. Dabei wird geprüft, ob eine besonders gute oder eine besonders schlechte Verkehrssituation Auswirkungen auf die Häufigkeit der Nutzung der Flexibilitätsoptionen besitzt und ob die positiven Effekte der Flexibilitätsoptionen unterschiedlich groß sind. Es werden unterschiedliche Unternehmensgrößen und verschiedene Schichtmodelle betrachtet, um Aussagen über deren Auswirkungen auf die Unternehmenskennzahlen auch in Kombination mit den Flexibilitätsoptionen treffen zu können. Es wird erwartet, dass kleine Unternehmen öfter umbuchen müssen, da größere Unternehmen mehr Tauschmöglichkeiten zur Anpassung ihrer Transporte besitzen und dass eine Kombination der Funktionen Umbuchen und Tauschen die positiven Effekte verstärkt.

Die Studie erweitert die Arbeit von Lange et al. [26] um die Möglichkeiten der Nutzung der Stornier- und Umbuchfunktion und verwendet folgende an [26] angelehnte Annahmen:

- Das Simulationsmodell ist auf den Bereich des Hamburger Hafens begrenzt.
- Ein Simulationslauf umfasst ein bis zwei Schichten. Alle erforderlichen Buchungen im TAS werden zu Beginn jedes Laufes getätigt.
- Im Tagesverlauf gehen keine neuen Transportanfragen ein.
- Zeitfensterbuchungen sind nur an den Containerterminals notwendig.
- Neben den Containerterminals haben nur vier Knoten ebenfalls 24/7 geöffnet.
- Transportaufträge gebuchter Zeitfenster können (auch kurzfristig) getauscht und bis vor Zeitfensterbeginn storniert und umgebucht werden, ein Hinzubuchen ist nicht möglich.
- Ein Transportauftrag besteht aus dem Abholen und Ausliefern eines Containers.
- Die Routenzeiten sind durch gesetzliche Lenkpausen und Schichtzeiten begrenzt.

Je nach Ankunftszeit am Terminal werden die Lkw unterschiedlichen Prioritätsgruppen zugeordnet, die sich auf die Art der Abfertigung beziehen. Die Zeitfensterbreite der Priorität 1-Abfertigung beträgt eine Stunde, wobei es eine Toleranz von je 30 Minuten vor und nach dem Zeitfenster gibt. Dadurch beträgt die Zeitfensterbreite, in denen ein Lkw mit der höchsten Priorität 1 abgefertigt wird, 120 Minuten. Sollte das Fuhrunternehmen erkennen, dass es sein Zeitfenster nicht einhalten kann, bietet das TAS den Fuhrunternehmen Flexibilitätsoptionen, um auf Veränderungen im Transportablauf reagieren und die Slots anpassen zu können. Bestehende Slots können storniert und umbucht oder getauscht werden. Das Stornieren und Umbuchen ist kurzfristig möglich und beeinflusst nur die Slotzeit des betreffenden Transportes. Beim Tauschen werden die Slotzeiten von zwei Transporten miteinander getauscht. Tauschen ist unabhängig von der Slotverfügbarkeit des Terminals immer möglich, eine Umbuchung kann nur dann erfolgen, wenn das Zeitfenster, in welches umbucht werden soll, noch freie Kapazitäten besitzt. Stornierungen nach Zeitfensterbeginn werden als No-Shows gezählt.

Mittels der durch Lange et al. [26] implementierten VBA wird die Auftragsliste für den untersuchten Tag in drei Schritten erstellt. Auf Grundlage der Struktur des Transportaufkommens werden Aufträge zufällig auf einzelne Verbindungen verteilt und in der Transportmatrix vermerkt. Für den Slotbuchungsprozess werden die Kapazitäten außerhalb der Betriebszeiten auf null gesetzt und die Kapazitäten in den verbliebenen Zeitfenstern je Terminal begrenzt. Die Kapazität je Zeitfenster berechnet sich aus der Multiplikation des einstellbaren Faktors mit der Anzahl der insgesamt benötigten Slotbuchungen und wird durch die Betriebszeit der Fuhrunternehmen dividiert. Die Zeitfensterbuchung erfolgt mit Zufallszahlen und in zufälliger Reihenfolge. Bei Aufträgen, die zwei Buchungen benötigen, wird versucht, das gleiche oder eines der beiden nachfolgenden Zeitfenster für die Senke zu buchen. Zusätzlich werden die Öffnungszeiten der Knoten und die Betriebszeiten des Fuhrunternehmens berücksichtigt.

Wie bei [26] erfolgt die dynamische Fahrzeugführung im Simulationsmodell und die Aufträge werden mit einer fahrzeuginitiierten Auftragszuweisung vergeben. Dabei werden neben der Dringlichkeit des Auftrages, welche von den Slotzeiten sowie etwaig vorhandenen Schließzeiten abhängt, auch die Fahrzeit zur Quelle des Folgeauftrages berücksichtigt. Bei der Vergabe des Auftrages wird nach Möglichkeit bereits ein passender Folgeauftrag vorgemerkt. Mögliche Tauschoptionen werden vor der Vergabe des nächsten durchzuführenden Auftrages

geprüft. Ob eine Umbuchung für den aktuellen und auch für einen eventuell bereits eingeplanten Folgeauftrag notwendig ist, wird erst nach der Auftragszuweisung untersucht. Sollte eine Umbuchung notwendig sein, so muss geprüft werden, ob noch Slots im gewünschten Zeitfenster verfügbar sind. Die Slotverfügbarkeit ändert sich im Zeitverlauf und ist abhängig von der Entfernung des Zeitpunktes der Buchung zum gewünschten Slottermin. Diese Dynamik bei der Slotverfügbarkeit wird durch die Erzeugung einer gleichverteilten Zufallszahl abgebildet.

Durch die unsicheren bzw. teilweise unvollständigen Informationen muss die Planung der Zeitfenster und des Transportablaufes unter Umständen mit den aktualisierten Informationen in einer rollierenden Planung modifiziert bzw. wiederholt werden [27]. Zu den unsicheren bzw. teilweise unvollständigen Informationen zählen z. B. die Slotverfügbarkeit im Tagesverlauf, die Auslastung der Terminals, die Möglichkeit einer Abfertigung auch außerhalb der gebuchten Slotzeit oder die reale Fahrzeit.

Um den Transport nachverfolgen zu können, wurden im Modell die Ereignisse Ankunft und Abfahrt am Knoten als sogenannte Meldepunkte definiert. Bei Eintritt dieses Ereignisses übermittelt der Lkw seine Position und die aktuelle Uhrzeit. Bei Ankunft an der Quelle wird die voraussichtliche Verweildauer, bestehend aus Warte- und Bearbeitungszeit, am Knoten zur Ermittlung der voraussichtlichen Abfahrtszeit (Estimated Time of Departure (ETD)) sowie die Fahrzeit zur Senke prognostiziert. Die damit ermittelte voraussichtliche Ankunftszeit (Estimated Time of Arrival, (ETA)) an der Senke, die hier ein Containerterminal darstellt, wird mit der vorhandenen Slotzeit verglichen.

Sollte der Lkw nicht innerhalb des Zeitkorridors seines Slots ankommen, so wird versucht, den Slot umzubuchen. Für alle bekannten Folgeknoten wird das Verfahren wiederholt. Ergebnisse etwaiger Umbuchungen der Vorgängerknoten werden bei der Planung berücksichtigt. Nach Abschluss des aktuellen Auftrages wird ein Folgeauftrag zugewiesen, der in der Regel dem bereits reservierten Folgeauftrag entspricht. Sollten sich spontan Änderungen hinsichtlich der Dringlichkeit eines durchzuführenden Auftrages ergeben, so kann der Disponent auch einen anderen Folgeauftrag zuweisen.

Mit der Google Distance Matrix API werden die tageszeitabhängigen Fahrzeitmatrizen erstellt, wobei diese neben der normalen Fahrzeit auch die Fahrzeiten für eine sehr gute und sehr schlechte Verkehrssituation ermitteln. Dadurch sollen die Auswirkungen unterschiedlicher Verkehrssituationen und die daraus resultierende Notwendigkeit einer Anpassung der Zeitfenster als zusätzlicher

Einflussfaktor der Transportdurchführung untersucht werden. Die Fahrzeiten werden im Modell vereinfacht als Verweilzeiten auf den jeweiligen Bausteinen betrachtet. Das dynamische Modell bezieht den Zeitaspekt und die Veränderungen der Fahrzeiten mit ein.

3 Experimente

Das Simulationsmodell untersucht drei verschiedene Unternehmensgrößen, von der die Anzahl der durchzuführenden Aufträge abhängt. Das kleine Unternehmen arbeitet im Ein-Schicht-Betrieb von 6 bis 15 Uhr, das große im Zwei-Schicht Betrieb von 4 bis 13 Uhr und von 13 bis 22 Uhr. Das mittlere Unternehmen nutzt beide Schichtmodelle. Die Fahrzeiten können für eine normale, gute und schlechte Verkehrslage berechnet werden.

Tabelle 1 zeigt die untersuchten Fuhrunternehmensgrößen, die Anzahl der Schichten sowie Fahrer und Aufträge. Bei zwei Schichten werden jeweils 50 % der Fahrer pro Schicht eingesetzt.

Nr.	Unternehmensgröße	Anzahl Schichten	Anzahl Fahrer	Anzahl Aufträge
1	Klein	1	12	65
2	Mittel	1	36	195
3	Mittel	2	36	195
4	Groß	2	60	325

Tabelle 1: Experimentplan mit ausgewählten Parametern

Es werden vier verschiedene Module untersucht (Tabelle 2). Neben einem Grundmodell werden eine Strategie, die nur Tauschen ermöglicht, und eine Strategie, die nur Umbuchen ermöglicht, betrachtet. Ebenso wird eine Kombination aus Tauschen und Umbuchen untersucht. Je Modul werden 100 Simulationsläufe durchgeführt.

Modul	Name des Moduls
1	Grundmodell mit Folgeauftrag
2	Tauschen mit Schwerpunkt Leerfahrt minimieren
3	Umbuchmodul
4	Gesamtmodul mit Umbuchen und Tauschen

Tabelle 2: Übersicht verschiedene Module und mit erlaubten Umbuchoptionen

4 Ergebnisauswertung

Tabelle 3 zeigt die Unterschiede bei der Auftragserfüllungsquote (AEQ) des Grundmodells (Modul 1) der drei Unternehmensgrößen und zwei Schichtmodelle.

	klein	mittel	mittel	groß
Fuhrunternehmen Nr.	1	2	3	4
AEQ normaler Verkehr	91,9 %	93,0 %	88,3 %	88,9 %
AEQ guter Verkehr	96,2 %	97,1 %	92,7 %	93,5 %
AEQ schlechter Verkehr	87,6 %	89,0 %	83,4 %	84,6 %

Tabelle 3: Vergleich verschiedener Auftragserfüllungsquoten im Grundmodell bei verschiedenen Verkehrssituationen

Es ist erkennbar, dass die AEQ bei einer Schicht (Unternehmen 1 und 2) besser ist als bei zwei Schichten (3 und 4), wobei die Unterschiede zwischen Unternehmen unterschiedlicher Größe aber gleichem Schichtmodell bei normalem Verkehr gering sind. Bei einer sehr guten Verkehrssituation steigt die AEQ um 4,1 % bis 4,6 % an. Die positiven Effekte fallen für Unternehmen mit zwei Schichten größer aus. Das große Unternehmen profitiert am meisten vom guten Verkehr. Eine schlechte Verkehrssituation reduziert die AEQ um 4,0 % bis 4,9 %, wobei die negativen Auswirkungen am größten für das mittlere Unternehmen mit zwei Schichten sind.

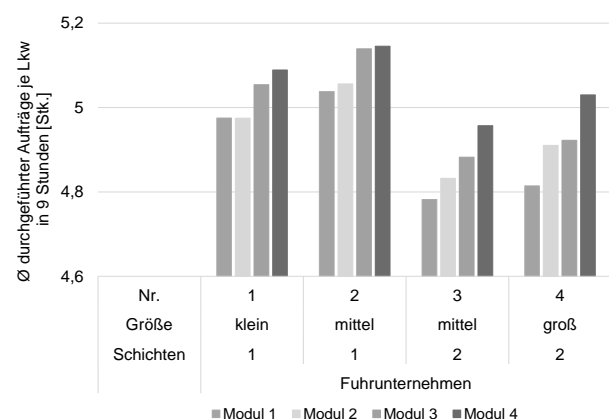


Abbildung 1: Anzahl durchschnittlich durchgeführte Aufträge je Lkw in 9 Stunden

Alle Flexibilitätsoptionen erhöhen die Anzahl der in 9 Stunden durchgeführten Aufträge im Vergleich zum Grundmodell. Modul 3 (nur Umbuchen) führt zu besseren Ergebnissen als Tauschen mit Leerfahrt minimieren (Abbildung 1). Das Gesamtmodul führt stets zu besseren Ergebnissen als Module mit nur einer Flexibilitätsoption.

Die Anzahl der durchgeführten Aufträge pro Lkw von Unternehmen mit einer Schicht ist größer als bei zwei Schichten. Bei gleichem Schichtmodell führt das größere Unternehmen mehr Aufträge durch.

Eine gute Verkehrssituation hat positive Auswirkungen auf die Anzahl der durchgeführten Aufträge. Bei einer schlechten Verkehrssituation sinkt die Anzahl der durchgeführten Aufträge bei allen Unternehmen, wobei die Auswirkungen auf Unternehmen mit zwei Schichten größer sind als auf Unternehmen mit einer Schicht.

Es wird deutlich, dass die Anzahl der Schichten bei der Anzahl der durchgeführten Aufträge entscheidender ist als die Größe des Unternehmens. Unternehmen mit einer Schicht können innerhalb ihrer Schicht mehr Aufträge durchführen. Grund hierfür ist die Struktur des Transportaufkommens, bei der 60 % einen und 20 % aller Aufträge zwei Logistikknoten beinhalten. Diese Logistikknoten haben größtenteils begrenzte Öffnungszeiten, die teilweise außerhalb der Schichtzeiten der zweiten Schicht liegen. Die Aufträge können daher nur in der ersten Schicht durchgeführt werden.

Bei zwei Schichten sind die Auswirkungen der Änderungen der Verkehrssituation in Prozent größer, da die Lkw von Transportunternehmen mit zwei Schichten neben der Morgenspitze des Verkehrs von 06:00 bis 07:00 Uhr auch während der Nachmittagspitze von 15:00 bis 16:00 Uhr im Einsatz sind.

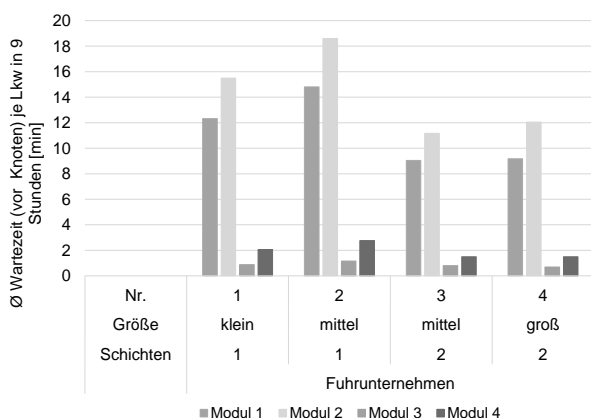


Abbildung 2: Durchschnittliche Wartezeit vor Knoten je Lkw in 9 Stunden

Die Wartezeit vor dem Knoten fällt nur an, wenn der Lkw am Terminal außerhalb seiner Slotzeit ankommt oder den Logistikknoten vor der Öffnung erreicht (Abbildung 2). Unabhängig von der Unternehmensgröße ist bei allen Modulen (3 und 4), die ein Umbuchen erlauben, die Wartezeit vor den Knoten am geringsten.

Beim Modul, welches nur das Tauschen erlaubt (Modul 2), ist die Wartezeit vor den Knoten größer als im Grundmodell. Wenn nur Umbuchen erlaubt ist (Modul 3)

ist die Wartezeit vor den Knoten am geringsten und insgesamt warten Unternehmen mit zwei Schichten kürzer als Unternehmen mit einer Schicht. Bei einer guten Verkehrssituation sind die Wartezeiten vor den Knoten länger, bei einer schlechten Verkehrssituation sind sie kürzer. Die Zunahme bei der guten Verkehrssituation beträgt bei einer Schicht rund 5 bis 10 Minuten bei den Modulen 1 und 2 sowie 3 bis 6 Minuten bei zwei Schichten. Bei den Umbuchmodulen beträgt die Zunahme maximal eine Minute. Bei einer schlechten Verkehrssituation nimmt die Wartezeit um 2 bis 5 Minuten bei den Modulen 1 und 2 bei einer Schicht und maximal 2 Minuten bei zwei Schichten ab. Die Wartezeiten bei den Umbuchmodulen sind annähernd gleich.

Die Wartezeit vor den Knoten ist, unabhängig von der Unternehmensgröße, bei allen Modulen, die ein Umbuchen erlauben (3 und 4), am geringsten und mit Wartezeiten unter vier Minuten innerhalb von neun Stunden zu vernachlässigen. Dies deckt sich mit dem Ziel der Erreichung des Terminals während der Priorität 1-Abfertigung, welches mit der Umbuchung verfolgt wird. Die Wartezeit vor den Knoten bei den anderen Modulen ist abhängig von der Anzahl der Schichten, wobei die Wartezeiten bei zwei Schichten geringer sind als bei einer Schicht. Bei einer guten Verkehrssituation sind die Wartezeiten insgesamt höher, da die Lkw schneller als geplant den Knoten erreichen. Da sowohl die initiale Slotbuchung als auch die Auftragszuweisung die erwarteten Fahrzeiten berücksichtigen, sind die Wartezeiten vor den Knoten bei Modulen ohne erlaubte Umbuchung höher. Bei einer schlechten Verkehrssituation sind die Wartezeiten vor den Knoten geringer. Da die Auftragszuweisung einen gewissen Puffer berücksichtigt und die Knoten damit tendenziell eher vor als nach dem Slottermin erreicht werden, fallen trotz schlechter Verkehrssituation noch Wartezeiten an.

Eine weitere wichtige Kennzahl ist der Anteil der Auftragsabbrüche bei der Durchführung aufgrund einer Ankunft außerhalb der Öffnungszeiten an den Logistikknoten oder außerhalb der erlaubten Abfertigungstoleranz an den Terminals (Abbildung 3). Beim Grundmodell werden anteilig die meisten Aufträge abgebrochen (2,5 % bis 3,5 %), wobei das große Unternehmen die meisten Auftragsabbrüche vorzuweisen hat. Bei einer Schicht führt das Tauschen (Modul 2) zu mehr Abbrüchen als das Umbuchen (Modul 3), bei zwei Schichten ist es umgekehrt. Eine gute Verkehrssituation führt insgesamt zu weniger Auftragsabbrüchen, eine schlechte zu mehr.

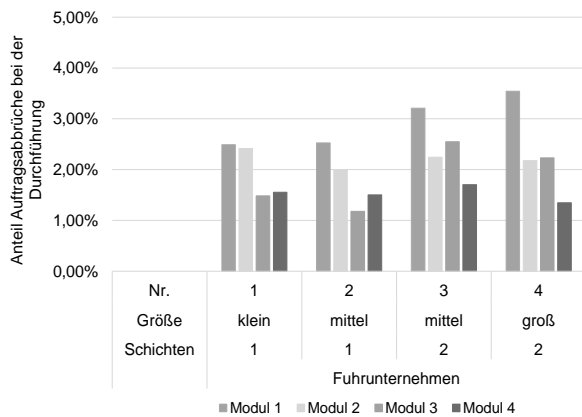


Abbildung 3: Anteil der Auftragsabbrüche bei der Durchführung

Der Anteil der Auftragsabbrüche ist bei zwei Schichten immer größer als bei einer Schicht, da dort beide Peaks des Verkehrs enthalten sind. Bei einer guten Verkehrssituation müssen weniger Aufträge abgebrochen werden, da auch bei der Zuweisung eines zeitlich kritischen Auftrages dieser mit einer höheren Wahrscheinlichkeit pünktlich durchgeführt werden kann. Bei einer schlechten Verkehrssituation müssen knapp zugewiesene Aufträge häufiger abgebrochen werden, da eine längere Fahrzeit eine Abfertigung verhindert. Innerhalb der zweiten Schicht liegen viele Schließzeiten der Logistikknoten, wodurch ein Abbruch des Auftrages nicht mehr nur an den Terminals, sondern auch an den Logistikknoten durch eine verspätete Ankunft möglich ist.

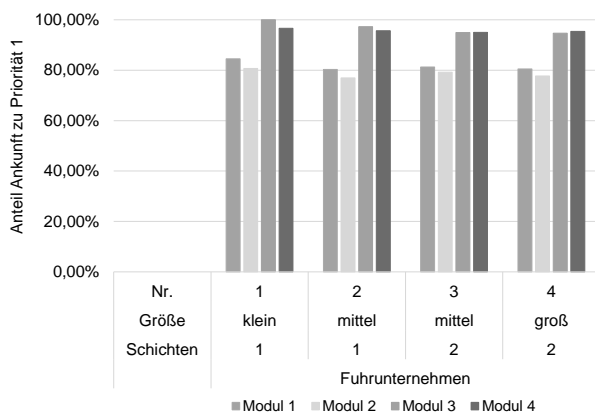


Abbildung 4: Anteil der Ankünfte innerhalb des Zeitfensters

Wie häufig die Lkw den Terminal innerhalb des Zeitkorridors mit der höchsten Abfertigungspriorität erreichen ist in Abbildung 4 dargestellt. Es ist ersichtlich, dass ein erlaubtes Umbuchen den Anteil der pünktlichen Ankünfte im Vergleich zum Grundmodell sowie Tauschmodell erhöht. Modul 2 führt zum geringsten Anteil der pünktlichen Ankünfte, wobei der Terminal häufig vor Zeitfensterbeginn erreicht wird. Eine Kombination der Möglichkeit des Umbuchens und Tauschens führt bei

zwei Schichten zur Erhöhung der Pünktlichkeit, bei einer Schicht sinkt die Quote leicht. Sollte nur ein Umbuchen erlaubt sein, so erreichen Lkw des kleinen Unternehmens den Terminal am häufigsten pünktlich.

Die Auftragszuweisung des Simulationsmodells weist die Aufträge so zu, dass eine Ankunft am Terminal zu Priorität 2 nach dem Slot eher vermieden wird. Unabhängig von der Verkehrssituation wird bei erlaubter Umbuchung der Terminal häufiger innerhalb von Priorität 1 erreicht, ebenso wird die Ankunft zu Priorität 2 nach dem Slot vermieden. Das Tauschmodell führt tendenziell zu einer zu frühen Ankunft am Terminal. Die Verkehrssituation hat kaum Einfluss auf den Anteil der Priorität 1-Ankünfte, allerdings wird der Terminal bei einer guten Verkehrssituation weniger häufig bzw. bei einer schlechten Verkehrssituation häufiger zu Priorität 2 nach dem Slot erreicht. Bei einem Auftragsabbruch an der Senke müsste der Lkw-Fahrer mit dem Container zum Depot fahren, um sein Chassis zu wechseln. Da dies vermieden werden soll, wird die Ankunft an der Senke mehrmals prognostiziert und der Auftrag im Simulationsmodell bereits an der Quelle abgebrochen, wenn ein Auftragsabbruch an der Senke wahrscheinlich ist. Die Senke wird unabhängig vom Fuhrunternehmen tendenziell eher zu früh erreicht. Bei erlaubtem Umbuchen wird die Senke fast immer innerhalb von Priorität 1 erreicht, da die Prognose häufiger vor Beginn des Zeitfensters durchgeführt wird, weshalb eine Umbuchung der Senke noch möglich ist.

Der Anteil umgebuchter Tourenpläne ist Abbildung 5 zu entnehmen.

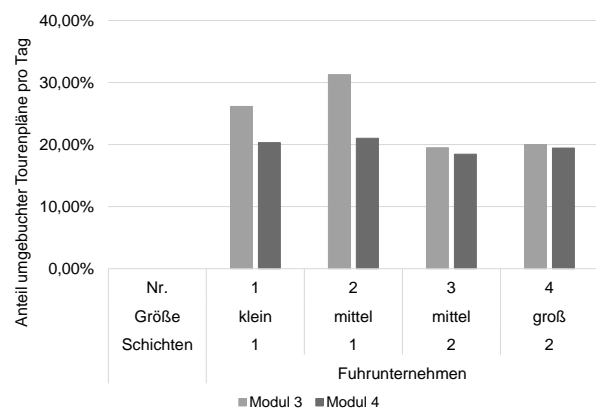


Abbildung 5: Anteil umgebuchter Tourenpläne pro Tag

Wenn nur Umbuchen erlaubt ist (Modul 3) ist der Anteil jeweils höher als bei dem kombinierten Modul 4. Bei einer Schicht sind weniger Restriktionen bei den Öffnungszeiten zu berücksichtigen, die eine Umbuchung teilweise verhindern. Daher buchen Unternehmen mit einer Schicht tendenziell häufiger um, wobei vor allem das

mittlere Unternehmen mit einer Schicht oft umbucht.

Eine gute Verkehrssituation führt vermehrt, eine pessimistische Verkehrssituation hingegen zu weniger Umbuchungen. Die Veränderung bei Modul 3 ist größer als bei Modul 4. Bei gutem Verkehr werden 5 % bis 10 % mehr Tourenpläne umbucht, da die Lkw zu früh ankommen würden. Bei schlechtem Verkehr wird weniger häufig umbucht, da die Tourenplanung einen gewissen Puffer enthält.

5 Fazit und Ausblick

Die Studie hat gezeigt, dass die Möglichkeit des Stornierens und Umbuchens in TAS einen positiven Einfluss auf die Fuhrunternehmen hat und daher eine wichtige Flexibilitätsoption darstellt.

Dabei ist die Nutzung der Umbuchfunktion stärker von der Anzahl der Schichten als von der Unternehmensgröße abhängig. Nur Umbuchen führt bei kleinen Unternehmen nicht zwingend zu besseren Lösungen als nur Tauschen. Bei einer Kombination von Tauschen und Umbuchen muss seltener umbucht werden und die Kombination der beiden Funktionen führt zu besseren Ergebnissen bei der AEQ als die Nutzung nur einer Flexibilitätsoption. Die Auswirkungen auf die AEQ fiel dennoch kleiner aus als erwartet.

Je mehr Flexibilitätsoptionen den Transportunternehmen zur Verfügung stehen, desto weniger Aufträge müssen abgebrochen werden. Zusätzlich konnte nachgewiesen werden, dass durch das Umbuchen die Wartezeit vor den Knoten sehr stark reduziert werden konnte und die Lkw sehr häufig während der Priorität 1-Abfertigung ankommen.

Eine besonders gute Verkehrssituation führt vermehrt zur Nutzung der Umbuchfunktion, eine besonders schlechte reduziert die Nutzung. Da die Auftragszuweisung tendenziell zu einer verfrühten Ankunft führt und genügend Puffer eingeplant wird, fallen die Auswirkungen einer durchschnittlich schlechten Verkehrssituation auf den kurzen Fahrstrecken nur wenig ins Gewicht. Neben den Vorteilen, die TAS für die Terminals haben, kann ein gut umgesetztes TAS auch Vorteile für das Fuhrunternehmen bieten, wenn er auch kurzfristig die vorhandenen Flexibilitätsoptionen nutzen kann und seine IT-Systeme so angepasst sind, dass er alle Funktionen des TAS richtig ausnutzen kann.

Die Arbeit konnte die vorangegangene Arbeit von Lange et al. [26] fortsetzen und das Modell realitätsnaher gestalten. Trotzdem mussten Annahmen getroffen werden, die teilweise von der Realität abweichen. Das Modell nutzt z. B. die vorhandenen Lenkzeiten der Fahrer

nicht vollständig aus. Ebenso warten Fahrer teilweise bis zu eine Stunde am Tag auf die Zuweisung eines neuen Auftrages. Darüber hinaus ist anzumerken, dass die Daten nicht gezielt für die Simulationsstudie erhoben wurden, sondern angepasst und aufbereitet wurden.

In zukünftigen Arbeiten könnte untersucht werden, wie sich zusätzliche Terminvorgaben an den Logistikknoten auswirken. Bei Transportaufträgen, die zwei Slots benötigen, gilt es zu untersuchen, wie sich eine Dynamisierung, d. h. eine automatische Anpassung des Slots an der Senke bei Verzögerungen an der Quelle, auswirkt. Bei Transporten zwischen zwei Terminals sollte bei Slotverfügbarkeit an einem der beiden Terminals die Anlieferung oder Abholung am anderen Terminal garantiert werden können.

Darüber hinaus könnte untersucht werden, welche Auswirkungen eine Veränderung der Struktur des Transportaufkommens oder eine Verlängerung der Öffnungszeiten der Logistikknoten hat. Ein großes Einsparpotential liegt in der Verkürzung der Wartezeiten auf die Auftragszuweisung, hierfür könnte eine verbesserte Vorwegeplanung oder eine dynamischere Anpassung sowie Veränderungen von mehreren Slotzeiten implementiert werden.

Bisher wird eine veränderte Verkehrssituation nur bei der Durchführung berücksichtigt, womit eine Lastgrenze untersucht wird. Die veränderte Verkehrssituation könnte bereits bei der Auftragszuweisung berücksichtigt werden, um bessere Dispositionsentscheidungen treffen zu können. Bei der initialen Slotbuchung könnte versucht werden, unter der Voraussetzung, dass ausreichend Slots vorhanden sind, diese immer nur in direkt angrenzende Zeitfenster bei Transporten mit zwei Terminals zu buchen, um Wartezeiten und zu frühe Ankünfte an der Senke zu vermeiden.

6 Literatur

- [1] DVV, "Mehr Umschlag, geringere Investitionen," URL: <https://www.thb.info/rubriken/single-view/news/mehr-umschlag-geringere-investitionen.html> [retrieved 5 July 2020].
- [2] Ramírez-Nafarrate, A., González-Ramírez, R. G., Smith, N. R., Guerra-Olivares, R., and Voß, S., Impact on yard efficiency of a truck appointment system for a port terminal. *Annals of Operations Research*, Vol. 258, No. 2, 2017. pp. 195–216. doi: 10.1007/s10479-016-2384-0.
- [3] Dekker, R., van der Heide, S., van Asperen, E., and Ypsilantis, P., A chassis exchange terminal to reduce truck congestion at container terminals. *Flexible Services and Manufacturing Journal*, Vol. 25, No. 4, 2013. pp. 528–542. doi: 10.1007/s10696-012-9146-3.

- [4] Fan, Ren, Guo, and Li, Truck Scheduling Problem Considering Carbon Emissions under Truck Appointment System. *Sustainability*, Vol. 11, No. 22, 2019. p. 6256. doi: 10.3390/su11226256.
- [5] Schulte, F., González, R. G., and Voß, S., Reducing Port-Related Truck Emissions: Coordinated Truck Appointments to Reduce Empty Truck Trips. *Computational Logistics*, edited by F. Corman, S. Voß and R. R. Negenborn, Springer International Publishing, Cham, 2015. pp. 495–509.
- [6] Davies, P., Container terminal reservation systems. *The 3 Rd Annual METRANS National Urban Freight Conference*, 2009. URL: <http://dtci.ca/wp-content/uploads/2011/10/Container-Reservation-Systems-> [retrieved 1 July 2020].
- [7] Huynh, N., Smith, D., and Harder, F., Truck Appointment Systems. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2548, No. 1, 2016. pp. 1–9. doi: 10.3141/2548-01.
- [8] Namboothiri, R., Planning Container Drayage Operations at Congested Seaports. Georgia Institute of Technology, 2006.
- [9] Jula, H., Dessouky, M., Ioannou, P., and Chassiakos, A., Container movement by trucks in metropolitan networks: modeling and optimization. *Transportation Research Part E: Logistics and Transportation Review*, Vol. 41, No. 3, 2005. pp. 235–259. doi: 10.1016/j.tre.2004.03.003.
- [10] Sterzik, S., and Kopfer, H., A Tabu Search Heuristic for the Inland Container Transportation Problem. *Computers & Operations Research*, Vol. 40, No. 4, 2013. pp. 953–962. doi: 10.1016/j.cor.2012.11.015.
- [11] Braekers, K., Ramaekers, K., and van Nieuwenhuys, I. e., The Vehicle Routing Problem: State of the Art Classification and Review. *Computers & Industrial Engineering*, Vol. 99, 2015. URL: https://www.researchgate.net/publication/287796502_The_Vehicle_Routing_Problem_State_of_the_Art_Classification_and_Review.
- [12] Yi, S., Scholz-Reiter, B., Kim, T., and Kim, K. H., Scheduling appointments for container truck arrivals considering their effects on congestion. *Flexible Services and Manufacturing Journal*, Vol. 16, No. 1, 2019. p. 87. doi: 10.1007/s10696-019-09333-y.
- [13] Chen, G., Govindan, K., Yang, Z.-Z., Choi, T.-M., and Jiang, L., Terminal appointment system design by non-stationary M(t)/Ek/c(t) queueing model and genetic algorithm. *International Journal of Production Economics*, Vol. 146, No. 2, 2013. pp. 694–703. doi: 10.1016/j.ijpe.2013.09.001.
- [14] Torkjazi, M., Huynh, N., and Shiri, S., Truck appointment systems considering impact to drayage truck tours. *Transportation Research Part E: Logistics and Transportation Review*, Vol. 116, 2018. pp. 208–228. doi: 10.1016/j.tre.2018.06.003.
- [15] Azab, A., Karam, A., and Eltawil, A., A Dynamic and Collaborative Truck Appointment Management System in Container Terminals. *Proceedings of the 6th International Conference on Operations Research and Enterprise Systems*, SCITEPRESS - Science and Technology Publications, 2017. pp. 85–95.
- [16] Phan, M.-H., and Kim, K. H., Negotiating truck arrival times among trucking companies and a container terminal. *Transportation Research Part E: Logistics and Transportation Review*, Vol. 75, 2015. pp. 132–144. doi: 10.1016/j.tre.2015.01.004.
- [17] Zehendner, E., and Feillet, D., Benefits of a truck appointment system on the service quality of inland transport modes at a multimodal container terminal. *European Journal of Operational Research*, Vol. 235, No. 2, 2014. pp. 461–469. doi: 10.1016/j.ejor.2013.07.005.
- [18] Namboothiri, R., and Erera, A. L., Planning local container drayage operations given a port access appointment system. *Transportation Research Part E: Logistics and Transportation Review*, Vol. 44, No. 2, 2008. pp. 185–202. doi: 10.1016/j.tre.2007.07.004.
- [19] Shiri, S., and Huynh, N., Optimization of drayage operations with time-window constraints. *International Journal of Production Economics*, Vol. 176, 2016. pp. 7–20. doi: 10.1016/j.ijpe.2016.03.005.
- [20] Huynh, N., Reducing Truck Turn Times at Marine Terminals with Appointment Scheduling. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2100, No. 1, 2009. pp. 47–57. doi: 10.3141/2100-06.
- [21] Do, N. A. D., Nielsen, I. E., Chen, G., and Nielsen, P., A simulation-based genetic algorithm approach for reducing emissions from import container pick-up operation at container terminal. *Annals of Operations Research*, Vol. 242, No. 2, 2016. pp. 285–301. doi: 10.1007/s10479-014-1636-0.
- [22] Huynh, N., [dissertation] Methodologies for reducing truck turn time at marine container terminals. 2005.
- [23] Huynh, N., and Walton, C. M. (eds.), *Improving Efficiency of Drayage Operations at Seaport Container Terminals Through the Use of an Appointment System*, 2011.
- [24] van Asperen, E., Borgman, B., and Dekker, R., Evaluating impact of truck announcements on container stacking efficiency. *Flexible Services and Manufacturing Journal*, Vol. 25, No. 4, 2013. pp. 543–556. doi: 10.1007/s10696-011-9108-1.
- [25] Speer, U., *Optimierung von automatischen Lagerkransystemen auf Containerterminals*, Springer Fachmedien Wiesbaden, Wiesbaden, 2017.
- [26] Lange, A.-K., Grafelmann, M., Schwientek, A., and Jahn, C., Effizientes Tauschen der Zeitfenster von Transportaufträgen in Truck Appointment Systems: Efficient Swapping of Time Windows of Transport Orders in Truck Appointment Systems. 1st ed., edited by M. Putz and A. Schlegel, Wissenschaftliche Scripten, Auerbach/Vogtl., 2019. pp. 295–304.
- [27] Arnold, D., Isermann, H., Kuhn, A., Tempelmeier, H., and Furmans, K. (eds.), *Handbuch Logistik*, 3rd ed., Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

Modeling Urban Transportation Using Tree-Attribute-Matrix Models

Kilian Nickel, Daniel Lückerrath, Oliver Ullrich

Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, Schloss Birlinghoven, 53757 Sankt Augustin, Germany; kilian.nickel@iais.fraunhofer.de

Abstract.

This paper applies the Tree-Attribute-Matrix (TAM) modelling method to a simplified model of an urban light-rail transportation system. The resulting model is a conceptual model that is beneficial for understanding, management and coordination of the system on a high level, in particular when different (interdisciplinary) stakeholders are involved. The paper briefly explains basic terms and terminology of railway systems as well as of the TAM modelling approach. It displays a simplified rail network and how it is translated into a TAM model. The resulting model contains the key physical and logical components of the system. In particular, the matrix depiction between line routes and the platforms they connect is found suitable for gaining oversight and identifying points of high complexity. In this case, there are five platforms that are serviced more by line routes than the other platforms and can be considered bottlenecks for service operation. The TAM model is considered less well suited when it comes to a complete description of realistic timetables and rail network plans, which require more detail (such as turn-outs and track sections) as well as more quantity of data in the model (e. g. the number of trips made per day).

The conceptual TAM model discussed could be generalized to include other urban sub-systems and their interactions, such as critical infrastructure systems. In that case, such a model would provide a common ground regarding understanding and terminology between different stakeholders, highlight points of strong interactions and allow to discuss the impacts of failures within the system on a high level.

Introduction

Urban system components (including water and electrical power supply grids, sewage and draining systems, street and transit networks) with their inter-dependencies constitute a system of high complexity. That “system of systems” forms the base of urban life; its functionality and reliability are essential for the well-being of the urban population.

Urban system components have long been subject to modelling, simulation, and optimization. One recurring challenge in modelling urban system components, especially when many domain experts and stakeholders from different professions are involved, is the generation of structured knowledge as a base for the system model itself [6]. All the domain experts bring their individual understanding of the system components and their crucial elements and behavior, with their respective points of

view strongly dependent on their own professional backgrounds.

Such a modelling project can benefit from the application of a structured method to collect information on the organization, characteristics, and inter-dependencies of urban system components. The Tree-Attribute-Matrix (TAM) modelling method [1][4] aims at supporting the assessment of the structure specifically of systems that are designed, managed and controlled by humans. Using TAM, these systems are represented as a collection of interconnected tree and matrix structures annotated with attributes. TAM is usually applied to facilitate a better understanding of administrative and business systems, and is utilized as a starting point to reduce their organizational complexity. TAM has been developed in the course of a number of business analysis projects, and is being routinely used as a tool for the analysis and improvement of administration and production processes, as well as in IT infrastructure and change management projects. TAM specifically focuses on transparency and simplicity of the structural modelling process, its purpose is to create a more understandable representation of the whole system and its complexities.

This paper examines the applicability of the TAM approach to model urban system components. As a first experiment, TAM is applied to assess the structure, components, and internal dependencies of a public transit system. Following on to this introduction to motivation, aims, and scope, the paper goes on to share some background on urban transit systems as well as the TAM modelling method (Section 1). To show the applicability of TAM in the context of urban system components, it then describes an exemplary model of an idealized transit system (Section 0). The paper closes with a short summary of the lessons learned and an outlook on further research (Section 3).

1 Background

1.1 Urban Transit Systems

Urban transit networks can be considered as a combination of physical and logical components. The *physical network* consists of tangible entities, e.g. stations, tracks and rail cars, whereas the *logical network* is comprised of concepts and plans, e.g. lines, trips or timetables. Figure 1 shows an extract of an example demonstrator network.

At the beginning of each *turn*, which is constituted by a number of concatenated *trips*, and is the planned movement of a *vehicle* through the network on a specific operational day, a tram leaves the maintenance and storage depot where it was stored over night. It then travels to the first platform of its first trip, where the *passenger exchange* takes place. *Platforms* are usually unidirectional and always part of a *station*, which combines adjacent platforms under a common name.

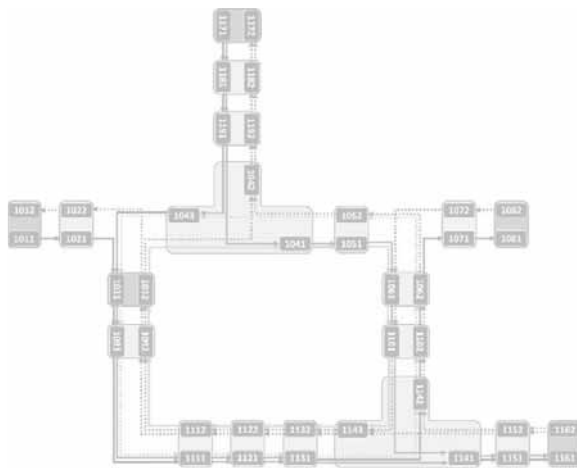


Fig. 1: A demonstrator light-rail network of four lines with eight routes (Source: [2]).

After executing the passenger exchange the vehicle travels to the next platform of the trip. The order of platforms to be visited is defined by the *line route*. Different line routes can be combined under a common name, thus constituting a *line*. For example Cologne's light-rail Line 1 (from Junkersdorf to Bensberg and back) actually consists of 27 line routes, 15 of which are eastbound and 12 are westbound. The type of rail cars used defines the maneuvering capabilities and hence the characteristics of it traversing the network. Table 1 depicts some important characteristics for two different light-rail car types in use in Cologne's tram network. As of 2020, Cologne operates a rail car fleet of 382 cars [9].

Characteristics	K4000	K5000
Length of car	29.2 m	29.3 m
Weight of car	35.0 t	37.8 t
Maximum velocity	80 kph	80 kph
Acceleration	1.3 m/s ²	1.2 m/s ²
Normal brake ability	1.4 m/s ²	1.2 m/s ²
Emergency brake ability	3.0 m/s ²	2.73 m/s ²

Table 1: Characteristics of two light-rail car types (Sources: [7][8])

The *tracks* between two locations of the network are usually unidirectional, but bidirectional tracks also exist. Some tracks may have speed limitations due to their environment, e.g. inner-city tracks may have a speed limit because of traffic regulations.

While the vehicle travels from one platform to another it may have to traverse track switches. These are locations where tracks meet; they can be differentiated between dividing and joining track switches. Like platforms and tracks, track switches are usually unidirectional. All but one of the tracks sharing one side of the track switch must form a curve, which leads to speed limitations that are usually lower than the speed limits on tracks. The access to track switches (as well as to platforms and track sections) is usually controlled by traffic lights.

At the end of the operational day, the vehicle travels once again to a maintenance and storage depot. The spatial and chronological order of the vehicles in use on a specific operational day is constituted by the timetable, i.e. the timetable assigns each tram a turn and each turn a set of line routes with starting times. The timetable defines the services as experienced by the prospective customers when they use the transit system.

1.2 Tree-Attribute-Matrix Models

The TAM method develops a representation of the structure of any examined real-world system, be it a public administration, a technical system, a business, or any other form of engineered organization. Usually, the model is created for a specific point of view of the person who is tasked to manage and oversee the whole system. Alternatively, it can serve as a common ground for multiple stakeholders who manage different parts of the system. Therefore, the depth of the model depends on the observers.

TAM models in a nutshell. A TAM model is essentially an information tree (think of a mind map) with nodes storing pieces of information, typically names, values or expressions. Nodes can be assigned one or more user-defined types - any concept desired to be stored in a variable can be defined as a type. They are a necessary tool to distinguish the meaning of nodes and to handle (filtered) subsets within the model. The nodes on the first level together with their corresponding sub-trees are called *aspects* or *trees*. They are meant to describe the components of the underlying system using hierarchical decomposition. Nodes within different trees can be linked to each other, representing interactions between real-life system components. These links are graphically represented as a matrix. For a more detailed description of TAM and its application in the complexity assessment in business and other organizations see [4].

Trees and Attributes. The first step in the formal modeling process is to identify the components of the system under consideration and model them as trees. The hierarchical lines connecting nodes to sub-nodes are usually understood as “contains”, “has property” or “depends on”, but the semantics are up to the user. For example, one would construct a tree containing all relevant classes of rail cars according to Table 1. The resulting tree is displayed in Fig. 2, where the classes have their own type “Rail car class” and their properties have the type “attributes”. In general, an attribute is a node which carries additional information, but is in itself not an object that interacts with other parts of the system.

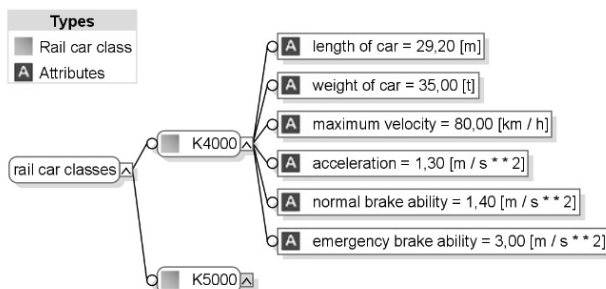


Fig. 2: A tree containing classes of rail cars with attributes.

Matrices. With the trees and types in place, the next piece of information to model are interactions and dependencies. This is done by defining matrices between any two trees. For example, the user might add the actual rail cars within an operator’s fleet to the model as a separate tree. They would in turn be mapped to the corresponding rail car classes to represent which type of car they are. The result is displayed as a matrix (Fig. 3).

Types			
		K4000	K5000
Rail car class			
Rail car			
	rail car K5000-01	□	■
	rail car K5000-02	□	■
	rail car K4000-01	■	□
	rail car K4000-02	■	□

Fig. 3: A matrix representing rail cars and their corresponding classes.

In this case, this (somewhat trivial) matrix represents a one-to-one mapping between two trees. In general, matrices can represent any n-to-m mappings. The layout of trees (T) and matrices (M) within a model are visualised via so-called TM diagrams (Fig. 4). Each matrix is named according to the semantics that they represent, in this case “rail cars are of class”, which also codes the direction of the connection made between the two trees. This depiction highlights the similarity to a knowledge graph [5] with trees representing nouns and matrices representing verbs.

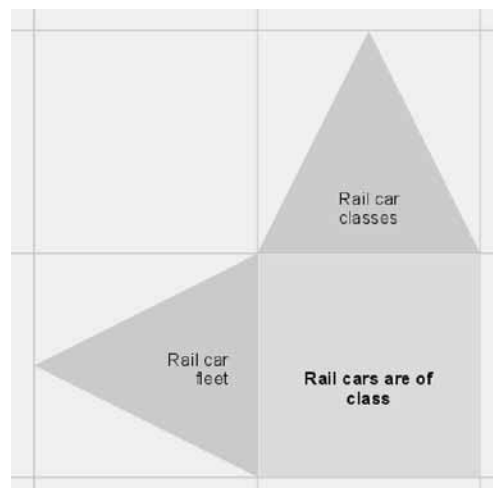


Fig. 4: A TM diagram showing trees as triangles and matrices as squares.

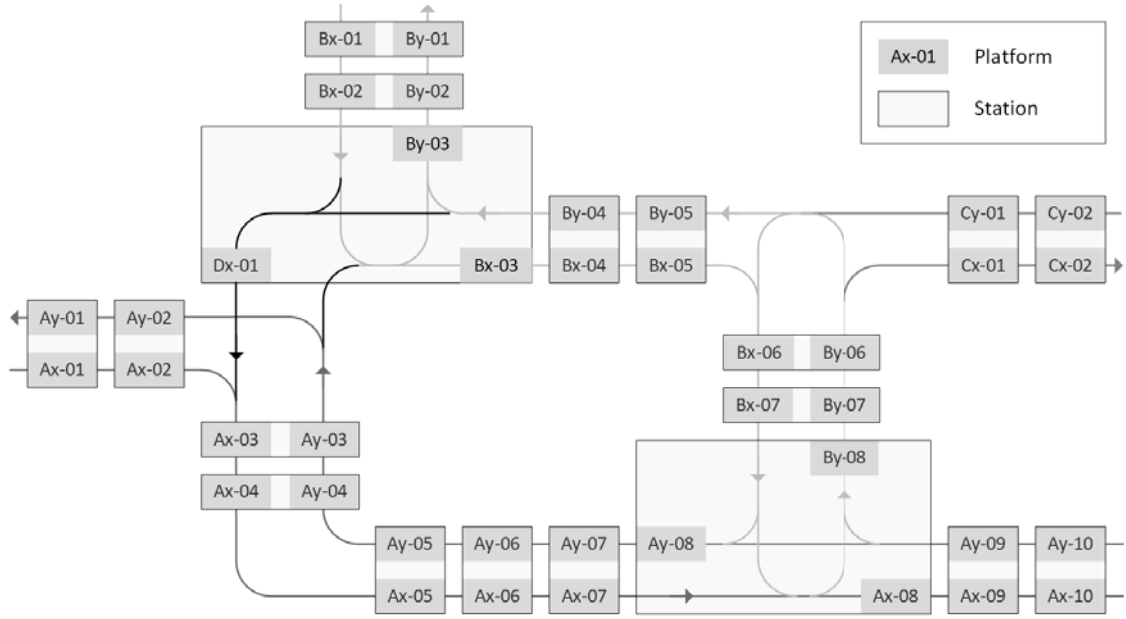


Fig. 5: A light-rail network based on [2] with renamed stations and platforms.

2 Modelling Urban Transit Systems Using TAM

In this chapter, the key components of the urban transit system shown in Fig. 1 are modelled using the TAM method. The key components include:

- Platforms
- Stations
- Line routes
- Lines

For an easier naming convention, Fig. 1 is translated into Fig. 5, which is used in the following.

2.1 Physical components of the network

The rail network under consideration is composed of 40 platforms. It has 19 stations, each consisting of two or three platforms, e.g. within walking distance of each other. To translate this statement into a TAM model, three components are required:

- a tree containing platforms
- a tree containing stations
- a matrix connecting the two

Now, placing all 40 platforms into a tree with a flat hierarchy would be the simplest tree design. However, more structure can improve human understanding. In our

example, the longest track line (blue) consists of two parallel tracks with 10 platforms on each side, labelled Ax (eastbound) and Ay (westbound). Similarly, there are 7 platforms along the green tracks labelled Bx (southbound) and 8 platforms labelled By (northbound). There are two platforms on each of the red tracks with labels Cx and Cy . In addition, completing the circle line there is only one platform Dx_1 along the black tracks. Using this naming scheme we can model a tree with six first-level nodes as in Fig. 6. Depending on the information within a tree, the order of the nodes can be deemed relevant or not. In this case, the order is relevant, as it corresponds to the order of platforms along the track.

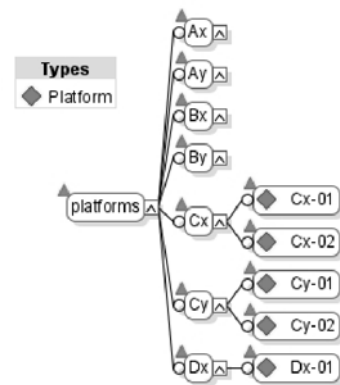
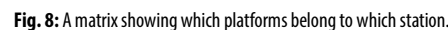
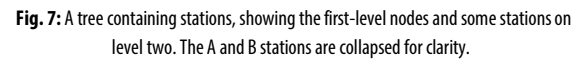


Fig. 6: A tree containing platforms, showing the first-level nodes and some platforms on level two. The A and B platforms are collapsed for clarity.

Concerning stations, the same naming convention using the letters A-D can be used. The first level has nodes labelled A , B and C , containing 10, 7 and 2 stations, respectively. In Fig. 7 the station tree is shown, together with link-nodes representing matrix entries, e.g. from station C_1 to their corresponding platforms Cx_1 and Cy_1 . The mapping of stations and platforms is also shown in the matrix in Fig. 8.



2.2 Logical components of the network

Rail cars move along line routes, passing through several platforms. Line routes will be modelled as entities within a separate tree. For simplicity, consider three types of line routes: (1) a circle line starting at station A_3 , (2) a linear east-to-west line (C_2 to A_1), and a linear north-to-east connection (B_1 to A_{10}), each with two-way connections. This results in six individual line routes passing through stations as indicated in Fig. 10. The line routes again combine to form lines (shown in Fig. 9), which are usually communicated to the passengers.

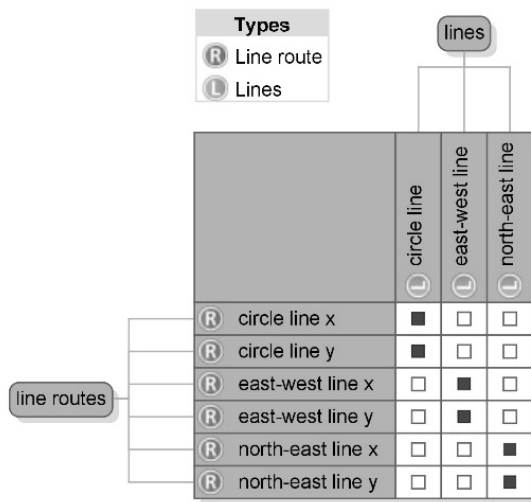


Fig. 9: Matrix showing lines and their corresponding line routes.

Next, the line routes need to be served by rail cars at specific points in time, forming a trip. This information object is at least a triplet: (rail car, line route, starting time) and can thus not be satisfied by just one matrix. A way to deal with this is by adding a separate tree for all the trips, which can store the starting time as an attribute (Fig. 11). Then, each trip can be mapped to rail cars and line routes respectively, creating two matrices (Fig. 12, Fig. 13). Let's assume for simplicity that each trips takes 55 minutes to be completed.

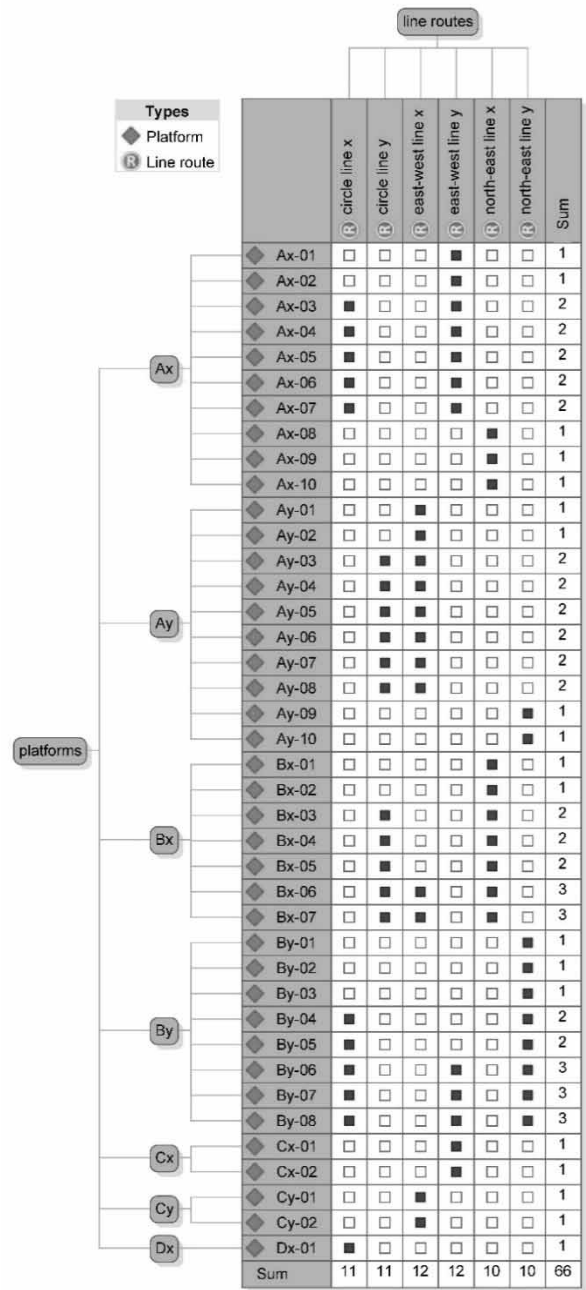


Fig. 10: Matrix showing line routes passing through platforms.

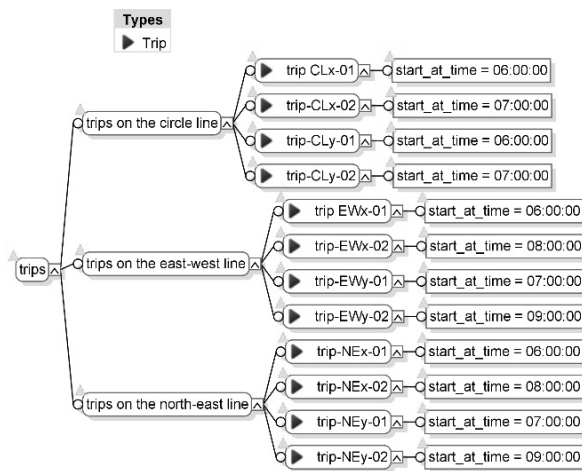


Fig. 11: A tree containing trips made by rail cars at a certain starting time.

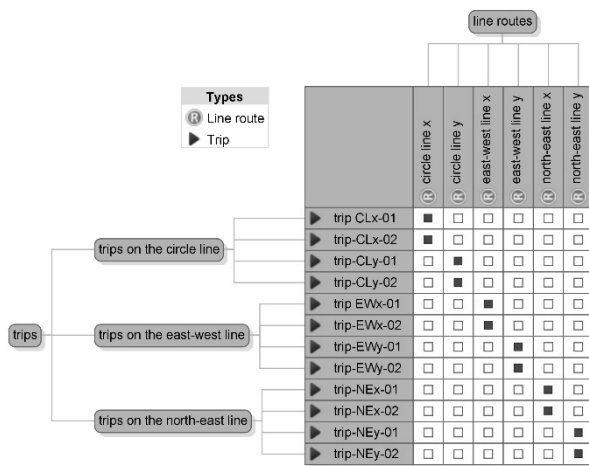


Fig. 12: A matrix showing trips serving different line routes.

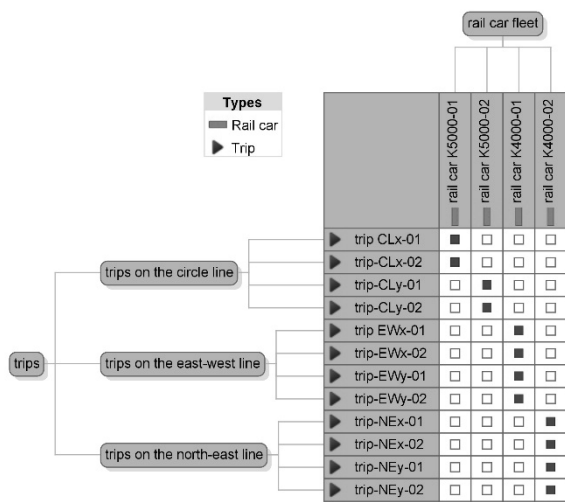


Fig. 13: A matrix showing which trips are being executed by which rail cars.

In real life, the number of trips will likely be much larger and would require a consistent naming scheme. In the example shown here, the circle line is served by two cars which each start a trip at the same time in different directions. After 60 minutes, they both start another cycle. The other two lines are each served by just one rail car, each serving both directions and oscillating between the first and the last stop. This basic model can be extended to include more cars, trips and timing information via additional attributes.

The big picture of the model is presented as a TM diagram showing all the trees and matrices between them (Fig. 14). There are seven trees and six matrices in total. The layout of triangles and squares is up to the user, as long as trees and associated matrices are separated only by white squares. It may be required that the same tree shows up multiple times in a TM diagram, however, this is not the case here.

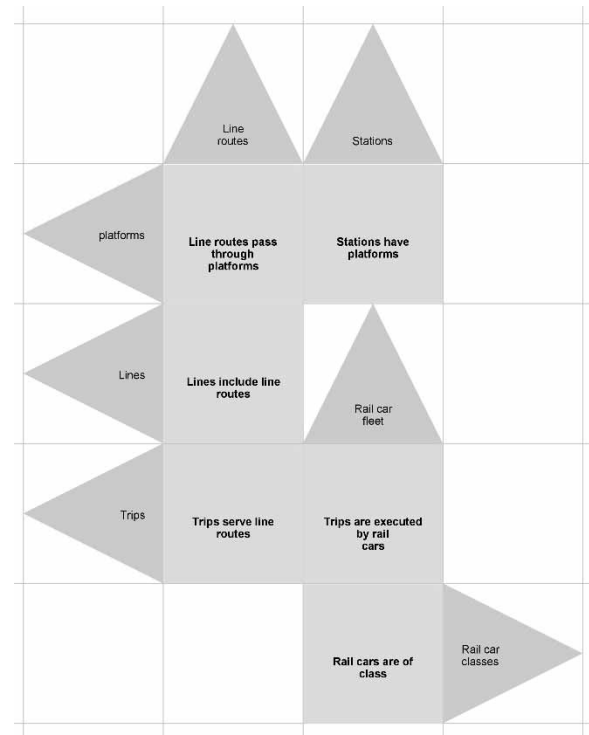


Fig. 14: Overall depiction of the model as a TM diagram.

2.3 Results and Discussion

We have set up a simplified TAM model comprising physical and logical components of an urban rail network and its operations. The model counts seven trees and six matrices. In a realistic model, the number of entries per tree and matrix would be larger, but the overall layout, as

shown in Fig. 14, would remain unaffected. Some components of the model have a static character, meaning that they do not change very often over time (i.e. rail car classes, platforms, stations, line routes, lines). Other components are more likely to experience changes during operations (rail car fleet, trips, timing, rail car assignment).

What are the advantages and disadvantages of this TAM model? Firstly, the model does not contain full topological information about the rail network, since it lacks the connection points (turnouts) between the track sections A, B, C and D. This information could be included by introducing a self-coupling matrix “platform is connected to platform”. However, this practice is discouraged in TAM modelling because it will likely introduce circular dependencies into the model, which inhibits queries along the graph. Another way to capture this information would be to introduce additional trees for turnouts and track sections respectively, allowing to link them to the platforms they connect. However, both of these approaches do not play to the strengths of a TAM model, which is to improve human understanding and oversight of a system. When it comes to comprehending the layout of a rail network, nothing beats a map such as Fig. 5. Therefore, information which is either too detailed or does not really benefit from being projected into trees and matrices should be left out of a TAM model.

The most useful matrix within the model is likely the mapping between platforms and line routes (Fig. 10). This is because it shows the key components of the physical assets (platforms) and the logical assets (line routes) in one planar map. By looking at single columns (line routes), one can see which platforms are connected by a line route. On the other hand, looking at a single row gives information about which line routes that stop at the same platform. The matrix can be used to “walk” vertically and horizontally from blue dot to blue dot, tracing paths that a rail car could reach if it only followed the paths allowed by line routes. This way, it can be noticed that the set of platforms decomposes into two separate sets that are disconnected this way. Technically, a rail car could reach any platform from any starting point, if it were allowed to move unrestricted.

Additionally, the sums of dots on the columns and rows provide useful information. Column sums show the number of stops along a line route. Row sums show how many line routes stop on the same platform, indicating

where to find bottlenecks. In fact, there are only five platforms ($Bx_6, Bx_7, By_6, By_7, By_8$) which have the maximum of three line routes stopping there. This limits the service frequency of those lines. For example, if a platform can only handle trains stopping at least five minutes apart, it can handle 12 stops per hour. As a consequence, only 4 stops per hour can be made on average by each of the three lines passing through them. TAM models are well suited for this type of top-level complexity analysis.

Another benefit is that the process of TAM modelling forces to create structures and nomenclature. In this case, this resulted in grouping the platforms and stations along lines. Other ways of structuring are of course also possible and this is what creates improved understanding and manageability.

For the purpose of simulation, TAM models can serve as a blueprint for defining the required objects and properties. Simple calculations can be included via expressions between different nodes, such as adding up the time taken for transit and stops along a line.

3 Conclusion

The TAM modelling method has been applied to a simplified network of an urban light-rail transport system in order to describe its physical and logical components. The resulting model provides a useful representation, in particular by the matrix mapping line routes to platforms (Fig. 10). This matrix provides a linearized view on the whole network including line routes served by trains, while allowing a top-level complexity analysis.

The model’s use is limited when it comes to (a) representing a full physical network including turnouts and rail tracks, (b) describing a full service schedule including a large number of trips per day. For these cases, expert software and dedicated models are likely to be better suited. The TAM model could be generalised to describe multi-mode transport networks or even different critical infrastructures components in an urban environment, operated by separate entities. The benefit would be to create a common ground between different stakeholders, improving their capability to coordinate and having a means to identify points of strong interactions (e.g. choke points) within the networks. The model could be used to consider certain emergency settings, such as the shutdown of a certain railway platform, and analyse the impacts on other parts of the system.

References

- [1] Beyer, U., Nickel, K., Hasenbeck, F., Zimmermann, A.: *Mensch und System - Ideen zu humanzentrischen Systemmodellen*. Springer Gabler, 2018.
- [2] Lückcrath, D.: Ein Simulationsmodell für Öffentlichen Personennahverkehr mit regelbasiertem Verkehrsmanagement. Dissertation, Universität zu Köln, 2017.
- [3] Lückcrath, D., Speckenmeyer, E., Ullrich, O., Rische, N.: A Mesoscopic Bus Transit Simulation Model Based on Scarce Data. In: *Simulation Notes Europe (SNE)*, Vol. 28, No. 1, 2018, to appear.
- [4] Nickel, K., Hasenbeck, F., Beyer, U., Ullrich, O., Zimmermann, A.: Assessing Organizational Complexity Using Tree-Attribute-Matrix Models. *Proceedings of 2019 IEEE 21st Conference on Business Informatics (CBI)*, 2019, pp. 124-129.
- [5] Sowa, J. F., "Top-level ontological categories". *Int. J. Human-Computer Studies*. 43(5-6), 1995, pp. 669-685.
- [6] Ullrich, O., Bogen, M., Lückcrath, D., Rome, E.: Co-operating with Municipal Partners on Indicator Identification and Data Acquisition. *Simulation Notes Europe (SNE)*, Volume 29, Number 4, 2019, pp. 159-168.
- [7] Vossloh Kiepe GmbH. *Elektrische Ausrüstung des Niederflur-Stadtbahnwagens K4000 der Kölner Verkehrs-Betriebe AG*. Druckschrift 00KV7DE, 2003.
- [8] Vossloh Kiepe GmbH. *Elektrische Ausrüstung der Hochflur-Stadtbahnwagen K5000 der Kölner Verkehrs-Betriebe AG*. http://www.vossloh-kiepe.com/vkproduktordner.2008-05-14.1154367607/vkproduktordner.2008-06-30.8585393121/vkproduktordner.2008-05-15.5609169940/vkprodukt.2008-06-04.1250026636/vkprodukt_download accessed on 24.05.2011.
- [9] Kölner Verkehrs-Betriebe AG. *Leistungsdaten 2020*. https://www.kvb.koeln/unternehmen/die_kvb/zahlen_daten_fakten/index.html accessed on 31.08.2020.

„Performance Evaluation of Timed Events in Railways“ in Österreich

Alexander Edthofer^{1*}, Martin Bicher^{2,3}, Felix Breiteneker¹

¹Institute of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Wien, Österreich; *alexander.edthofer@tuwien.ac.at

²dwh GmbH, Neustiftgasse 57-59, 1070 Wien, Österreich

³Institute of Information Systems Engineering, Vienna University of Technology, Favoritenstraße 8-11, 1040 Wien, Österreich

Abstract. Diese Arbeit befasst sich mit Methoden zur Verbesserung der Effizienz von Zugnetzen und wendet diese am Beispiel des österreichischen Güterzugverkehrs an. Ziel ist es, die Auslastung der Gleise zu analysieren und so den Fahrplan auf Realisierbarkeit zu untersuchen. Dies wird auf Güterzugfahrplandaten aus Österreich angewandt. Der theoretische Hintergrund ist die Max-Plus-Algebra, mit der ein Max-Plus Eigenwertproblem beschrieben werden kann. In dem Modell „PETER“ wird dabei die zu untersuchende Matrix erstellt.

Einleitung

In den letzten Jahrzehnten hat der Zugverkehr im Personen- wie auch im Gütertransport auf der ganzen Welt enorm zugenommen. Während 1998 noch 179.465.000 Menschen in Österreich die Bahn nutzten, waren es 2018 bereits 309.534.000 [7]. Güterzüge in unserem Land chauffierten 1998 noch 72.637.000 Tonnen, 2018 hingegen 105.271.000 Tonnen an Waren [7]. Da die Kapazität vieler Zugnetze bereits ausgeschöpft ist, müssen neue Möglichkeiten geschaffen werden, um die steigende Nachfrage weiterhin abdecken zu können.

Mit der Max-Plus Algebra lassen sich Modelle entwickeln, die insbesondere in Analyse und Optimierung von ereignisdiskreten Prozessen Anwendung finden, wie zum Beispiel bei Fahrplänen. In dieser Arbeit wird mit PETER ein solches Modell vorgestellt. PETER ist die Abkürzung für „Performance Evaluation of Timed Events in Railways“, das heißt, es ist zur Effizienzanalyse von Zugfahrplänen entwickelt worden. Am Beispiel des dänischen Zugnetzes des Personenverkehrs ist dies in [3] angewandt worden.

Diese Arbeit beschäftigt sich mit dem Fahrplan des österreichischen Güterzugverkehrs. Aus den gegebenen

Daten der Abfahrt und Ankunft lässt sich eine quadratische Matrix erstellen, wobei die Anzahl der Zeilen und Spalten definiert wird durch die Anzahl der befahrenen Stationen. Durch Eigenwertanalyse dieser Matrix wird so der Fahrplan analysiert und auf Realisierbarkeit überprüft. Das Ziel der Arbeit ist nun, den Anteil des österreichischen Güterzugverkehrs am gesamten Eisenbahnverkehr einzuschätzen, um zu überprüfen, ob dieser Verspätungen verursacht.

1 Max-Plus Algebra

Die mathematische Grundlage für das folgende Modell bietet die Max-Plus Algebra. Auf der Algebra sind die Operationen \oplus und \otimes für $a, b \in \mathbb{R}_{\max}$ folgendermaßen definiert, wobei \max und $+$ die herkömmlichen Operationen sind, welche aus den reellen Zahlen bekannt sind.

$$a \oplus b := \max\{a, b\}. \quad (1)$$

$$a \otimes b := a + b. \quad (2)$$

Die Operationen \oplus bzw. \otimes besitzen die neutralen Elemente $\varepsilon := -\infty$ bzw. 0. Weitere Eigenschaften der Max-Plus Algebra sind in [3, S. 13-20] nachzulesen.

Die Max-Plus Algebra lässt sich nun auch auf Vektoren und Matrizen in folgendem Sinn erweitern. Sei $\mathbb{R}_{\max}^{n \times m}$ die Menge aller $n \times m$ Matrizen mit Werten aus \mathbb{R}_{\max} . Die Operationen können analog übernommen werden.

$$[A \oplus B]_{ij} := a_{ij} \oplus b_{ij} = \max\{a_{ij}, b_{ij}\}.$$

Es wird also komponentenweise das Maximum der je-

weiligen Einträge berechnet.

$$[A \otimes B]_{ik} := \bigoplus_{j=1}^l a_{ij} \otimes b_{jk} = \max_{j \in \{1, \dots, l\}} \{a_{ij} + b_{jk}\}.$$

Diese Operationen besitzen nun wiederum neutrale Elemente. So ist eine $n \times m$ -Matrix, deren Werte alle gleich ε sind, neutral bezüglich \oplus und wird mit $\mathcal{E}(n, m)$ bezeichnet. Die Matrix $E(n, m)$, welche definiert ist als

$$[E(n, m)]_{ij} := \begin{cases} 0 & i = j, \\ \varepsilon & \text{sonst,} \end{cases}$$

bildet das neutrale Element bezüglich \otimes . Für $n = m$ wird $E(n, n)$ auch Einheitsmatrix genannt. Für weitere Eigenschaften wird wiederum auf [3, S. 13-20] verwiesen.

2 PETER - Performance Evaluation of Timed Events in Railways

Ein geplanter Fahrplan der Eisenbahn kann als discrete event dynamic system (DEDS) mittels der Max-Plus Algebra modelliert werden. Hierzu werden irreduzible Modelle erster Ordnung der Form $x(k) = A \otimes x(k-1) \oplus B \otimes u(k)$ betrachtet.

Um auch höhere Ordnungen, wie auch die nullte Ordnung, von Max-Plus linearen Systemen zu bearbeiten, wird im Folgenden eine „Polynom-Matrix-Repräsentation“ der Zustandsmatrix und der zugehörige zeitlich festgelegte Eventgraph vorgestellt. Dazu wird ein Shiftoperator γ und seine Erweiterung auf Matrizen $A(\gamma)$ eingeführt. Die Theorie stützt sich großteils auf [2, S. 179 - 194], einige Informationen wurden auch aus [3, S. 36 - 45] entnommen. Dieser Operator führt uns schließlich auf eine Beschreibung ähnlich zu der Max-Plus Theorie erster Ordnung. Dieser Zugang wird nun verwendet, um periodische Eisenbahnfahrpläne als geplante Max-Plus lineare Systeme zu modellieren, und sie in weiterer Folge zu analysieren, um vor allem die Realisierbarkeit zu untersuchen.

2.1 Aufbau des Modells

Periodische Fahrpläne der Eisenbahn werden durch die in regelmäßigen Intervallen befahrenen Gleisstrecken bestimmt. Die Periodenlänge wird im Folgenden mit T bezeichnet und beträgt üblicherweise 60 Minuten. In

dem Modell (DEDS) beschreibt ein Event i die Abfahrt eines Zuges der zugehörigen Linie L_i aus der Station S_i . Weiters wird mit $x_i(k)$ die k -te Abfahrtszeit von i bezeichnet, diese Zeit liegt also im Intervall $[k \cdot T, (k+1) \cdot T)$.

An $x_i(k)$ lassen sich nun einige Bedingungen stellen:

1. Fahrplanbedingung: Die Abfahrt darf nicht vor der geplanten Abfahrtszeit $d_i(k)$ erfolgen:

$$x_i(k) \geq d_i(k). \quad (3)$$

Da der Fahrplan eine gewisse Periodenlänge hat, gilt $d_i(k) = d_i(0) \otimes T^{\otimes k}$.

2. Linienbedingung: Die Abfahrt eines Zuges hängt von den vorhergehenden Events auf der Linie ab:

$$x_i(k) \geq a_{ij} + x_j(k - \mu_{ij}). \quad (4)$$

Dabei steht a_{ij} für die Summe der Fahrzeit eines Zuges von der Station S_j zur Station S_i und die minimale Wartezeit in der Zielstation. Die periodische Verschiebung zwischen den Abfahrten x_i und x_j wird von $\mu_{ij} \in \mathbb{N}_0$ beschrieben. Sie wird mit der Aufrundungsfunktion berechnet:

$$\mu_{ij} = \left\lceil \frac{a_{ij} + d_j^0 - d_i^0}{T} \right\rceil.$$

3. Synchronisationsbedingung: Weiterhin muss ein Zug oft auf andere warten. Berechnet wird dies analog zu (4), wobei a_{ij} sich nun aus der Fahrzeit des „Zulieferers“ und der Umstiegszeit in der Station S_i zusammensetzt.
4. Infrastrukturbedingung: Schlussendlich muss ein Zug noch warten, falls das zu benutzende Gleis vor ihm blockiert ist. Wiederum gilt (4), wobei a_{ij} nun für die Fahrzeit nach Abfahrt steht, bevor die blockierte Stelle für den Zug i freigegeben ist.

Zu beachten ist nun auch noch, dass $a_{ij} = a_{ij}(\mu_{ij})$ von μ_{ij} abhängt, da die vorhergehende Abfahrt eventuell noch in einer alten Periode stattgefunden hat. Wenn nun n die Anzahl der Abfahrts-events ist und für jedes Paar (j, i) , an das keine Bedingung geknüpft ist $a_{ij} = \varepsilon$ gesetzt wird, lassen sich die Bedingungen als Gleichung

$$x_i(k) = \bigoplus_{j=1}^n (a_{ij}(\mu_{ij}) \otimes x_j(k - \mu_{ij})) \oplus d_i(k), \quad (5)$$

$$\forall i = 1, \dots, n$$

formulieren. Die Werte a_{ij} lassen sich für gleiche periodische Verschiebungen zu Matrizen zusammenfassen: $A_l = (a_{ij}(l))$ wiederum mit $[A_l]_{ij} = \varepsilon$, falls keine Bedingung in der jeweiligen periodischen Verschiebung existiert. Weiters lässt sich selbiges mit $x_i(k)$ und $d_i(k)$ zu Vektoren machen. Die Ordnung des Systems wird im Folgenden mit $p \in \mathbb{N}$ bezeichnet. Damit kann nun (5) dargestellt werden:

$$\begin{aligned} x(k) &= \bigoplus_{l=0}^p (A_l \otimes x(k-l)) \oplus d(k), \\ d(k) &= d_0 \otimes T^{\otimes k}. \end{aligned} \quad (6)$$

Die Gleichung (6) beschreibt nun die allgemeine Zustandsraumdarstellung des geplanten Max-Plus linearen Systems mit periodischem Fahrplan, wobei $x(k)$ die wirkliche und $d(k)$ die geplante Abfahrtszeit ist.

Eine andere Formulierung von Max-Plus linearen Systemen höherer Ordnung kann mit einem „backward-shift“ γ erhalten werden. Dieser ist definiert als

$$\gamma x(k) = x(k-1) \quad \text{bzw.} \quad \gamma^l x(k) = x(k-l), \\ \gamma^0 x(k) = x(k).$$

Damit erhalten wir nun die alternative Beschreibung von (6)

$$x(k) = A(\gamma) \otimes x(k) \oplus d(k), \quad (7)$$

wobei $A(\gamma) = \bigoplus_{l=0}^p \gamma^l A_l$ eine polynomielle Matrix im Shift-Operator γ ist.

2.2 Eigenwertproblem

Analog zur reellen Spektralanalyse lässt sich auch in der Max-Plus Algebra ein Eigenwertproblem definieren.

Für eine Matrix $A \in \mathbb{R}_{\max}^{n \times n}$ heißt $\lambda \in \mathbb{R}_{\max}$ Eigenwert und $v \in \mathbb{R}_{\max} \setminus \{\varepsilon\}$ der zugehörige Eigenvektor, wenn

$$A \otimes v = \lambda \otimes v.$$

Das Tupel (λ, v) wird Eigenpaar und die Menge aller Eigenwerte Spektrum der Matrix A genannt. Ein Algorithmus, welcher diesen berechnet, ist beispielsweise der *Floyd-Warshall Algorithmus*, welcher in Abschnitt 3 vorgestellt wird.

Weiters lässt sich das Eigenwertproblem auf polynomielle Max-Plus Matrizen verallgemeinern.

Definition 2.1 Sei $A(\gamma) = \bigoplus_{l=0}^p \gamma^l A_l$ eine polynomielle quadratische Max-Plus Matrix. Ein Skalar $\lambda \in \mathbb{R}_{\max} \setminus \{\varepsilon\}$ heißt Eigenwert und $v \in \mathbb{R}_{\max}^n \setminus \{\varepsilon\}$ zugehöriger Eigenvektor \Leftrightarrow

$$A(\lambda^{\otimes -1}) \otimes v = v.$$

Eine homogenes Max-Plus lineares System $x(k) = A(\gamma) \otimes x(k)$ heißt *autonom*, wenn die polynomielle Zustandsmatrix $A(\gamma)$ keine Zeile besitzt, deren Werte alle gleich ε sind, also $\forall 1 \leq i \leq n \exists 1 \leq j \leq n : [A(\gamma)]_{ij} \neq \varepsilon$. Eine quadratische polynomielle Matrix heißt *irreduzibel*, wenn sie zu einem autonomen Max-Plus linearen System gehört.

Um Max-Plus Spektralanalyse zu vertiefen, werden nun zeitbehaftete Petri-Netze („timed petri nets“) vorgestellt. Dabei handelt es sich um einen Unterklasse von entscheidungsfreien Petri-Netzen mit nur einem Input beziehungsweise Output und ist ein DEDS. Das zeitbehaftete Petri-Netz ist ein Tupel $\mathcal{G} = (\mathcal{T}, \mathcal{P}, \mu, w)$, wobei \mathcal{T} die Menge der Übergänge, also $|\mathcal{T}| = n$, \mathcal{P} die Menge der Strecken mit $|\mathcal{P}| = m$ beschreibt. Die Variable $\mu \in \mathbb{N}_0^m$ ist eine Anfangsmarkierung, vergleichbar damit, wieviele Züge zu einem gewissen Startzeitpunkt verkehren bzw. verfügbar sind. Diese werden auch Token genannt. Mit $w \in (\mathbb{R}_+ \cup \varepsilon)^m$ wird der Vektor der Fahrzeiten der einzelnen Strecken bezeichnet. Die Events, also die Übergänge, werden mit Balken, Strecken mit Kreisen dargestellt.

Ein Max-Plus lineares System ist die Zustandsraumrealisierung des zeitbehafteten Petri-Netzes. Die Verteilung der Token beschreiben den Zustand des Systems und die Bewegung der jener das dynamische Verhalten. Diese Bewegung geschieht nach folgender Regel: Ein Übergang ist aktiviert, wenn jede Strecke dorthin einen Token enthält und deren Fahrzeit verstrichen ist. Danach „feuert“ dieser Übergang, indem jede zuführende Strecke einen Token verliert, und jede wegführende einen erhält.

Eine polynomielle Matrix $A(\gamma) = \bigoplus_{l=0}^p \gamma^l A_l$ entspricht nun solch einem zeitbehafteten Petri Netz \mathcal{G} , mit $\mathcal{T} = \{1, \dots, n\}$, für jeden Wert $[A_l]_{ij} \neq \varepsilon$ gibt es eine Strecke $(j, i) \in \mathcal{P}$ deren Fahrzeit $w_{ij} = [A_l]_{ij}$ ist. Die Anfangsmarkierung μ_{ij} beträgt l .

Weiters lässt sich nun der Begriff der Irreduzibilität auf die neue Darstellung anwenden: Die quadratische po-

lynomielle Matrix $A(\gamma)$ heißt *irreduzibel*, wenn der zugehörige Graph \mathcal{G} stark zusammenhängend ist. Solch eine Matrix hat nun einen einfachen verallgemeinerten Eigenwert, dem im Zusammenhang mit dem zeitbehafteten Petri-Netz eine besondere Rolle eingeräumt wird:

Satz 2.1 (Verallgemeinerter Eigenwert) Sei $A(\gamma) = \bigoplus_{l=0}^{\gamma} A_l$ eine irreduzible polynomielle Matrix mit einem gerichteten azyklischen Subgraphen. Dann hat $A(\gamma)$ einen einfachen verallgemeinerten Eigenwert $\lambda > \varepsilon$ und einen Eigenvektor $v > \varepsilon$, sodass $A(\lambda^{\otimes -1}) \otimes v = v$, und λ ist gleich der maximalen Zykluslänge des zugehörigen zeitbehafteten Petri-Netz $\mathcal{G}(A(\gamma))$,

$$\eta = \max_{\xi \in C} \frac{w(\xi)}{\mu(\xi)},$$

wobei C die Menge aller grundlegenden Kreisläufe in $\mathcal{G}(A(\gamma))$, $w(\xi)$ die Kosten des Kreislaufs ξ und $\mu(\xi)$ die Zahl der Token in dem Kreislauf beschreibt.

Der Beweis ist in [2, S. 187] zu finden. Die Bedingung des azyklischen Subgraphen ist notwendig, da sonst Kreisläufe auftreten können, die keinen Token enthalten, und daher nicht mehr feuern können, womit möglicherweise kein Übergang, der von dort erreicht wird, jemals wieder aktiviert wird (*festgefahren*). Ein *kritischer Kreislauf* hat maximale Zykluslänge.

Ein Eigenvektor kann dann als Anfangsfahrplan-Vektor $d_0 = v$ interpretiert werden, sodass das Max-plus lineare System eine Periodenlänge von $T = \lambda$ hat. Das Eigenwertproblem kann also auch folgendermaßen formuliert werden: Finde einen Wert λ , welcher die Zeiten der Wege v_i von einem kritischen Knoten zu jedem Knoten i minimiert (modulo λ).

3 Österreichischer Güterzugverkehr

Der Fahrplan des österreichischen Güterzugverkehrs ist sehr umfassend. Die Methode PETER wird nun auf diesen angewandt, welcher 1539803 Datensätze beinhaltet. Die Daten stehen wiederum im Rahmen des Projekts A&O mit den Österreichischen Bundesbahnen zur Verfügung. Der Fahrplan beinhaltet 40932 Züge, welche einerseits national, andererseits auch international verkehren.

Die Periode des untersuchten Fahrplans ist eine Woche, als Einheit sind Minuten gewählt. Damit ergibt sich für die Periode $T = 10080$ min. Es lässt sich nun

bereits erahnen, dass für diese Daten der Rechenaufwand um einiges länger ist, als bei der Analyse von Personenzügen, wie sie in [2] durchgeführt worden ist, bei der die Periode 60 Minuten ist, und außerdem pro Periodendurchgang viel weniger Züge zu untersuchen sind.

Für die Umsetzung der Methode PETER wird die Zugnummer, die Abfahrts- und Ankunftszeit, der Abfahrts- und Ankunftsort, die Anzahl der Zugzwischenstopps, wie auch etwaige Extrazeit aus dem Fahrplan benötigt. Letztere kommt bei Be- und Entladungen der Züge vor, oder aber auch wenn sie warten müssen, bis das Gleis vor ihnen frei wird. Um die Rechenzeit zu verkürzen, wird der Fahrplan soweit vereinfacht, dass nur mehr Start- und Endpunkt eines Zuges, sowie jene Informationen, wo eine Extrazeit stattfindet, gespeichert werden.

Von den gegebenen Daten werden nun zunächst die Anzahl der Züge bestimmt. Dann wird eine Matrix W erstellt, für die ein Eintrag w_{ij} das i -te wichtige Gleis des j -ten Zugs beschreibt. Diese Gleisstrecken werden in einen Vektor G abgespeichert. Abschließend werden noch die Abfahrtszeiten und Ankunftszeit am Endbahnhof an den Wartestellen in die Matrix Z geschrieben, deren Indizierung gleich der der Matrix W ist.

Nun sind die Vorbereitungen getroffen, um die zu analysierende Matrix zu erstellen. Zunächst wird eine quadratische Matrix $A \in \mathbb{R}_{max}^{n \times n}$, wobei n gleich der Anzahl der wichtigen Gleise ist, erstellt. Anschließend werden gemäß den Bedingungen (3) und (4) die Einträge der Matrix auf das Maximum der zu wartenden bzw. fahrenden Zeit gesetzt. Im Vergleich zum theoretischen Modell von Abschnitt 2 liegt ein Problem der Modellumsetzung darin, dass nun mehrere Züge im Laufe der relativ langen Periode ein Gleisstück befahren und dies teilweise mit unterschiedlichen Geschwindigkeiten. Durch die Wahl der maximalen Zeit, was der geringsten Geschwindigkeit entspricht, wird das Modell so eher auf Realisierbarkeit des Fahrplans überprüft.

Der Eigenwert der Matrix wird nun mit dem *Floyd-Warshall Algorithmus* berechnet. Er beträgt $\lambda_0 = 2882$ min. Der Algorithmus läuft dabei folgendermaßen ab.

Wie in [5] erläutert, wird zunächst eine erste untere Schranke μ für den Eigenwert gewählt. Ist ein Wert der Diagonale von A ungleich ε , so wird $\mu = \max_{i \in \underline{n}} a_{ii}$

gesetzt. Andernfalls ist $\mu = \min a_{ij}$ über alle $a_{ij} \neq \varepsilon$. In weiterer Folge wird überprüft, ob die Matrix $A - \mu J$, wobei $J \in \mathbb{R}^{n \times n}$ eine Matrix ist, deren Werte alle gleich 1 sind, Zyklen mit positiven Gewichten aufweist. Zyklen sind Pfade, deren Start- und Endpunkt derselbe ist. Dies wird untersucht, indem eine Folge von Matrizen A_k konstruiert wird. Die erste Matrix ist dabei $A_0 = A - \mu J$. Die weiteren Matrizen sind über

$$A_k(i, j) := \max\{A_{k-1}(i, j), A_{k-1}(i, k) + A_{k-1}(k, j)\}, \\ k = 1, \dots, n,$$

definiert. Ein Eintrag dieser Art kann als maximales Gewicht aller Pfade von j nach i der Länge $k + 1$ gesehen werden. Innerhalb dieser Pfade wird nun auf Zyklen geachtet, wobei jetzt nur mehr auf den Diagonalen der Matrizen A_k geachtet werden muss. Wenn kein Wert der Diagonalen größer 0 ist, haben wir bereits den maximalen Eigenwert gefunden mit $\lambda_0 = \mu$. Sollte es doch noch einen positiven Wert $(A_k)_{\tilde{i}\tilde{i}}$ an einer Diagonale geben, dann ist $\mu < \lambda_0$. Der zugehörige Zyklus, welcher von dem Knoten \tilde{i} ausgeht, wird dann zurückverfolgt und μ durch das durchschnittliche Pfadgewicht $\frac{\text{Summe der Pfadgewichte}}{\text{Länge des Zyklus}}$ ersetzt. Die Prozedur wird daraufhin mit dem aktualisierten Wert von μ durchgeführt.

Um nun zugehörige Eigenvektoren zu finden, müssen zunächst theoretische Überlegungen gemacht werden. Ein kritischer Zyklus ist, wie in Abschnitt 2.2 bereits erwähnt, ein Zyklus mit maximaler Länge, dessen Länge also gleich λ_0 ist. Eine notwendige Bedingung für die Existenz eines Eigenvektors ist in folgendem Satz zu finden (siehe [5, S. 236]).

Satz 3.1 Wenn ein Eigenvektor v des Problems $A \otimes v = \lambda_0 \otimes v$ mit $A \in \mathbb{R}^{n \times n}$ existiert, dann gibt es für alle $i \in \underline{n}$ Pfade im zur Matrix A gehörigen Graphen von einem Knoten eines kritischen Zyklus zu i .

Beweis. Das Eigenwertproblem $A \otimes v = \lambda_0 \otimes v$ lässt sich auch komponentenweise schreiben als

$$\max_{1 \leq j \leq n} (A_{ij} + v_j) = \lambda_0 + v_i, \quad 1 \leq i \leq n \\ \iff \max_{1 \leq j \leq n} (A_{ij} + v_j - v_i) = \lambda_0, \quad 1 \leq i \leq n. \quad (8)$$

Von der Gleichung (8) ausgehend ist nun bekannt, dass für jeden Knoten i mindestens ein j existiert, sodass diese erfüllt ist. Dieser Knoten sei mit \tilde{j}_1 bezeichnet. Wenn dies nun weitergeführt wird, so gibt es für \tilde{j}_1 wiederum ein \tilde{j}_2 , sodass (8) erfüllt ist. Iterativ entsteht so ein Pfad unendlicher Länge zu dem ursprünglichen

Knoten i . Da es aber nur n Knoten insgesamt gibt, muss ein Zyklus enthalten sein, und da für jeden Pfad die Gleichung (8) gilt, ist dieser ein kritischer Zyklus. ■

Wenn nun die Kontraposition dieses Satzes betrachtet wird, so gilt nun, dass, wenn ein Knoten \tilde{i} existiert, zu dem es keinen Pfad eines Knoten eines kritischen Zyklus gibt, dann gibt es keinen Eigenvektor. Dies ist in im gegebenen Beispiel des Güterzugmodells der Fall. Wenn nämlich Abbildung 1 betrachtet wird, fällt auf, dass darin drei Subsysteme vorhanden sind, was wiederum bedeutet, dass nicht alle befahrenen wichtigen Gleise miteinander verbunden sind. Der kritische Zyklus, welcher in dem linken Subsystem enthalten ist, erreicht zwar die meisten Stationen, jedoch eben nicht alle.

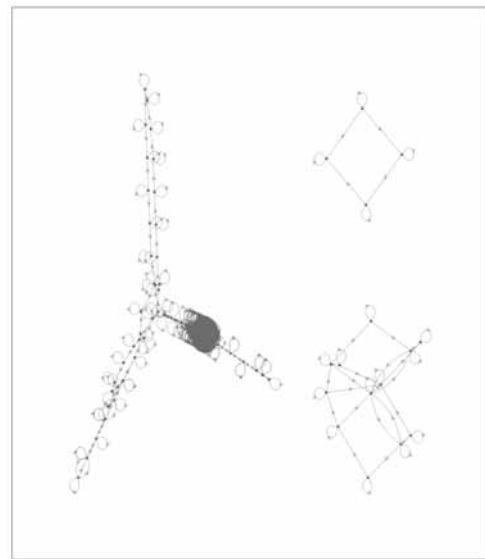


Abbildung 1: Der Güterzugfahrplan weist drei Subsysteme auf.

Was sagt uns der Eigenwert aber nun und in welchem Kontext muss er betrachtet werden? Zunächst einmal ist

zu bemerken, dass eine Woche 10080 Minuten hat. Dadurch wirkt $\lambda_0 = 2882$ min zunächst sehr klein. Hierzu muss jedoch bedacht werden, dass nur Güterzüge betrachtet. Personenzüge, welche auf den gleichen Gleisen unterwegs sind und insbesondere bei eventuellen Wartezeiten Vorrang haben, werden nicht untersucht, da hierzu keine Daten zu Verfügung standen. Dies ist ein entscheidender Faktor, da laut [4] Personenzüge Österreichs Gleise mit insgesamt 114,9 Millionen Kilometer viel mehr belegen. Güterzüge haben im Gegensatz dazu „nur“ 41,5 Millionen Kilometer zurückgelegt. Damit werden von den insgesamt gefahrenen 156,4 Millionen Kilometern 26,53% von Güterzügen zurückgelegt. Wenn nun der Eigenwert betrachtet wird, so wird 28,59% der Zeit von Güterzügen beansprucht.

Nun wirkt es zunächst so, als wäre das System instabil, da Güterzüge mehr Kilometer auf den Gleisen verbringen, als Personenzüge. Da sie durchschnittlich mit 90 bis 120 km/h zwar gleich schnell, wie Personennahverkehrszüge unterwegs sind, jedoch Personenfernverkehrszüge eine um einiges höhere Geschwindigkeit haben, ist die Zeit, welche Güterzüge auf den Gleisen verbringen, auch vergleichsweise länger (siehe [6, S.442-449]). Weiters besitzen manche Gleisabschnitte mehrere Gleise nebeneinander, sodass mehrere Züge zur selben Zeit in die gleiche Richtung fahren können, ohne sich zu blockieren, was mehr Ressourcen schafft. Diese sind jedoch nicht in dem angegebenen Modell eingebunden. Damit relativiert sich der Unterschied zwischen den Prozentsätzen und das Ergebnis wirkt plausibel, der Fahrplan also realisierbar.

4 Fazit

In dieser Arbeit wurden die Max-Plus Algebra und darauf aufbauend ein Modell zur Zugfahrplananalyse und -optimierung vorgestellt. Insbesondere wird die Kapazitätsbelastung der Gleisnetze bewertet.

Das Modell Performance Evaluation of Timed Events in Railways (PETER) verkörpert einen graphentheoretischen Ansatz in Kombination mit einem Max-Plus Eigenwertproblem. Die Vorteile liegen nun darin, dass Nebenbedingungen wie Wartungen, Wenden, Koppeln, usw. kein extriges Modellieren benötigen, sondern die Züge auf dem momentanen Gleis länger verharren. Was bei diesem Modell im Anwendungsfall aufgefallen ist, ist, dass bei der Analyse des Güterzugfahrplans keine polynomielle Matrix, wie sie in Abschnitt 2 eingeführt wurde, benötigt wird, da durch die lange Peri-

odendauer von einer Woche die Züge maximal mit jenen der nächsten Woche in Kontakt kommen, wodurch ein DEDS von erster bzw. nullter Ordnung entsteht.

Das Modell lässt sich nun noch in ein paar Aspekten erweitern. So lässt sich der Fahrplan weitergehend auf Stabilität und Robustheit untersuchen. Stabilität bedeutet dabei, ob ein Fahrplan Verspätungen produziert, und Robustheit, ob diese wieder kompensiert werden können. Damit wird die Effizienz des Fahrplans getestet. Durch den Wegfall des Max-Plus-Eigenvektors konnte dies in dem beschriebenen Beispiel aber nicht angewandt werden. Für weitergehende Analyse würden auch noch Daten zur Geschwindigkeit von Güter- und Personenzügen benötigt werden, um den berechneten maximalen Eigenwert mit dem tatsächlichen zeitlichen Anteil von Güterzügen auf den Gleisen in Verbindung zu bringen.

Referenzen

- [1] Borndörfer R., Klug T., Lamorgese L., Mannino C., Reuther M., Schlechte Th.: *Handbook of Optimization in the Railway Industry*. Bd. 268. Springer International Publishing; 2018.
- [2] Goverde R.M.: Railway timetable stability analysis using max-plus system theory. In: *Transportation Research Part B: Methodological*. Elsevier; 2007; 2: p 179–201.
- [3] Heidergott B., Olsder G.J., van der Woude J.: *Max Plus at Work: Modeling and Analysis of Synchronized Systems: A Course on Max-Plus Algebra and Its Applications*. Bd. 48. Princeton University Press; 2014.
- [4] OEBB-Infrastruktur-AG: *Zahlen, Daten, Fakten*. <https://infrastruktur.oebb.at/de/unternehmen/zahlen-daten-fakten/zahlen-daten-fakten-oebb-infrastruktur.pdf>. Letzter Zugriff am 7. Mai 2020.
- [5] Olsder G.J., Roos K., Egmond R.-J.: An efficient algorithm for critical circuits and finite eigenvectors in the max-plus algebra. In: *Linear Algebra and its Applications*. Elsevier; 1999; 295: p 231–240.
- [6] Röhl F.V.v.: *Enzyklopädie des Eisenbahnwesens : Vierter Band*. Vero Verlag; 2019.
- [7] United Nations Economic Commission for Europe: *UNECE Statistical Database - Transport - Railway Traffic*. https://w3.unece.org/PXWeb2015/pxweb/en/STAT/STAT_40-TRTRANS_05-TRRAIL/. Letzter Zugriff am 25. Jänner 2020.

Creating Cloud Simulations for Urban Logistics

Richard Pump^{1*}, Charline von Perbandt¹, Volker Ahlers¹, Arne Koschel¹

¹University of Applied Sciences and Arts, Hannover, Ricklinger Stadtweg 120,
30459 Hannover; *richard.pump@hs-hannover.de

Abstract. In the project USEFUL, simulations are used for the evaluation of novel logistics concepts, since real-world evaluations are very cost intensive. However, no simulation tool can provide a holistic solution to high-detail small-scale solution and low-detail large-scale simulation at the same time. To increase simulation capabilities, we enhanced two simulation tools, AnyLogic and MATSim, to run in lock step. This provides a single solution for different simulation needs. Furthermore, a job-based simulation cloud was designed to reduce simulation costs while preserving data security.

Simulation for urban logistics

Currently urban logistics are changing, novel concepts taking hold in cities to reduce emissions and improve efficiency. At the University of Applied Sciences and Arts Hannover, we are working with the city of Hannover and other institutions to explore the effects of novel logistics concepts on the current urban mobility. Within the project USEFUL [1], the project team evaluates multiple novel logistics concepts and presents the results in an easily understandable way to support decision making. Simulation models are used to reduce costs and avoid investing in concepts with adverse effects on the environment or the quality of life of residents. Two tools are used for the simulations, AnyLogic [2, 3] and MATSim [4], to simulate microscopic and macroscopic behavior respectively.

AnyLogic is a commercially available multi-method simulation tool using a graphical modeling language in combination with snippets of java code. While allowing fine-grained modeling of individual agents, AnyLogic does not scale easily to city-scale populations.

MATSim is a simulation framework purpose built for the simulation of large logistical scenarios, supporting a large amount of agents traversing a predefined road network. Published as an open-source Java library, MATSim can be easily extended with new mod-

ules, providing modeling support for e.g. trains. Within MATSim, agent behaviour is modeled through agent plans, containing a sequence of activities at certain locations and the routes the agents use to move from activity to activity.

For the evaluation, research areas within the city of Hannover were defined. The research areas coincide with four districts of Hannover, which were selected as representation of the city as a whole. Selecting smaller parts of the city allows fine-grained simulation within the research area and reduces administrative burden if a certain concept is to be tested in practice.

Within the project the following workflow has been established: First a baseline simulation is created, modeling the current state of traffic within Hannover. The baseline provides a current standard against which new logistics concepts can be evaluated. In the next step, changes in agent behaviour deriving from the logistical concept (e.g. online grocery shopping) are modeled within AnyLogic and simulated to evaluate the effects on activities and routes within the small-scale research areas [5]. Changes in activities and routes are then fed back into MATSim to evaluate the effects of the logistical concept on the entire city. Finally, all data is collected and evaluated to estimate the ecological and economic benefits and disadvantages of the concept.

The simulations are run on office laptops and other consumer-grade hardware, as mobile working is often necessary in a cooperative project without a centralized office location. A cost-efficient service is therefore needed to provide easily usable computational resources for simulation.

In the following we present our work on combining MATSim and AnyLogic into a joint simulation tool and constructing an automated platform to run simulations in a cost-efficient, privacy-conserving way. Section 1 will provide an overview of related works concerning the combination of AnyLogic and MATSim as well as cloud-based simulation solutions. In Section 2 we will describe the designed solution to run AnyLogic and MATSim in lock step, while Section 3 presents the

job-based simulation cloud. A conclusion is provided in Section 4.

1 Related Work

Co-simulation and bringing simulation to the cloud have been extensively researched in the past.

Gütlein et al. [6] present a co-simulation framework for MATSim and SUMO, both traffic-oriented simulation tools. In a similar setup, the authors use SUMO as microscopic traffic simulation tool and embed the SUMO-based microscopic simulation into a macroscopic simulation run in MATSim. However, SUMO does not support multiple simulation modeling principles and is not as accessible as AnyLogic.

Further work on co-simulation between traffic-oriented simulation tools was done by Kathes et al. [7]. In the publication the authors describe the combination of SUMO and DYNA4's virtual vehicle as test bed for certification of automated and connected driving. The proposed system concentrates on the fine-grained simulation of a single car's systems on small-scale traffic scenarios instead of the simulation of whole cities.

While all these works provide some guidelines and follow similar approaches, the combination of MATSim and AnyLogic has not been extensively researched. However exhaustive works have been published on the combination of different modeling and simulation approaches [8].

A flexible simulation framework for cloud computing is presented by Filho et al. in [9]. The presented framework CloudSim Plus is an improvement over an existing cloud simulation framework. The authors focus on the usage of software engineering principles to create a flexible, extensible framework, instead of a concrete application to execute simulations.

Fujimoto et al. [10] view cloud computing as a chance to provide non-computer-science users an easy access to efficient resource usage on distributed computer hardware. The authors present Aurora, a cloud platform that uses a master/worker system to divide the work needed to execute a parallel discrete event simulation. While the described system is of similar intent to our design, the authors concentrate on the technical execution of the simulation instead of a broader view. Security and privacy are not the focus of the design.

Another shift of simulations into the cloud to reduce simulation costs is presented by Wang et al. in [11]. The authors describe a very low-level approach

to the distributed execution of Monte-Carlo simulations within a cloud environment. While the work shares the goal of reducing simulation cost, neither privacy nor traffic-simulation is considered.

Zehe et al. [12] present a cloud-based simulation service for large-scale urban systems. Instead of importing off-the-shelf simulation tools, the authors construct SEMSim as a purely cloud-based alternative providing fine-grained agent based simulation for urban traffic. This however results in researchers having to learn another tool instead of continuing the usage of known tools, which also often provide extensive libraries.

Within the different cloud-systems for simulation, privacy is often mentioned but not design-centric. The reduction of cost and decreasing simulation runtime are often main goals. Our approach focuses on cost-savings and privacy, as the data used in the simulations is partially non-public.

2 Combining MATSim and AnyLogic

As previously described the project USEfUL uses MATSim for macroscopic traffic simulation and AnyLogic for the fine-grained microscopic simulation of inhabitants behaviour when exposed to novel logistics concepts.

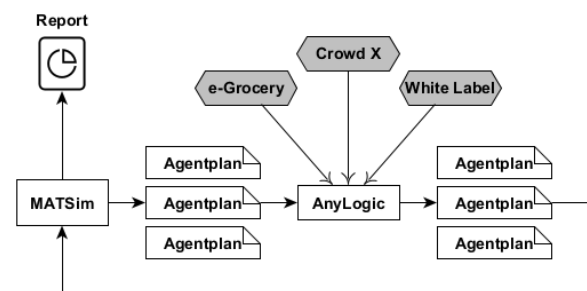


Figure 1: Workflow utilizing MATSim and AnyLogic.

Currently, simulation data is exchanged between AnyLogic and MATSim twice as shown in Fig. 1. First, MATSim simulates the current behaviour of the population of Hannover, creating plans for each agent to follow. Then, the plans are fed into the small-scale simulation of a logistical concept within a research area. This results in changes within the agent plans, which are in

turn integrated into the city-scale simulation in MATSim. A second simulation of the city is run to study the city-wide effects of introducing a concept into a research area.

This process does not allow for interactions between city-scale and research area-scale simulation, neglecting feedback between the different areas. The task thus is to combine both tools to work in tandem, running both simulations at the same time.

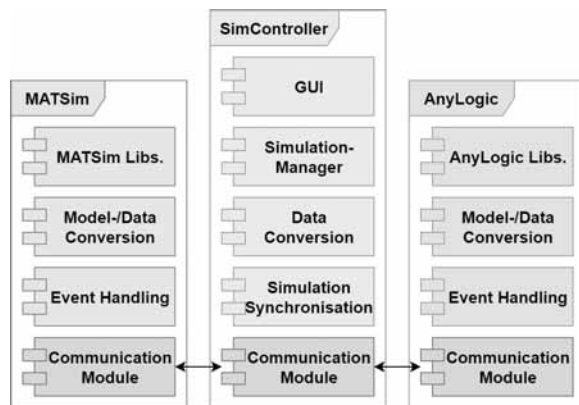


Figure 2: Architecture of the combined simulation.

Figure 2 shows the architecture of the combined simulation solution. MATSim and AnyLogic have been extended to accommodate inter-tool-communication by introducing special communication modules interacting with a third component. The SimController has been introduced to coordinate the simulation between MATSim and AnyLogic. This controller provides a graphical user interface to start and stop the combined simulation, controls the simulation execution by providing a single synchronizing point for both tools, and converts data between tool-specific formats.

To combine both simulations, a geographical split was utilized. The research area, simulated in fine detail with AnyLogic, was cut out from MATSim, as shown in Fig. 3. Agents that cross into the area are transferred with their plans (including a route to follow on the shared network) from MATSim to AnyLogic and vice versa. Furthermore, the simulations are sliced into fixed time units used as synchronization points. This allows for both tools to run in lock step as a single combined simulation.

For the communication between components remote procedure calls are utilized, which enables the distributed execution of simulation tools and coordinating components. By enabling a distributed execution, the

optimal hardware for the simulation can be chosen to fine-tune performance. Existing hardware in different geographical locations can be combined, reducing the cost of integrating the new system into existing projects.

The system has been shown to work with small-scale examples just including agent movement between the areas. Further work now focuses on building complex scenarios containing an entire city and complex behaviour, to test if the strengths of both tools have been successfully combined. Especially the performance of the combined solution will be evaluated. Simulations within the larger project have already shown run times of multiple days and a moderate (within a server context) usage of resources (e.g. 8 threads using 12 GB RAM), which prohibits running the simulations on standard office laptops normally used by researchers.

3 Creating a job-based simulation cloud

When large amounts of computing resources are needed, execution of software is usually shifted to server hardware. Often institutes have special simulation servers or a shared access to large server infrastructure. This however requires a lot of resources in the form of space and server administrators, restricting research budgets.

To circumvent acquisition and maintenance of large server infrastructures cloud computing has become increasingly popular. Therefore, a job-based simulation cloud was considered. This solution follows the idea of a mainframe, allowing users to submit jobs which are then executed on powerful hardware. Instead of executing different kinds of programs, only user-configurable predefined simulation models are executed.

3.1 Requirements

The goal of the job-based simulation cloud is however not exclusively saving costs. The system should also provide an easy-to-use interface to enable non-computer-science researchers to utilize the tool in their day to day research. Therefore, all of the tools functionalities should be accessible via a graphical user interface, which allows a user to create, edit, save and run simulation configurations.

The tool needs to provide configurable simulations that use standardized inputs, so researchers can configure scenarios (e.g. the usage of online grocery shop-

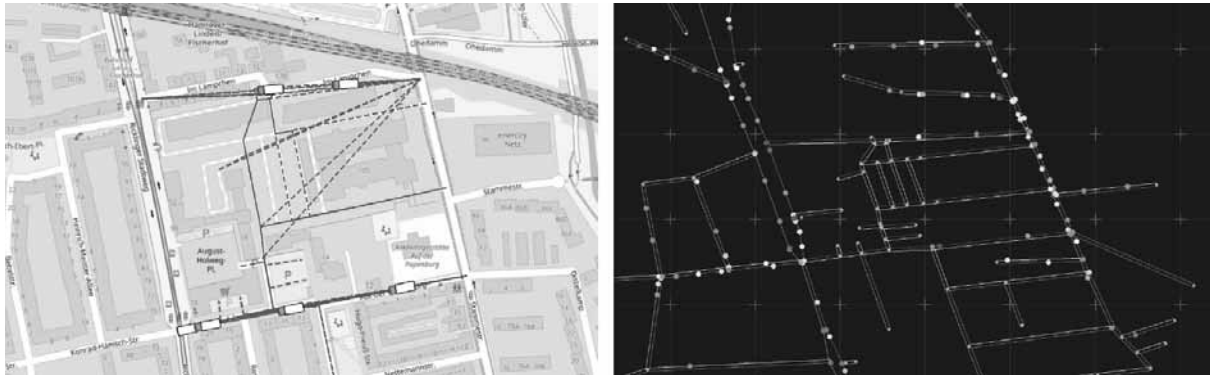


Figure 3: Left: Simulated area in AnyLogic. Right: Simulated area shown MATSim.

ping) for their specific needs. Depending on the simulation tool, this can be achieved via configuration files or an online value editor.

Furthermore, the simulations need to be organized and executed by the tool. Each simulation run can be seen as a single job, which is added to a queue and then executed when processing resources are available.

After the simulation run has concluded, the user needs to be able to view the results and compare them to other simulation runs of the same scenario. In case the simulation run has to be executed a second time, the complete configuration needs to be saved by the system. This ensures the ability to replicate simulations by other researchers, as configurations can easily be shared.

To provide this functionality, the system needs to manage user accounts, providing functionality to create, delete and change user details. The users need to be authenticated by password and grouped by research projects, to facilitate sharing of results within a project. Users should be able to use the system on all devices, allowing them to check the status and results of their simulations while on travel.

While cost-savings are a main driver of the system development, the systems stability and performance are paramount. If the tool can not reliably execute simulations in a timely manner, acceptance of the system will be very low. Furthermore, scalability is an issue, since multiple users can quickly demand large amounts of resources when parameter studies are needed.

Within the context of urban simulation, privacy is a major concern. The utilized simulation models are often based on non-public data (as shown in [15]), requiring a higher degree of security, often only provided by self-hosted solutions.

3.2 Cloud computing models

When switching from classic server based computing to a cloud service, first the appropriate cloud computing model needs to be chosen. Within cloud computing multiple different models have evolved. Most commonly known are the public cloud, the private cloud and the hybrid cloud [13], shown in Fig. 4. Each cloud model has advantages and disadvantages that need to be considered.

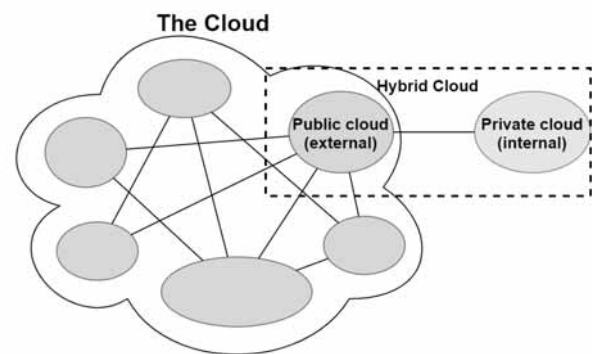


Figure 4: Overview of different cloud models [14].

Within a public cloud, all computation is done on shared hardware, owned by a service provider. This model often has large cost-saving potential for the clients, as no own hardware or support team is necessary. Furthermore, only utilized hardware is billed by the provider, reducing the cost of less computation intensive periods. However, a public cloud is also a very attractive target for an attacker, as access to the system also provides access to the client's system. With multiple users configuring their own systems in the cloud,

the likelihood of insecure configuration rises, potentially compromising the security of other users within the same cloud environment.

In a private cloud however, only a single client has access to the cloud infrastructure. This solution is comparable to a company owned mainframe or an institute's simulation server and is often the most expensive cloud model. However, a private cloud also provides the highest level of security, since only a single client's applications are executed on the hardware.

As a compromise between the public and the private cloud, the hybrid cloud was created. Within a hybrid cloud, some applications are run on private cloud-like infrastructure, while other applications are executed within the public cloud environment. By limiting the connections between public and private parts, security engineering becomes easier.

For the job-based simulation cloud a hybrid cloud approach was chosen to reduce costs and provide a secure platform. Long-term data storage is provided by the private part of the cloud, accessible only through clearly defined interfaces as to minimize the attack surface. User interface and computation power are provided by the public cloud, where data is only temporarily stored in memory while simulations are executed. By encrypting all communications between the public and the private part of the solution, confidentiality is assured while data is in transit. This split provides the best trade off between cost, security and usability.

3.3 Architecture

Figure 5 shows the proposed architecture of the job-based simulation cloud. Clients can access the Frontend through the internet and are authenticated via password based login. After configuring a simulation scenario through the parameter control, the user's simulation request is added to the simulation queue. The simulations are then automatically executed and the results logged in a database for later access and analysis. Users can access their results through the result analysis.

Utilizing the hybrid cloud approach, Frontend and Simulation components are executed within the public part of the cloud, User and Simulation Storage are secured within the private part. The communication between the different components is secured by encryption and authentication of the participating servers. The developed architecture is currently designed for the manual process of executing simulations one after another, feeding the results of the tools into each other.

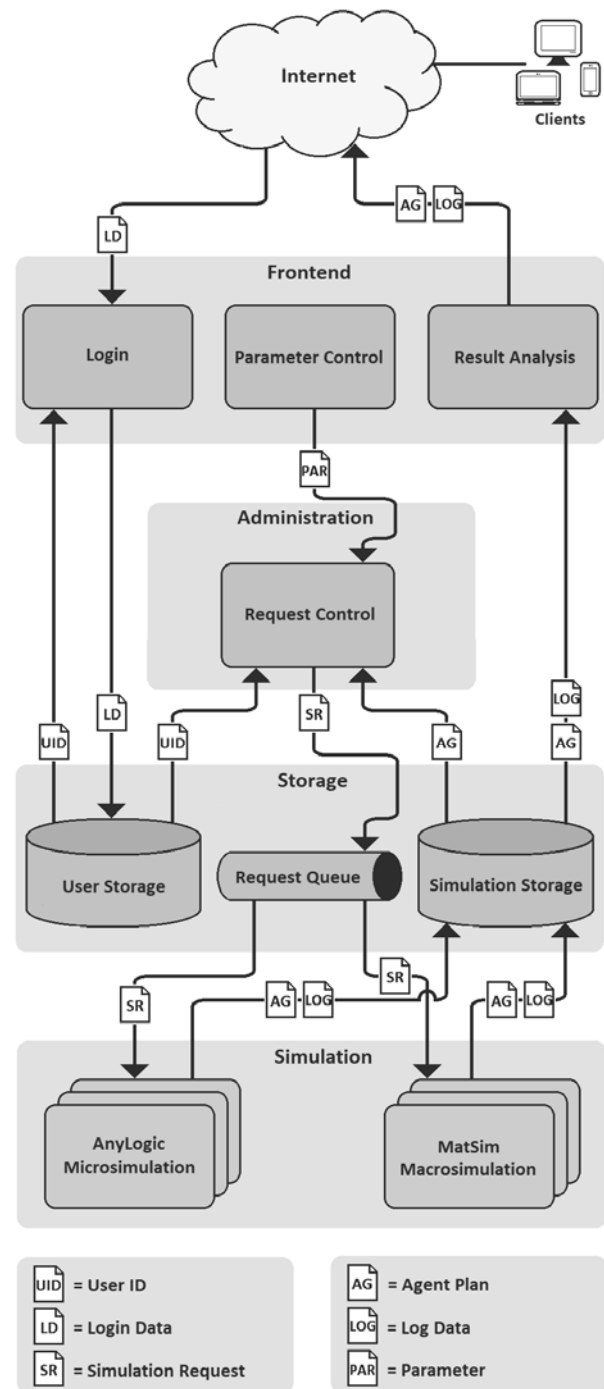


Figure 5: Design of the job-based simulation cloud.

As a proof of concept, the next step will be a first implementation and deployment to explore the performance of the solution.

4 Conclusion

In this paper we gave an excerpt of two subprojects within the context of the research project USEfUL, which aim to increase productivity in research concerning urban logistic simulation.

By combining the two simulation tools used within the research project, we increase the flexibility in modeling urban traffic and population behaviour. The tools are run in lock step by embedding a highly detailed, small-scale simulation within a low-detail, large-scale simulation, exchanging agents on predefined points in the network.

The second project aims to provide an easy-to-use platform for non-computer-science researchers. By utilizing a hybrid cloud approach, security requirements can be met while reducing the client's costs. The developed architecture reduces the technical burden for domain experts.

In further work, we plan to adapt the combined simulation to the developed job-based simulation cloud and create an all-in-one solution for the usage in projects building on the tools and workflows created in the project USEfUL.

Acknowledgement

This work was supported by the Federal Ministry of Education and Research of Germany (project USEfUL, grant no. 03SF0547). We thank our colleagues from the Departments of Mechanical Engineering and Business Information Systems, the colleagues from the City of Hannover and the other institutions collaborating within the research project USEfUL, as well as the participants of the student project CoSim.

References

- [1] *Urbane Logistik Hannover (urban logistics Hannover)*. Retrieved 24.06.2020. [Online]. Available: www.hannover.de/Urbane-Logistik-Hannoverp.
- [2] Grigoryev, I. *AnyLogic 6 in three days: a quick course in simulation modeling*. Any-Logic North America, 2012.
- [3] Borshchev, A. *The big book of simulation modeling: multimethod modeling with AnyLogic 6*. AnyLogic North America Chicago, 2013.
- [4] Horni, A., Nagel, K., Axhausen, K., editors. *Multi-Agent Transport Simulation MATSim*. Ubiquity Press, London, Aug 2016.
- [5] Auf der Landwehr, M., Trott, M., von Viebahn, C. *E-Grocery in Terms of Sustainability – Simulating the Environmental Impact of Grocery Shopping for an Urban Area in Hanover*. Simulation in Produktion und Logistik. 2019.
- [6] Gütlein, M., German, R., Djanatljev, A. *Towards a hybrid co-simulation framework: HLA-based coupling of MATSIM and SUMO*. 2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT). 2018.
- [7] Kathes, J., Schott, F., Chucholowski, F. *Co-simulation of the virtual vehicle in virtual traffic considering tactical driver decisions..* SUMO UserConference. 2019.
- [8] Gomes, C., Thule, C., Broman, D., Larsen, P., Vangheluwe, H. *Co-simulation: a survey*. ACM Computing Surveys (CSUR). 2018.
- [9] Silva Filho, M., Oliveira, R., Monteiro, C., Inácio, P., Freire, M. *CloudSim plus: a cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness*. 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). 2017.
- [10] Fujimoto, R., Malik, A., Park, A. *Parallel and distributed simulation in the cloud*. SCS M&S Magazine. 2010.
- [11] Wang, H., Ma, Y., Pratz, G., Xing, L. *Toward real-time Monte Carlo simulation using a commercial cloud computing infrastructure*. Physics in Medicine & Biology. 2011.
- [12] Zehe, D., Knoll, A., Cai, W., Aydt, H. *SEMSim Cloud Service: Large-scale urban systems simulation in the cloud*. Simulation Modelling Practice and Theory. 2015.
- [13] Jin, H., Ibrahim, S., Bell, T., Gao, W., Huang, D., Wu, S. *Cloud types and services*. Handbook of Cloud Computing. Springer. 2010.
- [14] Suppan, J. *Cloud Liefermodelle (Cloud delivery models)*. Retrieved 24.06.2020. [Online]. Available: www.comconsult-research.de/cloud-liefermodelle/
- [15] Bienzeisler, L., Lelke, T., Wage, O., Thiel, F., Friedrich, B. *Development of an agent-based transport model for the city of Hanover using empirical mobility data and data fusion*. Transportation Research Procedia 47, 99–106. 2020.

Simulation und Bewertung unterschiedlicher Boarding-Strategien am Beispiel des Airbus A320

Jürgen Wunderlich^{1*}

¹Fakultät Informatik, Hochschule für angewandte Wissenschaften Landshut, Am Lurzenhof 1, 84036 Landshut, Deutschland; *juergen.wunderlich@haw-landshut.de

Zusammenfassung. Der vorliegende Beitrag vergleicht auf Basis eines Simulationsmodells des Airbus A320 die Boarding-Zeiten bei Anwendung des Random-Boarding sowie der Boarding-Strategien Back-to-Front Boarding, Outside-In Boarding und einer Kombination von Back-to-Front und Outside-In Boarding. Dabei wird deutlich, dass sich einerseits durch das Outside-In Boarding eine Verkürzung der Boarding-Zeiten von über 12% realisieren lässt, aber andererseits hierfür auch eine hohe Disziplin erforderlich ist.

Einleitung

Boarding-Strategien sind immer wieder Gegenstand interessanter Diskussionen. So entwickeln sowohl Reisende als auch Fluggesellschaften und Flughäfen regelmäßig Ideen zur Verbesserung.

Umgesetzt wurden diese Ideen vor der Pandemie kaum. Eventuell ändern aber aktuelle Herausforderungen sowie ein schlankes Simulationsmodell diese Haltung.

1 Motivation

In Zeiten von Corona haben viele Fluggesellschaften den Boarding-Prozess angepasst, um den Mindestabstand sicherzustellen [3]. Das stellt nun eine Steilvorlage dar, generell über andere Boarding-Strategien als das bisher am häufigsten praktizierte Verfahren [2] – nämlich die Passagiere gruppenweise in die Maschine zu lassen und dabei, nachdem First-Class- und andere Vorzugsgäste eingestiegen sind, mit den hinteren Reihen anzufangen – nachzudenken.

Hinzu kommt, dass das Boarding einen Teilabschnitt des Turnaround-Prozesses darstellt. Dieser bezeichnet die Abfertigung eines Flugzeugs zwischen Landung und Start und sollte so schnell bzw. effizient wie möglich ablaufen, was umso wichtiger wird, je stärker der Flugverkehr wieder zunimmt. Denn bestimmte Aktivitäten, wie z.B. die Anweisungen zur Sicherheit durch die Flugbegleiter, können erst begonnen werden, wenn sich alle Passagiere an Bord befinden. Insofern gilt es, mehrere Kriterien und deren Wechselwirkungen zu

beachten, wofür sich die Ablaufsimulation als praktisches Experimentierfeld eignet.

2 Zielsetzung

Das finale Ziel der Untersuchung stellt die Verbesserung des Ablaufs und der Effizienz des Boarding-Prozesses durch die Auswahl einer geeigneten Boarding-Methode dar. Die Grundidee hierbei ist, durch eine Variation der Reihenfolge der einsteigenden Passagiere, Stauungen im Gang möglichst zu vermeiden, was zu einer Beschleunigung des Boardings und somit zu einer Verringerung der benötigten Boarding-Zeit bzw. der Turnaround-Prozesszeit führen soll.

Da der Turnaround-Prozess für die meisten Flugzeugtypen ähnlich von statten geht und sich lediglich der Ablauf und die Dauer einzelner Subprozesse unterscheiden, wird weiterhin angestrebt, die Studie so aufzubauen, dass die Kernaussagen leicht auf andere Flugzeugtypen übertragbar sind. Hierfür erfolgt zunächst die Definition eines realitätsgetreuen Referenzsystems, auf dessen Grundlage schließlich die Erstellung des Simulationsmodells sowie die Bewertung der Vor- und Nachteile der untersuchten Boarding-Strategien stattfindet.

3 Referenzsystem

Mit mehr als 14.000 verkauften Flugzeugen ist die A320-Familie der größte Erfolg von Airbus [4], weshalb dieser Flugzeugtyp für die Simulation gewählt wird. Dessen Sitzplätze werden in Business und Economy Class aufgeteilt und in einer Konfiguration von 154 Sitzen mit 30 Sitzreihen, wovon 2 Sitzreihen nicht für Passagiere vorgesehen sind, angeordnet. Davon befinden sich 28 Sitze mit den ersten 7 Reihen in der Business Class und 126 Sitze mit den Reihen 8-30 in der Economy Class [5]. Bei den Passagieren wird zwischen Geschäftsreisenden und Touristen bzw. danach unterschieden, ob sie im Gepäckfach zu verstauendes Handgepäck mitführen.

Weiterhin geht das Referenzsystem und in der Folge auch die Simulation von acht Annahmen aus:

- das Boarding beginnt mit dem Aufruf der Passagiere, wobei sich zu diesem Zeitpunkt sowohl alle eingetragenen Passagiere als auch das Flugzeug bereits am Gate befinden
- jeder Passagier ist bereits einem festen Sitzplatz zugewiesen, d.h. es erfolgt keine freie Sitzplatzwahl
- die Bordkartenkontrolle erfolgt durch das Flughafenpersonal und stellt die Einhaltung der Aufrufreihenfolge sicher
- die Passagiere betreten das Flugzeug in der Aufrufreihenfolge durch die vordere Flugzeugtür über eine Fluggastbrücke
- durch den Flugzeugtyp A320 ist ein Single-Aisle festgelegt, was bedeutet, dass im Flugzeug nur ein Gang zum Passieren zur Verfügung steht
- im Flugzeug verhalten sich die Passagiere höflich und überholen sich nicht
- die Passagiere sind im Besitz von einem oder keinem Handgepäckstück, das bereits die richtigen Maße aufweist
- es ist genügend Stauraum für das Handgepäck eines jeden Passagiers vorhanden, so dass eintretende Passagiere dieses ohne kapazitätsbedingte Zeitverzögerungen verstauen können

Als einfachste Boarding-Methode wird das Random-Boarding angewandt, das beispielsweise die Airlines Lufthansa und Eurowings am Flughafen München praktizieren. Bei der Random Boarding-Methode haben alle Passagiere einen fest gebuchten Sitzplatz, können das Flugzeug aber ohne Vorgaben in zufälliger Reihenfolge betreten. Lediglich die einzelnen Buchungsklassen (Zone 1 für Business und Zone 2 für Economy Class) werden nacheinander geboardet.

Diese Methode besticht grundsätzlich durch ihre Einfachheit. Ein zusätzlicher Vorteil liegt in der verteilten Auslastung des Flugzeuggangs. Es drängen nicht die ganze Zeit Passagiere in die gleichen Reihen bzw. an die selben Gepäckfächer. Bei dieser Boarding-Methode kommt es zwar auch zu Staus – z.B. weil immer wieder Personen aufstehen müssen, um andere durchzulassen –, aber diese verteilen sich wenigstens über den gesamten Flugzeuggang. Nachteil dieser Methode ist, dass keine Einflussnahme auf die Boarding-Reihenfolge stattfinden kann.

Der Boarding-Prozess selbst beginnt mit der Bordkartenkontrolle. Danach durchlaufen die Passagiere

den Pufferärmel, bevor sie an die erste Gangreihe gelangen. Dort prüfen sie, ob sich ihr Sitzplatz darin befindet. Ist das der Fall, wird – sofern vorhanden – zunächst das Handgepäck verstaut. Im Anschluss daran erfolgt die Einnahme des Sitzplatzes. Die hierfür benötigte Dauer hängt davon ab, wo genau sich der Sitzplatz befindet und wie viele Plätze vor dem zugeordneten Sitzplatz bereits belegt sind. Solange ein Fluggast noch nicht seine Zielreihe erreicht hat, läuft er eine Gangreihe weiter. Der genaue Ablauf ist in Abb. 1 dargestellt. Bei einem Business Class Passagier ändert bzw. verkürzt sich im Vergleich zu einem Economy Class Passagier lediglich die Zeit für die Sitzplatzeinnahme, da es in der Business Class nur zwei Sitze auf jeder Seite gibt.

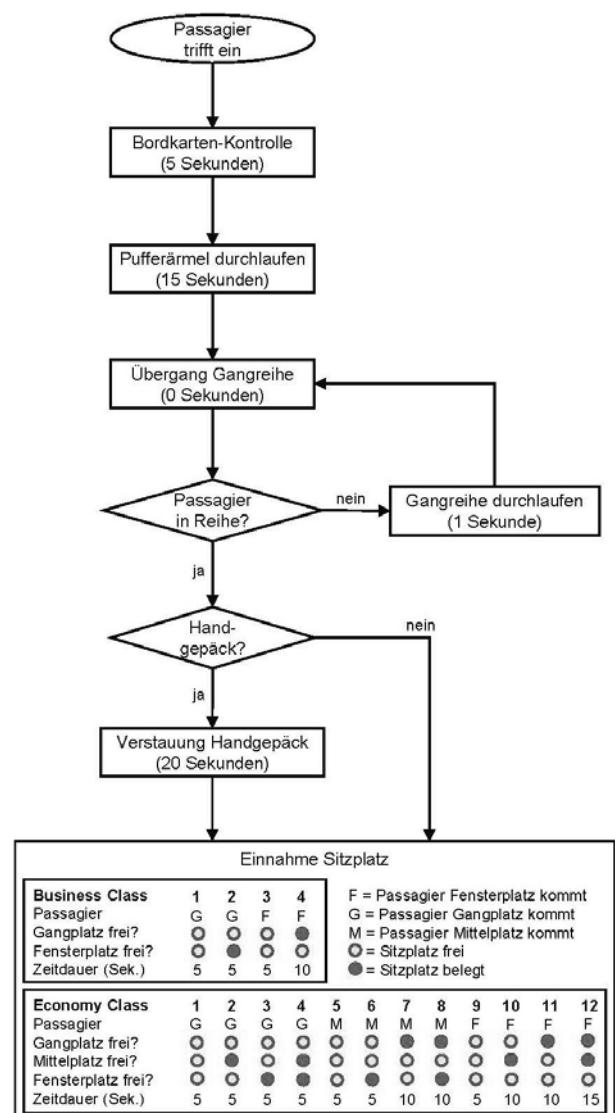


Abbildung 1: Flussdiagramm des Boarding-Prozesses aus Sicht eines Fluggastes

Die Zeiten für die einzelnen Prozessschritte wurden in Zusammenarbeit mit Herrn Martin Bertling, einem Prozessplaner des Flughafen Münchens, ermittelt. Dabei erfolgte eine Orientierung am offiziellen Dokument für Flughafenplaner „AIRBUS A320 AIRCRAFT CHARACTERISTICS AIRPORT AND MAINTENANCE PLANNING Chapter 5-2-0“ (Feb. 2018) [1]. Weiterhin fand eine Ergänzung noch fehlender Daten auf Basis der Dissertation „Analyse der Verzögerungen beim Boarding von Flugzeugen und Untersuchung möglicher Optimierungsansätze“ von Holger Stefan Appel (2014) [2] statt.

Bezeichnung	Zeitangaben	Quelle
mittlere Boardingdauer	18 Minuten	Prozessplaner Flughafen München, offizielles Dokument für Flughafenplaner
Bordkarten-Kontrolle	5 Sekunden	Prozessplaner Flughafen München
Eintrittszeit Flugzeug ohne Warteschlange	15 Sekunden	Prozessplaner Flughafen München
Einnahme Sitzplatz	keine Person: 5 s eine Person: 10 s zwei Personen: 15 s	Prozessplaner Flughafen München
Handgepäck-anteil	Geschäftsreisende: 95% Touristen: 90%	Prozessplaner Flughafen München, Diss. Holger Appel
Handgepäckzeit	20 Sekunden	Prozessplaner Flughafen München, Diss. Holger Appel

Tabelle 1: Datengrundlage für den Boarding-Prozess

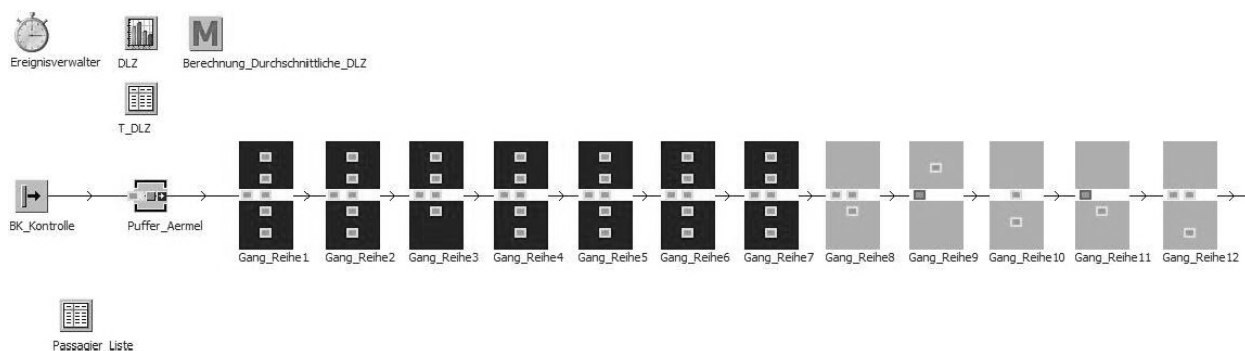


Abbildung 2: Simulation Random-Boarding (Ausschnitt)

4 Simulationsmodell

Ausgangspunkt für die Simulation in Plant Simulation ist eine Passagierliste. Diese legt die Reihenfolge der eintreffenden Passagiere bei der Bordkartenkontrolle fest und enthält für jeden Passagier den Eintrittszeitpunkt, den zugewiesenen Sitzplatz sowie die Anzahl an Handgepäckstücken, wobei die Reihenfolge der Passagiere und die Anzahl an bzw. das Vorhandensein von Handgepäckstücken auf Zufallszahlen basieren.

Der Flugzeugtyp A320 ist in 28 belegbare Sitzreihen unterteilt, welche im Modell als separate Anwendungsbausteine dargestellt sind. Jeder Anwendungsbaustein umfasst entweder vier (Business Class) oder sechs (Economy Class) Sitzplätze, die in Form von Einzelstationen nachgebildet sind. Sowohl die Belegung der Sitzplätze, als auch die sich im Gang beziehungsweise einer Sitzreihe befindlichen Passagiere, werden mithilfe von Animationen veranschaulicht. Jeder Passagier wird als Fördergut in die Simulation eingesetzt. Das Eintreffen des ersten Passagiers in der Bordkartenkontrolle stellt den Anfangszeitpunkt der Simulation dar. Die Simulation ist beendet, sobald der letzte Passagier seinen Sitzplatz (Einzelstation) eingenommen hat.

Das Simulationsmodell wurde anhand der Gesamtdurchlaufzeit für einen Boardingprozess validiert. Die aus mehreren Simulationsläufen resultierende durchschnittliche Durchlaufzeit für das Random Boarding betrug 17 Minuten und 43 Sekunden. Damit beträgt die Abweichung von den 18 Minuten, die der Prozessplaner des Flughafens München als mittlere Boarding-Dauer angegeben hat, lediglich 1,57%, womit das Modell als valide angesehen werden kann.

5 Experimente

Als Alternativen zum Random Boarding werden in der Literatur v.a. die Boardingstrategien Back-to-Front Boarding, Outside-In Boarding und die Kombination von Back-to-Front und Outside-In Boarding genannt. Daher erfolgt nun nach einer kurzen Erläuterung eine simulationsgestützte Untersuchung dieser drei Methoden.

Die Grundidee des **Back-to-Front Boardings** ist es, Passagiere von hinten nach vorne einsteigen zu lassen [2]. Damit soll vermieden werden, dass der hintere Teil des Ganges zu Beginn des Boardings eine Zeit lang ungenutzt bleibt, weil im vorderen Teil bereits die ersten Passagiere den Gang blockieren, um ihr Handgepäck zu verstauen. Es betreten somit die hinten sitzenden Passagiere das Flugzeug als erstes (ausgenommen Business Class), damit jeder Passagier mit möglichst wenigen Unterbrechungen zu seinem Sitzplatz gelangen kann. Dabei ergab sich in der Simulation eine mittlere Durchlaufzeit von 17 Minuten und 55 Sekunden für das Back-to-Front Boarding, was eine Verlangsamung um zwölf Sekunden im Vergleich zum Random Boarding bedeutet.

Bei der **Outside-In Boarding-Methode** wird die Maschine von außen nach innen, also zuerst mit den Fensterplätzen, anschließend mit den Mittelplätzen und als letztes mit den Gangplätzen geboardet [2]. Dabei spielt es keine Rolle, in welcher Reihe die Passagiere sitzen. Jedoch gilt auch hier, dass die Passagiere der Business Class eine höhere Priorität genießen und damit beginnen. In der Simulation konnte für diese Boarding-Methode eine mittlere Durchlaufzeit von 15 Minuten und 34 Sekunden erzielt werden. Verglichen mit dem Ausgangswert stellt dieses Ergebnis eine Verbesserung von zwei Minuten und neun Sekunden bzw. 12,1% dar.

Genauso wie bei der Back-to-Front Boarding-Methode betreten bei der **Kombination aus Back-to-Front und Outside-In Boarding** die Passagiere von hinten nach vorne das Flugzeug. Zeitgleich werden nach dem Prinzip des Outside-In Boardings zuerst die Fensterplätze, dann die Mittelplätze und zu guter Letzt die Gangplätze belegt [2].

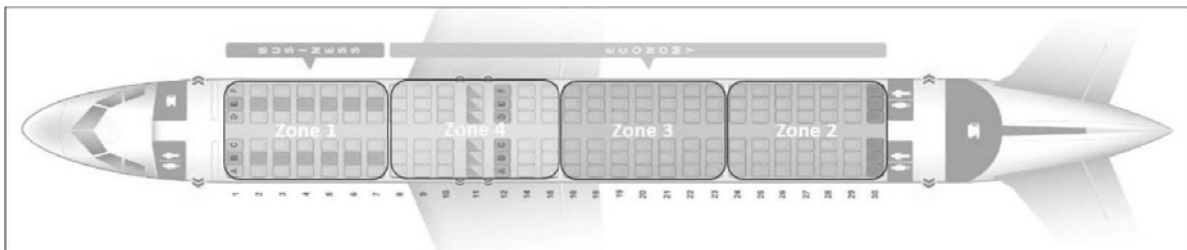


Abbildung 3: Back-to-Front Boarding

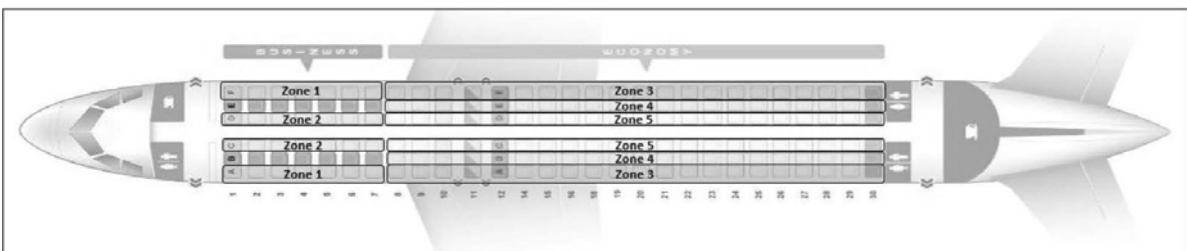


Abbildung 4: Outside-In Boarding

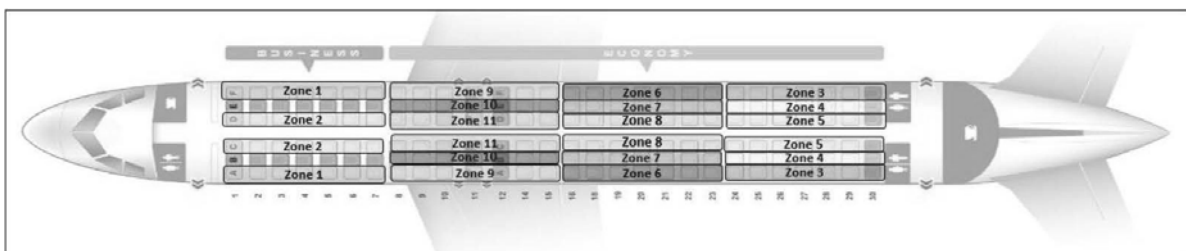


Abbildung 5: Kombination aus Back-to-Front und Outside-In Boarding

Auch hier werden als Erstes die Business Class Passagiere und als Zweites die Economy Class Passagiere geboardet. In den Simulationsläufen ergab sich für diese Boarding-Methode eine durchschnittliche Durchlaufzeit von genau 16 Minuten. Dieser Wert bedeutet zwar immer noch eine Verbesserung von einer Minute und 43 Sekunden gegenüber dem Random Boarding, aber eine Verschlechterung von 26 Sekunden im Vergleich zum reinen Outside-In Boarding.

In der Gesamtbetrachtung ist das Back-to-Front Boarding am ineffizientesten. Das Hauptproblem liegt darin, dass Passagiere viel Zeit mit Warten auf dem Gang verbringen, weil relativ viele Passagiere gleichzeitig versuchen, wenige Reihen zu besetzen. Zum einen kommt es hierbei zu Störungen in der Sitzreihe, wenn ein bereits sitzender Passagier wieder aufstehen muss, weil ein wartender Passagier am Fenster oder in der Mitte sitzt und zum anderen können lediglich die ersten an der Sitzreihe ankommenden Passagiere ihr Handgepäck verstauen, während alle nachrückenden Passagiere den Gang nicht zum Verstauen des Gepäcks, sondern häufig nur als Wartebereich nutzen können. Im Ergebnis verlagert sich dadurch die Warteschlange von der Gangway in das Flugzeug. Vorteilhaft erscheint, dass diese Methode, genauso wie das Random Boarding, einfach zu verstehen ist, da das Flugzeug in nur wenige Bereiche mit gleichzeitig boardenden Passagieren unterteilt wird.

Für das Outside-In Boarding spricht, dass sich keine Stauungen im Flugzeuggang aufgrund von im Weg stehender Passagieren ergeben. Im Gegensatz zum Back-to-Front Boarding erfolgt hier eine bessere Verteilung hinsichtlich der Auslastung des Gangs (ähnlich zum Random Boarding) gegeben. Es müssen nämlich Passagiere, die einmal sitzen, nicht wieder aufstehen und blockieren somit nicht erneut den Gang. Die Vorteile der Boarding Methoden sind in der verbesserten Durchlaufzeit erkennbar. Als entscheidender Nachteil wird in der Literatur genannt, dass die Plätze einer Sitzreihe nicht gemeinsam geboardet werden. Das heißt, dass sich Reisegruppen bzw. Familien beim Einstieg in das Flugzeug kurzzeitig trennen müssen. Demnach ist die Akzeptanz gegenüber dieser Boarding Methoden gering, da die meisten Passagiere einen gewissen Komfort beim Fliegen erwarten. Somit verwarfen viele Fluggesellschaften diese Boarding-Methode nach einer kurzen Testphase. Eine Lösung hierfür wäre, dass z. B. Familien – ähnlich wie Business Class Passagiere – beim Einsteigen bevorzugt werden.

Mit der Kombination der Back-to-Front und der Outside-In Boarding Methode wird versucht, die Vorteile dieser beiden Methoden zu verbinden. Das Boarding des Flugzeugs erfolgt von hinten nach vorne, so dass die Auslastung im Flugzeuggang möglichst gleichmäßig ist. Gleichzeitig wird das Flugzeug aber auch von außen nach innen geboardet, um Störungen innerhalb der Reihen, also Reiheninterferenzen, zu vermeiden. In der vorliegenden Simulation waren diese Vorteile bei einer Flugzeugauslastung von 100% jedoch nicht nachweisbar (sondern erst bei weiteren Versuchen mit einer Auslastung von 90% und weniger). Hinzu kommt, dass diese Methode in der Praxis einige Nachteile mit sich bringt. Zum einen stellt die richtige Anordnung der Passagiere vor dem Flugzeugeintritt eine Herausforderung dar und zum anderen müssen sich wie bereits unter der Boarding Methode Outside-In beschrieben, auch hier Familien, bzw. Reisegruppen generell, trennen.

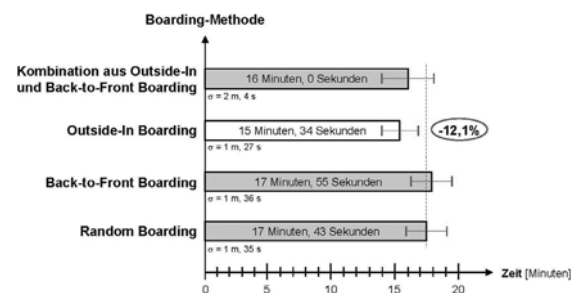


Abbildung 6: Mittlere Boarding-Dauer und Standardabweichung der einzelnen Boarding-Methoden nach jeweils 35 Simulationsläufen

Sollten sich Fluggesellschaften aufgrund der bisherigen Ergebnisse und der aktuellen Situation überlegen, ihre bisherige Boarding-Strategie zu ändern, bietet das vorliegende Simulationsmodell eine gute Ausgangsbasis für zusätzliche bzw. zukünftig mögliche Erweiterungen, um den simulierten Boarding-Prozess noch weiter an die Realität anzunähern und damit für konkrete Umgebungs- bzw. Auslastungssituationen die optimale Boarding-Strategie zu ermitteln. Ansätze für mögliche Erweiterungen sind z.B. die Aufnahme zusätzlicher Boarding-Methoden, die Differenzierung nach Anzahl und Art des Handgepäcks in Trolley, Rucksack und Jacke einschließlich Kapazitätsbeschränkungen der Gepäckfächer sowie die Berücksichtigung unterschiedlicher persönlicher Laufgeschwindigkeiten und einer Fehlerquote für verspätete Passagiere.

6 Conclusio

Bereits mit einem einfachen Simulationsmodell gelang es, die unterschiedlichen Boarding-Strategien anschaulich gegenüberzustellen. Auffällig war, dass sich zumindest bei einer Auslastung von 100% die einfachere Outside-In Boarding-Strategie gegenüber der Kombination aus Outside-In und Back-to-Front Boarding als vorteilhafter erwies. Dabei bestand der Vorteil nicht nur in einer um 26 Sekunden kürzeren mittleren Boarding-Dauer, sondern vor allem in einer mit einer Minute und 27 Sekunden deutlich niedrigeren Standardabweichung im Vergleich zur Standardabweichung von zwei Minuten und vier Sekunden bei der Boarding-Kombination. Dies erhöht die Planungssicherheit erheblich, was wiederum v.a. bei einer Auslastung von 100% wichtig erscheint, die bei den wenigen Flügen in der aktuellen Zeit durch die Fluggesellschaften angestrebt wird.

Im Zuge der Umsetzung in die Praxis werden zusätzlich mindestens visuelle Hilfsmittel erforderlich sein, um das Verständnis der Passagiere gegenüber der angewandten Boarding-Strategie und ihren Vorgaben zu erhöhen. Weiterhin gilt, sich in Erinnerung zu rufen, dass die erste Voraussetzung für eine erfolgreiche Realisierung ist, dass sich alle Fluggäste spätestens zu einer fest definierten Zeit am Gate eingefunden haben. Da sich das so gut wie nie zu 100% gewährleisten lässt, scheiterte schon daran die Umsetzung von noch ausgeklügelteren Boarding-Strategien, als sie in diesem Beitrag vorgestellt wurden.

Ein Vertreter der Lufthansa hält das Boarding sogar für einen zu komplexen Ablauf, um es allein mit mathematisch-informatischen Methoden optimieren zu können. So müssten Experten, die modellieren und simulieren können, mit Psychologen zusammengebracht werden, die Gruppenphänomene verstehen und erklären. Insofern steht vor einer Verfeinerung des Simulationsmodells immer die Frage, welcher Aufwand damit verbunden ist und welcher Nutzen im Hinblick auf die Übertragbarkeit in die Praxis tatsächlich entsteht, sofern nicht bereits die Voraussetzungen des Modells als zu restriktiv eingeschätzt werden. Da das aus der Vor-Corona-Zeit bekannte Gedränge in Flugzeugen während der Pandemie auch aus Gründen des Infektionsschutzes vermieden werden muss, böte sich jetzt immerhin die Chance, einen Versuch in Richtung des vergleichsweise einfach verständlichen Outside-In-Boardings zu wagen.

References

- [1] Airbus. *A320 – Aircraft Characteristics Airport and Maintenance Planning*; 2018.
- [2] Appel, H. *Analyse der Verzögerungen beim Boarding von Flugzeugen und Untersuchung möglicher Optimierungsansätze* [Dissertation]. Rheinisch-Westfälische Technische Hochschule Aachen; 2014
- [3] Condor. *Reisen in Zeiten von Corona*; <https://www.condor.com/de/blog/reisen-in-zeiten-von-corona/> (abgerufen am 01.07.2020)
- [4] Flug Revue. *Top 10 – Die größten Kunden der A320-Familie von Airbus*; [https://www.flugrevue.de/zivil/bestseller-aus-europa-top-10-die-groessten-kunden-der-a320-familie-von-airbus/Reisen in Zeiten von Corona](https://www.flugrevue.de/zivil/bestseller-aus-europa-top-10-die-groessten-kunden-der-a320-familie-von-airbus/Reisen%20in%20Zeiten%20von%20Corona); (abgerufen am 01.07.2020)
- [5] Seatguru. https://www.seatguru.com/airlines/Lufthansa/Lufthansa_Airbus_A320-200_NEK.php; (abgerufen am 01.07.2020)

Modellieren mit Raumbezug: Spezifikation dynamischer Topologien mit den Mitteln von Graphersetzungs-systemen

Jochen Wittmann¹

¹ Hochschule für Technik und Wirtschaft Berlin, Studiengang Umweltinformatik,
Wilhelminenhofstraße 75A, 12459 Berlin, Germany, wittmann@htw-berlin.de

Abstract. Viele Anwendungen im Bereich der Umweltsimulation beschränken sich nicht auf die Dynamik von eindimensionalen Bestandsgrößen, sondern versuchen zusätzlich, die räumliche Dimension der untersuchten Objekte mit ihren dynamischen Veränderungen zu beschreiben. In dieser Situation zeigt der Artikel, dass es einer allgemeinen Spezifikationsebene bedarf, die es erlaubt, anwendungsnah und problemspezifisch die Dynamik von Objekten mit Raumbezug abzubilden und andererseits die Möglichkeit gibt, diese Spezifikation algorithmisch sauber in einem Simulationsprogramm abzuarbeiten. Dabei sollen insbesondere die aus den topologischen Eigenschaften der Objekte abgeleiteten semantische Konsistenzbedingungen eingehalten werden. Zu diesem Zweck werden die Möglichkeiten der Dynamik für die GIS-Primitive Punkt, Linie und Polygon klassifiziert. Anschließend wird der Ansatz der Graphgrammatiken aus dem Bereich der Formalen Sprachen auf die Probleme der Dynamikspezifikation von Topologien von raum-zeitlichen Objekten übertragen und das algorithmische Optimierungspotential für die Implementierung dieses Ansatzes aufgezeigt.

1 Motivation

Die Anforderungen an Analyse, Modellierung und Simulation von dynamischen Prozessen haben sich in den letzten Jahren grundlegend verändert. Einerseits durch eine zunehmende Verbreitung von Smartphones mit automatischer Positionsermittlung über GPS-Satelliten auf Seite der Datenerfassung, die selbst für den Consumer-Bereich zum Standard geworden ist, und andererseits mit dem freien, komfortablen und schnellen Zugriff auf geographisches Karten- und Bildmaterial z.B. durch das Web-GIS Google-Maps (Google Maps, 2019) oder OpenStreetMap (OpenStreetMap, 2019). Die Ansätze

zur Modellierung können nun geographisch differenziert mit hoher räumlicher Auflösung erfolgen. Von Seiten der Informatik wird dieser Trend durch die Konzepte der objektorientierten Programmiersprachen bzw. der individuenbasierten Modellierungstechniken unterstützt, die die Behandlung einer Vielzahl auch räumlich differenzierter Objekte bzw. Individuen auf relativ einfache und anschauliche Weise möglich machen (siehe z.B. Ortmann, 1999) oder eine Einführung in die objektorientierte Programmierung bei (Balzert, 1999). Während die Erfassung und Speicherung von Raum-Zeit-Daten durch entsprechende objektorientierte Datenbankkonzepte im Wesentlichen gelöst ist, gestaltet sich die Spezifikation von dynamischen Modellen mit Raum- und Zeitbezug schwierig. Eine detaillierte Analyse der entsprechenden Ansätze aus dem Bereich der Modellspezifikation einerseits und den Geoinformationssystemen andererseits findet sich in vorausgehenden Arbeiten des Autors, z.B. in (Wittmann, 2019).

In diesem Beitrag sollen nun zunächst die GIS-Primitive Punkt, Linie und Polygon auf ihre potenziellen dynamischen Eigenschaften hin untersucht und klassifiziert werden. Anschließend wird der Ansatz der Graphgrammatiken, der aus dem Bereich der formalen Sprachen stammt, auf das die Dynamikspezifikation der Geo-Objekte übertragen. Am Ende steht eine Abschätzung des Potenzials dieses Ansatzes in Bezug auf laufzeit-technische und modellierungstechnische Aspekte.

2 Dynamikbeschreibung auf der Basis der GIS-Primitive

Nach der vorausgehenden Analyse soll nun ein konstruktiver Vorschlag entwickelt werden, wie Dynamik

von Geoobjekten beschrieben und algorithmisch behandelt werden kann. Im Folgenden Abschnitt wird für die Primitive des GIS Punkt, Linie, Polygon und Topologie klassifiziert und spezifiziert, wie sich Bewegung ausdrücken kann.

2.1 Dynamik von Geo-Objekten

Für jedes der Primitive soll hier klassifiziert werden, welche Möglichkeiten der dynamischen Veränderungen auftreten und wie diese im Modell zu beschreiben sind. Dabei fließen sowohl die Klassifikation von Yattaw (Yattaw, 1999) als auch die Klassifikation von Modellbeschreibungsmethoden aus (Wittmann, 2019) ein und werden zu einer Dynamikbeschreibung für Geoobjekte zusammengeführt. Abbildung 1 zeigt die Klassifikation im Überblick.

Typ Point

Im einfachsten Fall liegt ein Punkt-Feature vor. Je nach Modellziel kann sich ein Punktobjekt entweder durch eine kontinuierliche Bewegung im Raum bewegen oder aber plötzliche, sprunghafte Positionsveränderungen durchführen. Im ersten Fall kann diese kontinuierliche Bewegung durch einen Bewegungsvektor, also die Angabe von Richtung und Geschwindigkeit, dargestellt werden, mathematisch lässt sich die Beschreibung auf eine Differentialgleichung zurückführen.

Im Fall der sprunghaften Positionsänderung muss man auf die Konzepte der diskreten Simulation zurückgreifen und die Positionsänderung als diskretes Ereignis interpretieren, dessen Ausführung ohne Zeitverzug (also eben sprunghaft) erfolgt und das durch eine wie auch immer geartete logische Bedingung ausgelöst wird.

Neben diesen Bewegungsformen eines zum aktuellen Zeitpunkt im System bzw. im Modell existierenden Punkt-Objektes sind dynamische Veränderungen in der Topologie möglich, die in diesem Fall das Erzeugen bzw. das Löschen eines Punkt-Objektes abbilden.

Beispiele sind offensichtlich: für den kontinuierlichen Fall ein kontinuierlich fahrendes Fahrzeug oder ein Vogel, der kontinuierlich im 3-D-Raum fliegt. Eine nur zu diskreten Zeitpunkten gemessene bzw. beobachtete Position eines Tieres bei Tierwanderungen oder aber alle Bewegungen zwischen Haltestellen eines ÖPNV-Verkehrsmittels, bei dem der eigentliche Fahrweg nicht wesentlich für den Modellzweck ist sondern nur die Info

wann sich das Fahrzeug laut Fahrplan an welcher Haltestelle befindet. Das Erscheinen und Verschwinden vom Punktobjekten bedarf wohl keiner weiteren Beispiele, es kann sowohl mit dem kontinuierlichen Fall als auch mit dem diskreten Fall kombiniert werden.

Typ Line

Für Linienobjekte ist die Dynamik weiter zu differenzieren: Es kann sich um eine Bewegung des Linienobjektes als Ganzes handeln oder aber das Linienobjekt ändert seine Form nur in Teilen, indem sich nur eine Teilmenge der definierenden Punkte bewegt, die übrigen Stützstellen ihre Position jedoch beibehalten. Beide Varianten sind sowohl für den kontinuierlichen als auch für den diskreten Fall möglich. Hinzu kommen die Veränderungen in der Topologie, die sich durch die elementaren Methoden „Stützstelle hinzufügen“ und „Stützstelle entfernen“ abbilden lassen.

Beispiel für den kontinuierlichen Fall könnte die dynamische Entwicklung einer Küstenlinie sein, der diskrete Fall kann beispielsweise eine Absperrung oder einen Zaun abbilden, der einmal jährlich neu an geographische Gegebenheiten angepasst wird, beispielsweise als Schutz für Spaziergänger an die kontinuierliche Veränderung einer Steilküste.

Wird die Liniendynamik durch die Dynamik einer Teilmenge der Stützstellen beschrieben, so ist für die Modellierung bedeutsam, dass durch die für die Einzelpunkte getrennt spezifizierte Dynamik die Semantik und Topologie der Linie als Ganze nicht verletzt wird (z.B. die Eigenschaft kreuzungsfrei). Entsprechend muss bei der Beschreibung oder zumindest bei der anschließenden Abarbeitung der Beschreibung in der Simulation darauf geachtet werden, dass semantische Verstöße zu einer Fehlermeldung und zum Abbruch der Simulation führen.

Typ Polygon

Für Objekte des Typs Polygon gelten analog dieselben Bemerkungen wie für Linienobjekte. Polygone können sich mit allen ihren Stützstellen synchron kontinuierlich oder plötzlich bewegen. Zeigt nur eine Teilmenge der Stützstellen des Polygons Dynamik, so kann diese wiederum durch einen Bewegungsvektor (kontinuierlich) oder durch ereignisartige, sprunghafte Veränderungen der Position der Stützstellen spezifiziert werden. Auch hier kann es durch die Einzeldynamik von Stützstellen zur Verletzung der topologischen Eigenschaften

des Polygons kommen. Für solche Fälle sind entsprechende Vorkehrungen bei Modellbeschreibung und Simulation zu treffen.

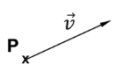

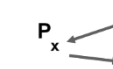
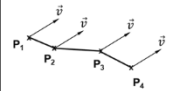
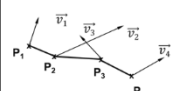
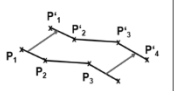
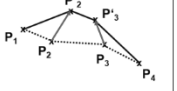
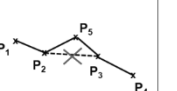
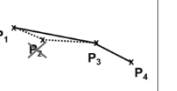
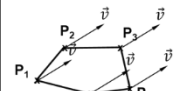
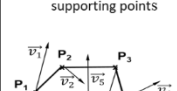
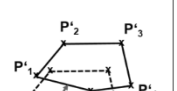
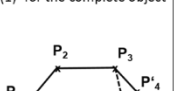
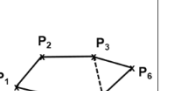
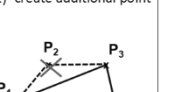
	continuous movement	sudden movement	topological change
point	 <p>\vec{v} with velocity direction</p>	 <p>triggered by event condition</p>	 <p>create Point delete point</p>
line	 <p>(1) \vec{v} identical for all supporting points</p>  <p>(2) \vec{v}_i for supporting point i</p>	 <p>(1) for the complete line</p>  <p>(2) individually for a subset of supporting points</p>	 <p>(1) create additional point</p>  <p>(2) delete point of the line</p>
polygon	 <p>(1) \vec{v} identical for all supporting points</p>  <p>(2) \vec{v}_i for supporting point i</p>	 <p>(1) for the complete object</p>  <p>(2) individually for a subset of supporting points</p>	 <p>(1) create additional point</p>  <p>(2) delete point of the object</p>

Abbildung 1: Dynamikspezifikation für GIS-Primitive

Die entsprechenden topologischen Dynamikalternativen verhalten sich analog zu den zuvor besprochenen.

Beispiel für ein diskretes Bewegen eines Polygonobjektes ist das tägliche Umstellen eines Schafpferchs als Ganzes oder mit Anpassung der Pfähle an geographische Gegebenheiten. Die kontinuierliche Entwicklung eines Polygons bildet beispielsweise die Entwicklung eines Siedlungsgebietes. Bei diesem Beispiel werden die Stützstellen unterschiedliche Bewegungsmuster aufweisen. Die kontinuierliche Bewegung einer durch einen kontinuierlich fahrenden Traktor gezogenen Egge über ein Feld kann als Beispiel für eine kontinuierliche aber form-erhaltende Dynamik gelten.

Bemerkungen zur vorausgehenden Klassifikation

Bemerkung 1: Grundsätzlich ist die Modellierung über die Angabe eines Bewegungsvektors oder durch die Angabe eines diskreten Ereignisses möglich. Der kontinuierlich zu interpretierende Bewegungsvektor reicht zur Dynamikspezifikation aus und wird für physikalisch-naturwissenschaftliche Modelle bevorzugt. Er ermöglicht die Simulation mit Zeitschritten, die gegen Null gehen, also eine sehr hohe zeitliche Auflösung. Diese bedingt jedoch auch einen erheblichen Rechenzeitbedarf für so beschriebene Modelle. Daher ist die Alternative der diskreten Modelle zu bedenken, die zwar immer nur sprunghafte Änderungen des Systemzustands zulassen, in der Regel jedoch wesentlich weniger Rechenzeit beanspruchen. Im Fall der diskreten Ereignisse ist die Angabe einer logischen Bedingung (wann das Ereignis stattfindet.) und eines in beliebiger formalen Sprache zu formulierenden „Effekts“ des Ereignisses (Was zum Ereigniszeitpunkt geschieht.) notwendig.

Bemerkung 2: In der tabellarischen Übersicht der unterschiedlichen Dynamik-Spezifikationsvarianten fällt auf, dass eine kontinuierliche Spezifikation von Änderungen in der Topologie nicht vorgesehen ist. Dies ist durch die mengentheoretische Definition der Topologie begründet: ein Element einer Menge (Ecke oder Kante) ist entweder vorhanden oder nicht. Ein „Wachsen einer Kante muss demnach durch die Bewegung der die Kante definierenden Stützstellen nachgebildet werden.

Bemerkung 3: Beim Erzeugen von neuen Objekten (Punkten oder Stützstellen) wurde in der Beschreibung allein der Dynamikaspekt diskutiert und die Auswirkung auf die Topologie beschrieben. Selbstverständlich müssen beim Erzeugen auch sämtliche Attribute der neu erzeugten Objekte sinnvoll parametrisiert werden.

Bemerkung 4: In diesem Abschnitt sollten allein und ausschließlich die Spezifikationsmöglichkeiten für die dynamischen Beziehungen für Geo-Objekte klassifiziert werden. Die Frage, wie sich Betrag und Richtung eines Bewegungsvektors berechnen lassen und wie weit und wohin eine Stützstelle bei einem Ereignis springt, hängt von diversen anderen Größen, oder ganz allgemein formuliert, vom aktuellen Systemzustand ab.

3 Graphen Grammatiken als Ansatz zur Formalisierung von topologischen Veränderungen

Mit den Standardverfahren der kontinuierlichen und diskreten Simulation lassen sich eine Reihe der in Abbildung 1 klassifizierten Fälle behandeln. Allerdings ergeben sich Schwierigkeiten, wenn die Bewegung für Teile eines Objekts unterschiedlich erfolgt, weil durch diese Bewegung die Konsistenz der Topologie der Objekte gegebenenfalls verletzt wird (neue Schnittpunkte bei Linien entstehen, ein Polygon entwickelt sich zu zwei Teilpolygonen, ...). Daher erscheint es sinnvoll, die Spezifikation dynamischer Veränderungen nicht an die Beschreibung der Dynamik einzelner Bestimmungstücke eines Objektes (z.B. einzelne Punkte als Stützstellen) zu binden, sondern die Dynamikspezifikation auf höherer Ebene für eine komplexere Situation anzugeben. Damit können, wie sich zeigen wird, Konsistenzprobleme an der Wurzel vermieden werden.

Zu diesem Zweck wird im Folgenden ein Ansatz von Schneider (Schneider, 2019) referiert, der aus dem Bereich der formalen Sprachen stammt und sich mit sogenannten Graphersetzungssystemen bzw. Graphgrammatiken auseinandersetzt. Die Definitionen und das Beispiel der folgenden zwei Abschnitte sind wörtlich mit nur kleinen Anpassungen aus dem genannten Skript von Schneider übernommen, das den Stand der Forschung kompetent und kompakt zusammenfasst. Die Übertragung auf die Situation der Geo-Objekte erfolgt dann im dritten Abschnitt dieses Kapitels.

3.1 Definitionen

Die Definition der Graph-Grammatik ist motiviert durch die allgemein bekanntere Definition der Chomsky-Grammatik, die nicht auf Graphen, sondern auf Zeichenketten arbeitet:

Definition (Chomsky Grammar (Chomsky, 1959)):

A Chomsky grammar (phrase structure grammar) is a quadruple $G = (T, N, P, S)$ where T and N are disjoint finite sets (alphabets), S is a distinguished element of N , and P is a finite subset of $L^*NL^* \times L^*$ with $L = T \cup N$.

Die Elemente der Mengen T und N werden Terminalsymbole bzw. Nicht-Terminalsymbole genannt. S ist das Startsymbol. P ist eine Menge von Produktionen bzw. Ersetzungsregeln. Statt der Tupelschreibweise (u, v) werden die Produktionen für gewöhnlich in der sogenannten Backus-Naur-Form als $u ::= v$ angegeben.

Definition (Chomsky Language):

Jede Chomsky-Grammatik definiert eine Menge von Zeichenketten, die unter Verwendung der Produktionen und ausgehend vom Startsymbol abgeleitet werden können und ausschließlich aus Terminalsymbolen bestehen. Diese Menge wird Chomsky-Sprache zur Grammatik G genannt:

$$L(G) := \{w \mid w \in T^* \wedge S \xRightarrow{*G} w\}$$

In einem zweiten Schritt können wir Chomskys Ansatz zur Formalisierung des Begriffs einer Graphengrammatik verallgemeinern. Der Hauptpunkt ist, Produktionen anzuwenden, bis eine Art Normalform erreicht ist. Dazu hat Chomsky Terminalsymbole von nicht-terminalen unterschieden. Produktionen werden angewendet, bis die Zeichenkette keine nicht-terminalen Symbole mehr enthält. Wir können diese Idee leicht auf Graphen-Grammatiken übertragen: (hier wieder die wörtlichen Definitionen aus (Schneider, 2019):

Definition (Graph grammar):

A graph grammar is given by a quadruple $G = (L, T, P, S)$ with

P being a finite set of graph productions in a category of labeled (hyper-)graphs using L as the labeling alphabet.

$T \subseteq L$ is called the terminal alphabet,

S is the starting graph.

Definition (Graph language):

If G is a graph grammar, then the set

$$L(G) := \{G \mid S \xRightarrow{*G} G \wedge \\ l_{EG}[E_G] \subseteq T_E \wedge \\ l_{VG}[V_G] \subseteq T_V\}$$

is called the language of G .

Da diese Definitionen mathematisch anspruchsvoll sind, muss an dieser Stelle zur weiteren Erläuterung und Vertiefung auf die Originalquelle verwiesen werden. Die Bedeutung des Ansatzes soll im Folgenden zunächst durch ein etwas ausführlicheres Beispiel veranschaulicht werden, bevor die Übertragung auf die Situation der Geo-

Objekte erfolgt.

3.2 Beispiel Graphersetzung

Abbildung 2 zeigt die Situation am Beispiel der Relation “is_mother”. Die Knoten des gegebenen Graphen sind Personen von weiblichem (f), männlichen (m) oder beliebigem (x) Geschlecht. Die Knoten sind zur Identifikation mit hochgestellten ganzen Zahlen durchnummeriert. Zwischen den Knoten besteht die Relation “is_mother”, ebenfalls mit hochgestelltem Identifikator.

Als Produktion wird ein dreiteiliges Ersetzungsschema angegeben: Auf der linken Seite wird ein Ausschnitt eines Graphen als parametrisierte Ist-Situation beschrieben. In der Mitte steht der sogenannte Interface-Graph und auf der rechten Seite muss die Situation nach Anwendung der Produktion angegeben werden. Sinn der Produktion ist es, die Relation “sister_of” zu ergänzen.

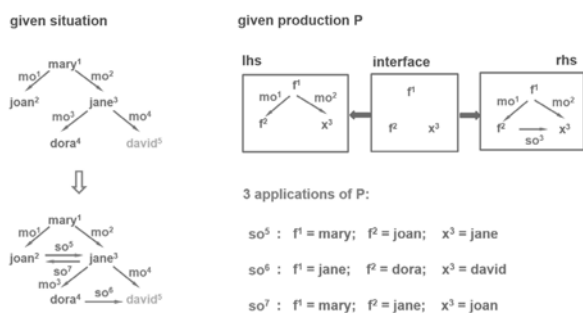


Abbildung 2: Beispiel für Graphersetzung (aus (Schneider, 2019))

Das linke Objekt der Produktion muss nicht nur sicherstellen, dass beide Söhne die gleiche Mutter haben, sondern auch, dass die Person an der Quelle der neuen Kante weiblich ist. Wir stellen dies sicher, indem wir w für mary, joan, jane, dora und x für mary, joan, jane, dora, dora, david definieren. Dann haben wir drei Möglichkeiten, diese Produktion auf den gegebenen Graphen (so⁵, so⁶ und so⁷) anzuwenden, wie in der Abbildung dargestellt.

Es ist zu beachten, dass durch die Spezifikation der Produktion sämtliche semantische Bedingungen automatisch eingehalten werden: passendes Geschlecht der beteiligten Personen, Beziehung zur gemeinsamen Mutter. Darüber hinaus wird bereits an dieser Stelle deutlich, dass das Finden geeigneter Passungen der linken Seite im aktuell gegebenen Graphen eine nicht-triviale Aufgabe darstellen kann.

3.3 Beispiele für eine Anwendung auf Geo-Objekte

Es kann aus Platzgründen das Potential dieses Ansatzes für die Beschreibung raum-zeitlicher Geo-Objekte nur angedeutet werden. Drei Beispiele sollen wenigstens illustrieren, wie komplexe Zusammenhänge durch geeignete Graphersetzungsgesetze in allgemeiner Form angegeben werden können. Topologisch lassen sich derartige Dynamiken als Wachstumsregeln für Graphen beschreiben. Abbildung 3 zeigt drei einfache Beispiele:

1. Stößt eine vorbestimmte Flugroute auf ein Hindernis, so soll dieses durch eine lokale Änderung der Route mit einer zusätzlichen Stützstelle umflogen werden.
2. Stellt sich in einem Netzwerk eine Punkt-zu-Punkt-Verbindung als stark nachgefragt, eine andere als schwach nachgefragt dar, so soll erstens eine Direktverbindung für die stark nachgefragte Strecke eingerichtet werden und zweitens die schwach nachgefragte Strecke gelöscht werden.
3. Im dritten Beispiel können die Objekte sowohl als Polygone als auch als Graphen interpretiert werden. In jedem Fall geht es darum, Wachstumsprozesse zu beschreiben. Eine Form, wie sie links des Pfeiles dargestellt ist, kann sich zu einer Form, wie sie rechts des Pfeiles dargestellt ist, verwandeln. Interpretiert als Polygone könnte auf diese Weise die Entwicklung eines Siedlungsgebietes modelliert werden. Als Graph könnte das Wachstum eines Versorgungsnetzes auf diese Weise beschrieben sein.

Die prinzipielle Vorgehensweise lässt sich dabei in 4 Schritten sehr einfach beschreiben:

1. Erstelle einen Satz von Produktionen, die die Dynamik der behandelten Objekte wiedergeben.
2. Finde in einem bestehenden Graphen eine Repräsentation der linken Seite einer Produktion und löse diese linke Seite als Teilgraphen heraus.
3. Führe in dem Teilgraphen, der durch die linke Seite gegeben ist, die Veränderung entsprechend der rechten Seite der Produktion aus.
4. Binde den veränderten Teilgraphen wieder in seinen Kontext des bestehenden Graphen ein.

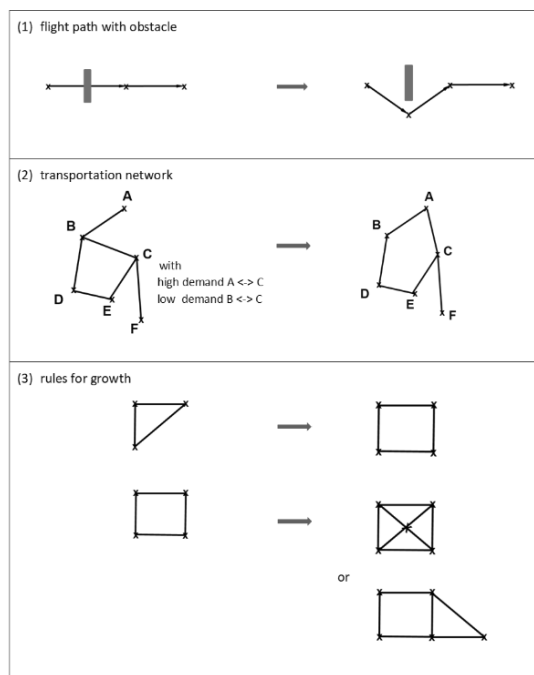


Abbildung 3: Beispiele zur Dynamik von Topologien

Allerdings stellen sich bei Detailbetrachtung einige Fragen, die die beschriebene Vorgehensweise verkomplizieren:

1. Wie weit muss die Übereinstimmung beim Vergleich der aktuellen Situation mit der linken Seite der Produktion gehen? Nur topologisch? Mit allen Attributen der beteiligten Knoten? ... Hier handelt es sich im Wesentlichen um ein Spezifikationsproblem, das durch den Formalismus der Graphgrammatiken gelöst werden kann. (Im Beispiel durch die Einbeziehung des Geschlechts der Knoten.)
2. Wie findet man alle Passungen im gegebenen Graphen? Dies ist ein Suchproblem, das näherer Betrachtung bedarf, durch die Formalisierung aber algorithmisch gut zu fassen ist.
3. Wie sind die Regeln (Produktionen) zu formulieren, um Konsistenz zu erhalten? Hierbei hilft ebenfalls der Formalismus, indem er Eigenschaften einer Produktionenmenge mathematisch ableitbar macht (analog zu den Eigenschaften der Chomsky-Grammatiken).

4. Wie ist das Zeitverhalten einer derartig spezifizierten Dynamik? Hier schlägt der Autor eine Interpretation gemäß dem Paradigma der diskreten Modelle vor, indem die Ausführung einer Produktion als Effekt eines diskreten Events behandelt wird.

4 Ansätze zur Optimierung der algorithmischen Behandlung der Dynamikspezifikation durch Graphgrammatiken

Vielmehr sollen hier zwei Ansatzpunkte zur Optimierung, die sich allein aus dem vorgestellten Simulationskonzept ergeben, als Themen für weitergehende Forschungsarbeiten andiskutiert werden.

Ein wesentlicher Unterschied zur Behandlung der Geo-Primitive im GIS liegt darin, dass sich ein Objekt nicht mit allen seinen bestimmenden Stützstellen gleich bewegen muss, sondern dass die Dynamik differenziert für jede individuelle Stützstelle spezifiziert werden kann. Das führt zu erheblichen Problemen bei Speicherung und Zugriff auf diese Objekte, weil ein effizienter Zugriff auch auf die einzelnen Stützstellen die Kapselung der höheren, zusammengesetzten Objekte (z.B. eines Polygons) verletzt. Hier sind Überlegungen notwendig, wie diese Zugriffe effizient ermöglicht werden können.

Neben diesen Aspekten der effizienten Speicherung soll hier aber besonders das Optimierungspotential betrachtet werden, das durch die Einführung der Graphersetzung entsteht. Im konzeptuellen Teil wurde zu diesem Schritt zwar die Spezifikation in Form der Ersetzungsregeln angegeben, die Teilschritte, die eine Implementierung umfassen muss und die im Wesentlichen für die Rechenzeiten verantwortlich sind, sind nicht aufgeführt worden. Es handelt sich dabei um die Schritte:

- Suche nach dem durch die linke Seite einer Ersetzungsregel gegebenen Musters im aktuellen Modellzustand.
- Generieren der Menge mit erlaubten Ersetzungsvarianten.
- Variation der Position neuer Knoten im Raum.

Neben den offensichtlich benötigten Suchstrategien werden Methoden zum effizienten Aufspannen des Suchraums durch die systematische Generierung von Alternativen benötigt. Für beide Methoden müssen effektive und

effiziente Datenstrukturen gefunden werden. Für lokal begrenzte Suchen wären geeignete Hashing-Strategien in Betracht zu ziehen. Gefundene Alternativen müssen bewertet und einem Ranking unterzogen werden. Effiziente Zugriffe auf das Bewertungsmodell sind dazu notwendig. Die Bewertung ist durch entsprechende statistische Verfahren abzusichern (Konfidenzintervalle, ...). Dies wiederum legt es nahe, die Simulationsläufe für unterschiedliche Parametrisierungen durch Parallelisierung zu beschleunigen. Da ähnliche Probleme häufig auftreten werden (z.B. das Finden einer optimalen Position für einen neuen Knoten), ist es wahrscheinlich, dass selbstlernende Verfahren diesen Verfahrensschritt erheblich verkürzen können.

Bevor ein Satz von Ersetzungsregeln zum Einsatz kommt, erlauben die Erkenntnisse aus der Theorie der Graphgrammatiken syntaktische und semantische Tests auf Vollständigkeit und Widerspruchsfreiheit des Regelsatzes. Es ist denkbar und wünschenswert, diese Tests um Bedingungen zur Sicherstellung der korrekten Topologie und Topographie zu erweitern. Können derartige Aussagen bereits formal aus dem Regelsatz abgeleitet werden, werden topologische und topographische Fehler zur Laufzeit ausgeschlossen und damit die Menge der zu vergleichenden Alternativen von Anfang an reduziert.

5 Zusammenfassung

Ausgangspunkt für die Arbeit war die Beobachtung, dass im interdisziplinären Bereich der Modellierung und Simulation von räumlichen Objekten eine Spezifikationsebene fehlt, die einerseits beschreibend genug ist, um komplexe dynamische Veränderungen auch für die Nicht-Informatik-Experten aus den jeweiligen Anwendungsbereichen darzustellen und andererseits formal genug ist, um einer algorithmischen Behandlung im Sinne eines Simulationsalgorithmus zugänglich zu sein. Eine Analyse bestehender Modellierungsparadigmen und Softwaresysteme zeigt, dass die üblichen Modellierungstechniken wenig Unterstützung bei der Spezifikation der räumlichen Dynamik bieten. Obwohl die objektorientierten, einzelbasierten Ansätze auch zur Modellierung räumlicher Prozesse genutzt werden können, werden die Nutzer bei der Beobachtung räumlicher Konsistenzbedingungen auf die proprietären Lösungen zurückgreifen.

Basierend auf dieser Analyse wird der Formalismus der Graphensubstitutionssysteme auf die Anwendung im Bereich der räumlichen Modelle übertragen. Der Nutzen,

der dem nicht unerheblichen Aufwand durch Formalisierung entgegensteht, besteht vor allem in der sauberen algorithmischen Handhabung und Verarbeitung der Modelldynamik, die von Grafikproduktionen beschrieben wird. Konsistenzbedingungen als Folge des räumlichen Bezugs können auf der Metaebene der Produktionen berücksichtigt werden und vermeiden topologisch sinnlose Dynamik. Darüber hinaus erweist sich der Formalismus als vorteilhaft, wenn es darum geht, die algorithmische Verarbeitung der zahlreichen alternativen Möglichkeiten zur Entwicklung von Objekten im Raum beherrschbar zu machen und die Komplexität und Konsistenz der Lösungen zu optimieren.

Die Integration dieses Ansatzes in ein Simulationslaufzeitsystem wird in einem separaten Beitrag diskutiert. Die Idee ist es, die Ausführung eines von der Produktion vorgegebenen Ersatzes als diskretes Ereignis im Sinne einer diskreten Ereignissimulation zu behandeln. Darüber hinaus wird die Praktikabilität des Ansatzes und insbesondere die Eignung der vorgeschlagenen Dynamikbeschreibung für die Kommunikation mit Nicht-Informatikern in geeigneten Praxisprojekten aus dem Anwendungsbereich der Umweltinformatik getestet.

6 Literaturverzeichnis

- Brown, D., Riolo, R., Robinson, D., North, M., & Rand, W. (2005). Spatial Process and Data Models: Toward Integration of . *Journal of Geographical Systems* 7(1), S. 25-47.
- Dransch, D. (1997). *Computer-Animation in der Kartografie: Theorie und Praxis*. Heidelberg: Springer.
- ESRI. (25. 6 2018). *ArcGIS API for Python*. Von <https://developers.arcgis.com/python/> abgerufen
- Esri ArcMap. (25. 6 2018). *Esri ArcMap Online-Hilfe: Erstellen von Animationen in ArcGIS*. Von <https://desktop.arcgis.com/de/arcmap/10.3/map/animation/about-building-animations-in-arcgis.htm> abgerufen
- GAMA. (12. 3 2019). Von GAMA Platform: <https://gama-platform.github.io/> abgerufen
- Kimerling, A., Buckley, A., Muehrcke, P., & Muehrcke, J. (2016). Map Use, Chapter 8, Quantitative

- Thematic Maps. 206. Redlands California: ESRI Press.
- MARS. (04. 03 2019). *MARS-Group*. Von <https://mars-group.org/features/#gis> abgerufen
- O'Sullivan, D. P. (2013). *Spatial Simulation – Exploring Pattern and Process*. Chichester: Wiley.
- OGC. (25. 6 2018). *OGC® Open Geospatial APIs - White Paper OGC*. Von <http://docs.opengeospatial.org/wp/16-019r4/16-019r4.html> abgerufen
- Ortmann, J. (1999). *Ein allgemeiner individuenorientierter Ansatz zur Modellierung von Populationsdynamiken in Ökosystemen unter Einbeziehung der Mikro- und Makroebene*. Rostock: Dissertation am Fachbereich Informatik, Universität Rostock.
- Rechenberg, P., & Pomberger, G. (2006). *Informatik Handbuch, Stichwort Architektur von SW-Systemen*. München: Hanser.
- Schneider, H. (6. 3 2019). *Graph Transformations - An Introduction to the Categorical Approach, Vorlesungsunterlagen*. Von <https://www2.cs.fau.de/staff/schneider/gtbook/index.html> abgerufen
- Torrens, P., & Benenson, I. (2005). Geographic automata systems. *International Journal of Geographical Information Science* 19(4), S. 385-412.
- Uhrmacher, A. (2001). Dynamic Structures in Modeling and Simulation - A Reflective Approach. *ACM Transactions on Modeling and Simulation, Vol.11. No.2.*, S. 206-232.
- Wittmann. (2000). Simulationsmodell und Geographisches Informationssystem Kopplungsalternativen am praktischen Beispiel. In A. Cremers, & K. Greve, *Umweltinformatik 2000, 12.Internationales Symposium, Bonn, 2000* (S. 45-58). Marburg: Metropolis.
- Wittmann, J. (2017). *Simulation in Umwelt- und Geowissenschaften - Workshop Hamburg 2017*. Aachen: Shaker.
- Wittmann, J., & Thiel-Clemen, T. (2016). *Simulation in Umwelt- und Geowissenschaften - Workshop Hamburg 2016*. Aachen: Shaker.
- Yattaw, N. J. (1999). Conceptualizing Space and Time: A Classification of. *Cartography and Geographic Information Science*, 26:2, S. 85-98.
- Zeigler, B. (1990). *Object-Oriented Simulation with Hierarchical, Modular Models*. London: Academic Press.

Erreichbarkeitsgraphen als Werkzeug zur Visualisierung des Treibhausgasausstoßes für die Verkehrsmittel Flugzeug, Auto, Bahn und Reisebus bei der Dienstreiseplanung

Malte Christiansen¹, Jochen Wittmann¹

¹ Hochschule für Technik und Wirtschaft Berlin, Studiengang Umweltinformatik, Wilhelminenhofstraße 75A, 12459 Berlin, Germany, wittmann@htw-berlin.de

Abstract. Die Angestellten der HTW Berlin (Hochschule für Technik und Wirtschaft) besuchen Konferenzen und sind im Rahmen internationaler Projekte in aller Welt unterwegs. Eine Forschungsgruppe der HTW Berlin hat die Reisedaten des Jahres 2017 gesammelt, um die Verteilung der für die Dienstreisen gewählten Verkehrsmittel zu analysieren. Deren Rohdaten bilden die Ausgangslage für dieses Projekt, das die ausgestoßene Treibhausgasmenge der Verkehrsmittel Flugzeug, Auto, Reisebus und Zug miteinander vergleicht und visualisiert.

Ausgangswert bilden die ausgestoßenen Treibhausgase für einen Flug von Berlin nach München. Die meistgenutzte Verbindung der Angestellten im Jahre 2017. Dieser Wert bildet das Limit an Treibhausgasen, das den anderen Verkehrsmitteln zur Verfügung steht.

Die Strecke zwischen Berlin und München beträgt 528 km und stößt 10,6 kg Treibhausgase aus. Mit der gleichen Menge könnten im Auto 764 km, im Reisebus 3317 km und im Zug 2948 km zurückgelegt werden. Der Unterschied ist deutlich zu erkennen. Mit der Treibhausgasmenge, welche bei einem Flug von Berlin nach München freigesetzt wird, würde der Zug bis nach Lissabon und der Reisebus tief ins Innere von Russland (z. B. Ufa) kommen.

Ziel ist es, diese Informationen in den Prozess der Planung und Buchung von Dienstreisen zu integrieren, um bei den Beteiligten das Problembewusstsein für CO₂-sparende Verkehrsmittel zu wecken.

1 Motivation

Dienstreisen gehören für Hochschulmitglieder zum beruflichen Alltag. Zum wissenschaftlichen Austausch ist der Besuch von internationalen Konferenzen unverzichtbar. Dazu kommen Projekte mit den entsprechenden

Verpflichtungen zu Dienstreisen anlässlich von Meetings und Präsentationen. Dies gilt auch für die Angestellten an der HTW Berlin (Hochschule für Technik und Wirtschaft). Sie unternehmen Dienstreisen in alle Welt. Diese Strecken werden häufig und wie selbstverständlich mit dem Flugzeug zurückgelegt. Laut den Umweltleitlinien der HTW Berlin sollen zwar öffentliche Verkehrsmittel vorgezogen werden, allerdings gilt dies nur, wenn es wirtschaftlich vertretbar ist. (HTW-Berlin, Umweltleitlinien, 2020) (HTW-Berlin, Mobilität, 2019)

Eine Projektgruppe der HTW Berlin hat die Reisedaten des Jahres 2017 gesammelt und die Verteilung der Verkehrsmittel veranschaulicht (Fronk, Güccük, Höhne, Motuz, & Zagorski, 2019). Deren Rohdaten bilden die Ausgangslage für das vorliegende Paper.

Bei der Gegenüberstellung von verschiedenen Verkehrsmitteln wird meistens nur in den Kategorien Zeit und Kosten argumentiert. So bietet zum Beispiel der wohl am häufigsten genutzte Routenplaner von Google (Google, 2020) nur genau diese beiden Zielkriterien für eine Routenoptimierung an. Sicherlich sind Zeit und Strecke zwei wichtige und nachvollziehbare Argumente, allerdings könnte die Gegenüberstellung noch um die Auswirkungen auf Klima und Gesundheit erweitert werden, um auch die ökologische Dimension bei der Transportmittelwahl transparent zu machen.

Um Angestellten, die ihre Geschäftsreisen mit alternativen Verkehrsmitteln zurücklegen, eine Argumentationshilfe an die Hand zu geben, möchte dieses Projekt anhand einer Visualisierung auf Kartenbasis den Unterschied zwischen Flugzeug einerseits und seinen Alternativen beim Faktor Treibhausgase andererseits darstellen. Als zum Fliegen alternative Verkehrsmittel wurden das

Auto, der Reisebus und der Zug gewählt. Bei denen es sich um gängige und leicht verfügbare Verkehrsmittel handelt.

Ziel ist es, Informationen über diese Alternativen in den Prozess der Planung und Buchung von Dienstreisen zu integrieren, um bei den Beteiligten das Problembewusstsein für CO₂-sparende Mobilität zu wecken.

In den folgenden Abschnitten wird daher zunächst kurz das Visualisierungsverfahren der Erreichbarkeitsgraphen selbst erklärt. Darauf folgt die Vorgehensweise zur Datenauswahl und -integration sowie die Möglichkeiten und Probleme der Nutzung der entsprechenden Tools unter ArcMap. Dabei wird auch auf die Randbedingung der Studie eingegangen, besonders auf die Beschränkung auf open-source Datenmaterial. Erste Resultate und eine Diskussion der Schwierigkeiten bei der Durchführung beschließen das Paper.

2 Methode

Als Grundlage für das angewandte Visualisierungsverfahren dienen Erreichbarkeitsgraphen. Erreichbarkeitsgraphen werden in der Kartografie angewandt, um darzustellen, welches Gebiet bei vorgegebener Distanz von einem gegebenen Punkt aus erreicht werden kann. Zur Demonstration zeigt Abbildung 1 als Beispiel die älteste bekannte Darstellung eines Erreichbarkeitsgraphen. Ausgehend von London als zentralem Punkt und Ausgangspunkt für die Entfernungsmessung zeigt sie die Reisezeit in die verschiedenen Regionen der Welt. Dabei wird ausgehend von London berechnet, wie viel Zeit auf einem gegebenen Wegenetz benötigt wird, die entsprechende Stelle auf dem Globus zu erreichen. Dargestellt ist dies in der historischen Karte durch in gewisser Weise aufeinander aufbauende, farblich gestaffelte Darstellungsschichten. Der dunkelgrüne Bereich wird am schnellsten erreicht, gefolgt von einem hellgelben Ring und so weiter. (Galton, 2020)

Auch heute noch wird diese Technik angewandt, um zu visualisieren, welche Orte innerhalb einer gegebenen Zeit erreicht werden können. In dieser Arbeit wird als Parameter für die Bestimmung der Ausdehnung der Erreichbarkeitszonen jedoch nicht das Attribut „Zeit“ angegeben vielmehr sollen die für die Reise benötigte Emissionsmenge an Treibhausgas zugrunde liegen. Dazu soll zunächst eine Obergrenze an Emissionen für die zu fah-

rende Strecke berechnet werden und anschließend angegeben und visualisiert werden, wie weit man auf einem gegebenen Wegenetz unter Einhaltung dieses Emissionsmaximums kommen kann. Der folgende Abschnitt erklärt die Vorgehensweise im Detail.

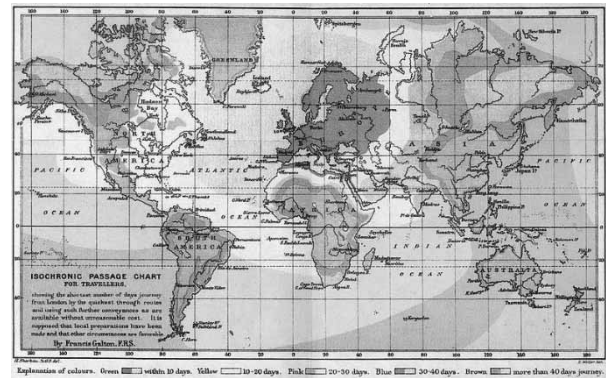


Figure 1: Die erste bekannte Erreichbarkeitskarte aus dem Jahre 1881. Sie zeigt die Reisezeit zu verschiedenen Regionen der Welt. (Galton, 2020)

3 Vorgehensweise

Nach der Veranschaulichung der Methode folgt nun die Schilderung der einzelnen Arbeitsschritte. Als Erstes wurde ein Wegenetz benötigt. Als Zweites musste der Grenzwert bzw. Maximalwert an Emissionen berechnet werden, der den Erreichbarkeitsgraphen für die Transportmittel als Limit dienen soll. Im Dritten und letzten Schritt konnten die Erreichbarkeitsgraphen dann erstellt werden.

3.1 Basisschicht erstellen

Ein Erreichbarkeitsgraph berechnet die mögliche Strecke anhand eines gegebenen Wegenetzes. Dieses Projekt vergleicht die Flugstrecke mit den Verkehrsmitteln Auto, Reisebus und Zug, sodass zwei Wegenetze benötigt werden. Einmal ein Straßennetz für Auto und Reisebus und einmal ein Schienennetz für den Zug.

Als Basis dienen Daten des OpenStreetMap-Projektes, die von den Downloadservern der Geofabrik bezogen wurde (Geofabrik, 2019). Dabei wurden die europäischen Länder und Regionen als Shape-Files heruntergeladen. Aus diesen Shape-files wurden die für das Projekt benötigten Linienfeatures herausgesucht. Als Orientierung dienten die Wiki-Seiten des OpenStreetMap-Projektes.

- Für das Straßennetz wurde der Key "highway" mit den Werten „motorway“, „motorway_link“,

“trunk“, “trunk_link“, “primary“, “primary_link“, “secondary“, „secondary_link“, „tertiary“, „tertiary_link“, “unclassified“ verwendet.

- Für das Schienennetz wurde der Key "railway" mit dem Wert „rail“ verwendet.

Ziel dabei war, bewusst auch niederrangige Verkehrswege aufzunehmen, aber andererseits dennoch die Datenmenge auf eine handhabbare Größe zu reduzieren. Die Auswahl der hier angeführten Attributwerte ist für diesen Prototypen im Sinne einer Machbarkeitsstudie sicher praktikabel. Für die Integration in das Zielsystem, das den gesamten Dienstreiseplanungsprozess umfassen soll, muss allerdings noch geprüft werden, ob sich die Menge der notwendigen Werte nicht weiterhin reduzieren lässt, ohne wesentliche Abstriche in der Genauigkeit der Ergebniskarte zu erzeugen. So könnte es sinnvoll sein, das zugrundeliegende Wegenetz in Abhängigkeit vom gewünschten Maßstab der Zielkarte zu skalieren: Bei einer weiteren Reise werden kleinste Wegeverbindungen sicherlich weniger relevant für das Endergebnis sein als bei einer kürzeren Strecke.

Nachdem die Datenmenge deutlich geschrumpft war, konnten die Länder und Regionen zu einer Karte zusammengefügt und in einer Geodatabase gespeichert werden, um die weitere Verarbeitung zu erleichtern.

3.2 Berechnung der CO2 Faktoren

Für jedes gewählte Verkehrsmittel wurde ein Grenzwert berechnet. Dieser Wert berechnet sich aus dem Ausgangswert und dem jeweiligen Verbrauch pro Pkm (Personenkilometer). Als Ausgangswert wird die ausgestoßene Menge Treibhausgase für einen Flug von Berlin nach München definiert. Dies war die meistgenutzte Verbindung der HTW-Angestellten im Jahre 2017 (Fronk, Güccük, Höhne, Motuz, & Zagorski, 2019). Die Flugstrecke von Berlin nach München beträgt 528 km. Der Tabelle 1 wird die entsprechende Treibhausgasmenge pro Pkm entnommen und miteinander verrechnet.

Eine Person stößt demnach 10,6 kg Treibhausgase auf einem Flug von Berlin nach München aus. Ausgehend von diesem Wert konnte nun berechnet werden, welche Entfernung andere Transportmittel mit dem durch den Flugverkehr vorgegebenen Emissionsmaximum zurücklegen könnten.

Tabelle 1 enthält die entsprechend berechneten Distanzen, die als Grenzwert für das jeweilige Verkehrsmittel

eingetragen wurden. Basis waren dabei die inzwischen allgemein anerkannten durchschnittlichen Emissionen gemäß der Zusammenstellung des Umweltbundesamtes (Umweltbundesamt, 2020). Die Verwendung dieser Durchschnittswerte wird in einem folgenden Abschnitt diskutiert werden.

Tabelle 1: Durchschnittliche Emissionen der Verkehrsmittel im Jahre 2017. (Umweltbundesamt, 2020)

Verkehrsmittel	Treibhausgas (g/Pkm)	Auslastung	Distanz (km)
Flugzeug	201	82 %	528
Auto	139	1,5 Pers. pro PKW	764
Reisebus	32	60 %	3317
Zug (Fernverkehr)	36	56 %	2948

3.3 Distanzgraf erstellen

Nach der Berechnung der Grenzwerte für die erreichbare Distanz und der erfolgreichen Erstellung des Streckennetzes, konnte damit begonnen werden die Distanzgraf zu berechnen. Dazu wurde die Erweiterung „ArcGis Network Analyst“ von ArcGis verwendet.

Der „ArcGis Network Analyst“ benötigt ein Netzwerk-Dataset, das aus miteinander verbundenen Kanten (Linien) und Verbindungsknoten (Punkten) besteht. In diesem Projekt wird dazu einmal das Straßennetz und einmal das Schienennetz in ein Netzwerk-Dataset umgewandelt, um danach eine Netzwerkanalyse durchführen zu können.

Bei der Analyse können, wie bereits geschrieben, zeitliche Faktoren sowie Distanzen angegeben werden. Dieses Projekt arbeitet mit vorgegebenen Distanzen. Als Distanz wird für das jeweilige Verkehrsmittel der Wert in der Spalte Distanz in Tabelle 1 entnommen.

Aufgrund von der erheblichen Datenmenge konnten in Anbetracht der zur Verfügung stehenden Zielplattform, keine Polygone für den Reisebus und den Zug erstellt werden. Diese wurden nachträglich mittels des Tools „Feature in Polygon“ aus den Liniennetzen erzeugt.

4 Ergebnisse und Diskussion

4.1 Erreichbarkeitsgraphen

Die Abbildungen 2, 3 und 4 stellen den jeweiligen Erreichbarkeitsgraphen für die Verkehrsmittel Auto, Reisebus und Zug dar. Die Abbildungen 3 und 4 zeigen insbesondere, dass sich die mögliche Wegstrecke stark erweitert, wenn für die Reise der Reisebus oder der Zug gewählt wird. Dagegen fällt der Unterschied zwischen PKW und Flugzeug erstaunlich gering aus. (Abb. 2)

Während des Projektes gab es einige Faktoren, die die Qualität der Visualisierung wesentlich beeinflusst haben. Das war zum einen die bereits genannte Beschränkung in der Rechenleistung, andererseits die Qualität des zugrundeliegenden Wegenetzes, sowie schließlich auch die verwendeten pauschalierten Treibhausgaswerte. Auf diese Punkte soll im Folgenden eingegangen werden.

4.2 Problem Rechenleistung

Die genutzten Rechner stießen an die Grenzen ihrer Berechnungsmöglichkeiten für den „Network Analysten“. Es war nicht in vertretbarer Zeit möglich, ein Polygon für den Zug und den Reisebus zu berechnen. Die Berechnung eines Erreichbarkeitsgraphen wurde auf einen Standard-PC nach etwa 36 Stunden Rechenzeit abgebrochen. Stattdessen musste ein Umweg über die berechneten Linien gegangen werden, was eine potenzielle Fehlerquelle ist, da bei dieser Vorgehensweise einige Bereiche nicht korrekt dargestellt werden. Das betrifft zum Beispiel Bereiche in Norwegen und Schweden.

Tatsächlich liegen aber in den nicht berücksichtigten Zwischenräumen zwischen den durch das Liniennetz erreichbaren Punkten ja gerade keine Verkehrswege, die es ermöglichen würden, in diese Zwischenräume vorzudringen. Die Polygonbildung glättet lediglich den Rand des Ergebnisses der Erreichbarkeitsrechnung. Besonders in Bezug auf die Genauigkeitsanforderungen des Prototypen stellt dies keine wesentliche Einschränkung dar.

4.3 Problem Basis-Geodatendaten

Das Ausgangsmaterial basiert auf Ländern, die Geofabrik dem europäischen Raum zuordnet. So werden Georgien, Russland (der nicht europäische Teil) und die Türkei zu Europa zugerechnet, Aserbaidshan oder Kasachstan jedoch nicht. Diese Grenzziehung durch die Geofabrik ist dabei nicht ganz klar und wird auch auf den Seiten des Anbieters nicht näher erläutert (Geofabrik, 2019).

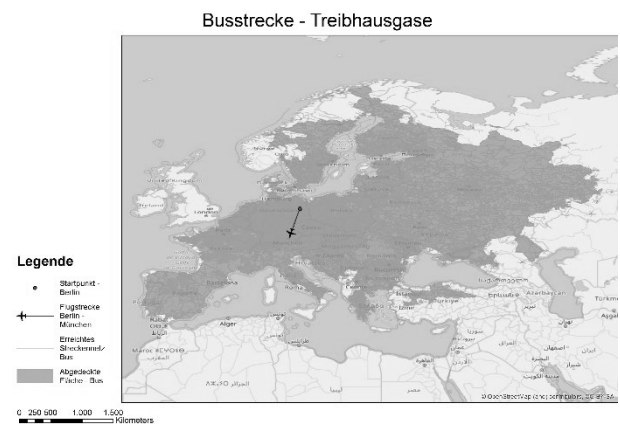


Figure 2: Erreichbarkeitsgraf für einen PKW, wenn 10,6 kg Treibhausgase zur Verfügung stehen.

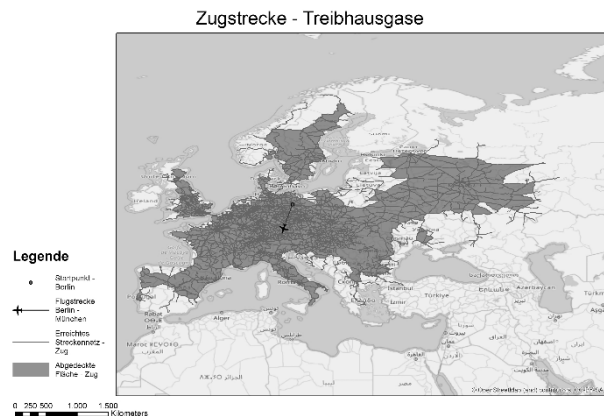


Figure 3: Erreichbarkeitsgraf für einen Reisebus, wenn 10,6 kg Treibhausgase zur Verfügung stehen

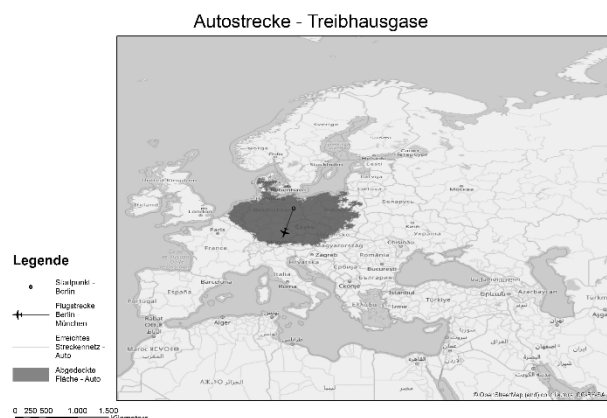


Figure 4: Erreichbarkeitsgraf für einen Zug, wenn 10,6 kg Treibhausgase zur Verfügung stehen.

Die Auswahl der hier berücksichtigten Länder erscheint somit recht willkürlich. Bei nachfolgenden Anwendungen sollten daher die ungefähr benötigten Reichweiten vorab abgeschätzt werden und als Richtwert dienen, welche Länder bei der Erstellung der Basisschicht berücksichtigt werden sollten.

Ein auffälliger Unterschied zwischen Abbildung 3 und 4 ist, dass es eine Verbindung zwischen Frankreich und Großbritannien für das Schienennetz gibt, aber nicht für das Straßennetz. Sodass die Reisebuskarte nicht die theoretisch mögliche Strecke bis nach Großbritannien abbildet. Ursache hierfür ist, dass der Reisebus nur mittels des Zuges den Tunnel oder mittels einer Fähre den Kanal durch-/überqueren könnte. Der Reisebus muss selbst auf andere Verkehrsmittel ausweichen. Da keine Straße den Ärmelkanal durchquert, kann der „Network Analyst“ diese Strecke nicht ohne Weiteres berücksichtigen. Hier könnte man im Einzelfall durch eine individuelle Nachbearbeitung des Streckennetzes abhelfen, ein Aufwand der für den Prototypen nicht getrieben wurde.

4.4 Problem Emissionswerte für die jeweiligen Verkehrsmittel

Die gewählten Werte beruhen auf Durchschnittswerten herausgegeben vom Umweltbundesamt. Die tatsächlich ausgestoßene Menge an Treibhausgasen für einen Flug zwischen Berlin und München könnte höher sein, weil während des Starts und der Landung tendenziell mehr Treibhausgase ausgestoßen werden, als wenn die Reiseflughöhe erreicht ist. Daher haben Kurzstreckenflüge (unter 750 km) im Schnitt einen höheren Treibhausgasausstoß als Langstreckenflüge. Bei der Tabelle des Umweltbundesamtes ist allerdings nur ein Wert für das Verkehrsmittel Flugzeug angegeben. (Umweltbundesamt, 2020) Bei der Übertragung auf sämtliche Dienstreisen sollte daher unbedingt eine entsprechende Differenzierung gemäß der Länge des Fluges bzw. gemäß einer detaillierteren Klassifizierung des Fluges vorgenommen werden. Für die Machbarkeitsstudie des hier vorgestellten Prototyps wurde zunächst auf diese Präzisierung verzichtet, zumal bei den Reisen der Datenbasis auch nicht abgefragt wurde, ob es sich bei der geflogenen Verbindung um einen Direktflug mit oder ohne Zwischenlandung handelte.

Des Weiteren kann die Auslastung eines PKWs bei

einer tatsächlichen Dienstreise höher sein, als die hier angegebenen 1,5 Personen pro PKW. Die Reichweite des Fahrzeugs erhöht sich, umso mehr Personen im Fahrzeug sitzen. Dies ist allerdings eine Information, die im betrachteten Use-Case bei der Reiseplanung einer konkreten Dienstreise durchaus zur Verfügung steht und als zusätzlicher Parameter vom Benutzer abgefragt werden könnte. Damit kann dann ein für die aktuell untersuchte Dienstreise spezifischer Emissions-Wert und damit eine spezifische potenzielle Reichweite leicht ermittelt werden.

5 Fazit und Ausblick

Trotz der genannten Einschränkungen ist der Unterschied der möglichen Strecke zwischen Flugzeug und anderen Verkehrsmitteln, speziell zum Reisebus und Zug, deutlich zu erkennen. Mit der Treibhausgasmenge, welche bei einem Flug von Berlin nach München freigesetzt wird, würde der Zug bis nach Lissabon und der Reisebus bis weit hinter Moskau ins Innere von Russland kommen. Dieses Projekt möchte niemanden dazu auffordern, statt mit dem Flugzeug nach München, lieber mit dem Bus nach Lissabon zu fahren. Allerdings haben erste Präsentationen im Rahmen der Hochschule gezeigt, dass diese Art der Visualisierung durchaus das Problembewusstsein fördert, indem es anschaulich aufzeigt, welches Einsparpotenzial an Treibhausgasen bei dem Verzicht auf Flugreisen vorhanden ist.

Im Weiteren ist daher geplant, eine im Funktionsumfang reduzierte, jedoch für die jeweils aktuell zu planende Dienstreise individuell parametrisierbare Version zu erstellen und diese Version standardmäßig bei jeder Beantragung einer Flugreise zur Verfügung zu stellen.

6 Literaturverzeichnis

- Fronk, M., Güccük, A., Höhne, M., Motuz, A., & Zagorski, A. (2019). Erfassung und Auswertung der mit Dienstreisen verbundenen Umweltauswirkungen der HTW Berlin. In J. Wittmann, *Simulation in den Umwelt- und Geowissenschaften, Workshop Kassel 2019* (S. 15-26). Shaker: Aachen.
- Galton, F. (31. 03. 2020). *Wikipedia*. Von https://en.wikipedia.org/wiki/File:Isochronic_Passage_Chart_Francis_Galton_1881.jpg abgerufen

- Geofabrik. (09.. 07. 2019). *OpenStreetMap Europe*. Von <https://download.geofabrik.de/europe.html>. abgerufen
- HTW-Berlin. (09.. 07. 2019). *Mobilität*. Von [htw-berlin.de/einrichtungen/zentrale-hochschulverwaltung/technische-dienste/organisation-atd/umweltmanagement/aktivitaeten-und-tipps/mobilitaet/](https://www.htw-berlin.de/einrichtungen/zentrale-hochschulverwaltung/technische-dienste/organisation-atd/umweltmanagement/aktivitaeten-und-tipps/mobilitaet/) abgerufen
- HTW-Berlin. (31.. 03. 2020). *Umweltleitlinien*. Von https://www.htw-berlin.de/fileadmin/HTW/Zentral/ZHV_IIQM_-_Qualitaetsmanagement/08_Umweltleitlinien_final.pdf. abgerufen
- Umweltbundesamt. (31.. 03. 2020). *Emissionsdaten*. Von <https://web.archive.org/web/20190718134549/https://www.umweltbundesamt.de/themen/verkehr-laerm/emissionsdaten> abgerufen

Applying Simulation to Advance Resilience of Historic Areas to Climate Change and Natural Hazards

Katharina Milde^{1*}, Sonia Giovinazzi², Daniel Lückerrath¹, Oliver Ullrich¹, Maurizio Pollino²,
Erich Rome¹, Vittorio Rosato²

¹Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, Schloss Birlinghoven, Sankt Augustin, Germany; *katharina.milde@iais.fraunhofer.de

²Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA), Lungotevere Thaon di Revel, 76, 00196 Rome, Italy

Abstract. The EU-H2020 project *ARCH* aims to develop and adapt tools and methods for assessing and improving the resilience of historic areas to climate-related and other natural hazards [1]. One of these tools is CIPCast, a scenario simulation and decision support system for the analysis and forecast of risks and vulnerabilities of critical infrastructure components and their interdependencies. In this paper, we describe the basic functionalities of CIPCast, as far as the application to seismic risk assessment is concerned and we provide an overview of the models behind it. Furthermore, a brief discussion on how we plan to extend CIPCast to model and simulate potential risks and impacts induced by climate change to historic areas, and how this is intended to support resilience assessment strategies, is provided in the conclusions.

Introduction

Historic towns, old urban quarters, villages and hamlets, as well as historic landscapes make up a significant part of Europe: Natural heritage sites cover roughly 18% of the European land territory [2] and on average 22% of the European housing stock was constructed before 1946 [3]. These historic areas are deeply embedded in larger urban and rural environments (in which 72% of the European population live [4]), serving a role in preserving local identity and personality as well as local knowledge, while relying on interdependent infrastructure services to keep functioning. Historic areas are a major component of quality of life and play an important role in society and community well-being [5], as well as providing important environmental and economic functions.

Although climate change has become one of the most significant and fastest growing threats to people and their cultural heritage [6] the impacts of climate-related and

other natural hazards on historic areas have not been studied extensively enough [7], and disaster risk reduction seldomly registers as a priority area for management of World Heritage property [8].

Therefore, there is a need for specific methods and tools for climate change adaptation and disaster risk reduction that take the unique physical, environmental, economic, social, cultural, and governance aspects of historic areas, as well as the enabling conditions they provide for taking action into account.

The EU Horizon 2020 research project *ARCH* (*Advancing resilience of historic areas against climate-related and other hazards*) [1] aims to take a step in this direction by providing a suite of tools for assessing and improving the resilience of historic areas, combined within a unified disaster risk management framework.

One of the tools developed within the project is an extension of the scenario simulation and decision support system CIPCast [9] in order to enable the assessment of impacts and risks to historic areas induced by climate change and natural hazards. This is an essential input for assessing the resilience of historic areas and identifying suitable resilience building strategies.

This paper gives an overview on how CIPCast functions and describes the extensions necessary to maximize its utility in the project context. The first section (sec. 1) gives a brief introduction to the *ARCH* project, followed by a general overview of the basic functionalities of CIPCast (sec. 2) and how these can already be employed to assess damage and impacts induced by seismic hazards (sec. 3). Following these explanations, the planned extensions of CIPCast (sec. 4), and how its results supports re-

silience assessments (sec. 5) are described, before the paper closes with conclusions and an outlook (sec. 6).

1 The ARCH project

Advancing resilience of historic areas against climate-related and other hazards (ARCH) is an EU Horizon 2020 research project that aims to better protect historic areas from climate-related and other natural hazards induced risks. The project started in June 2019 and will run until May 2022.

Within a co-creation process, the project team of eleven research partners and the cities of Bratislava, Camerino, Hamburg, and València will create tools and methods to provide cities with better information and decision support for improving the resilience of historic areas. The results will be applied in pilot sites within the cities covering a diverse spectrum of historic areas: the historic old towns of Bratislava and Camerino, the Devin Castle ruin in Bratislava, the Speicherstadt and Kontorhaus World Heritage sites in Hamburg, as well as the La Huerta peri-urban farmland and Albufera national park in València. These areas are affected by a multitude of different hazards, amongst them earthquakes, heat-waves, fluvial and pluvial flooding, storm surges, erosion, and landslides.

The technical work in ARCH includes the preparation of a hazard object information management system that captures data on hazards and object conditions using newly deployed sensors and readily available open data platforms; an impact risk assessment framework that provides methods and tools for risk and impact assessment, including hazard models and scenario simulation for what-if analyses; the design of implementation pathways that identify potential resilience measures enriched with effectiveness scores, supported via a tool for graphical implementation planning; and a multi-stakeholder resilience assessment framework integrating the methods and tools as well as a platform for collaboration and sharing.

The remainder of this paper focuses on describing the simulation and decision support system adapted within the project and how this will be employed for the resilience assessment.

2 CIPCast Simulation and Decision Support System

CIPCast is a GIS-based Decision Support System (DSS)

developed as part of the EU-funded FP7 project *CIPRNet* (*Critical Infrastructures Preparedness and Resilience ResearchNetwork*) [10]. CIPCast provides a database, an interoperable platform and a user-friendly WebGIS interface. These are conceived as a combination of free/open source software environments, for the real-time and operational (24/7) monitoring and risk analysis of built and natural environments, with special focus on interdependent critical infrastructures (such as electric power, water, telecommunication and road networks) and buildings [9][11][23].

CIPCast is based on a four-layer architecture:

- Within the **data preparation layer** basic data is collected, harmonized and organised for the following processing step.
- In the **data repository layer**, data and metadata are stored in a geospatial database implemented in PostgreSQL/PostGIS.
- Within the **analysis and elaboration layer** stored data and metadata are managed and published online to enable geo-processing and risk analysis.
- Within the **front-end layer**, data and functions from the previous layers are exposed to end-users via a WebGIS application.

Within this architecture, CIPCast provides five distinct functional blocks that feed each other:

- **B1 – Monitoring of Natural Phenomena** acquires data from different data sources.
- **B2 – Prediction of Natural Events** houses different hazard models to estimate the expected intensities for predictable events.
- **B3 – Prediction of Damage Scenarios** correlates the (estimated) hazard intensity with the vulnerability of elements located in an affected area to estimate potential direct damages (e.g. breakage of a transformer in an electric substation).
- **B4 – Prediction of Impacts and Consequences** correlates the potential direct damages to exposed elements with their (inter-)dependencies with other elements and the general system characteristics to estimate larger consequences (e.g. loss of service in an electrical network).
- **B5 – Support of efficient strategies** enables what-if analysis of different strategies to counter the effects of examined hazards.

The CIPCast **GeoDatabase** stores data related to ex-

posed elements and hazards. For seismic hazards the database includes information on epicenter location, hypocenter depth and magnitude; for exposed elements it includes both static data, like structural characteristics of buildings, and dynamic data, like population dynamics. The data model used in the GeoDatabase differentiates between different classes of exposed elements, e.g. networks, like telecommunication, electricity, transport networks, and groups of buildings. Detailed information for exposed elements is stored to estimate vulnerabilities, e.g. build material, construction age, and number of inhabitants.

Currently, CIPCast includes hazard data collected from seismic sensors, weather stations (for precipitation, temperature, humidity, wind, etc.), and hydrometers (for inundation levels of river basins).

3 CIPCast-ES for Seismic Risk Assessment in Italy

CIPCast-ES is an extension of CIPCast specifically aimed at simulation of seismic hazards and at the assessment of related physical damage and impact scenarios [17]. This section provides an overview of the models embedded within CIPCast-ES that enable these functions and some explanatory case studies.

3.1 CIPCast-ES Seismic Hazard assessment

To allow the assessment and representation of *ground motion* and *earthquake-induced geotechnical hazards*, available data, layers and information were collated in the **GeoDatabase**.

This data was sourced from previous studies as well as from external web services. They include services provided by the Italian National Earthquake Center (<http://cnt.rm.ingv.it/en>) managed by the Italian national Institute of Geophysics and Volcanology INGV; hydrogeological risk maps provided by “Idrogeo” (<https://idrogeo.isprambiente.it/>), and a web platform on landslide and flood risk provided by the Italian Institute for Environmental Protection and Research ISPRA.

For assessing *ground motion* hazards CIPCast-ES includes the following data: known faults locations (see Figure 1); catalogues of historical earthquakes; and seismic microzonation maps. The latter provide, at the local scale, spatial information about the effect of the local geological conditions on ground-shaking.

For assessing *earthquake-induced geotechnical hazards* CIPCast-ES includes the following data: surface faulting; seismic-induced landslide potential (see Figure 1); seismic-induced rock-fall potential; liquefaction potential; and potential for permanent soil deformation.

Based on this data a *seismic hazard simulation* allows to model and represent the location, extension and intensity of expected ground shaking generated by real or user-defined (artificial) events.

The simulation of real events is undertaken to support emergency management. In this case, a quasi-real time estimation of the extent and severity of the seismic ground shaking after an earthquake event is fundamental to provide a rapid, efficient and effective response. The simulation of end-user defined events is instrumental to support risk mitigation planning as explained in Section 3.3.

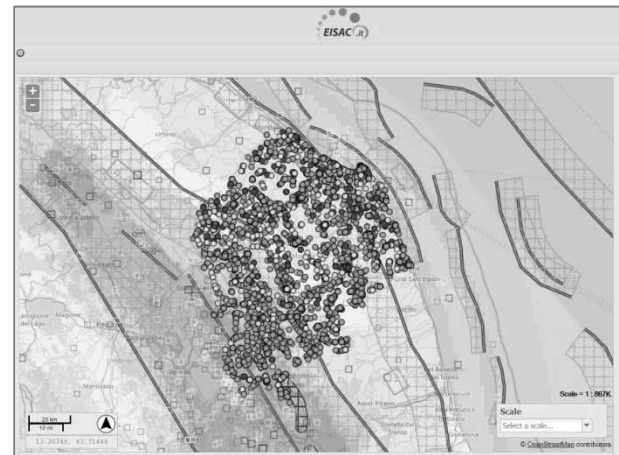
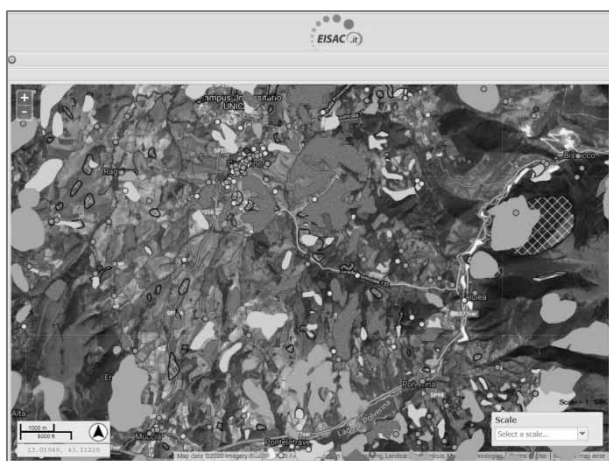


Figure 1: CIPCast-ES screenshot showing: seismic probabilistic hazard map and known-fault location maps from INGV, overlaid with cultural heritage assets (point locations).

In both cases the required inputs are:

- the location of the epicentre, i.e. latitude and longitude, X_E , Y_E ;
- the depth of the hypocentre in kilometre D_H [km]; and
- the magnitude, M , expressed according to the Richter scale.



For real events, location and magnitude of any seismic event with a magnitude larger than $M=3$ are acquired automatically and represented in real time within CIP-Cast-ES.

For end-user defined events, the user provides the relevant parameters, usually based on a catalogue of historic events and known fault locations (Figure 4), both accessible in CIPCast-ES.

Once the parameters are defined CIPCast-ES calculates where, to what extent and with which intensity ground shaking will propagate using *Ground Motion Prediction Equations (GMPEs)*, or “attenuation” relationships. GMPEs provide a means of predicting the level of ground shaking and its associated uncertainty at any given site or location, based on magnitude, source-to-site distance (i.e. distance between the epicentre and the location of an exposed element), local soil conditions, topology of the fault mechanism, etc. GMPEs are empirical-based equations derived after post-processing of recorded accelerations or observed damages generated by historical earthquake events¹.

In CIPCast-ES different GMPEs can be selected by the end-users allowing for the calculation and representation of seismic hazard maps with different metrics, i.e.:

- *Macroseismic Intensity, I*, [24];
- *Peak Ground Acceleration PGA and Spectral Acceleration, S_a (T)*, [25];
- *Peak Ground Velocity, PGV*, [27]
- *Spectral Displacements S_d (T)*, [27].

The selection of the most appropriate metric to represent the seismic hazard depends on the focus of the analysis; for example, *PGA* and *S_a* (T) have been observed to be more appropriate when the focus of the analysis is the structural performance of above-ground structures such as buildings while *PGV* and *S_d* (T) are suitable when the focus is on buried infrastructures. *Macroseismic intensity* on the other hand, is a qualitative descriptor of the effects of an earthquake at a particular location, as evidenced by observed damage on the natural and built environment and by the human and animal reactions at that location. Although a qualitatively metric, it is still used when adopting empirical-based models for assessing seismic vulnerability, such as the one described in section 3.2 for residential and monumental buildings.

Figure 3 provides an example of the official ground motion map after the L'Aquila earthquake on April 6, 2009 at 03:32 CEST, represented in *PGA* [%g].

While official ground shaking maps are released by INGV around 45 minutes after an earthquake event occurs, the basic parameters necessary for simulation are usually published only a few minutes after an event. This allows to easily check and validate simulation results with actual event data.

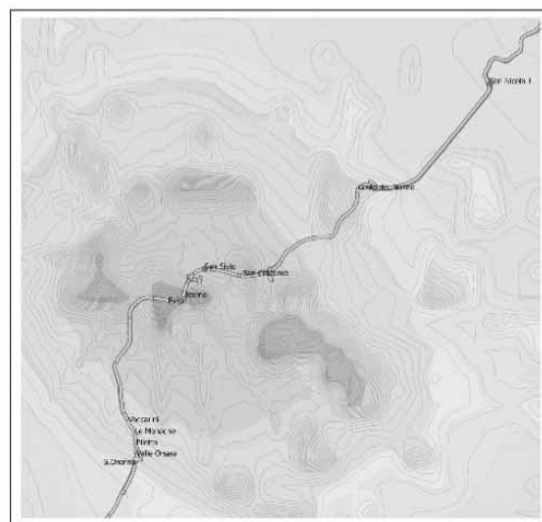


Figure 3: Official INGV ground motion map after April 6, 2009 L'Aquila earthquake overlaid with A24 Highway route.

For example, to simulate a ground-shaking map for the L'Aquila earthquake a user would choose the following parameters:

¹ An exhaustive compilation of GMPEs defined in the period

range 1964-2019 can be found here: <http://www.gmpe.org.uk>.

- epicentre location 42.3476° N, 13.3800 °E,
- magnitude $M = 6.3$,
- hypocentre D_H [km] = 9.46 km.

Because CIPCast-ES allows to calculate ground shaking maps in quasi real time it is able to support emergency management operations with first estimates of the location, extent and level of ground shaking in the affected territories. Simulated maps are substituted in CIPCast-ES with official maps, as soon as they become available.

3.2 CIPCast-ES Vulnerability and Physical Damage assessment for buildings

To allow the assessment and representation of built-environment elements and of potentially exposed population the **GeoDatabase** includes: Administrative borders (regions, municipalities and census tracks) and associated data on population (including gender, age, occupation, etc.) sourced from the Italian National Institute of Statistics; location and basic information on critical infrastructures like electrical transmission systems, gas transmission systems, main sources of electricity production, transport systems (road network, railways, airports), as well as the locations of strategic buildings like hospitals, barracks and schools.

Data about residential buildings is stored at single building level, when possible, or as aggregated data linked to geographical units otherwise. This includes data on construction age, construction material (masonry, reinforced concrete, timber, prefabricated,), type of structural system (e.g. frame versus shear walls for reinforced concrete buildings, bricks versus stones for masonry buildings), and adoption of seismic codes for design or retrofitting.

For the **Prediction of Damage Scenarios** (i.e. B3 Module) for buildings, the *Macroscopic-Mechanical cross-calibrated Method* [28][29] is implemented. This method allows to assess the seismic vulnerability of building groups, statistically aggregated in a geographical unit, and of single buildings as a function of their *seismic vulnerability* and of the *ground-motion* at their location (see Section 3.1).

The *seismic vulnerability* of buildings is measured by the *vulnerability index* V and the *ductility index* Q , calculated based on building typology and constructive, geometrical or additional features able to affect and modify building behaviour when subjected to earthquake shaking. One way to calculate the *vulnerability index* V is to combine a *basic vulnerability index* V^* and a *vulnerabil-*

ity index modifier ΔV , where V^* reflects the building typology and ΔV the sum of influencing features:

$$V = V^* + \Sigma \Delta V \quad (1)$$

Table 1 lists basic vulnerability indexes for different building categories (I to VII), construction materials (masonry or reinforced concrete) and construction periods. Figure 3 shows an example visualisation of vulnerability indexes on census tract level.

	Masonry	V*	RC	V*	
I	< 1919	0.79	-	-	
II	1919 - 1945	0.73	-	-	
III	1945-1971	0.69	V	< 1971	0.59
IV	> 1971	0.65	VI	1971-1981	0.55
	-	-	VII	> 1981	0.42

Table 1: Basic vulnerability indexes V^* for different building categories, construction periods and construction material. RC: Reinforced Concrete

Once the seismic vulnerability is assessed, the expected damage can be estimated as follows:

$$\mu_D = 2.5 \left[1 + \tanh \left(\frac{I + 6.25V - 13.1}{Q} \right) \right] \quad (2)$$

where:

- μ_D is the *average expected damage for a group of buildings or an individual building*;
- V is the value of the *vulnerability index*;
- Q is the *ductility index* assumed to be 2.3 for ordinary building categories like the ones from Table 1;
- I is the *Macroscopic Intensity* as described in the previous section.

$D_0 = \mu_D < 0.5;$	$D_3 = 2 \leq \mu_D < 3;$
$D_1 = 0.5 \leq \mu_D < 1;$	$D_4 = 3 \leq \mu_D < 4;$
$D_2 = 1 \leq \mu_D < 2;$	$D_5 = 4 \leq \mu_D \leq 5.$

Table 2: Categorization of expected damages based on [30]

In order to categorize the expected damage μ_D the EMS-98 damage grade scale [30] is applied. This scale differentiates between six different levels D_k : D_0 no damage, D_1 slight, D_2 moderate, D_3 heavy, D_4 very heavy, D_5 collapse/destruction. Table 2 lists how expected damages are categorized and Figure 4 shows an example visualization on census tracks level.

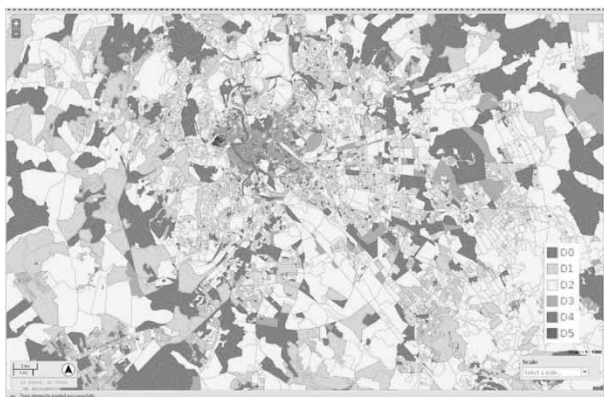


Figure 4: Example of the earthquake-induced damage assessment at census tract levels [29]

4 CIPCast Extensions for ARCH

In order to employ CIPCast in ARCH, the **GeoDatabase** will be extended to include maps and data representing historic areas and heritage buildings, classified into moveable heritage, archaeological resources, buildings and structures, cultural landscapes, associated and traditional communities and intangible heritage as described in [31].

To allow the assessment of climate change induced scenarios the **GeoDatabase** will need to be extended to include related hazard maps (e.g. for floods or extreme temperatures) under different climate scenarios. These maps will be derived from numerical simulations, climate hazard indicators, optical/thermal earth observation maps, high-resolution aerial and satellite maps, and from existing data services, e.g. Copernicus Climate Change Service [21].

4.1 CIPCast extensions for assessing damage to historic buildings

In order to estimate seismic damages to historical buildings the same function as for residential building (see equation 2) will be employed, using adapted V and Q index values. These values will be calibrated using earthquake-induced damage sustained by cultural heritage areas and buildings during the last twenty years.

It is important to note that a vulnerability index assigned to a monument simply by a typological classification represents an average value that does not account for the distinctiveness of the single building and does not allow singling out the most vulnerable structures among buildings of the same type. Therefore, the vulnerability assessment will be refined to reflect peculiar characteristics and features of historic buildings that might increase

or decrease their vulnerability, e.g. via a survey that collects relevant parameters like maintenance conditions, quality of materials, structural regularity (in plan and in elevation), size and slenderness of relevant structural elements, possible interaction with adjacent structures, presence of retrofitting interventions, etc.

4.2 CIPCast extensions for assessing climate change induced scenarios

To allow for the simulation of damage and impact scenarios induced by climate-related hazards CIPCast needs to be extended with the capability to manage additional input data and additional simulation modules. The basic framework for CIPCast-CC (*CIPCast Climate Change module*, also referred to as *ARCH DSS*), will be similar to CIPCast-ES, i.e. physical damage induced by climate change on the built environment in historic areas will be assessed as a function of hazard, exposure and vulnerability. Specifically, ARCH DSS will include

- models for index-based vulnerability assessment at area and single building level (e.g. compare to [32][33]);
- models for physical damage assessment that combine a) hazards parameters; b) position and typology of heritage; and c) heritage vulnerability; and
- models for the estimation of functional, economic and societal impacts.

For the latter, cause-effect models are necessary, which can for example be derived by developing impact chains in multi-stakeholder workshops [34].

5 Integration of CIPCast in the ARCH resilience assessment

The ARCH project adapts the Urban Adaptation Cycle [35] to describe the resilience management process of historic areas. One step in this process is the assessment of hazards, vulnerabilities, risks, and resilience. The resilience assessment is based on the UNDRR Disaster Resilience Scorecards for cities [22] and buildings [35]. As part of the resilience assessment, users need to identify the most relevant hazard and risk scenarios for the historic area that is being assessed and should formulate resilience enhancing measures to eliminate resilience weak spots. Here, CIPCast will be employed to enable users to identify and simulate hazard scenarios, assess potential impacts and identify the most suitable measures to raise the resilience.

The resilience assessment will be implemented as a web-based, semi-quantitative, multi-stakeholder self-assessment questionnaire that covers topics like governance processes to increase resilience, financing resilience, restoration and recovery for resilience, social justice in resilience management, and environmental issues in resilience. The resilience assessment is intended to guide users through this process, link to relevant tools at appropriate steps and support better coordination among relevant actors. The result of the resilience assessment will be given as a weighted resilience score for the historic area with linked resilience enhancing measures and additional information for decision-makers.

6 Conclusion

We described the planned use of modelling and simulation for assessing the resilience of historic areas against the impact of climate change and other extreme events. Aims and scope of the ARCH project were introduced, the CIPCast Simulation and Decision Support System, its planned extensions, as well as a brief application example were described.

As next research steps, CIPCast will be extended as described in section 4. The hazard models and simulation approaches will be integrated with a database of resilience building measures to support formulation and comparison of resilience building strategies. These functionalities will be integrated in a resilience assessment framework based on the UNDRR Disaster Resilience Scorecards for cities and buildings that include further – non-physical – resilience aspects (e.g. community resilience) to support the formulation of comprehensive resilience actions plans for historic areas.

Acknowledgements

This paper has been prepared in the framework of the European project ARCH – Advancing Resilience of historic areas against Climate-related and other Hazards. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 820999. The sole responsibility for the content of this publication lies with the authors. It does not necessarily represent the opinion of the European Union. Neither the EASME nor the European Commission are responsible for any use that may be made of the information contained therein.

References

- [1] ARCH Advancing Resilience of Historic Areas Against Climate-related and Other Hazards, <https://savingculturalheritage.eu/>, accessed on Sep. 15, 2020.
- [2] European Commission, *Europe's Cultural and natural Heritage in Natura 2000*. Publications Office of the European Union, Luxembourg, 2018 .
- [3] Nicol S., Roys M., Ormandy D., Ezratty V. *The cost of poor housing in the European Union*. University of Warwick, 2016.
- [4] Nabielek K., Hamers D., Evers D. *Cities in Europe – Facts and Figures on cities and urban areas*, PBL Publishers, The Hague, 2016.
- [5] Tweed C., Sutherland M. Built cultural heritage and sustainable urban development. *Landscape and urban planning*, 2007, 83(1): 62-69.
- [6] ICOMOS, 19GA 2017/30, Resolutions of the 19th General Assembly, 11.-15.12.2017, New Delhi, India.
- [7] Bigio A. G., Ochoa M. C., Amirtahmasebi R. *Climate-resilient, Climatefriendly World Heritage Cities*. Urban Development Series Knowledge Papers, Bd. 19, World Bank, Washington, DC, 2014.
- [8] Global Platform for Disaster Risk Reduction, *Heritage and Resilience. Issues and Opportunities for reducing disaster risks*, Global Platform for Disaster Risk Reduction, Geneva, Switzerland, 2013.
- [9] Di Pietro, A., Lavalle, L., La Porta, L., Pollino, M., Tofani, A., Rosato, V.: *Design of DSS for Supporting Preparedness to and Management of Anomalous Situations in Complex Scenarios*. in: Setola R., Rosato V., Kyriakides E., Rome E. (Eds.): *Managing the Complexity of Critical Infrastructures, A Modelling and Simulation Approach*, Springer, 2016, pp 195-232.
- [10] Rome E., Doll T., Rilling S., Sojeva B., Voÿ N., Xie J.: *The Use of What-If Analysis to Improve the Management of Crisis Situations Chapter 10*, in: Setola R., Rosato V., Kyriakides E., Rome E. (Eds.): *Managing the Complexity of Critical Infrastructures, A Modelling and Simulation Approach*, Springer, 2016.
- [11] Giovinazzi, S., Pollino, M., Ciarallo, F., Rosato, V., Luigi La Porta, L., Di Pietro, A., Clemente, P., Buffarini, G., (2019). *A Decision Support System for the Emergency Management of Highways in the Event of Earthquakes*. ANIDIS XVIII, Ascoli Piceno, Settembre 2019.
- [12] INSPIRE Knowledge Base, <https://inspire.ec.europa.eu/>, accessed on March 4, 2020.
- [13] D'Alessandro, A., Costanzo, A., Ladina, C., Buongiorno, F., Cattaneo, M., Falcone, S., La Piana, C., Marzorati, S., Scudero, S., Vitale, G., Stramondo S., and Doglioni C.. *Urban Seismic Networks, Structural Health and Cultural Heritage Monitoring: The National Earthquakes Observatory (INGV, Italy) Experience*. Front. Built Environ., 05 November 2019.

<https://doi.org/10.3389/fbuil.2019.00127>

- [14] *RAMSES Science for cities in transition*, <https://ramses-cities.eu/home/>, accessed on March 4, 2020.
- [15] Giovinazzi, S., Pollino, M., Kongar, I., Rossetto, T., Caiaffa, E., Di Pietro, A., La Porta, L., Rosato, V., Tofani, A., *Towards a Decision Support Tool for Assessing, Managing and Mitigating Seismic Risk of Electric Power Networks*. Computational Science and Its Applications - ICCSA 2017. In: Lecture Notes in Computer Science, Part III, LNCS 10406, pp. 399-414. Springer International Publishing, 2017a.
- [16] Lagomarsino, S., Giovinazzi, S., *Macroseismic and mechanical models for the vulnerability and damage assessment of current buildings*. Bull. Earthq. Eng. 4, 415-443, 2016.
- [17] Matassoni, L., Fiaschi, A., Giovinazzi, S., Pollino, M., La Porta, L., Rosato, V., *A geospatial decision support tool for seismic risk management: Florence (Italy) case study*. Computational Science and Its Applications - ICCSA 2017. In: Lecture Notes in Computer Science. Part II, LNCS 10405, pp. 278-293, Springer International Publishing, 2017.
- [18] D'Agostino, G., Di Pietro, A., Giovinazzi, S., La Porta, L., Pollino, M., Rosato, V., Tofani, A., *Earthquake Simulation on Urban Areas: Improving Contingency Plans by Damage Assessment*. In: Luijckx, E., Zaitouni, I., Hammerli, B. (eds) Critical Information Infrastructures Security. CRITIS 2018. Lecture Notes in Computer Science, vol 11260, 72-83. Springer, Cham, 2019.
- [19] Dolce, M., Nicoletti, M., De Sortis, A., Marchesini, S., Spina, D., and Talanas, F. *Osservatorio sismico delle strutture: the Italian structural seismic monitoring network*. Bull. Earthq. Eng. 15, 621-641. doi: 10.1007/s10518-015-9738-x, 2017.
- [20] Giovinazzi, S., Di Pietro, A., Mei, M., Pollino, M., Rosato, V.: *Protection of Critical Infrastructure in the Event of Earthquakes: CIPCast-ES. Proc. XVII ANIDIS Conference*, Pistoia, Italy, 2017, pp. 62-70.
- [21] *Copernicus Climate Change Service*, <https://climate.copernicus.eu/>, accessed March 4, 2020.
- [22] United Nations office for Disaster Risk Reduction: *Disaster Resilience Scorecard for Cities*, <https://www.undrr.org/publication/disaster-resilience-scorecard-cities>, 2017.
- [23] Taraglio, S., Chiesa S., La Porta, L., Pollino, M., Verdecchia, M., Tomassetti, B., Colaiuda, V., Lombardi, A., *Decision Support System for smart urban management: resilience against natural phenomena and aerial environmental assessment*. International Journal of Sustainable Energy Planning and Management, Vol. 24, 2019.
- [24] Faccioli, E., Cauzzi, C., *Macroseismic intensities for seismic scenarios estimated from instrumentally based correlations*. In: Proc. of the First European Conference on Earthquake Engineering and Seismology - Geneva, Switzerland, 3-8 September 2006. ECEES, 2006.
- [25] Ambraseys, N.N., Simpson, K.A. And Bommer, J.J.. Prediction of horizontal response spectra in Europe. Earthquake Eng. Struct. Dyn., 25: 371-400, 1996.
- [26] Sabetta, F., Pugliese, A., *Estimation of Response Spectra and Simulation of Nonstationary Earthquake Ground Motions*. Bull. Seismol. Soc. Am. 86, 337-352, 1996.
- [27] Cauzzi, C., Faccioli, E., *Broadband (0.05 to 20 s) prediction of displacement response spectra based on worldwide digital records*. J Seismol 12, 453, 2008.
- [28] Giovinazzi S., *The vulnerability assessment and the damage scenario in seismic risk analysis, Ph.D Thesis of the doctoral course "Risk Management on the built environment"* jointly organized by University of Florence (I) and TU-Braunschweig (D), 2005.
- [29] Lagomarsino, S., Giovinazzi, S. *Macroseismic and mechanical models for the vulnerability and damage assessment of current buildings*. Bull Earthquake Eng 4, 415-443 (2006). <https://doi.org/10.1007/s10518-006-9024-z>
- [30] Grünthal G (ed) *European Macroseismic Scale 1998* (EMS-98). Cahiers du Centre Européen de Géodynamique et de Séismologie 15, Centre Européen de Géodynamique et de Séismologie, Luxembourg, 99 p, 1998.
- [31] V. Rebollo, V. Latinos, I. Balenciaga, R. Roca. *ARCH D7.2 Mapping and characterisation of good practices in cultural heritage resilience*. ARCH H2020 Project GA No. 820999, <https://savingculturalheritage.eu/resources/deliverables> (accessed on September 15, 2020), 2020.
- [32] RESIL, KP, NCRS, ENG, INOV, TROIA, BU, UWA, *D6.6: STORM Damage Assessment and Decision Support Services*, STORM H2020 Project Grant Agreement No.: 700191, 2016.
- [33] A Chiabrando, F., Colucci, E. & Lingua, A., Matrone, F., Noardo, F., Spano, A., *A European Interoperable Database (EID) to increase resilience of cultural heritage*. ISPRS Journal of Photogrammetry and Remote Sensing. XLII. 151-158. 10.5194/isprs-archives-XLII-3-W4-151-2018, 2019.
- [34] Lückerrath, D., Streberova, E., Bogen, M., Rome, E., Ullrich, O., Pauditsova, E.: *Climate Change Impact and Vulnerability Analysis in the City of Bratislava: Application and Lessons Learned*. In: Nadjm-Tehrani S. (ed) *Critical Information Infrastructures Security*. CRITIS 2019. Lecture Notes in Computer Science, Vol. 11,777. Springer, 2019, pp. 83-94
- [35] *European Climate Adaptation Platform Climate-ADAPT* <https://climate-adapt.eea.europa.eu/knowledge/tools/urban-ast>, (accessed Sep 15, 2020), partnership between the European Commission and European Environment Agency.
- [36] UNDRR ARISE, *Disaster resilience scorecard for industrial and commercial buildings. For use by building owners, operators and managers*, <https://www.preventionweb.net/publications/view/69845> (acc. Sep 14, 2020)

Modellierung der Ausbreitung von Baumschädlingen nach aerochemischer Insektizidanwendung mit den Mitteln eines Geoinformationssystems

Colja Krugmann¹, Majdi Abusaleh¹, René Krüger¹, Jochen Wittmann¹

¹ Hochschule für Technik und Wirtschaft Berlin, Umweltinformatik, Wilhelmshofstraße 75A, 12459 Berlin, Germany, wittmann@htw-berlin.de

Abstract. Im Rahmen eines Studentenprojektes für das Modul Umwelt- und Geoinformationssysteme greift der vorliegende Beitrag die Diskussion um eine aerochemische Bekämpfung von Waldschädlingen auf und versucht allein auf der Basis von öffentlich zugänglichen Daten ein Modell zu entwickeln, das die für eine Besprühung erlaubten Flächen aufzeigt, das aber darüber hinaus auch versucht, im Rahmen der Entwicklung eines dynamischen Modells die erneute Ausbreitung der Waldschädlinge nach der Bekämpfungsmaßnahme einzuschätzen. Grundlegende Annahme für diese Modellierung ist die Tatsache, dass bei einer Besprühung Schutzzonen um besiedeltes Gebiet aber auch um Oberflächengewässer und Waldränder eingehalten werden müssen, die nicht behandelt werden dürfen und damit zu Zonen werden, in denen die Schädlinge überleben. Nimmt man nun noch eine bestimmte Dauer für den Vermehrungszyklus der Schädlinge sowie einen mittleren Verbreitungsradius an, kann man mit einfachen Methoden eines Geoinformationssystems die dynamische Ausbreitung der Schädlingspopulation über mehrere Vermehrungszyklen simulieren und darstellen. Das Paper zeigt als Machbarkeitsstudie, dass ein solches, einfaches Modell sinnvolle Ergebnisse liefern kann und stellt für konkrete Anwendungen eine Liste mit den unbedingt notwendigen Parametern auf, die erhoben werden müssen, um eine praxisnahe Simulation durchführen zu können.

1 Motivation

2019 wurde in Brandenburg seit langem wieder in Erwägung gezogen, aerochemisch mit “Karate Forst flüssig” von Syngenta gegen Waldschädlinge, in erster Linie

gegen die Nonne, vorzugehen. Von Seiten von Umweltschützer*innen und Anwohner*innen gab es regen Widerstand (Mit „Karate Forst“ gegen Raupen in Brandenburgs Wäldern, 2019). Brandenburgs Wälder bestehen zu 70% aus Kiefern. Diese Monokultur zusammen mit den dürreren Sommern der vergangenen Jahre erhöhte die Anfälligkeit der Bäume für Schadorganismen wie Kiefernspinner, Kiefernspanner, Kiefernbuschhornblattwespe, Forleule und Nonne. Das Waldmonitoring prognostizierte bei günstigen Bedingungen für die Schädlinge wie warmes und trockenes Wetter eine große Gefahr des Kahlfraßes. Kiefern können einen einmaligen Kahlfraß von ca. 90% mit wenig Verlusten überstehen, kommt es jedoch zu stärkeren Schäden ist von einem Absterben des Waldes auszugehen (Landesbetrieb Forst Brandenburg (Hrsg.), kein Datum). Auch in Bezug auf den CO₂-Haushalt stellen die Wälder einen wesentlichen Faktor dar und ein Kahlfraß würde die Forste von einer positiv zu bewertenden CO₂-Senke in eine für die Bilanz negative CO₂-Quelle verwandeln. Dem allen wollte das Landeskompetenzzentrum Forst Eberswalde (LFE) mit aerochemischer Schädlingsbekämpfung mit Karate Forst flüssig von Syngenta begegnen (Landesbetrieb Forst Brandenburg (Hrsg.), kein Datum). Bei diesem Insektizid handelt es sich um ein Kontaktgift, das hoch verdünnt in die Baumkronen der befallenen Gebiete ausgebracht wird. Es tötet die Nonne im Raupenstadium (Landesbetrieb Forst Brandenburg (Hrsg.), kein Datum).

Naturschützer zweifelten stark an der Sinnhaftigkeit und Umweltverträglichkeit dieser Vorgehensweise, da auch die natürlichen Feinde der Schädlinge mit betroffen

wären. Laut LFE kann die Konzentration aber gerade so hoch eingestellt werden, dass allein die Nonnenraupen abgetötet werden und nicht deren Gegenspieler oder beispielsweise Maikäfer oder andere Laufkäfer.

Die vorliegende Machbarkeitsstudie greift an dieser Stelle in die Diskussion ein. Es soll geprüft werden, ob man durch eine Modellentwicklung, die ausschließlich auf öffentlich zugänglichem Datenmaterial beruht, Aussagen über die Auswirkungen einer aerochemischen Besprühung auf die Entwicklung des Schädlingsbefalls machen kann. Besondere Berücksichtigung bei der Modellierung soll die Bedeutung der Schutzzonen finden, die bei der Ausbringung des Insektizids um Siedlungen, Gewässer usw. eingehalten werden müssen und damit Rückzugsgebiete für die Schädlingspopulation darstellen aus denen heraus sich diese sowohl zahlenmäßig erholen als auch neu verbreiten kann.

2 Modellidee und Aufbereitung für eine GIS-Analyse

Die Grundidee der vorliegenden Arbeit ist es zunächst, ausschließlich öffentlich verfügbares Kartenmaterial als Basisdaten für den Anwendungsfall aufzubereiten, das die für eine Besprühung erlaubten und geeigneten Flächen aufzeigt, das aber darüber hinaus auch versucht, im Rahmen der Entwicklung eines dynamischen Modells die Verbreitung der Waldschädlinge nach der Bekämpfungsmaßnahme einzuschätzen. Grundlegende Annahme für diese Modellierung ist die Tatsache, dass bei einer Besprühung Schutzzonen um besiedeltes Gebiet aber auch um Oberflächengewässer und Waldränder eingehalten werden müssen, die nicht behandelt werden dürfen und damit als Rückzugsgebiete für die Schädlingspopulation dienen können. Nimmt man nun noch eine bestimmte Dauer für den Vermehrungszyklus der Schädlinge sowie einen mittleren Verbreitungsradius an, soll allein mit den Standardmethoden zur räumlichen Analyse, die ein Geoinformationssystem zur Verfügung stellt, die dynamische Ausbreitung der Schädlingspopulation über mehrere Vermehrungszyklen simuliert und dargestellt werden. (s. Abb. 1)

Ziel der Modellierung ist es dabei, den Einfluss und die Bedeutung der Schutzzonen herauszuarbeiten und einer quantitativen Untersuchung zu öffnen. Denn sicherlich wird der Insektizideinsatz in einer topographisch stark strukturierten und damit stark mit Schutzzonen

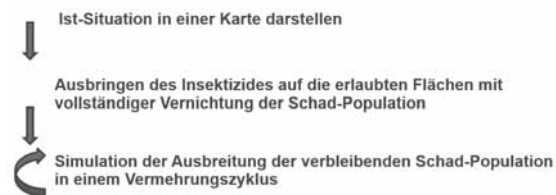


Abb. 1: Modellidee

durchsetzten Landschaft andere Effekte zeigen als in einer großflächig strukturierten Landschaft, die weniger Schutzzonen ausweist.

Mit dieser Modellidee als Basis stellen sich für die Analyse im Geoinformationssystem die folgenden Teilaufgaben:

1. Finden und Aufbereiten geeigneter öffentlich zugänglicher Geobasisdaten
2. Ermittlung der für die Modellierung notwendigen Parameterwerte
 - a. Bezüglich der Anwendungshinweise des Insektizids
 - i. Schutzzone um Siedlungen
 - ii. Schutzzone um Gewässer
 - iii. Schutzzone an Waldrändern
 - b. Bezüglich der biologischen Daten des Schädlings
 - i. Dauer eines Vermehrungszyklus
 - ii. Ausbreitungsreichweite in einem Vermehrungszyklus

3 Der Analyse-Workflow

Nach der Veranschaulichung der Methode folgt nun die Schilderung der einzelnen Arbeitsschritte. Als Erstes wurde ein Wegenetz benötigt. Als Zweites musste der Grenzwert bzw. Maximalwert an Emissionen berechnet werden, der den Erreichbarkeitsgraphen für die Transportmittel als Limit dienen soll. Im Dritten und letzten Schritt konnten die Erreichbarkeitsgraphen dann erstellt werden.

3.1 Schritt 0: Grundkarte und Geobasisdaten erstellen

Die für die Machbarkeitsstudie getroffene Beschränkung auf öffentlich zugängliches Kartenmaterial stellt eine gewisse Herausforderung dar, die einerseits zwar zusätzliche Arbeitsschritte erfordert, andererseits jedoch

die Unabhängigkeit des Modellansatzes von bereits vorliegendem, jedoch nur beschränkt und/oder unter Auflagen verfügbarem Datenmaterial unterstreicht.

Als Beispielfläche wurde eine Region in Brandenburg zwischen Bad Belzig und Werder (Havel) gewählt, für den sich die Datenlage folgendermaßen darstellt:

- Als Grundkarte wird ein Ausschnitt aus openstreetmap gewählt (OpenStreetMap, 2019)
- Die Gewässerdaten als Shape-Dateien stammen vom Geoportal Brandenburg (LGB (Landesvermessung und Geobasisdaten Brandenburg), 2019).
- Die Straßennetzdaten für Brandenburg stammen von Inspire Brandenburg und wurden über ein API in Form einer GML-Datei heruntergeladen (LGB (Landesvermessung und Geobasisdaten Brandenburg), 2019) .
- Die Walddaten stammen vom Landeskompetenzzentrum Forst Brandenburg und lagen als GML-Datei vor (Landesbetrieb Forst Brandenburg (Hrsg.), kein Datum).
- Idealerweise liegen Siedlungen und landwirtschaftlich genutzte Flächen als Shape-Dateien, in vorliegendem Beispiel allerdings nur als Lücken in den Forstgebieten vor.



Abb. 2: Skizze zu Schritt 1: zusammenhängende Waldflächen erzeugen

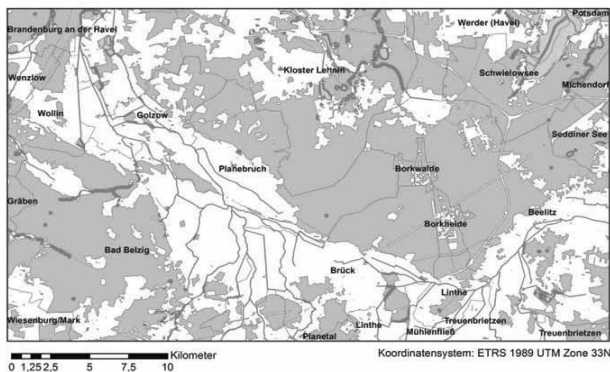


Abb. 3: Basiskarte nach Schritt 1

3.2 Schritt 1: zusammenhängende Waldflächen erzeugen

Innerhalb eines Forstes befinden sich Waldwege und kleinere Straßen, welche zu einer Unterbrechung der Forstfläche in den Geodaten führen und damit fiktive Waldränder darstellen, die als Schutzzonen relevant wären. In Bezug auf einen Insektizideinsatz zählen Waldwege jedoch nicht als Grenze eines Forstgebietes und müssen deswegen für dieses Projekt ignoriert werden, um mit geschlossenen Forstflächen weiterarbeiten zu können.

Um das zu erreichen, wird um die Forstgebiete ein Puffer gelegt, der die Waldwege überdeckt. Die daraus entstandenen überlappenden Waldpolygone werden anschließend zusammengeführt (Dissolve). Ein anschließender negativer Puffer um die gleiche Breite reduziert die Waldgebiete schließlich auf ihre ursprüngliche, reale Größe. (Skizze des Verfahrens Abb. 2, Basiskarte nach diesem Bearbeitungsschritt in Abb. 3)

3.3 Schritt 2: einheitliche Gewässerflächen mit Sicherheitsabstand erzeugen

Geodaten für Seen liegen in Form von Polygonen und für Flüsse und Kanäle in Form von Linien vor. Linien haben keine Breite, weshalb sie vor einer Vereinigung mit den Seeflächen zunächst in Polygone umgewandelt werden müssen. Hierfür wurde ein Puffer um die Flüsse als line-Feature erstellt. Anschließend können Seen und Flüsse zu einem Layer "Gewässer" zusammengeführt werden, der für die Bestimmung der Schutzzonen relevant ist. Sämtliche Gewässer-Features werden nun erneut um die Breite der Schutzzone gepuffert, damit Polygone entstehen, die die Gewässer inklusive ihres Sicherheitsabstands repräsentieren. (Abb. 4)

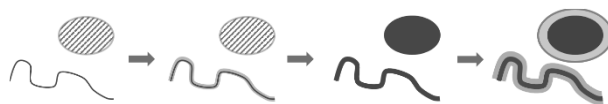


Abb. 4: Schritt 2: einheitliche Gewässerflächen mit Sicherheitsabstand erzeugen

3.4 Schritt 3: Sicherheitsabstand zu großen Straßen; Zusammenführung der Schutzzonen

Über größere Straßen, wie Bundes- und Kreisstraßen oder Bundesautobahnen darf nicht hinweg gesprüht werden. Die Geodaten für Straßen liegen wiederum als line-Features vor. Um Schutzzonen entlang von Straßen zu ermitteln werden diese wiederum entsprechend einer gemittelten Straßenbreite und des vorgeschriebenen Schutzabstands gepuffert. Diese Schutzzonen um Straßen werden anschließend mit den Schutzzonen um Gewässer zusammengeführt. (Abb. 5)



Abb. 5: Schritt 3: Sicherheitsabstand zu Straßen; Zusammenführung der Schutzzonen

3.5 Schritt 4: nicht besprühte Waldfläche

Um die Gebiete zu bestimmen, die nicht besprüht werden dürfen, wurde zunächst die besprühbare Waldfläche identifiziert. Die Waldpolygone aus Schritt 1 werden um den Sicherheitsabstand zum Waldrand negativ gepuffert. Diese Fläche wird mit den Flächen der Gewässer und großen Straßen aus Schritt 2 und 3 verschneitten. Wird das Ergebnis dieser Verschneidung nun noch durch eine symmetrische Differenz von der gepufferten Waldfläche abgezogen, erhält man eine Waldfläche, bei der alle Sicherheitsabstände berücksichtigt wurden, also die Waldfläche, die mit dem Insektizid besprüht werden darf. Jetzt kann man erneut eine symmetrische Differenz mit der gesamten Waldfläche bilden und man erhält schließlich diejenige Waldfläche, die nicht besprüht werden darf. (Abb. 6)



Abb. 6: Schritt 4: nicht besprühte Waldfläche ermitteln

3.6 Schritt 5: Ausbreitung des Schädlings

Man geht davon aus, dass die Schädlinge in dem Gebiet, in dem nicht gesprüht werden darf, vollständig überleben. Von diesem Gebiet aus können sie sich folglich erneut in die benachbarten, behandelten Gebiete ausbreiten. Dieser Ausbreitungsprozess kann durch einen Puffer um die nicht behandelten Gebiete modelliert werden. Die Pufferbreite entspricht dabei der Ausbreitungsgeschwindigkeit des Schädlings in einem Vermehrungszyklus. Da die angenommene Ausbreitung in alle Richtungen geht reicht sie über die Waldgrenzen hinaus, weshalb sie miteinander verschnitten werden müssen. Zudem kann es zu einer Überlagerung von Teilen der Ausbreitzungszone kommen, die durch ein Zusammenführen entfernt werden. (Abb. 7)

Nun kann diese gesamte Prozedur für genau einen Vermehrungszyklus wiederholt abgearbeitet werden, um die Abfolge und die Auswirkungen mehrerer Vermehrungszyklen zu simulieren.

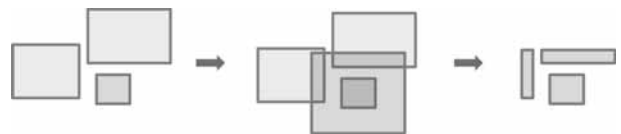


Abb. 7: Ausbreitung des Schädlings

4 Ergebnisse

4.1 Machbarkeitsstudie für die Beispielregion

Die im vorausgegangenen Abschnitt beschriebenen Analyseschritte wurden mit dem Model Builder in Esri ArcMap 10.5 als Analyseworkflow angelegt und auf die Beispielregion aus Abbildung 3 angewandt. Für die diversen in den Einzelschritten genannten Parameter wurden die Werte plausibel geschätzt, die exakten Parameterwerte sollen an dieser Stelle jedoch bewusst nicht genannt werden, da es sich bei dem Modell ausdrücklich um einen Prototypen handelt, der auf den prinzipiellen Analyseworkflow abzielt und die Ergebnisse daher zunächst nur einem Plausibilitätstest unterzogen werden können. Im nachfolgenden Unterabschnitt soll die Frage der Parametrisierung separat diskutiert werden.

Als Ergebnis der Modellierung und der Simulation

können daher zunächst nur drei qualitative Schlüsse gezogen werden: Erstens ist der beschriebene schrittweise Analyseworkflow mit den Standardmethoden von Arc-Map vollständig und transparent abbildbar und steht als frei parametrisierbares neues Tool zur Verfügung. Zweitens ist es möglich, eine einfache dynamische Simulation durch Iteration der in Schritt 5 erläuterten Ausbreitungsmethode zu implementieren. Und drittens führt das durch den Workflow beschriebene Vorgehen tatsächlich zu plausiblen Ergebnissen, wie aus der Abbildung 8 ersichtlich ist.

Man erkennt in der Ergebniskarte, dass bereits nach drei defensiv parametrisierten Vermehrungszyklen (gekennzeichnet durch unterschiedlich stark rot symbolisierte Pufferflächen um die Schutzzonen) ein großer Teil der Waldfläche ausgehend von den Schutzzonen neu besiedelt ist. Die Abhängigkeit von der Landschaftsstruktur wird durch das Modell offensichtlich nachvollziehbar.

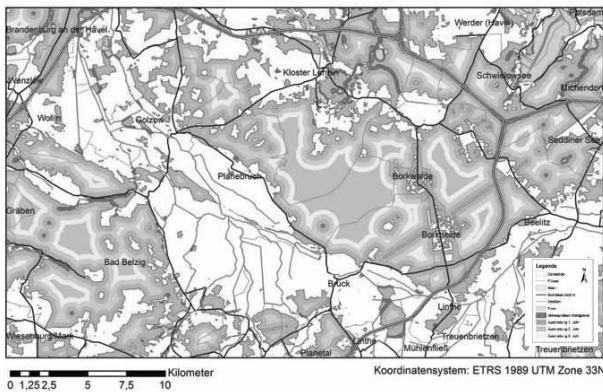


Abb. 8: Simulierte Situation für die Beispielregion nach drei Ausbreitungszyklen

4.2 Notwendige Parameterwerte für eine exakte Modellstudie

Wie bereits erwähnt, basiert das vorgestellte Modell auf öffentlich zugänglichen Daten, ist aber an einigen Stellen von Parameterwerten abhängig, die in der Machbarkeitsstudie lediglich plausibel geschätzt werden konnten für eine auch quantitativ verwertbare Modelluntersuchung aber mit validen Werten besetzt werden müssen. Allerdings ist anzumerken, dass die Modellierung darauf abzielte, dass es sich um eine verhältnismäßig kleine Zahl von Parametern handelt, die zudem relativ einfach zu bestimmen zu sein scheinen. Hier die Liste dieser Mo-

dellparameter, die für eine konkrete Anwendung zahlenmäßig zu besetzen sind:

1. Geobasisdaten:
 - a. Grundkarte (die Machbarkeitsstudie arbeitet bewusst ausschließlich mit öffentlich zugänglichem Material, hier sind sicherlich bessere thematische Basiskarten möglich)
 - b. Breite von kleinen Fließgewässern
 - c. Breite von Straßen
2. Anwendungshinweise des Insektizids:
 - a. Mindestabstand zu Waldrand
 - b. Mindestabstand zu Oberflächengewässern
 - c. Mindestabstand zu Siedlungen
 - d. Wirksamkeit
3. Daten zur Biologie des Schädlings:
 - a. Dauer eines Vermehrungszyklus
 - b. Ausbreitungsreichweite in einem Vermehrungszyklus

Dies ist die vollständige Parameterliste, die für das vorgestellte Modell benötigt wird. Selbstverständlich sind für eine detailliertere Modellierung noch viele andere Einflussfaktoren denkbar und sinnvoll. Bei der Beschäftigung mit dem Modell stößt man zum Beispiel sehr schnell auf den Einfluss des Wetters und die Zusammensetzung des Baumbestandes als zwei Größen, die eigentlich nicht unberücksichtigt bleiben dürften, die aber mit Blick auf das Ziel, zunächst nur die prinzipielle Machbarkeit zu demonstrieren, trotzdem vernachlässigt wurden.

5 Fazit

Die vorliegende Arbeit soll zeigen, dass bereits ein sehr einfacher Modellansatz aussagekräftige Ergebnisse generieren kann, die es erlauben, die Auswirkungen eines Eingriffs durch aerochemische Insektizidanwendung abzuschätzen und die Fläche der unbehandelten Waldgebiete zu bestimmen. Mit einer überschaubaren Menge von dazu benötigten Parameterwerten und einem sowohl algorithmisch als auch softwaretechnisch geringen Aufwand lässt sich durch Parametervariationen und Szenariountersuchungen die Diskussion sicherlich versachlichen.

Auf methodologischer Ebene zeigt das vorgestellte Modell, dass die häufig diskutierte Kopplung und/oder

Integration von Geoinformationssystem und Simulationssystem bei einem engen Anforderungsprofil einerseits und einem sehr pragmatischen Modellierungsansatz andererseits für Einzelprojekte pragmatisch und erfolgreich gelingen kann.

6 Quellen

Geofabrik. (09.. 07. 2019). *OpenStreetMap Europe*. Von <https://download.geofabrik.de/europe.html>. abgerufen

Landesbetrieb Forst Brandenburg (Hrsg.). (kein Datum). Abgerufen am 03. 04 2020 von Waldschutzmaßnahmen gegen Nonnenraupen: <https://forst.brandenburg.de/lfb/de/lfe/waldschutzinformationen/waldschutzmassnahmen-gegen-nonnenraupen/>

Landesbetrieb Forst Brandenburg. (17. 01 2018). *Forstgrunddaten - Flaechen - Waldbedeckung - Land Brandenburg*. Abgerufen am 23. 04 2019 von http://www.brandenburg-forst.de/inspire/dls/ifgk_wld/

LGB (Landesvermessung und Geobasisdaten Brandenburg). (2019). *Geoportal Brandenburg*. Abgerufen am 14. 05 2019 von <https://geoportal.brandenburg.de/startseite/>

Majunke, C., Möller, K., & Funke, M. (2004). Die Nonne (*Lymantria monacha* L., Lepidoptera, Lymantriidae). (L. Eberswalde, Hrsg.) *Walschutz-Merkblatt* 52.

Mit „Karate Forst“ gegen Raupen in Brandenburgs Wäldern. (26. 04 2019). Abgerufen am 03. 04 2020 von [rbb24.de: https://www.rbb24.de/panorama/beitrag/2019/04/brandenburg-beelitz-mit-karate-forst-fluessig-gegen-insekten.html](https://www.rbb24.de/panorama/beitrag/2019/04/brandenburg-beelitz-mit-karate-forst-fluessig-gegen-insekten.html)

OpenStreetMap. (2019). Von <https://www.openstreetmap.org> abgerufen

Modellbildung und Simulation als Grundlagenfach

Werner Maurer

Zürcher Hochschule für Angewandte Wissenschaften (ZHAW); werner@pegaswiss.ch

Zusammenfassung. Modellbildung und Simulation öffnen neue Türen, sobald der explorative Charakter dieser Methode erkannt und zum Kompetenzerwerb eingesetzt wird. Unsere Erfahrungen aus den letzten dreissig Jahren zeigen, wie sich die studentischen Aktivitäten vom Lösen von Standardaufgaben hin zur kreativen Auseinandersetzung mit den Grundgesetzen verschieben, wenn man die entsprechenden Aufgaben stellt und die Ergebnisse mündlich und schriftlich einfordert. Zur Hydrodynamik, Translationsmechanik und Thermodynamik wird nachfolgend je eine kurze Einführung in die Theorie gegeben und anhand einer Aufgabenstellung erläutert. Wie man all diese Strukturen in eine Modelica-Bibliothek einbringt und so für das weitere Studium nutzbar macht, wird mangels Erfahrung nur als Idee skizziert.

Einführung

Viele Studiengänge speziell an Fachhochschulen sind stark fragmentiert, was die Stoffmenge aufbläht und nach jedem Semester zu einem Prüfungsmarathon führt. In Folge der fortschreitenden Spezialisierung und unter dem Einfluss neuer Technologien sind die klassischen Fächer marginalisiert worden, ohne dass den neuen genügend Platz für eine angemessene Entfaltung eingeräumt worden ist. Wem nützen wenige Lektionen Kommunikation, wenn gleichzeitig in der Physik Sprache durch Formelhüpfen ersetzt wird?

Modellbildung ist ein zentrales Element wissenschaftlicher Tätigkeit. Die Fähigkeit, Daten und Fakten zu kausalen Mustern zusammen zu fassen, ist urmenschlich und generiert andauernd neue Wissenschaften und leider auch Scheinwissen wie Astrologie, Homöopathie oder Radiästhesie.

Visionäre Studiengänge wie Datenanalyse und Prozessdesign oder Aviatik stehen weniger unter Druck von Tradition und vorgegebenem Berufsbild. Entsprechend bieten sie Raum für neue Fächerkombinationen und grosszügige Stundendotation. Die nachfolgende Darstellung basieren auf den Erfahrungen im Fach "Physik und Systemwissenschaft für Aviatik". Dieses Modul, das von

Lehrkräften aus der Physik und dem Sprachbereich gemeinsam unterrichtet worden ist, hatte folgenden Aufbau

- Plenum: Flipped Classroom im Hörsaal
- Labor: Übungen und Modellbildung in Kleinklassen
- Kommunikation: Bericht, Vortrag, Poster, Reportage in Kleinklassen.

Ziel des vorliegenden Aufsatzes ist nicht ein Resümee zu diesem mittlerweile schon wieder fragmentierten Modul, sondern eine Vision zu Modellbildung und Simulation als integrierendes Element eines zeitgemässen Studienganges für Ingenieure, Naturwissenschaften und Medizinalberufe.

1 Hydrodynamik

Der herkömmliche Einstieg über die Punktmechanik in die Physik verhindert, dass bei den Studierenden kreative Prozesse aktiviert werden. Das Modell von Massenpunkten, die sich unter der Fernwirkung von Kräften in einem gegebenen Raum bewegen, war damals im Barock eine geniale Leistung, ist aber derart weit von unseren Erfahrungen entfernt, dass es von den Studierenden kaum verstanden wird [1]. Leider scheitert oft auch die Übertragung auf alltägliche Problemstellungen, wie das Beispiel der falsch eingezeichneten Kräfte bei der schiefen Ebene zeigt [2].

Dieser kaum zu behebbende Mangel der Newtonschen Mechanik sowie der in dieser Theorie nur implizit vorhandene Energiebegriff haben uns bewogen, über die Hydrodynamik in die Naturwissenschaften einzusteigen. Mit dem Volumen oder der schweren Masse als bilanzierfähige Mengen sowie dem Druck bzw. dem Gravitationspotential als zugehörige Energiebeladung stehen Grössen im Zentrum, die jeder kennt oder von denen eine ausbaufähige Vorstellung vorhanden ist. Im Gegensatz zur Punktmechanik oder zur klassischen Thermostatik werden nicht nur Speicher, sondern gleichberechtigt auch Stromglieder untersucht. Die Hydrodynamik liefert zudem das Vorbild für das Flüssigkeitsbild [3], eine arche-

typische Darstellung, die in vielen Zweigen der Naturwissenschaften eingesetzt werden kann.

Die Hydrodynamik ermöglicht einen intuitiven Einstieg ins systemdynamische Modellieren. Man kann darüber streiten, ob ein Räuber-Beute-Modell mit Schneehasen und Luchsen anschaulicher ist als ein Modell zu den drei Seen im Berner Seeland. Von der Physik über die Chemie bis zur Pharmakokinetik ist aber das Bilanzieren kontinuierlicher Mengen eine derart zentrale Tätigkeit, dass diese Fähigkeit nicht früh genug geübt werden kann. Im Studiengang Aviatik sind wir mit dem Ausflussgesetz von Torricelli eingestiegen [4], haben danach die Dynamik kommunizierender Gefässe studiert [5], um zuletzt in einer etwas ausführlichen Übung den Druckausgleich zwischen zwei PET-Flaschen zu untersuchen [6]. Dieser Aufbau erlaubt eine systematische Einführung in die grundlegenden Strukturen wie Volumen- und überlagerte Energiebilanz sowie das Studium von kapazitiven, resistiven und induktiven Gesetzen, wobei speziell in der Hydrodynamik viele Abhängigkeiten nichtlinear sind. In zwei der drei Beispiele haben wir das gespeicherte Wasservolumen indirekt über eine Kraftmessung erfasst, indem wir die Töpfe und Flaschen am Haken eines entsprechenden Messgerätes aufgehängt haben.

2 Energieträger

Neben den bilanzierbaren Mengen, den zugehörigen Potentialen sowie den konstitutiven Gesetzen bildet die Energie die Buchhaltungsgrösse. Über die Energie werden die verschiedenen Gebiete miteinander vernetzt, d.h. die Energie ist eine Art Währung der Physik. Tabelle 1 gibt einen Überblick über die Energieträger und die zugehörigen Potentiale.

Für alle Mengen kann eine Bilanzgleichung formuliert werden

$$\sum_i I_{M_i} + \Sigma_M + \Pi_M = \dot{M} \quad (1)$$

I steht für Stromstärke, Σ für Quellenstärke, Π für Produktionsrate und M für eine der Mengen. Einige Mengen wie Masse oder Volumen werden nur zusammen mit der Materie transportiert, andere wie die Ladung, der Impuls oder die Entropie fliessen auch durch die Materie hindurch.

Der Energietransport ist in den meisten Fällen durch folgende Beziehung an die entsprechende Menge gebunden

$$I_W = \varphi_M I_M \quad (2)$$

W steht für Energie und φ für Potential. Formel (2) liefert beim Impuls- und beim Drehimpulstransport ein Skalarprodukt. Bei einigen konvektiven Transporten wie Impuls in bewegten Stoffen oder dem Elektronenstrahl kann Formel (2) nicht angewendet werden.

Menge	Potential	Gebiet
Volumen V	Druck	Hydraulik
Masse m	Gravitationspotential	Hydraulik
Ladung Q	elektrisches	Elektrodynamik
Impuls p	Geschwindigkeit	Mechanik
Drehimpuls L	Winkelgeschwindigkeit	Mechanik
Entropie S	Temperatur	Thermodynamik
Stoffmenge n	chemisches	Chemie

Tabelle 1: Die Energieträger und ihre Potentiale. Impuls und Drehimpuls werden durch ein raumfestes Koordinatensystem in je drei Mengen unterteilt. Diese sechs Mengen sind wie die elektrische Ladung vorzeichenfähig. Volumen, Entropie und Stoffmenge sind keine Erhaltungsgrössen.

Fällt ein Mengenstrom über eine zugehörige Potentialdifferenz, wird Energie aufgenommen oder abgegeben. Die Prozessleistung P folgt aus Formel (2)

$$P = (\varphi_{M_1} - \varphi_{M_2}) I_M \quad (3)$$

Gemäss Formel (3) ist die Leistung positive, wenn Energie freigesetzt wird, wenn der Mengenstrom I_M vom hohen zum tiefen Potential fällt. Eine freigesetzte Leistung wird meist nur teilweise von einem zweiten Prozess aufgenommen. Der Rest dient der Entropieproduktion. Die zugehörige Produktionsrate ist gleich dissipierte Leistung geteilt durch die dort herrschende absolute Temperatur. Das hier verwendete Energieträgerbild des Karlsruher Physikkurses (KPK) [7] hat grosse Ähnlichkeit mit der Bond-Graphen-Theorie [8] oder dem Basiskonzept von Modelica [9].

3 Kontinuumsmechanik und Relativitätstheorie

Der KPK welcher der Systemphysik zugrunde liegt, ist vor Jahren durch eine Expertengruppe der Deutschen

Physikalischen Gesellschaft massiv verunglimpft worden [10]. Dieses Ereignis, das wie ein Damoklesschwert auch über der Systemphysik schwebt, verlangt nach einer wissenschaftlichen Begründung der nachfolgenden Ausführungen. Die Mechanik hat seit Newton einen intensiven Ausbau erfahren und mit der Kontinuumsmechanik einen vorläufigen Abschluss gefunden. Der Hinweis, dass die hier dargelegte Translationsmechanik auf der Impulsbilanz im Sinne der Navier-Stokes-Gleichung beruht, sollte als Begründung genügen. Weil weder Claude Louis Marie Henri Navier noch George Gabriel Stokes im physikalischen Olymp nicht annähernd so hoch anzusiedeln sind wie Sir Isaac Newton, rufe ich Albert Einstein als Zeuge auf. Die wohl grösste Leistung Einsteins war die Verallgemeinerung der Newtonschen Gravitationstheorie. Wo bei Newton gemäss der Formulierung von Carl Friedrich Gauß die Massendichte steht, setzte Einstein den Energie-Impuls-Tensor. Diese Grösse beschreibt die Verteilung von Energie und Impuls in der Raum-Zeit, wobei die Gesamtenergie eines Körpers gleich seiner Masse mal die Lichtgeschwindigkeit im Quadrat ist. Schreibt man den Energie-Impuls-Tensor bezüglich eines ausgewählten Koordinatensystems, erhält man eine vier-mal-vier Matrix

$$T^{\alpha\beta} = \begin{bmatrix} T^{00} & T^{01} & T^{02} & T^{03} \\ T^{10} & T^{11} & T^{12} & T^{13} \\ T^{20} & T^{21} & T^{22} & T^{23} \\ T^{30} & T^{31} & T^{32} & T^{33} \end{bmatrix} \quad (4)$$

Nimmt man als nullte Koordinate die Lichtgeschwindigkeit mal die Zeit, steht in der ersten Zeile die Energiedichte sowie die Energiestromdichte geteilt durch die Lichtgeschwindigkeit. Die restlichen drei Zeilen beinhalten die Impulsdichte mal die Lichtgeschwindigkeit sowie die Impulsstromdichte.

Als einfachen Anwendungsfall setzen wir uns auf den Riemen eines Riemetriebes und formulieren den Energie-Impuls-Tensor für einen materiellen Punkt. Indem wir die x -Achse längs des Riemens ausrichten, können wir den Tensor in nur zwei Dimensionen formulieren

$$T^{\alpha\beta} = \begin{bmatrix} \rho c^2 & 0 \\ 0 & -\sigma \end{bmatrix} \quad (5)$$

Die Energiedichte ist gleich Massendichte mal Lichtgeschwindigkeit im Quadrat. Weil die Zugspannung σ als positive Grösse definiert wird, die zugehörige Impulskomponente bei dieser Belastung gegen die eigene Koordinatenrichtung fliesst, muss ein negatives Vorzeichen beigelegt werden.

Transformiert man den mit Formel (5) beschriebenen Energie-Impuls-Tensor ins Ruhesystem des Riemetriebes, erhält man den folgenden Ausdruck

$$T^{\alpha\beta} = \gamma^2 \begin{bmatrix} \rho c^2 - \beta^2 \sigma & \rho v c - \beta \sigma \\ \rho v c - \beta \sigma & \rho v^2 - \sigma \end{bmatrix} \quad (6)$$

$$\beta = \frac{v}{c} \quad \gamma = \frac{1}{\sqrt{1 - \beta^2}}$$

Weil das Verhältnis β von Riemengeschwindigkeit v zur Lichtgeschwindigkeit c etwa 10^{-7} beträgt, ist der Lorentzfaktor γ sehr nahe bei eins. Nimmt man nur die relevanten Terme, erhält man folgende Näherung

$$T^{\alpha\beta} \approx \begin{bmatrix} \rho c^2 & \rho v c - \frac{v \sigma}{c} \\ \rho v c & \rho v^2 - \sigma \end{bmatrix} \quad (7)$$

Die Massendichte ändert sich nicht messbar, dafür speichert der Riemen Impuls mit der Dichte ρv . Die zweite Kolonne liefert zwei bemerkenswerte Einsichten: neben der Massensromdichte ρv erscheint im oberen Element noch eine Energiestromdichte $-v \sigma$; im unteren Element taucht neben der leitungsartigen Impulsstromdichte $-\sigma$ noch eine konvektive Impulsstromdichte ρv^2 auf.

Impuls kann leitungsartig durch und konvektiv mit der Materie transportiert werden. Bei Zugspannung fliesst der Impuls rückwärts und bei Druckspannung vorwärts. Konvektiv wird der Impuls in beiden Riemen vorwärts transportiert. Würde man die Geschwindigkeit des Riemens erhöhen, bis das vierte Element in Formel (7) gleich null ist, stünde der Riemen unter Zugspannung, würde aber über den Riemenscheiben abheben. Formel (7) zeigt uns, dass ein mechanischer Energietransport nur zusammen mit einem leitungsartigen Impulsstrom erfolgen kann, wobei die Geschwindigkeit den Zusammenhang herstellt. Dieser Transport kann mit einem punktmekanischen Modell nicht erklärt werden. Die kinetische Energie, die im Riemen gespeichert ist, findet man in Formel (7) erst, wenn man den Lorentzfaktor γ nach v entwickelt.

4 Translationsmechanik

Stellvertretend für das ganze Kapitel sei hier ein Praktikumsversuch beschrieben. Zwei Wagen auf einer Rollbahn sind über Gummifaden untereinander sowie mit den beiden Enden der Bahn verbunden. In Vorversuchen wird die Reibung der Wagen und die statische Kennlinie der Gummifaden bestimmt. In einem ersten Experiment wird

ein Wagen, der mit zwei Gummifaden an der Bahn fixiert ist, in Schwingung versetzt. Gemessen werden der Ort mittels eines Distanzsensor (Ultraschall oder Laser), die Impulsstromstärken in beiden Gummifaden sowie die Beschleunigung des Wagens [11]. Parallel dazu wird ein systemdynamisches Modell erstellt. Die Parameter eines rheologischen Gummifaden-Modells werden so lange verändert, bis die Mess- und Simulationsdaten optimal übereinstimmen. Zur Validierung werden zwei Wagen mit einem Gummifaden untereinander sowie mit zwei weiteren mit der Bahn verbunden. Diese Anordnung wird wiederum in ein systemdynamisches Modell abgebildet. Diesmal dürfen die Parameter nicht mehr verändert werden. Ein Vergleich der Mess- mit den Simulationsdaten zeigt, wie präzise das Modell die Bewegung der beiden Wagen vorherzusagen vermag.

In einem Bericht müssen die Studierenden die Zusammenhänge im Impulsstrombild, im Flüssigkeitsbild und im Kraftbild erklären [12]. Eine ausführliche Beschreibung des systemdynamischen Modells und des experimentellen Aufbaus sowie eine gründliche Diskussion der Mess- und Simulationsergebnisse werden ebenfalls eingefordert. Bild 1 zeigt das Systemdiagramm (Flowchart) dieses Experiments, Bild 2 die überlagerte Energieebene.

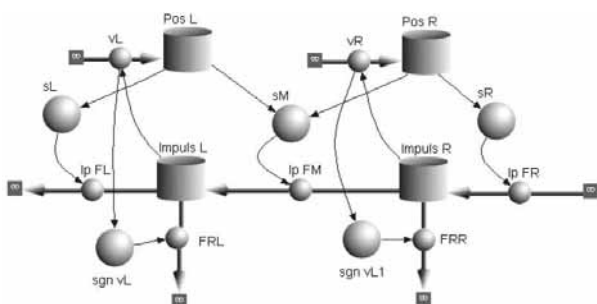


Bild 1: Systemdiagramm der zwei Wagen mit drei Gummifäden sowie den beiden reibungsbedingten Impulsströmen. Die oberen beiden Töpfe integrieren die Geschwindigkeit zum Ort.

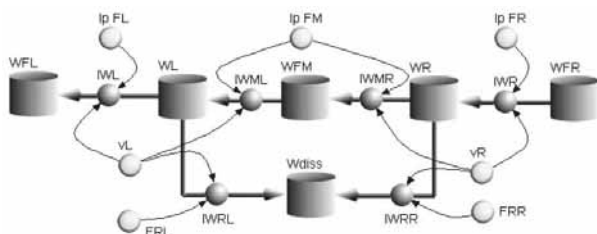


Bild 2: Die Energiebilanz rekapituliert die Impulsbilanz, wobei die Gummifäden und die Reibschicht zusätzlich Energie speichern können. Deshalb enthält die Energieebene mehr Töpfe als die Impulsebene.

Dieses Experiment lässt sich in vielen Varianten durchführen. Die Gummifäden können durch Stahlfedern oder durch starke Magnete ersetzt werden. Als Reibelement darf auch mal eine Wirbelstrombremse oder ein Segel eingesetzt werden. Anhand dieses Beispiels erkennt man deutlich den Unterschied zwischen der Punktmechanik und der Systemphysik, befasst sich erstere nur mit der Bewegung der Wagen, so verschiebt letztere den Fokus auf die Impuls- und Energieströme.

5 Thermodynamik

Wer die Wärme auf die Energie reduziert, hat gemäss dem ersten Hauptsatz der Thermodynamik recht, verletzt aber die dort getroffene Definition, sobald er die Wörter Wärmespeicher, Wärmedurchgang und Wärmeproduktion in den Mund nimmt. Wer dagegen Entropiespeicher, Entropietransport und Entropieproduktion sagt, trifft den Kern der Thermodynamik. Entropie steht für die thermische Basismenge. Entropie wird in der Wärmepumpe von tiefer zur hohen Temperatur gefördert. Entropie wird im Haus gespeichert und Entropie fliesst wieder an die Umgebung weg. Weil die Energie eine Erhaltungsgrösse ist, eignet sie sich für gewisse Fragestellungen wie Wärmedurchgang oder Wärmespeicher besser als die vermehrungswillige Entropie. Doch spätestens beim Optimieren von Sonnenkollektoren, Wärmepumpen oder auch Verbrennungsmotoren muss die Entropie wieder ins Spiel gebracht werden.

Die Thermodynamik beschäftigt sich oft mit der Schnittstelle zwischen Thermik und Mechanik. Um ein gewisses Grundverständnis dafür zu erwerben, modellieren die Studierenden in einem ersten Schritt den Carnotor [13]. Danach wenden sie sich einer speziellen Fragestellung zu, zum Beispiel dem Laborversuch von Rüchardt [14]. Bei der an der ZHAW verwendeten Version schwingt eine zweckentfremdete Milchpumpe über einem Erlenmeyer-Kolben. Das dynamische System besteht demnach aus einem vertikal beweglichen Körper und einer Gasfeder. Indem man den Erlenmeyer mit Stahlwolle, Aluminiumfolie oder Watte füllt, verändert sich der Wärmeleitwert. Das zugehörige systemdynamische Modell formuliert die Bilanz für Entropie und Volumen sowie für die beide Mengen verbindende Energie [15].

Bild 3 zeigt zwei Simulationsergebnisse im Temperatur-Entropie-Diagramm. Die Extremfälle wären die isentrope (vertikale Linie) und die isotherme Schwingung

(horizontale Linie). Schaltet man die mechanische Reibung im Modell aus, tritt in diesen beiden Grenzprozessen keine Dämpfung auf. Das Dämpfungsmaximum liegt irgendwo dazwischen und kann mittels einer Parameterstudie schnell gefunden werden. Zudem verändert sich die Schwingungsdauer zwischen den beiden Extremfällen um etwa 20 %. Beide Phänomene lassen sich auch experimentell nachweisen, indem die Füllmenge im Erlenmeyer variiert wird. Der Erfolg dieser Lernmethode ist augenfällig. So haben die Studierenden bei der Präsentation ihrer Arbeit in der Regel ein erstaunlich gutes Verständnis für die Thermodynamik gezeigt.

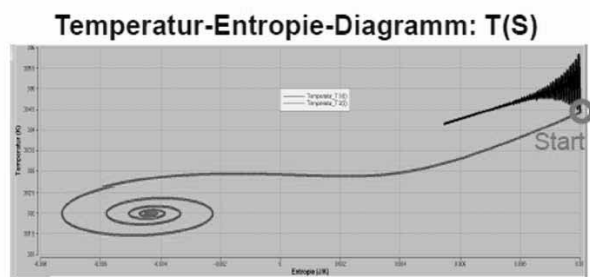


Bild 3: Das T - S -Diagramm für kleinen (schwarz) und grossen (roten) Leitwert.

6 Modelica

Studierende setzen heute die unterschiedlichsten Simulationstools ein, welche für spezielle Fragestellungen entwickelt worden sind. Dazu müssen sie weder die Theorie noch die Mathematik verstanden haben. Steht für ein gewisses Problem kein geeignetes Werkzeug zur Verfügung, wird oft Simulink benutzt. Simulink ist kein Modellierungstool im engeren Sinn, was sich in der Anwendung zeigt. So werden die Differentialgleichungen meist händisch aufgestellt und nur zur Lösung in Simulink eingegeben. Systemdynamische Tools wie STELLA, Berkeley Madonna und Vensim eignen sich besser als Simulink, um dynamische Systeme zu analysieren und abzubilden, wie auch Anwendungsbeispiele aus der Ökonomie und der Ökologie zeigen. Man kommt mit der graphischen Eingabe und der Numerik aber schnell an Grenzen, wenn die Systeme komplexer werden. Zudem ist die Wiederverwendbarkeit einzelner Teilmodelle nur beschränkt möglich. Modelica kann die Lücke zwischen der hier dargelegten, auf Verständnis und Einsicht beruhenden Basisausbildung und den Ansprüchen vieler Anwender schliessen.

Komplexe Systeme können auch in Modelica ohne tiefes Verständnis der Theorie abgebildet werden, indem

man die einzelnen Komponenten nach dem Lego-Prinzip der Bibliothek entnimmt, graphisch zusammengefügt und entsprechend den Vorgaben parametrisiert. Wenn das nicht funktioniert, liegt es meist am fehlenden Verständnis des Anwenders, manchmal aber auch an den in den Komponenten hinterlegten Gleichungen.

Zu Modelica gehört eine Standardbibliothek, die von der Hydraulik über die Mechanik und die Elektrik bis zur Thermodynamik reicht. Statt diese Bibliothek zu benutzen und zu ergänzen, plädiere ich für eine spezielle Bibliothek, welche die zentralen Ideen der Systemphysik rekapituliert [16]. Obwohl schon die Standardbibliothek methodisch der Systemphysik sehr nahesteht, können ein paar Dinge besser gemacht werden. Dazu gehört die Definition der Konnektoren, die konsequente Trennung von Speicher und Stromglieder sowie der konsequente Umgang mit der Energie. Dazu zwei Beispiele und eine allgemeine Anregung.

Die Unterbibliothek Translation enthält die Masse als Speicher sowie Feder und Dämpfer als Stromglieder. Daneben findet man aber auch eine Masse mit Reibung. Konsequenter wäre ein Modell für die Masse und verschiedene Modelle für die Impulsleiter, womit auch die unterschiedlichsten Reibmodelle gemeint sind. Zudem sollte die Masse auch eine Impulsquelle für die Gravitationskraft aufweisen, was ganz im Sinne von Albert Einstein wäre.

Die Konnektoren für die Thermodynamik weisen als «flow»-Grösse den Energiestrom aus. Hier sollte wie in allen anderen Domänen die Stromstärke der Basismenge, also der Entropie, eingesetzt werden. Was auf den ersten Blick nach l'art pour l'art aussieht, liefert auf den zweiten eine wesentliche Verbesserung. Solange man maximal dissipative Systeme wie Motorenkühlung oder Gebäudeisolation modelliert, reicht eine auf dem Energiebegriff aufgebaute Bibliothek völlig aus. Nicht aber, wenn man thermodynamische Systeme wie Wärmepumpen oder das weiter oben erwähnte Rückardt-Experiment modellieren will. Die Energieerhaltung für die Wärmeleitung lässt sich auch formulieren, wenn man die Entropie als Menge und die absolute Temperatur als Potential nimmt.

Die Konnektoren für die verschiedenen Gebiete sollten wenn möglich entsprechend Tabelle 1 ausgeführt werden. Zudem sollte man die Energie überall mitmodellieren, damit die Energiebetrachtung immer dann zur Verfügung steht, wenn sie gebraucht wird. Bei der Wärmeleitung und in allen anderen dissipativen Elementen sollte zusätzlich die Entropieproduktion gerechnet wer-

den. Aus didaktischer Sicht sind diese Anpassungen notwendig, um die Erkenntnisse aus der Grundausbildung zu repetieren und zu vertiefen. Welchen Gewinn man mit der konsequenten Umsetzung der systemphysikalischen Ideen in Bezug auf Variationsbreite und Stabilität erhält, kann nicht zum Voraus gesagt werden. Die Erfahrungen beim Aufbau der Anwenderbibliothek DyMoRail haben mir gezeigt, dass man sich viel Ärger ersparen kann, wenn man sich konsequent an die Struktur der Systemphysik hält [17].

Referenzen

- [1] Hestenes D, Wells M, Swackhamer D. Force concept inventory. *The Physics Teacher*. 1992; 30. 141 - 158. doi 10.1119/1.2343497.
- [2] Meschede, D. *Gerthsen Physik*. 24. Auflage. Berlin Heidelberg: Springer; 2010. Seite 78.
- [3] Maurer, W: Der Impuls im Flüssigkeitsbild. *Praxis der Naturwissenschaften – Physik für die Schule*. 1996. 4/45: 12-16.
- [4] Maurer, W. Ausflussgesetz von Torricelli. Youtube 24.09.2015: <https://youtu.be/-giLthDilGY>
- [5] Maurer, W. Dynamik kommunizierender Gefäße. Youtube 23.09.2014: <https://youtu.be/xNki45iOe3c>
- [6] Maurer, W. Druckausgleich modellieren. Youtube 03.05.2019: <https://youtu.be/NT0adWQMMoo>
- [7] Herrmann, F. Der Karlsruher Physikkurs. <http://www.physikdidaktik.uni-karlsruhe.de/>
- [8] Karnopp D, Margolis D, Rosenberg R. *System Dynamics – A Unified Approach*. 2ed. New York: Wiley; 1990.
- [9] The Modelica Association. <https://www.modelica.org/>
- [10] Lehn R. und andere. *Gutachten über den Karlsruher Physikkurs*. In Auftrag gegeben von der Deutschen Physikalischen Gesellschaft. 28. Februar 2013. https://www.dpg-physik.de/veroeffentlichungen/publikationen/stellungnahmen-der-dpg/bildung-wissenschaftlicher-nachwuchs/kpk/stellungnahme_kpk.pdf
- [11] Maurer, W. Modellbildung: zwei Wagen auf der Rollbahn. Youtube 05.12.2016: <https://youtu.be/5a31p3uWBXQ>
- [12] Maurer, W. Rangierstoss. Youtube 23.05.2019: <https://youtu.be/2LH3LHhMQg0>
- [13] Maurer, W. Der Carnotor. Youtube 19.06.2011: <https://youtu.be/1QAQJV2C6B0>
- [14] Wikipedia. Rüchardt-Experiment. <https://de.wikipedia.org/wiki/R%C3%BCrhardt-Experiment>
- [15] Maurer, W. Rüchardt-Experiment. Youtube 08.04.2014: https://youtu.be/8Tcw_yKcpsM
- [16] Maurer, W. PhyDynSys – eine Modelica-Bibliothek zur Physik der dynamischen Systeme. ASIM 19. Symposium Simulationstechnik. Hannover 12.-14. September 2006. https://www.academia.edu/32931337/PhyDynSys_eine_Modelica-Bibliothek_zur_Physik_der_dynamischen_Systeme
- [17] Maurer, W. Simulationsgestützte Entwicklung von Puffern und Dämpfern für Eisenbahnzüge. ASIM 18. Symposium Simulationstechnik. Erlangen 12.-15. September 2006.

Teaching Application Area Oriented Mathematics in Engineering

Andreas Körner¹, Stefanie Winkler¹

¹Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria;
*andreas.koerner@tuwien.ac.at

Abstract. The article focuses on the symbiosis between mathematics in engineering education and problem based approaches. Traditional approaches focus on the idea to teach first basic math and establish on this basics the theoretical subjects out of the application area, e.g. theoretical foundation, theoretical physics, mechanics, etc.

The obvious benefit is the formal foundation and the structured development of the theory. The disadvantage is a delay in developing the core subject after the mathematical and physical foundations are grounded.

In the subject of technical physics at TU Wien the approach is different. The first semester consists of three different mathematics subjects, analysis, linear algebra and a subject called practical mathematics. The practical math is supplying the fourth subject with the necessary tools for understanding the basic concepts of physics.

This paper is introducing this concept in detail and analyzing the possibilities to apply this approach in engineering mathematics. The last section of the paper addresses the relation to modelling and simulation as a subject in STEM studies and points out the parallel concepts and their application.

Introduction

The basic concept in including the topics and methods of mathematics in one engineering subject is introduced at the HAW Hamburg since 2014, see [1]. The core idea is to provide mathematical topics and apply them the next week in one of the subjects of engineering. This article goes one step further, that mathematics is split in theory and application and the applied math course builds the foundation for engineering subject.

A new approach would be to combine applied mathematics and a basic subject from the engineering field in the same semester, almost parallel. Math content, which has been taught 1 or 2 weeks ago, would be applied in the engineering subject immediately after. The benefit is that students see the application just in time.

For lecturers a benefit is that the examination during the semester can be less, because the corresponding subject is using the methods and students need to know them to pass this subject.

A particular example to illustrate the basic idea is the subject physics foundation. Almost all engineering studies have a basic physics education in the first semester. These physics courses usually are taught on a high school level, because the necessary higher mathematics for teaching e.g. mechanics, are missing.

The splitting into applied and theoretical mathematics would result in the effect that the methods, which are required in the physics course and the structural theoretical mathematics, shall be given in separate courses. Without studying the one subject it will not be possible to understand the other subject, see [2] as well. This interconnection will substitute the motivation aspects of the mathematics. Hence, students need to study methods and algorithms for the applied engineering courses.

This setup decreases the overall efforts for students, if the mathematics courses and the applied courses are coordinated regarding their requirements. Furthermore, the connection between mathematics and the applied courses can be supported by online examples in exercises in the math course and in the best case in the applied courses as well.

1 Physics Studies in contrast to Engineering Studies

First, it should be argued, why physics is more accessible for a different curriculum approach. This can be shown by analyzing the content of the basic physics course.

The covered physics topics are:

1. Mechanics of a particle
2. Collisions of particles
3. Rigid body dynamics
4. Kinematic reference systems
5. Oscillations and waves

These headlines cover several mathematical aspects, like the concept of forces and potentials in (1) and (2). Students have to deal with line integrals and vector fields after few weeks of the start of the first semester. For chapter (3) higher dimensional integration techniques are required and for (5) even ordinary and partial differential equations.

To compare to an engineering subject, e.g. electrical engineering, the basic subject of electrodynamics is given in the fourth semester of bachelor study. The basic math courses are accommodated in the first three semesters, so in the fourth semester allow to apply the whole math package.

Disadvantage in the engineering education is the lack of application. The three sequential math courses are preparing the whole bunch of methods on a stock. Nevertheless, the mathematical requirements are comparable to the ones in physics.

Covered topics in electrodynamic are:

1. Electromagnetic fields
2. Energy and impulse
3. Stationary and quasi stationary fields
4. Magnetic induction
5. Electromagnetic waves

The requirements from the mathematical point of view is comparable. For (3) and (4) vector calculus and vector analysis is needed. The difference is, that the physics subject is offered in the first semester, meanwhile the electrodynamic subject is offered in the fourth.

2 Setup Requirements and Curriculum

For applying this approach of collaborative teaching, some requirements are necessary:

- Flexibility in Curriculum
- Collaborative Teacher in the first semesters
- Itemized requirements for related subjects
- Perfect synchronized lectures and exercises
- Harmonized content

2.1 Traditional Curriculum Setup

Most Curriculums are going the traditional way and form, first the mathematical basics (and basic in physics) and continue, based on these basics, to construct the applied engineering theory.

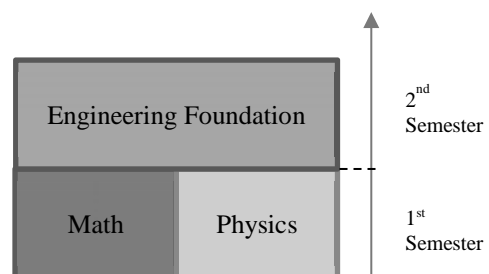


Figure 1: Each Box is illustrating a subject, e.g. Math and Physics, as a basis for a particular Engineering Subject, e.g. electrodynamics or analytic mechanics.

The strategy of *building up*, for an engineering subject, is illustrated in Figure 1. Mechanical and electrical engineering studies are typically structured in this traditional approach.

2.2 Integrated Curriculum Setup

Physics studies are different in this aspect. The basics in physics are concerning much more fields like mechanics, thermodynamics, optics, electrodynamics, relativity, etc. than the typical engineering studies. To wait for the math subjects to build the basics would last too long, so the math is split in two complementary subjects.

The theoretical math subjects, like analysis and linear algebra, on the one side and practical mathematics on the other.

The last subject equips the students with methods and practices to compute different mathematical problems and the theoretical subjects are going in the conceptual deepness of the content.

The resulting curriculum needs to have small granularity, to equip the physics course with the mathematical methods needed. Figure 2 is illustrating this structure of the curriculum.

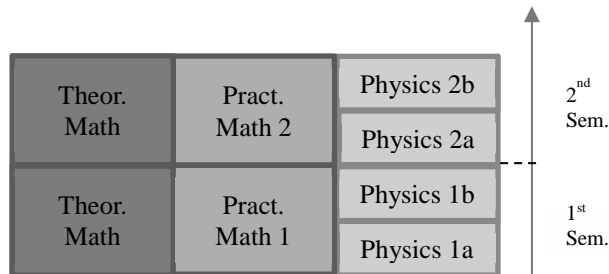


Figure 2: Each Box is illustrating a subject, e.g. Math and Physics, as a basis for a particular Engineering Subject, e.g. electrodynamics or analytic mechanics.

Of course, the curriculum of physics studies is different to the classical engineering studies, but the approach to split the calculus skills, from the theoretical math knowledge provides faster math concepts for engineering application.

Moreover, the mentioned benefits, the presented way of interconnecting mathematical skills and physics is benefiting the whole physics education, as presented in [4]. The same aspects are addressed in the highschool level in [5].

2.3 A demonstrative Example

A particular example in electrical engineering is the electric flux. To understand the relation between flux Ψ and flux density \bar{D} , which is the representing quantity, who is measured in practical examples, one needs vector calculus. In particular, the relation is given by a surface integral

$$\Psi = \int_{\mathcal{A}} \bar{D} \cdot dA. \quad (1)$$

The development of this mathematical concept is a long one and needs in the classical setup 3 semesters. If there is a practical mathematics, which is motivating the surface integral by geometrical facts and reduces them to the computation of particular examples, the development of such a method could work faster.

Especially in engineering subjects, mathematical concepts are often reduced by using symmetries,

simplifications and specialisations; there is never a situation to apply an abstract theory.

3 Learning Perspective versus Academic Buildup

Discussing these issues about the setup of the curriculum raises the argument of academic buildup. Of course, the curricula of higher education institutions should have a certain buildup, but the proposed setup would split this buildup just in two tracks: Skills and deductive math. The skills equip the students earlier in to be able to understand engineering applications and the deductive math path is illustrating the power of an abstract approach and illustrates the benefit of such access. Additionally, the previous learned skills can make the step to the abstract deductive approach much easier.

Apart from the academic discussion, several applied fields of STEM studies are applying a similar concept. One famous field is robotics. Robotics is starting in a simple setup, understandable in a geometrical way by almost all people of the society. More in detail, the robotics field is much more physics and math related, than it looks like in the beginning. In [3] is presented the idea of practical robot education. This approach is following some ideas presented in this paper, which is well accepted in the scientific setting.

4 Modelling and Simulation

A similar setup is given in the interdisciplinary subject of modelling and simulation. That subject is in the methodological layer influenced by mathematic knowledge and in the application layer by physics and engineering.

The related teaching approach is closely related to the situation faced in a modelling and simulation lecture. Students need to apply the math basics in a particular application field. Even in this setup computer simulations are used to help students get familiar with the application field and gather experiences in modelling.

4.1 Matheamtical Modelling

Addressing mathematical modelling is a similar scenario to the basic subject physics and engineering. Students need to study basic analysis and calculus, advanced linear algebra, numeric and computer numeric, differential equations, dynamic systems and partial differential equations before starting with modelling and simulation subjects. Normally this requires at least 5 semester of bachelor studies in mathematics. To attract students earlier to modelling and simulation, in the traditional way, is problematic. One approach to succeed in it would be the earlier introduction of calculus and physical modelling.

4.2 Numerical Simulation

Numerical Simulation requires more than solely mathematical skills. In several simulation environments are required programming skills.

A simple use case is the implementation of a particular mathematical model in MATLAB. Numerical mathematics meets programming on a higher level to implement a model, which is given by a mathematical environment, e.g. differential equation, differential-algebraic equations, automata, etc.

A problem-based teaching approach in this field can lead to developing the necessary mathematical concepts for the specific example. Developing the corresponding skills in solving math and implementing the related simulation model would generate a completed knowledge in this area.

5 Conclusion and Outlook

The paper presented a new approach on how to design the math education in engineering subjects. The motivation of this approach is the curriculum of technical physics at TU Wien.

The proposed structure of a linked course system would benefit from the progress in the linked subjects and is not relying on the topics presented in the own course. There are several advantages and disadvantages.

One advantage is obvious, the evolution in methods and techniques in the math courses, that can be applied in the technical course immediately. Students

could have more insight, why to study math and what will they benefit from a stable math education in the field of STEM subjects.

One major disadvantage is, that if there is any serious problem in the math course to understand particular methods or techniques, students may have also troubles in the applied subject. Being that math is considered as one of the difficult subjects, it should be investigated in a survey over one semester.

Other problems can cause the requirement of timed and synchronized course progress. If one subject is delayed, due to holidays or canceled lectures, the other subjects are suffering from this.

In the last section, the concept was extended to the subject of modelling and simulation. Several similar requirements can be applied in this field of interest and the ideas of teaching an integrated subject of a smaller field, can lead to an earlier access to the field of (mathematical) modelling and simulation.

Further work will concern some survey on the success of the students in technical physics at TU Wien. Moreover, in a subject of mathematical modelling in system simulation, the problem based approach will be applied to the exercises part of the course.

References

- [1] Ditzel B, Dahlkemper J, Landenfeld K, Renz W. *Integratives Grundstudium in den Ingenieurwissenschaften durch Themenwochen – vom Konzept zur Umsetzung*. ZfHE 9-4, 2014
- [2] A. Renkl, R. K. Atkinson, U. H. Maier, und R. Staley, „From Example Study to Problem Solving: Smooth Transitions Help Learning“, *The Journal of Experimental Education*, Bd. 70, Nr. 4, S. 293–315, 2002.
- [3] K. Nagai, "Learning while doing: practical robotics education," in *IEEE Robotics & Automation Magazine*, vol. 8, no. 2, pp. 39–43, June 2001, doi: 10.1109/100.932756.
- [4] Uhden, O., Karam, R., Pietrocola, M. *et al.* Modelling Mathematical Reasoning in Physics Education. *Sci & Educ* 21, 485–506 (2012). <https://doi.org/10.1007/s11191-011-9396-6>
- [5] Teodoro, V., Neves, R. Mathematical modelling in science and mathematics education in *Computer Physics Communications*, Volume 182, Issue 1, 2011, p. 8–10, <https://doi.org/10.1016/j.cpc.2010.05.021>.

Data Science und Lineare Algebra

Didaktisch-Methodische Überlegungen

Thomas Schramm

HafenCity Universität Hamburg, Geodäsie & Geoinformatik, Überseeallee 16, 20457 Hamburg, Deutschland,
thomas.schramm@hcu-hamburg.de

Abstract. Mit dem Projekt „Linear Algebra driven by Data Science“ soll eine Selbstlernereinheit in der Form von Open Educational Resources (OER) geschaffen werden, die mit dem Anwendungsfokus „maschinelles Lernen“ die Konzepte der Linearen Algebra vermittelt.

Einführung

Lineare Algebra (LA) wird im MINT-Studium im ersten Studienjahr vermittelt und stellt neben der Differenzialrechnung das Fundament der Ingenieur-Mathematik dar. Hier werden mathematische Objekte wie Vektoren und Matrizen mit ihren Operationen und Ordnungsbegriffe wie Vektorraum, Dimension, Norm, Metrik mit Konstruktionen wie Basis, Linearkombination, Abbildung, Kern etc. und deren Anwendungen eingeführt, die es ermöglichen komplexe Sachverhalte zu modellieren bzw. zu simulieren. Ein in zwei oder drei Dimensionen gewonnenes geometrisches Verständnis kann in der Abstraktion auf höhere Dimensionen und andere Objekte wie Polynome übertragen werden. Die vorherrschende Anwendung ist die Lösung linearer Gleichungssysteme und die analytische Geometrie. Sie bildet die Grundlage für numerische Anwendungen, wie maschinelles Lernen, insbesondere mit künstlichen neuronalen Netzen.

Das Projekt trägt der zunehmenden Relevanz des maschinellen Lernens insbesondere der künstlichen neuronalen Netze als nicht-symbolischen Ansatz Rechnung. Es soll versucht werden, es nicht als nachgeordnete, wenn auch wichtige Applikation zu sehen, sondern als führendes und motivierendes Beispiel, das die Einführung der aufgezählten Begriffe zwingend notwendig macht. Es soll mit einem qualitativen Verständnis für neuronale Netze beginnend, die Konzepte maschinellen Lernens und die benötigten Elemente der Linearen Algebra simultan entwickeln.

Dieses Projekt wird für die Hamburg Open Online

University (HOOU) [1] entwickelt und von dieser gefördert. Die Lerneinheit wird dann über das Portal der HOOU als OER für jedermann zur Verfügung stehen und ist damit ein zweites Modul, das neben der Differenzialrechnung dem Projekt oHMiNT (online Higher-Mathematics for MINT-Students) [2] zugeordnet und vom sog. OMB+-Konsortium [3] beraten und evaluiert wird.

Dieses vorerst einjährige Projekt ist eine Kooperation des Autors, seiner Mitarbeiter Sören Schwenker und Kay Zobel, mit Ingenuin Gasser, Alexander Lohse der Universität Hamburg und Ruedi Seiler mit Mitarbeitern der Firma Integral Learning GmbH [4] aus Berlin als Kooperationspartnern.

1 Konzept

Die Notwendigkeit, sich mit Linearer Algebra auseinanderzusetzen, wenn maschinelles Lernen professionell betrieben werden soll, wird an vielen Stellen hervorgehoben (vergl. Z.B. [5]).

Wie betont, wollen wir aber über ein technisches Verständnis hinaus, eine kritische Grundhaltung vermitteln, die es ermöglicht, die Reichweiten der neuen Technologien des maschinellen Lernens (ML) jenseits der Hochglanzbroschüren einzuschätzen und mögliche Risiken einzuordnen. Fragen, die in diesem Zusammenhang erläutert werden sollen sind z.B.

- Wer ist für Entscheidungen verantwortlich, die auf der Basis von ML getroffen werden?
- Lassen sich ML-Entscheidungen immer verstehen?
- Wo und wie ist das gelernte Wissen repräsentiert?

Hierbei müssen die verwendeten allgemeinen Begriffe wie Data Mining, künstliche Intelligenz, maschinelles Lernen, überwachtes vs. unüberwachtes Lernen, klassifizieren (Support Vector Machines) etc. erst einmal geordnet und zumindest qualitativ eingeführt werden.

Im Weiteren beschäftigen wir uns dann mit der wesentli-

chen Methode des nicht-symbolischen maschinellen Lernens durch künstliche neuronale Netze. Mit einem qualitativen, modellhaften Verständnis für die Funktionsweise neuronaler Netze beginnend, werden deren Begrifflichkeiten, deren algorithmische Umsetzung und die genutzten mathematischen Methoden parallel entwickelt.

2 Technische Implikationen

Die Selbstlerneinheit wird auf Servern Integral Learnings auf der Basis eines vorhandenen online Linear Algebra-Kurses on-the-fly auf der speziell für Mathematik entworfenen Mumie-Plattform [6] entwickelt. In diesem Kurs stehen viele der benötigten Einheiten in der Form von Vorlesungstexten, interaktiven Beispielen und Übungen bereits zur Verfügung. Später wird diese Einheit wie der Differenzialrechnungskurs über die HOOU-Plattform verlinkt.

3 Exemplarische Umsetzung

Die neuentwickelten Kurslemente wurden exemplarisch in einem Lineare-Algebra-Kurs der HafenCity Universität Hamburg im Studiengang Geodäsie und Geoinformatik im zweiten Semester eingesetzt. Es war zwar nicht geplant, aber das aktuelle nur-digitale Covid-Semester zwang alle Studierenden in das Home-Office, was erfreulicher Weise zu intensiver Bearbeitung der Studieninhalte führte. Über die allgemeine Moodle-Plattform wurden die jeweiligen zu bearbeiteten Inhalte aufgezeigt, kommentiert und verlinkt. Diese wurden wöchentlich in einer Zoom-Sitzung im *flipped classroom*-Stil besprochen und Ergänzungen eingewoben. Dazu gab es ein verpflichtendes, wöchentliches formatives eAssessment auf der Plattform Möbius [7], die ebenfalls speziell für mathematisches eAssessment zur Verfügung steht. Auf dieser Plattform wird auch die finale, summative Klausur durchgeführt.

Die Aufteilung des Stoffes erfolgte nach den klassischen Mustern eines Lehrbuchs, oder einer „normalen“ Vorlesung. Hierbei wurde darauf geachtet, die essentiellen mathematischen Inhalte beizubehalten, auch wenn diese nicht zwingend für das Verständnis der künstlichen neuronalen Netze notwendig waren. Anders herum, wurden einige Elemente der Analysis (z.B. Gradientenabstieg) kurz und plausibel eingeführt, auch wenn diese nicht zwingend in eine Einführung der Linearen Algebra gehören.

Die erste Idee, ein manuell rechenbares Beispiel heranzuziehen, wurde fallen gelassen zugunsten eines Standardbeispiels zur Erkennung handschriftlicher Ziffern. Im Wesentlichen hielten wir uns hier an die Darstellung einer Folge von zitierten YouTube-Videos von 3BLUE1BROWN [8].

Dieses Beispiel wurde mit wachsender Komplexität parallel entwickelt und die Algorithmen als Python-Code vorgestellt. In einer späteren Version, sollen die Code-Schnipsel direkt auf der Mumie-Plattform ausgeführt werden. Aktuell kann der komplette Code als Jupyter-Notebook heruntergeladen, modifiziert und ausgeführt werden. Dies erfordert aktuell eine gewisse Grundkenntnis der Programmiersprache Python. Dazu wird zu Beginn auf eine kleine Lerneinheit hingewiesen. Später sollen kleine Übungen eingeführt werden, die mathematischen Konstrukte direkt auf der Mumie-Plattform in auszuführenden Code umzusetzen. Die notwendige Funktionalität dazu ist bereits vorhanden.

Um dem Anspruch gerecht zu werden, die gesellschaftlichen kritischen Aspekte der KI-Technologien zu vermitteln, wurden diverse wissenschaftliche und populäre Artikel, sowie entsprechende Videos als Zusatzmaterial angeboten und in den *flipped sessions* kritisch besprochen. Als Beispiel seien einige Youtube-Videos von Gert Scobel [9] oder zwei Kapitel aus dem aktuellen Werk Richard David Prechts [10] genannt. Die dort gestellten und oben schon genannten *großen Fragen* stießen bei den Studierenden auf großes Interesse. Besonders genannt sei die Frage nach der Verantwortlichkeit beim autonomen Fahren, der Unmöglichkeit autonomer Killerdrohnen oder die Frage, ob wir uns zu Knechten einer KI machen könnten.

4 Fazit

Wir glauben an eine sinnvolle Verbindung von Mathematik und KI im technischen und naturwissenschaftlichen Anfängerstudium und vielleicht nicht nur dort. Neben den algorithmischen Kompetenzen, die dabei erworben werden, wird die ethische Dimension und auch die damit verbundene Verantwortung der Bildungseinrichtungen deutlich. Unser erster Versuch wurde von den meisten Studierenden freundlich evaluiert, wenn auch hier und dort eine gewisse Überforderung sichtbar wurde, sich mit den verschiedenen Ebenen des Kontextes und gleichzeitig mit den auch für die Lehrenden völlig neuen Anforderungen eines digitalen Studiums auseinanderzusetzen.

5 Referenzen

- [1] Hamburg Open Online University, <https://www.hoou.de>
- [2] oHMint, www.hoou.de/materials/ohmint
- [3] OMB+, www.ombplus.de/
- [4] Integral Learning GmbH, www.mumie.net/about-il/
- [5] Donges N., Basic Linear Algebra for Deep Learning, towards data science, 2018, towardsdatascience.com/linear-algebra-for-deep-learning-f21d7e7d7f23
- [6] Mumie-Plattform, www.mumie.net/platform/

- [7] Möbius, DigitalEd, www.digitaled.com/products/assessment
- [8] Aber was *ist* nun ein neuronales Netzwerk? | Teil 1, Deep Learning, <https://youtu.be/aircAruvnKk>
- [9] Scobel, Gert, Künstliche Intelligenz philosophisch betrachtet, <https://youtu.be/oYc4p3o6IFA>
- [10] Precht, Richard David, Künstliche Intelligenz und der Sinn des Lebens, Goldmann Verlag, München 2020

Heuristische Untersuchung der Abhängigkeit von Übungen und Vorlesung für den Prüfungserfolg

Corinna Modiz^{*1}, Franziska Gorgas¹, Stefanie Winkler¹, Andreas Körner¹

¹Inst. of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria; ^{*}corinna.modiz@tuwien.ac.at

Abstract. In dieser Studie werden unterschiedliche Jahrgänge und deren Leistungen abhängig von dem Lehrveranstaltungs-konzept untersucht. Dabei wurde eine mathematische Grund-lehrveranstaltung bezogen auf den Prüfungserfolg in Abhängigkeit der Leistung in den zugehörigen freiwilligen Übungen evaluiert. Dabei wird gezeigt, dass der Jahrgang mit statischen Aufgaben schlechter abschneidet, als die Jahrgänge mit variierten online Beispielen.

1 Einführung

Das Erlernen von Fähigkeiten und Fertigkeiten auf einer Universität stützt sich im Grunde auf drei Schwerpunkte. Erstens, das Besuchen der Vorlesung, wobei aktives Zuhören und Mitarbeiten von Vorteil ist. Zweitens, regelmäßiges Anwenden der theoretischen Inhalte in Form von Übungsaufgaben, in denen jene Inhalte auf verschiedene Fragestellungen, beschränkt auf kleinere Teilgebiete angewandt werden [1]. Drittens, das Selbststudium, welches ein Auseinandersetzen mit den Inhalten der Vorlesung und Übung inkludiert, mit dem Ziel den Inhalt gut zu verstehen oder zumindest ein Formulieren jeglicher Unklarheiten ermöglicht [2]. Diese drei Schwerpunkte sind maßgeblich für die Gestaltung der Lehrveranstaltung und sollten daher für den Vortragenden ausschlaggebend sein. Sie sollten bei der Planung und Durchführung des Kurses beachtet werden und Studierenden die Möglichkeit zum Mitlernen und Üben garantieren.

Im folgenden Kapitel wird, basierend auf diesem Grundprinzip, das Konzept der ausgewählten Lehrveranstaltung beschrieben. Dabei werden die unterschiedlichen Umsetzungen der begleitenden freiwilligen Übung erläutert. Anschließend werden die Lehrveranstaltungsergebnisse der Jahrgänge 2016-2019 in Abhängigkeit der drei unterschiedlichen Umsetzungen des Lehrkonzepts unterschieden. Die gesammelten Daten ermöglichen einen direkten Vergleich zwischen dem Einsatz von statischen und variierenden online Beispielen. Im Conclusio werden die Schlussfolgerungen der Datenevaluierung zusammengefasst wiedergegeben.

2 Lehrkonzept & Tools

An der TU Wien wird im Bereich der Grundausbildung in Mathematik für Ingenieurwissenschaften seit einigen Jahren ein Lehrkonzept angewendet, welches Studierenden das Erlernen von mathematischen Fähigkeiten, basierend auf den oben genannten Grundlagen, ermöglicht. Neben einer Vorlesung, die frontal vorgetragen wird, umfassen Lehrveranstaltungen, welche sich der mathematischen Grundausbildung von Ingenieuren widmen, zusätzlich auch eine wöchentlich abgehaltene Präsenzübung. In den meisten Lehrveranstaltungen dieser Art, wird für die Übungsaufgaben das Online-System Möbius (ehemalig Maple T.A.) zur Verfügung gestellt. Diese Plattform basiert auf dem Computer-Algebra-System Maple und ermöglicht, sowohl Variablen sowie auch Funktionen in einem Beispiel einfach zu randomisieren, als auch eine automatisierte Bewertung der Aufgaben. Über die letzten Jahre wurden etliche Beispiele zu verschiedenen praktischen sowie theoretischen Fragestellungen entwickelt. Mit der Randomisierung der Beispiele haben Studierende die Möglichkeit, zu unterschiedlichen Lehrinhalten eine Vielzahl an Beispielen zu üben [3].

Verwendung findet Möbius in den Lehrveranstaltungen sowohl in Form von Übungssammlungen zum freien Üben, als auch zur Leistungsüberprüfung bei Klausuren und Vorlesungsprüfungen. Die Bepunktung der Beispiele wird voreingestellt, wodurch die Benotung für alle Teilnehmer gleich und fair durchgeführt werden kann. Bei den Lehrveranstaltungen handelt es sich um Mathematikvorlesungen für Studierende der Elektrotechnik, Geodäsie und Geoinformation, Raumplanung und Technischen Physik. Für diesen Beitrag, wurde die Vorlesung „Mathematische und statistische Grundlagen der Raumplanung“ ausgewählt, welche im nächsten Abschnitt näher beschrieben wird.

3 Lehrveranstaltungsmodus

Die Vorlesung „Mathematische und statistische Grundlagen der Raumplanung“ wird im zweiten Semester des Bachelorstudiums für Raumplanung angeboten. Ursprünglich wurde diese Vorlesung ohne eine zugehörige Übung angeboten. Nachdem die Arbeitsgruppe „Mathematics in Simulation and Education“ die Vorlesung mit Kollegen aus der Raumplanung übernommen hatte, wurde eine begleitende Übung eingeführt. Diese ist allerdings ein freiwilliges Zusatzangebot. An den anderen Ingenieursdisziplinen werden verpflichtende Übungen zu den Mathematiklehrveranstaltungen angeboten.

Die Vorlesung wurde in den angegebenen Jahren als zweistündige Präsenzveranstaltung mit Tafel bzw. Folien abgehalten. Die zugehörige Übung bestand in jedem der Jahrgänge aus den folgenden Komponenten:

- wöchentliche Übungen,
- Tafelleistungen,
- Klausuren und
- Mitarbeit.

In den nachstehenden Abschnitten werden die Unterschiede der einzelnen Jahrgänge herausgearbeitet.

3.1 Konzeptbeschreibung 2016

Die begleitende Übung wurde wöchentlich als Präsenzlehrveranstaltung durchgeführt. Für die Übungsvorbereitung wurden übungsrelevante Beispiele der Möbius Plattform in Moodle als Link zur Verfügung gestellt. In den Präsenzeinheiten wurden diese Beispiele von dem/der Übungsleitung vorgerechnet. Studierende hatten die Möglichkeit, vorbereitete Beispiele selbst vorzuführen bzw. mitzuarbeiten, um Bonuspunkte für die Abschlussnote zu sammeln. Im Laufe des Semesters wurden drei Klausuren angeboten, von denen mindestens zwei positiv abgelegt werden mussten. Zusätzlich gab es auch eine Mindestpunktezahl in jeder Klausur. Ein weiteres Angebot waren die Hausübungen, welche eine Woche vor den Klausuren online gestellt wurden und als Vorbereitung gedacht waren. Die Ergebnisse dieser Hausübung gingen ebenfalls als Bonuspunkte in die Bewertung ein.

3.2 Konzeptbeschreibung 2017

Die Veränderung zum Übungsmodus aus 2016 bestand darin, dass die Beispiele für die wöchentliche Übung als ausformuliertes pdf-Dokument anstatt als online Beispiele zur Verfügung gestellt wurden. Dies hatte natürlich zur Folge, dass Studierende in der Übungsvorbereitung lediglich mit den statischen Beispielen und dem

Skriptum beschäftigt waren. In der Klausur als auch der Prüfung wurden aber variierende online Beispiele gegeben. Diese Änderung wird in der Evaluation im nächsten Kapitel näher diskutiert.

3.3 Konzeptbeschreibung 2018

Im Jahr 2018 wurden diese Übungsbeispiele dann wieder als online Assignment abgebildet. Zusätzlich wurden weitere, zum Vorlesungsstoff passende Beispiele thematisch zur Verfügung gestellt.

In allen drei Jahrgängen, wurde die Vorlesungsprüfung aus dem Pool der Übungsbeispiele mit teilweise geringfügigen Adaptionen zusammengestellt. Im folgenden Kapitel wird der Erfolg dieser Prüfung unter diversen Nebenbedingungen genauer beleuchtet.

4 Evaluation

In den ersten zwei Diagrammen werden Aktivitäten und Ergebnisse der Übungen aus dem Sommersemester 2018 und 2019 dargestellt. Die darauffolgenden Histogramme stellen die Übungs- und Prüfungserfolge der Jahrgänge 2016 bzw. 2017 dar. Die Tabellen am Ende des Kapitels führen den Zusammenhang der einzelnen Übungselemente mit der positiven bzw. negativen Vorlesungsprüfung näher aus.

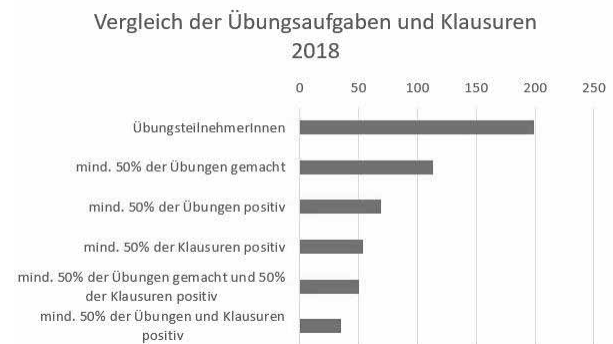


Abbildung 1: Vergleich der Übungsaufgaben und der Klausuren der Übung 2018S



Abbildung 2: Vergleich der Übungsaufgaben und der Klausuren der Übung 2019S

Bei der Gegenüberstellung der Übungsaufgaben und der Klausuren fällt auf, dass in beiden Jahrgängen 2017 und 2018 die Zahl jener Studierenden, die mindestens 2 Klausuren erfolgreich abschließen konnten, nicht sehr von der Zahl jener, die zusätzlich die Hälfte der Übungsaufgaben gemacht haben, abweicht. Im Abbildung 2 stimmt letztere sogar mit der Anzahl der Studierenden, die mindestens die Hälfte der Übungsaufgaben positiv absolviert haben, beinahe überein. Dieser Vergleich lässt den Schluss zu, dass eine positive Korrelation zwischen Übungsaufgaben und erfolgreich abgelegten Klausuren besteht.



Abbildung 3: Vergleich des Übungs- und Prüfungserfolgs der ÜbungsteilnehmerInnen 2016

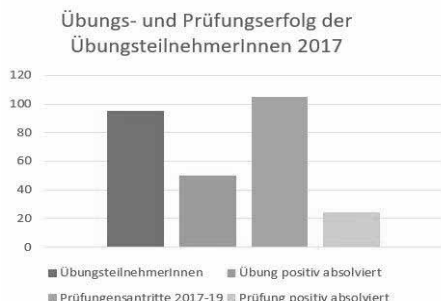


Abbildung 4: Vergleich des Übungs- und Prüfungserfolgs der ÜbungsteilnehmerInnen 2017

Der Vergleich der Jahre 2016 und 2017 zeigt, dass eine höhere Abschlussquote der Übung nicht auf eine höhere Erfolgsrate der Prüfungen schließen lässt. So haben in der Übung des Sommersemesters 2016, siehe Abbildung 3, nur 36 von 104 TeilnehmerInnen die Übung erfolgreich beendet, jedoch ist der Prüfungserfolg bei 53 positiven Ergebnissen von 80 deutlich höher. Gegenteiliges ist beim Vergleich des Übungs- und Prüfungserfolgs der Studierenden im Sommersemester 2017 zu erkennen, siehe Abbildung 4. Der Anteil der positiven Prüfungsergebnisse liegt bei fast 24%, während der Anteil jener ÜbungsteilnehmerInnen, die die Übung bestanden haben, bei fast 53% liegt.

Übung 2016S	Prüfung positiv	Prüfung negativ	gesamt
Prüfungsantritte gesamt			
2016-19 inklusive Mehrfachantritte	38.76%	61.24%	485
Prüfungsantritte von TeilnehmerInnen der Übung inklusive Mehrfachantritte	61.25%	38.75%	80
TeilnehmerInnen der Übung *	65.38%	34.62%	104
TeilnehmerInnen der Übung	50.96%	49.04%	104
70% der Übungsaufgaben angesehen	63.41%	36.59%	41
2 Klausuren bestanden	69.45%	30.55%	36
3 Klausuren bestanden	100%	0%	4

*Die Ergebnisse beziehen sich auf mindestens einmal angetreten (erste Spalte) bzw. nie angetreten (zweite Spalte).

Tabelle 1: Vergleich des Prüfungserfolgs der TeilnehmerInnen der Übung 2016S mit Übungskomponenten.

Übung 2017S	Prüfung positiv	Prüfung negativ	gesamt
Prüfungsantritte gesamt			
2017-19 inklusive Mehrfachantritte	32.42%	67.58%	309
Prüfungsantritte von TeilnehmerInnen der Übung inklusive Mehrfachantritte	22.86%	77.14%	105
TeilnehmerInnen der Übung *	77.89%	22.11%	95
TeilnehmerInnen der Übung	25.26%	74.74%	95
2 Klausuren bestanden	21.05%	78.95%	19
3 Klausuren bestanden	29.03%	70.97%	31

*Die Ergebnisse beziehen sich auf mindestens einmal angetreten (erste Spalte) bzw. nie angetreten (zweite Spalte).

Tabelle 2: Vergleich des Prüfungserfolgs der TeilnehmerInnen der Übung 2017S mit Übungskomponenten.

Übung 2018S	Prüfung positiv	Prüfung negativ	gesamt
Prüfungsantritte gesamt 2016-19 inklusive Mehrfa- chantritte	39.92%	60.08%	238
Prüfungsantritte von Teilneh- merInnen der Übung inklusive Mehrfachantritte	42.41%	57.59%	158
TeilnehmerInnen der Übung *	62.81%	37.19%	199
TeilnehmerInnen der Übung	33.67%	66.33%	199
70% der Übungsaufgaben angesehen	48.45%	51.55%	97
2 Klausuren bestanden	31.43%	68.57%	35
3 Klausuren bestanden	63.16%	36.84%	19

* Die Ergebnisse beziehen sich auf mindestens einmal angetreten (erste Spalte) bzw. nie angetreten (zweite Spalte).

Tabelle 3: Vergleich des Prüfungserfolgs der TeilnehmerInnen der Übung 2018S mit Übungskomponenten.

In den Tabellen 1, 2 und 3 wurden die Auswirkungen der einzelnen Übungselemente auf das Prüfungsergebnis untersucht. Während von insgesamt 485 Antritten in den Jahren 2016 bis 2019 fast 39% positiv und 61.24% negativ waren, fielen 61.25% der 80 Antritte von ÜbungsteilnehmerInnen im selben Zeitraum positiv aus, siehe Tabelle 1. Die Ergebnisse der gesamten Kohorte zeigen ein gegenteiliges Verhalten im Vergleich zu den Übungsteilnehmern. Ebenfalls kann aus Tabelle 1 abgelesen werden, dass viele jener Studierenden, die gute Ergebnisse bei den Übungsaufgaben und den Klausuren erreichen konnten, auch die Prüfung positiv abschließen konnten. Je mehr Übungselemente von den Studierenden behandelt wurden, desto höher ist der Prozentsatz der erfolgreichen Prüfungen. So haben zum Beispiel in der Übung im Jahr 2016 von 41 Studierenden, die 70% der Übungsaufgaben gemacht haben, 63.41% die Prüfung bestanden. Von jenen, die auf 2 Klausuren mindestens die Hälfte der Punkte erreicht haben, waren es 69.45% und von jenen, die 3 Klausuren bestanden haben, 100%.

Die Übung des Sommersemesters 2017 zeigt ähnliche Ergebnisse, siehe Tabelle 2. Der Prozentsatz der Studierenden, die an der Übung teilgenommen und die Prüfung bestanden haben, ist zwar geringer als im Vorjahr, jedoch steigt der Anteil der Studierenden mit positivem Prüfungsergebnis mit der Anzahl der bestandenen Klausuren.

Dieses Verhalten ist auch in Tabelle 3 zu erkennen. Der Prozentsatz der Studierenden, die in mehr als drei Klausuren und in der Prüfung 50% der Punkte erreichen konnten, ist mehr als doppelt so hoch wie jener, der sich auf 2 Klausuren bezieht.

Im Jahr 2016 konnte die Hälfte der ÜbungsteilnehmerInnen die Prüfung positiv abschließen, während es 2018 zirka ein Drittel war. Zu erwähnen ist allerdings, dass bei vielen ÜbungsteilnehmerInnen erst in den Folgejahren die Vorlesungsprüfung abgelegt wird. Da für diesen Beitrag nur Daten aus den Jahren 2016 bis 2019 miteinbezogen wurden, liegen für den Jahrgang 2018 noch vergleichsweise wenige Prüfungsdaten vor.

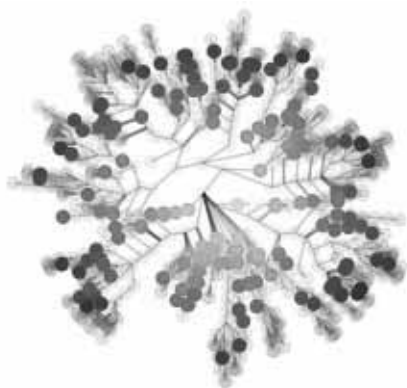
5 Conclusio

Aus den erhobenen Daten lässt sich eine positive Korrelation zwischen einem erfolgreichen Abschluss der Übung und dem Bestehen der Vorlesungsprüfung feststellen. Studierende, die die Lehrveranstaltung bei einem Vortragenden mit der zur Vorlesung angepassten Übung besucht haben, schneiden im Vergleich besser ab, als jene, die lediglich das Angebot der Vorlesung wahrgenommen haben. Die nicht so eindeutigen Ergebnisse des Jahrgangs 2017 könnten darauf zurückgeführt werden, dass dies der einzige Jahrgang ist, welcher statische Übungsaufgaben zur Verfügung gestellt bekommen hat. Dies könnte in der Folge dazu geführt haben, dass Studierende den statischen Beispielen zu viel Relevanz in der Prüfungsvorbereitung zugesprochen und dadurch die variierenden online Beispiele vernachlässigt haben.

Im Großen und Ganzen bestätigen die Ergebnisse allerdings die These, dass Lehrveranstaltungen als abgestimmtes System aus Vorlesung und Übung geplant werden sollten. Die Abstimmung der unterschiedlichen Elemente ist hierbei essentiell. Die Daten haben bestätigt, dass, im Sinne des „Constructive Allignments“, sowohl Übungen, als auch Klausuren auf die Prüfung einerseits inhaltlich andererseits auch methodisch vorbereiten sollen.

References

- [1] O. Uner und H. L. Roediger, „The Effect of Question Placement on Learning from Textbook Chapters“, *Journal of Applied Research in Memory and Cognition*, Bd. 7, Nr. 1, S. 116–122, März 2018, doi: 10.1016/j.jarmac.2017.09.002.
- [2] H. L. Roediger und A. C. Butler, „The critical role of retrieval practice in long-term retention“, *Trends in Cognitive Sciences*, Bd. 15, Nr. 1, S. 20–27, Jan. 2011, doi: 10.1016/j.tics.2010.09.003.
- [3] A. Körner, S. Winkler, R. Leskovaar, F. Gorgas: "Online-Komponenten der Lehre an der TU Wien", in: *Tagungsband ASIM 2018, 24. Symposium Simulationstechnik*, ISBN: 978-3-901608-12-4; 65 S.



ISBN eBook 978-3-901608-93-3

DOI 10.11128/arep.59

© by ARGESIM / ASIM, Wien, 2020