



8. und 9. März 2018 an der Hochschule Heilbronn

Tagungsband ASIM Workshop STS/GMMS 2018

Tobias Loose (Hrsg.)

ASIM Workshop STS/GMMS 2018

Tagungsband

Hochschule Heilbronn, 8. und 9. März 2018

Herausgegeben von
Tobias Loose
Hochschule Heilbronn

ARGESIM Report AR 54
ISBN 978-3-901608-19-3 (print) – ISBN 978-3-901608-20-9 (ebook)
ARGESIM Publisher – Vienna – www.argesim.org
DOI: 10.11128/arep.54

ASIM Mitteilung AM 165

Bibliographic Data:

Verlag: ARGESIM Publisher, Vienna

Titel: Tagungsband ASIM Workshop STS/GMMS 2018

Untertitel: Workshop ASIM Fachgruppen Simulation Technischer Systeme (STS) und Grundlagen und Methoden in Modellbildung und Simulation (GMMS), 8.3.-9.3.2018, Hochschule Heilbronn

Herausgeber: Tobias Loose, Hochschule Heilbronn

Series: ARGESIM Reports

Series Editors: Felix Breitenecker, Thorsten Pawletta, ASIM

Volume: ARGESIM Report no. 54

ISBN: 978-3-901608-19-3 (print), 978-3-901608-20-9 (ebook)

DOI: 10.11128/arep.54

ASIM ID: ASIM Mitteilung AM 165

Publikationsdatum: print 7.3.2018, ebook 5.4.2018

Seitenanzahl: 265 + x pages

Titelblatt: Mit freundlicher Genehmigung entstand die Titelseite durch das Foto von Hrn. Roland Schweizer sowie die Diagramme, Bilder und Skizzen aus den Beiträgen von Tobias Loose et al., Ralf Mikut, Katrin Reblinsky, Christian Richter et al., Volker Waurich et al., Karlheinz Wolfmüller.

Copyright © 2018 ARGESIM Publisher, Wien (print & ebook)

Copyright © 2018 Hochschule Heilbronn (print)

Hochschule Heilbronn, Fakultät für Technische Prozesse, Campus Sontheim

Max-Planck-Str. 39, 74081 Heilbronn, Deustchalnd

Das Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, der Entnahme von Abbildungen, der Funksendung, der Wiedergabe auf photomechanischem oder ähnlichem Weg und der Speicherung in Datenverarbeitungsanlagen bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Der vorliegende Tagungsband enthält Beiträge, für welche die Autoren selbst verantwortlich sind.

Copyright Information / Regulations ARGESIM

ARGESIM is a non-profit scientific society generally aiming for dissemination of information on system simulation - from research via development to applications of system simulation. ARGESIM's primary publication is the journal SNE – Simulation Notes Europe with open access to all contributions; generally, the authors retain the copyright of their SNE contributions. This copyright regulation holds also for *ARGESIM Reports* and *ARGESIM Advances in Simulation* publishing conference volumes for ASIM, MTHMOD, and EUROSIM (in consideration of copyright regulations for related conference publications) and monographs on system simulation (mainly PhD theses).

About ARGESIM

ARGESIM is a non-profit society generally aiming for dissemination of information on system simulation from research via development to applications of system simulation. ARGESIM is closely co-operating with EUROSIM, the Federation of European Simulation Societies, and with ASIM, the German Simulation Society. ARGESIM is an 'outsourced' activity from the Mathematical Modelling and Simulation Group of TU Wien, there is also close co-operation with TU Wien (organisationally and personally). *ARGESIM Publisher* organizes publishing activities, with ISBN root 978-3-901608-xx-y and DOI root 10.11128/xx...x.

ARGESIM's activities are:

- Publication of the scientific journal SNE - Simulation Notes Europe (Membership Journal of EUROSIM, the Federation of European Simulation Societies) → www.sne-journal.org
- Organisation and Publication of the ARGESIM Benchmarks for Modelling Approaches and Simulation Implementations → www.argesim.org/benchmarks/
- Publication of the series ARGESIM Reports (for monographs in system simulation, and proceedings of simulation conferences and workshops) → www.argesim.org/publications/
- Publication of the special series FBS Simulation - Advances in Simulation / Fortschrittsberichte Simulation (monographs in co-operation with ASIM, the German Simulation Society)
- Organisation of the Conference Series MATHMOD Vienna (triennial, in co-operation with EUROSIM, ASIM, and TU Wien) → www.mathmod.at
- Organisation of Summerschools and Seminars on Mod&Sim
- Administration and support of ASIM (German Simulation Society → www.asim-gi.org) and of EUROSIM (Federation of European Simulation Societies → www.eurosim.info)

ARGESIM – Arbeitsgemeinschaft Simulation News – Working Committee Simulation News – SNE Publication
Mommsengasse 19/8, 1040 Vienna, Austria; Tel +43-1-58801-10111, -10115; Fax +43-1-58801-910111
Email: office@argesim.org, office@sne-journal.org; WWW: www.argesim.org, www.sne-journal.org
Incorporated Austrian Society ZVR No 213056164 – EU VAT ID No ATU 72054279
Bank Account: ARGESIM, IBAN AT07 2011 1828 9115 0800, BIC GIBAATWWXXX, ERSTE BANK VIENNA

Vorwort

Die Workshops bzw. Tagungen der beiden ASIM/GI Fachgruppen „Simulation technischer Systeme“ (STS) sowie „Grundlagen und Methoden in Modellbildung und Simulation“ (GMMS) finden seit vielen Jahren jährlich an verschiedenen Standorten statt. Die Veranstaltungen dieser Reihe tragen dazu bei, die Diskussion, den Informations- und Erfahrungsaustausch zwischen Fachleuten auf den Gebieten der Simulation, der Modellbildung und des maschinellen Lernens zu fördern.

Am 8. und 9. März 2018 findet das ASIM Fachgruppentreffen erstmalig an der Hochschule Heilbronn statt. Über 40 Beiträge aus Wissenschaft und Industriepraxis werden präsentiert. Unter anderen konnten wir als Gastredner

- Prof. Dr.-Ing. Ralf Mikut, Karlsruher Institut für Technologie:
Maschinelles Lernen für das Energiesystem: Gegenwart und Zukunft
- Hr. Werner Schempp, Ministerialdirigent a. D.:
Baden-Württemberg zu Gast in der Welt - die Welt zu Gast bei uns
- Dr.-Ing. Hans Martin Ritt, Manager Application and Pilot Engineering, MathWorks:
Data Science für Ingenieure

gewinnen, siehe dazu auch die entsprechenden Beiträge in diesem Tagungsband.

Das Treffen wird begleitet von einer Ausstellung, auf der die anerkannten Fachfirmen ihre Produktpalette präsentieren. Wir begrüßen herzlich die VertreterInnen aus Forschungsinstituten und Industrie zum diesjährigen Workshop in Heilbronn und freuen uns gemeinsam mit Ihnen auf interessante Diskussionen und einen regen Erfahrungsaustausch. Die Beiträge sind alphabetisch nach der jeweiligen Institution bzw. anschließend nach den Nachnamen der AutorInnen sortiert.

Sehr herzlich möchten wir uns für die Unterstützung und die Mithilfe bei der Organisation der Tagung bei Fr. Forough Lang, Fr. Marion Strattner, Hr. Thomas Kriehn, Hr. Prof. Thomas Pospiech, Fr. Christina Deatcu, Hr. Prof. Walter Commerell, Hr. Prof. Thorsten Pawletta bedanken. Ebenso möchten wir uns bei allen Gastrednern, AutorInnen und Unternehmen für das Engagement und die Ausgestaltung der Tagung bedanken.

Es ist uns eine sehr große Freude, alle TeilnehmerInnen an unserer Hochschule und unserer Fakultät für Technische Prozesse begrüßen zu dürfen. Viele KollegInnen unserer Fakultät arbeiten, lehren und forschen auf dem Gebiet der Simulation und Modellbildung, was auch durch die zahlreichen Beiträge in diesem Tagungsband zum Ausdruck kommt. Durch die Tagung an unserer Fakultät unterstützen wir daher gerne die Vernetzung, Kooperation oder Zusammenarbeit zwischen den ExpertInnen in diesen sehr interessanten Fachgebieten.

Prof. Tobias Loose, Organisation der Tagung
Prof. Juliane König-Birk, Dekanin der Fakultät für Technische Prozesse
Heilbronn, im März 2018

Tagungsleitung:

Tobias Loose, Hochschule Heilbronn
Walter Commerell, Hochschule Ulm
Thorsten Pawletta, Hochschule Wismar

Thematische Schwerpunkte und Ansprechpartner:

Echtzeitsimulation und HiL	Robert Buchta
Grundlagen und Methoden in Modellbildung und Simulation	Thorsten Pawletta
Maschinelles Lernen	Tobias Loose
Mathematische Verfahren in Modellbildung und Simulation	Michael Striebel
Modellbasierte Funktionsentwicklung	Walter Commerell
Modellierungssprachen (Modelica, VHDL, AMS, etc.) und Standards	Klaus Panreck
Simulation in der Elektronikentwicklung	Joachim Haase
Simulation in der Energietechnik	Walter Commerell
Simulation in der Luft-und Raumfahrt	Umut Durak
Simulation in der Robotik / Steuerungstechnik	Thorsten Pawletta
Simulation mechatronischer Systeme	Heinz-Theo Mammen
Simulation thermischer Systeme	Leo Gall
Simulation und Modellbildung für die Ausbildung	Tobias Loose
Simulation von Elektro- und Hybridfahrzeugen	Xiaobo Liu-Henke
Technische Prozesse	Annett Großmann

Tagungsorganisation:

Hochschule Heilbronn
Campus Sontheim
Max-Planck-Str. 39
74081 Heilbronn
Tel.: +49 (0)7131 - 504 0
www.hs-heilbronn.de

Tagungsort:

Hochschule Heilbronn
Campus Sontheim
Max-Planck-Str. 39
74081 Heilbronn

Veranstalter:

ASIM/GI-Fachgruppe Simulation technischer Systeme

Die Fachgruppe Simulation technischer Systeme (STS) befasst sich innerhalb der Arbeitsgemeinschaft Simulation (ASIM) mit der Modellbildung und Modellstudien für die Simulation neu zu entwickelnder oder zu verbessernder technischer Geräte und Bauteile. In der Fachgruppe finden diejenigen einen Ansprechpartner, die sich mit der Bereitstellung und Anwendung von Werkzeugen und Programmen zur Modellerstellung bei der Simulation der genannten Systeme beschäftigen. Dazu werden Fachgruppentreffen und Fachgespräche zu aktuellen Themen organisiert. Themen sind u. a. Modellierung und Simulation mechatronischer und thermischer Systeme, Simulation in der Elektronikentwicklung, in der Medizintechnik und im Automobilbau, Echtzeitsimulation, neue Methoden der Regelungstechnik, Modellierungssprachen wie Modelica und VHDL-AMS, Modellaustausch - Stichwort Functional Mockup Interface (FMI) sowie auch der Einsatz von Simulationsverfahren in der Ingenieurausbildung.

www.asim-gi.org/sts

ASIM/GI-Fachgruppe Grundlagen und Methoden in Modellbildung und Simulation

Die Fachgruppe Grundlagen und Methoden in Modellbildung und Simulation (GMMS) befasst sich in enger Zusammenarbeit von Industrie und Forschungseinrichtungen mit methodischen Entwicklungen zu Modellierungsansätzen, numerischen und softwaretechnischen Verfahren, Algorithmen, Simulationswerkzeugen sowie simulationsgestützter Experimentiermethoden (z.B. Optimierung). Weitere Schwerpunkte bilden: die Kombination von Methoden der Simulation und KI, Methoden der Verkehrssimulation und die Anwendung von Methoden der Modellbildung und Simulation in der Ausbildung. Letztere werden teilweise in speziellen Arbeitsgruppen behandelt. Generell sind von besonderem Interesse Methoden und Werkzeuge, die über mehrere Anwendungsdomänen hinweg eingesetzt werden.

www.asim-gi.org/gmms/

Hochschule Heilbronn

Im Zentrum einer der wirtschaftlich stärksten Regionen Deutschlands, unweit der Ballungszentren um Stuttgart, Heidelberg und Mannheim, liegt die staatliche Hochschule Heilbronn.

Mit ca. 8.300 Studierenden ist die Hochschule Heilbronn eine der größten Hochschulen für Angewandte Wissenschaften in Baden-Württemberg. Ihr Kompetenz-Schwerpunkt liegt auf den Bereichen Technik, Wirtschaft und Informatik. An drei Standorten in Heilbronn, Künzelsau und Schwäbisch Hall bietet die Hochschule mehr als 50 Bachelor- und Masterstudiengänge an. Die Hochschule pflegt enge Kooperationen mit Unternehmen aus der Region und ist dadurch in Lehre, Forschung und Praxis gut vernetzt, siehe

<https://www.hs-heilbronn.de/2026130/die-hochschule>.

Inhaltsverzeichnis

	Seite
Einrichtung Ihrer Co-Simulation! Peter Hoffmann Chiastek GmbH	1
Design and Development of Research Aviation Training Device Haitao Wang, Umut Durak, Sven Hartmann Clausthal University of Technology, German Aerospace Center (DLR)	5
Prädiktion des Reglerverhaltens in industriellen Zuluftanlagen im Lackierbetrieb durch maschinelle Lernverfahren Philipp Oetinger, Simon Alt, Tobias Schlotterer, Martin Weickgenannt Dürr Systems AG	11
Modellierung und Simulation hochviskoser, nicht-isothermer Fluide mit freier Oberfläche Dimitri Harder, Edmond Skeli, Dirk Weidemann, Klaus Panreck FH Bielefeld	15
The RESIN climate change adaptation project and its simple modeling approach for risk-oriented vulnerability assessment Daniel Lückerrath, Manfred Bogen, Erich Rome, Betim Sojeva, Oliver Ullrich, Rainer Worst, Jingquan Xie Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS	21
Integration eines Multi-Agenten-Systems in eine Ereignisdiskrete Materialflusssimulation Michael Scholz, Toni Donhauser, Jörg Franke Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)	27
E Melt: Erweiterung einer unternehmensspezifischen Materialfluss- und Energiesimulation zur Abbildung variabler Betriebsstrukturen der Nichteisen- Schmelz- und Druckgussindustrie Andreas Buswell, Wolfgang Schlüter Hochschule Ansbach	33
Numerische Untersuchung einer Diffusor-Kontraktion und Beeinflussung des Strömungszustandes durch Wandkonturierung Konstantin Zacharias, Wolfgang Schlüter Hochschule Ansbach	39
Simulationsbasierte Bestimmung optimaler Betriebsparameter von Biogasanlagen David Wagner, Wolfgang Schlüter Hochschule für angewandte Wissenschaften Ansbach	45
Strategien zur Reorganisation von Mischstapeln in automatischen Blocklägern Markus Fittinghoff Hochschule Heilbronn	51
Alterssimulation anhand eines sensorgesteuerten Transportwagens Annett Großmann, Forough Lang Hochschule Heilbronn	53

WILLE: Wissenschaftliches Lehren und Lernen zur Förderung der Gründungskultur an Hochschulen	59
Jana Heimel, Joachim Allhoff, Martin Vural Hochschule Heilbronn	
Dynamische Simulation von Mini-Fahrzeugen mit Modern C++, Boost-Odeint und Robot-Operating-System	69
Niclas Heining, Uwe Ingelfinger, Ansgar Meroth, Frank Tränkle Hochschule Heilbronn	
Towards State of the Art in open-set Face Identification	75
Oliver Huettner, Carsten Lanquillon, Sigurd Schacht Hochschule Heilbronn	
Mechanische CAD-Simulationsverfahren; Anlagen-Engineering mit DMU	81
Werner Kalliwoda Hochschule Heilbronn	
Predictive Convolutional Autoencoding: Anomalieerkennung für eine automatisierte bildgestützte Qualitätsprüfung	83
Carsten Lanquillon, Jerome Tagliaferri, Marina Früh, Jananika Ragunathan, Inna Nenja Hochschule Heilbronn	
Einbindung von Labormodellen mit industriepraktischen Fragestellungen in der Hochschullehre	89
Tobias Loose, Thomas Pospiech Hochschule Heilbronn	
Lernerfahrung bei der Realisierung eines autonomen Transportfahrzeugs als Regelungstechnik-Laborarbeit	95
Christoph Obasohan, Thomas Pospiech, Tobias Loose Hochschule Heilbronn	
Das Zentrum für Maschinelles Lernen (ZML) an der Hochschule Heilbronn	97
Alexandra Reichenbach, Wendelin Schramm, Nicolaj Stache Hochschule Heilbronn	
Predictive IT-Service Operation	103
Sigurd Schacht, Carsten Lanquillon, Simon Vollert, Naftali Mishiev, Artjom Fink, Roland Licht, Bianca Parusel Hochschule Heilbronn	
Regelungstechnische Projektierung und Realisierung mechatronischer Systeme am Beispiel von Hochschul-Laboranlagen	109
Christian Siegle, Thomas Pospiech, Tobias Loose Hochschule Heilbronn	
Standardisierung dynamischer Modelle zur Analyse, Synthese, Simulation und Echtzeitrealisierung	113
Karlheinz Wolfmüller Hochschule Heilbronn	
Generierung von Simulationsmodellen zur Ermittlung des Systemverhaltens von Shuttle-Systemen	119
Thomas Kriehn, Franziska Schloz, Karl-Heinz Wehking, Markus Fittinghoff Hochschule Heilbronn, Universität Stuttgart	
MbedTarget - A Simulink Target for Cortex-M Microcontrollers	125
Olaf Hagendorf Hochschule Wismar	
Modellbasiert lernende Steuerung für Gelenkarmroboter (Work in Progress)	131
Georg Kunert, Thorsten Pawletta Hochschule Wismar	
Praxisbeispiel: Emulation mit SAP EWM	133
Alexander Klemm, Dominik Wittwer io-consultants GmbH & Co. KG	

Maschinelles Lernen für das Energiesystem: Gegenwart und Zukunft	137
Ralf Mikut Karlsruher Institut für Technologie	
Smooth virtual commissioning mit industrialPhysics	143
Isabelle Kautz, Florian Wegmann machineering GmbH	
Data Science für Ingenieure	145
Hans Martin Ritt, Marcus Maurer, Alexander Diethert MathWorks GmbH	
Baden-Württemberg zu Gast in der Welt - die Welt zu Gast bei uns	147
Werner Schempp Ministerialdirigent a.D.	
Deep Reinforcement Learning zur Bewegungsplanung und -regelung mobiler Roboter	149
Jan-Niklas Nieland, Marco Beesk, Frank Tränkle Neobotix GmbH, Hochschule Heilbronn	
Konzeption des Entwicklungsprozesses zur Berücksichtigung von Genderaspekten in fahrzeugmechatronischen Systemen	155
Xiaobo Liu-Henke, Haoqi Tao, Sven Jacobitz Ostfalia Hochschule für angewandte Wissenschaften	
Systemkonzept einer durchgängigen Low-Cost RCP-Plattform	163
Xiaobo Liu-Henke, Sven Jacobitz, Marian Göllner, Sören Scherler Ostfalia Hochschule für angewandte Wissenschaften	
Simulationsgestützte Konzeption der Antriebstopologie eines fahrerlosen Transportfahrzeugs	169
Xiaobo Liu-Henke, Sören Scherler, Marian Göllner, Johannes Maisik, Matthias Fritsch Ostfalia Hochschule für angewandte Wissenschaften	
Gesamtkonzept des Forschungsprojekts iREX 4.0	175
Sören Scherler, Xiaobo Liu-Henke Ostfalia Hochschule für angewandte Wissenschaften	
Modellbasierte Entwicklung einer Spurfolgeregelung mittels zeitdiskreten modellprädiktiven Regelalgorithmen (DMPC)	181
Jie Zhang, Marian Göllner, Xiaobo Liu-Henke Ostfalia Hochschule für angewandte Wissenschaften	
IoT gestütztes Maschinen-Monitoring - „Wearables“ für Maschinen	187
Martin Strube Ostfalia Hochschule für angewandte Wissenschaften, Hochschule Braunschweig Wolfenbüttel	
Approximation von Materialflussmodellen durch neuronale Netze und Gradient-Boost Verfahren	191
Bernd Allmendinger, Stefan Pfaff PPI-Informatik	
Implementing the Argesim C21 benchmark with Modelica components	197
Jan-Philipp Disselkamp, Peter Junglas, Alexander Niehüser, Phillip Schönfelder Private Hochschule für Wirtschaft und Technik Vechta/Diepholz (PHWT)	
Regressionsmethoden zur Ersatzmodellbildung und analytischen Beschreibung von Erdstoffmodellen	203
Christian Richter, Frank Will Technische Universität Dresden	
Simulation der automatisierten Schaufelbefüllung eines Radladers mit Modelica	209
Volker Waurich, Frank Will Technische Universität Dresden	

Modeling and Simulation Based Development of Autonomy Features for Drones	215
Shihui Chen, Umut Durak, Sven Hartmann TU Clausthal, German Aerospace Center (DLR)	
Analyse, Simulation und Optimale Steuerung eines mathematischen Dengue-Fieber Modells mit Impfung	223
Gaby Albrecht, Anne Fischer, Kurt Chudej Universität Bayreuth	
Ein Ansatz zur objektorientierten Modellierung und Simulation technischer Systeme mit dem Physolator	231
Dirk Eisenbiegler Universität Furtwangen	
Model-driven development and simulation of Integrated Modular Avionics (IMA) architectures	233
Björn Annighöfer Universität Stuttgart	
Variability Modeling for Engineering Applications	239
Hendrik Folkerts, Thorsten Pawletta, Christina Deatcu, Umut Durak University of Applied Sciences Wismar, German Aerospace Center (DLR)	
Experimental Adaptation of a Training Simulator for Manual Welding Processes towards the Teach-In of Welding Robots	249
Benjamin Knoke, Mareike Voskuhl, Marcel Tebbe, Markus Häveker , Christian Gorldt, Klaus-Dieter Thoben University of Bremen, Daimler AG	
Maschinelles Lernen beim visuellen HMI in der Testautomatisierung vernetzter Fahrzeugfunktionen bei Volkswagen	253
Robert Buchta, Adrian Heller Volkswagen AG	
MATLAB unterstützte Auslegung von Signalleitungen	259
Katrin Reblinsky VWA-Hochschule für berufsbegleitendes Studium	

Einrichtung Ihrer Co-Simulation!

Peter Hoffmann

Chiastek GmbH

peter.hoffmann@chiastek.com

Unterstützung Ihres Wissen und Vertrautheit mit der verwendeten Simulationsumgebung ist wesentlich bei der Kopplungsfrage von Simulatoren, Werkzeugen und Modellen. Um notwendige Interdependenzdaten zu erhalten, braucht man eine unabhängige Instanz, die durch native Kopplung bereits im Einsatz befindlicher Simulationswerkzeuge erreicht wird. Wahrscheinlich am wichtigsten für Sie, ohne ein weiteres Werkzeug neu zu lernen und Ihr geistiges Eigentum teilen zu müssen. Diese Lösung versteht sich als Hintergrundaufgabe und erlaubt die dynamische Steuerung des bi-direktionalen Datenaustausches von Abhängigkeiten zwischen Ein- und Ausgängen zur Optimierung oder Erforschung neuer Designs. Am Beispiel von Leistungselektronik profitieren auch thermische und elektromagnetische Experten und deren Modelle von Interdependenzdaten. Dieser Beitrag zeigt Details der verwendeten Co-Simulation Architektur, erläutert Hintergründe der Kopplung, beschreibt Möglichkeiten der Tool-Unterstützung und gibt ein Beispiel aus der Multiphysik Co-Simulation.

Die Einleitung

Nehmen wir mal an, dass Sie der Experte in Sachen Leistungselektronik sind, Simulation ist ein wichtiger Meilenstein in Ihrem Entwicklungsprozess und Sie liefern gute Ergebnisse bei der Entwicklung elektrischer Antriebsstränge. Wäre es nicht aufregend, herausragende Designs und Lösungen zu finden, indem wir alle Interdependenzen Ihrer Leistungselektronik simulieren, die elektromagnetische, thermische und mechanische Effekte beeinflussen oder bewältigen. Ihre persönliche Erfahrung ermöglicht es Ihnen, Leistungsverluste in Elektromotoren abzuschätzen, die durch nicht optimale Induktionsentwürfe, unnötige Wärmeverluste und mechanische Abnutzung verursacht werden können und die alle langfristige Zuverlässigkeit beeinträchtigen. Wäre es nicht großartig, Ihr Design mit direkten simulierten Daten für Wärme, Verschleiß und elektromagnetische Effekte zu betrachten, die Ihre Schätzungen unterstreichen? Das klingt schwer zu erreichen, aber ist es wirklich?

Unterstützung Ihrer Erfahrung und Vertrautheit mit der verwendeten Simulationsumgebung ist ein wesentlicher Aspekt. Der fehlende Teil, um die oben beschriebenen Interdependenzdaten zu erhalten, kann durch eine native Kopplung mit anderen bereits im Einsatz befindlichen Simulationswerkzeugen erreicht werden. Wahrscheinlich am wichtigsten für Sie, ohne ein weiteres Werkzeug neu zu lernen und die Notwendigkeit, Ihr geistiges Eigentum zu teilen. Dank unserer Kopplungsmöglichkeiten können Sie Ihr Simulationswerkzeug sehr einfach bereichern, die Effizienz der Entwicklung steigern und Ihre Ziele schneller erreichen.

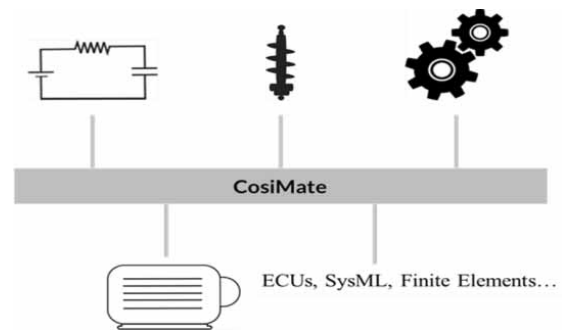


Abbildung 1. zeigt den Co-Simulation Bus.

Die Plattform-Architektur

Am ausgewählten Beispiel wie Leistungselektronik elektromagnetische, thermische und mechanische Effekte beeinflusst wird der Wert von nativer Simulationskopplung besonders deutlich. Sie ermöglicht Ihnen alle relevanten Effekte, die Ihre Leistungselektronik beeinflussen kann, zu simulieren ohne tiefe Expertisen anderer Physik, Domänen oder Simulationstools haben zu müssen. Die nativ-gekoppelten Simulatoren liefern Ihnen die Ergebnisse, die für Ihre Entwicklungsaufgabe von Bedeutung sind. Außerdem ermöglicht die native Kopplung netzwerkbasierter Zusammenarbeit in der Simulation zwischen Kollegen und Zulieferer auf regionaler oder globaler Ebene.

Zur Erreichung einer nutzerfreundlichen Co-Simulation innerhalb existierender Entwicklungsprozesse und Engineering-Tools gehört eine offene Plattform. Im Gegensatz zu einer Punkt-zu-Punkt-Lösung, bei der nur ein Simulator mit einem anderen Simulator

kommunizieren kann, kommt hier eine offene Co-Simulation Bus-Architektur zum Einsatz. Die Lösung bietet wesentliche Vorteile:

1. Uneingeschränkte Integration und Kommunikation zwischen heterogener Simulatoren
2. Flexible Simulation im Netzwerk um CPU-Nutzung und Simulationsleistung zu optimieren
3. Dynamische Konfiguration von Master-/Slave- und anderen Simulations-Steuerungen

Bereits Anfang 2000 wurde die Idee der nativen Simulationskopplung am TIMA-Forschungslabor in Grenoble geboren. Die Hauptidee bestand darin, einige Methoden zur Validierung heterogener Systeme zu entwickeln. Die erste Veröffentlichung wurde von Prof. A. Jerraya auf der MEDEA Konferenz zum Thema Embedded System Design gemacht. Im Jahr 2005 hatte die Groupe PSA ähnliche Simulationsherausforderungen und konnte von der Entwicklung einer offenen Co-Simulation Bus-Architektur namens CosiMate erstmalig profitieren. Diese frühe Produktbasis entwickelte sich zu einer ausgereiften Plattform, jetzt in der 9. Revision und einer robusten IT-Lösung, welche heute weltweit hauptsächlich bei Kunden aus der Luft- und Raumfahrt und der Automobilindustrie täglich eingesetzt wird. Die Möglichkeiten des Einsatzes der Co-Simulation Plattform sind unabhängig von der Industrie, da heterogene Simulation überall stattfindet.

Die Lösung um Simulatoren koppeln zu können leistet nicht allein die Kommunikationsschnittstelle zur Interkonnektivität. Die reale Herausforderung liegt in der korrekten Interpretation der ausgetauschten Daten in Bezug auf Zuverlässigkeit, Komplexität und Richtigkeit. Bi-direktionale Abhängigkeiten zwischen Realzeit- und Nicht-Realzeit-Systemen, zwischen langsamen, schnellen oder diskreten Daten sowie zwischen Uhrzeit-, Zeit- oder Trigger-gesteuerten Ereignissen brauchen spezifisch entwickelte Algorithmen. Verfügbare Technologien nennen sich Numerical Solution, Signal-based Extrapolation, Model-based Extrapolation und Waveform Relaxation. Eine effiziente Kombination von Technologien im Kopplungs-Algorithmus ist das Geheimnis um heterogene Subsysteme interagieren zu lassen. Unsere entwickelten Algorithmen unterstützen die Besonderheiten und Herausforderungen der Automotive & Aerospace Industrie. Unsere Bemühungen, die familiäre Entwicklungsumgebung bei zu behalten, die einfache Nutzbarkeit und schnelle

Konfigurierbarkeit zu unterstützen, verbessert Ihr neues oder existierende Simulation Modell in allen Phasen des Entwicklungsprozesses.

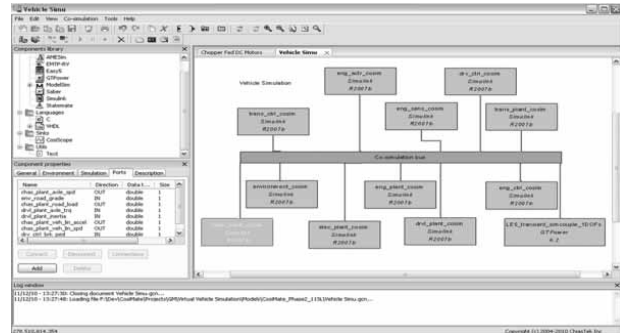


Abbildung 2. zeigt die CosiMate Benutzeroberfläche

CosiMate unterstützt intuitiv das Setup und die Kopplung jedes Simulators. Benutzern konfigurieren die Co-Simulationssitzung mit einer grafischen Drag & Drop-Schnittstelle und müssen lediglich folgenden Parameter definieren: Auswahl der Simulationstools aus einer Liste (Typ, Netzwerkadresse), Verbindungstypen zwischen Simulatoren (Ein- und Ausgänge) und gewünschter Simulationsstartmodus (automatisch oder manuell). Ein Datenmanager steuert die Datensynchronisation zwischen den verschiedenen Umgebungen während der Co-Simulation. Interne Applikationen sind ebenfalls verfügbar, um zum Beispiel die Aktivität auf dem Co-Simulation Bus zu verfolgen. In den meisten Fällen sind CosiMate Komponenten in den Komponentenbibliotheken der unterstützten Simulatoren verfügbar und können im Benutzermodell durch eine einfache Drag-und-Drop-Funktion hinzugefügt werden. Sie verbinden das Modell mit dem Co-Simulation Bus und damit mit den anderen Modellen. Sie können ihr Modell weiterhin über Ihr eigenes Simulationstool starten, als wäre es eine typische Simulation oder Sie nutzen eine vereinfachte Steuerungsschnittstelle der Plattform Applikation.

Die hier beschriebene Bus-Architektur ist entscheidend für eine nachhaltige und langfristig offene Lösung. Sie versteht sich generell als Hintergrundaufgabe und ermöglicht eine dynamische Steuerung des bi-direktionalen Datenaustausches von Interdependenzen (Ein- & Ausgänge) zur Optimierung oder Erforschung neuer Designs. Am gewählten Beispiel werden auch thermische und elektromagnetische Experten von Interdependenzdaten der Leistungselektronik zusätzlich profitieren und so ihr eigenes thermisches oder elektromagnetisches Simulationsmodell optimieren können.

Die Tool-Unterstützung

Die Zeit für Collaborative Design ist überfällig, aber vielleicht zu schwierig, um in einer bunten Welt von Simulationswerkzeugen Realität zu werden. Diese leistungsfähigen Simulationswerkzeuge sind über Jahrzehnte konstanter Erfahrung gewachsen und liefern heute die besten Ergebnisse in ihrem spezifischen Bereich der Mono-Physik. Die Bereicherung dieser Werkzeuge in Richtung Multi-Physik Co-Simulation lässt Ihr Simulationsprojekt Realität werden. Die native Kopplung ist unser Fachwissen. Jeder Simulationsanwender kann von dieser nativen Unterstützung profitieren, indem er System-Simulationsanwendungen in einer immer schnelllebigen und elektrifizierten Welt erhöht.

Es ist klar, dass die Komplexität einer Veränderung zu einem anderen Simulationswerkzeug ein sehr riskantes Unterfangen und eine große Herausforderung für die Ingenieurswelt darstellt, wo Zeit bis zur Marktreife am aller wichtigsten ist. Und warum sollte jemand sein Werkzeug wechseln, solange es seine technischen Bedürfnisse abdeckt. All das Simulationswissen basiert auf dem verwendeten Simulationswerkzeug und der im Laufe der Zeit erstellten Modellbibliothek, die vom selben Werkzeug abhängt.

Da die Komplexität in der Entwicklung stetig zunimmt und Interdependenzen nicht mehr ausgeschlossen und/oder zu einem späteren Zeitpunkt berücksichtigt werden können, ist es Zeit die Simulation von der Mono-Physik zur Multi-Physik Co-Simulation wachsen zu lassen. Ohne die Vertrautheit in der Simulation aufzugeben und ein neues Werkzeug erlernen zu müssen, bereichert die CosiMate Plattform Ihre Simulationsumgebung durch native Kopplung zu jedem anderen Simulations-Werkzeug, das von Kollegen, Partnern und Lieferanten genutzt wird.

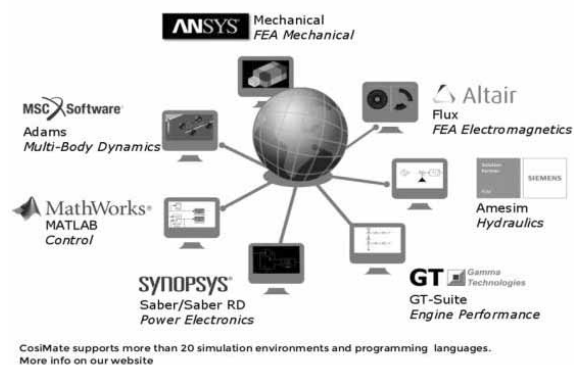


Abbildung 3. zeigt Beispiele des Partner-Netzwerks

Die Möglichkeiten einer Tool-Unterstützung sind theoretisch grenzenlos und werden durch native Schnittstellen und Bereitstellung von APIs umsetzbar. Die Implementierung neuer Tools auf den Co-Simulation Bus ist schnell und unproblematisch realisierbar. Die Liste unterstützter Tools umfasst schon heute mehr als 25 und wächst mit jeder neuen System-Simulation Anforderung.

Native Kopplung ermöglicht es Ihnen, alle Design Experimente durch Ihre bevorzugte Benutzeroberfläche selbst zu realisieren. Diese nicht störende Vorgehensweise macht Sie zum Master der Multiphysik Co-Simulation, bereichert Ihre Ingenieurleistung und nutzt Ihre Bibliothek von Simulationsmodellen

Das Applikationsbeispiel

Das verwendete Beispiel einer Multi-Physik Co-Simulation stammt von der Groupe PSA und nutzt CosiMate zur Stop/Start Funktionsoptimierung. Es ging letztendlich darum eine neue Plattform der Zusammenarbeit für die Dimensionierung von Baugruppen zu entwickeln, die unter anderem Saber und Matlab Simulation nutzen. Zu den Herausforderungen zählten eine neue Vorgehensweise im Design-Prozess, eine Modell-Generierung mit Monte-Carlo Ansatz und eine Einrichtung von zugehörigen Tools und Prozessen der Co-Simulation.

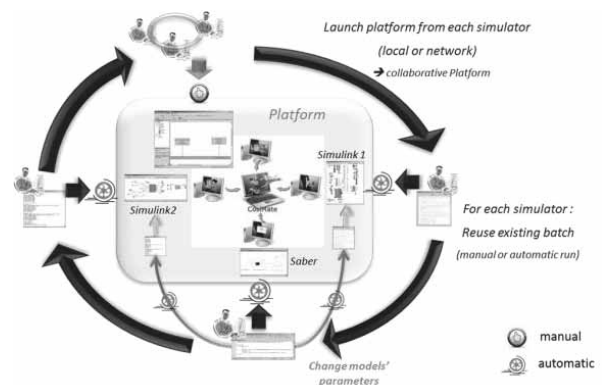


Abbildung 4. zeigt die von Groupe PSA eingerichtete Co-Simulation Plattform

Hierbei war es das Ziel der Groupe PSA eine optimierte Dimensionierung von Baugruppen zu erreichen. Um Studien über die Auswirkung der Motor Stop-/Start-Funktion auf andere Baugruppen durchführen zu können, kombiniert PSA ihr Motor-/Starter-

Simulationsmodell Matlab Simulink basierend und ihr Stromversorgungs-Simulationsmodell Synopsys Saber basierend in einem Co-Simulation Multiphysik-Modell. Eine unproblematische Tool-Kopplung dank CosiMate ermöglichte Methoden wie „Worst Case Size“ Annäherung mit überdimensionierten Baugruppen und „Statistical Sizing“ Annäherung mit schwellwert-basierenden Baugruppen zusammenhängend durchzuführen.

Allgemein gibt es viele gute Gründe Monte-Carlo-Analysen (was, wenn Studien) durchzuführen, vor allem für Optimierungsaufgaben. Große Ideen mit früher Nutzung von neuesten Technologien erfordern stetige Optimierungen nicht nur während des gesamten Produktlebenszyklus, vor allem aber für die nach vorn gerichtete Nutzung.

Unter der Prämisse von garantierten Motor-Leistungen und diversen elektrischen Einschränkungen wurden alle möglichen Analysen durchgeführt. Reduzierte Kosten, geringes Gewicht und breitere Plattformnutzung sind Optimierungsergebnisse, auf die Groupe PSA sehr stolz sein kann.

Die Zusammenfassung

Zur Etablierung von Zusammenarbeit zwischen Entwicklungsteams, multiplen Standorten und Regionen wie auch zur Unterstützung von parallelen Entwicklungsprozessen kommen vernetzte Lösungen zum Einsatz. Integrierte Test & Measurement Tools, Prozesse zur Optimierung von Genauigkeiten, Unterstützung von multiplen Versionen sowie Offenheit zur Kooperation mit anderen Simulations-Herstellern ermöglichen eine gewinnbringende System Simulation.

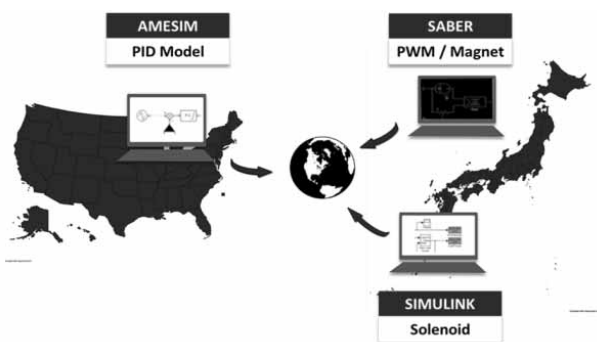


Abbildung 5. zeigt eine über VPN weltweit vernetzte Co-Simulation Plattform

Der Themenkomplex System-Simulation ist gerade heute von besonderer Bedeutung, wo diversifizierte Entwicklungsschritte zwischen Konzern internen, ausgelagerten Unternehmensbereichen und externen Zulieferern vorliegen; also besonders in der Aerospace & Automotive Industrie. Um komplexe Entwicklungsziele systemkompatibel zu realisieren braucht man eine Co-Simulation Plattform, die zuverlässig und unmittelbar allen Beteiligten zur Verfügung steht und ohne dabei Geistiges Eigentum preisgibt. Bei der Vielzahl von verwendeten Tools, Sprachen und Standards ist eine neutrale Instanz, welche unabhängig und ohne zusätzliche Beeinflussung Simulatoren koppelt, eine unabdingbare Grundvoraussetzung. Nur so kann man einen wirklich nachhaltigen Beitrag der Simulation liefern, mit den Zielen:

- 1) Verbesserung der Zusammenarbeit zwischen Entwicklungsteams und Lieferanten
- 2) Unterstützung der gewohnten Entwicklungs-umgebung
- 3) Steigerung der Entwicklungseffizienz und Beschleunigung des Marktreife-Prozesses
- 4) Rechtzeitige Erkennung von System-Designfehlern

CosiMate unterstützt viele einzigartige Simulationen, die auf die Kreativität der Simulation Anwender zurück zu führen sind. Welche Simulationswerkzeuge auch immer gekoppelt werden müssen, die Co-Simulation Plattform ist dafür gerüstet. Unsere Unabhängigkeit erlaubt es Ihnen, alle Arten von Werkzeugkombinationen von jedem Hersteller nutzen zu können. Wir überlassen dem Ingenieur seine bevorzugte Simulationsumgebung, bereichern den Werkzeugeinsatz und unterstützen den erforderlichen Datenaustausch. Dieser einzigartige Ansatz hat CosiMate in vielen Entwicklungsumgebungen so beliebt und nützlich gemacht Die frühzeitige Erkennung von unzulänglichen Technologien, riskanten Designs und möglichen Leistungsprobleme während des Produkt-Lebenszyklus sind Ergebnisse der Mono-/Multi-Physik Co-Simulation.

Design and Development of Research Aviation Training Device

Haitao Wang¹, Umut Durak^{1,2}, Sven Hartmann²

¹Clausthal University of Technology, Department of Informatics

²German Aerospace Center (DLR), Institute of Flight Systems

haitao.wang@tu-clausthal.de

Abstract: Though there are numerous Aviation Training Devices (ATDs) available on the market, it is still hard to find a specifically designed one for research and education purposes. Based on the requirements of researchers and students, we designed and developed a Research Aviation Training Device (RATD) for the Aeronautical Informatics Research Group at Clausthal University of Technology. The RATD not only arouses the interest of students in aeronautics but also provides them means to teach them basics of flight and aeronautical informatics. It comprises a training syllabus and corresponding brief flight training manual to support basic flight training with generic cockpit elements. It enables Man-in-the-Loop simulator testing with its open architecture software infrastructures. Training syllabus further includes a hands-on tutorial that demonstrates simulator-based testing of flight systems.

1 Introduction

Aeronautical informatics is becoming a more and more important topic of interest in both industry and academia. Durak and colleagues present the advances in this applied informatics field in their new recent book [1]. In the meanwhile, with the unique capabilities that they offer, the flight simulations [2] are becoming an indispensable part in aeronautical research and education. While there are many previous efforts that focus on how the flight simulation helps in aeronautical engineering research and education, there are almost no example of how we can use them in aeronautical informatics. Though there are numerous Aviation Training Devices (ATDs) available on the market, it is still hard to find a specially designed ATD for research and education purposes. It is significant for aeronautical informatics researchers and students to have a specially designed flight simulation system, which could obtain some basic flight training and verify their designs in a realistic environment.

There are many motivations to design and develop our own flight simulator for aeronautical informatics research and education some of which are budget limitations, flexibility requirements, providing basic flight training, creating interest and motivation and provid-

ing reinforcement.

1. *Budget limitations* are facts in academic setting. It is hard to purchase a sophisticated flight simulator with a limited budget and the increased maintenance costs make it worse;
2. *Flexibility requirements* are essential concerns for the design of simulator architecture which needs to be adapt to different circumstances. A simulator shall provide a controlled experimentation platform where the qualities of the developing system can be evaluated before deploying it into the real world. The major issue of simulators is that there is no such an open platform designed for aeronautical informatics research which can be easily configured to conduct different types of flight simulation testing;
3. *Providing basic flight training* is a crucial requirement in aeronautical informatics education to teach the basics of flight and aircraft control. Lots of students do not have any opportunities to fly an aircraft even in a flight simulator. On the other hand, current ATDs on the market are designed for professional pilots training purposes, their training processes are usually too complex and not necessary for students;

4. *Creating interest and motivation* is the key for acquiring students' attention and motivating them to study aeronautical informatics topics.
5. *Providing reinforcement* is the most obvious method to strengthen memory in education. Students implement their design in flight simulation system can reinforce what they learned in the lecture. The major issue to consider for reinforcement is that how to merge tedious theoretical knowledge into more interesting experimental practices for aeronautical informatics novices.

This project proposes a ground-based Research Aviation Training Device (RATD) which aims to fill in that gap and offers aeronautical informatics researchers and students, especially aeronautical novices, an easier and lower-cost opportunity to explore and experiment general aeronautical informatics knowledge. Our research takes place at the Aeronautical Informatics Laboratory in Clausthal University of Technology. We elicit the following requirements which should be fulfilled in the design and development of this RATD system.

1. The development and maintenance of the whole RATD system should be easy and low-cost;
2. The RATD system architecture should be loosely coupled and easily extensible to fit different types of flight simulation testing requirements;
3. The RATD system should simulate a realistic environment to provide some basics of flight training;
4. The RATD system should have user-friendly interfaces and be more interactive with users;
5. The RATD system should be easily configurable to provide various experimental practices of aeronautical informatics.

The RATD system is mainly designed and developed to provide a low-cost flight training device. It provides not only means for better understanding of how an aircraft flies in the air, but also a convenient developing and testing platform where some basic flight simulator testing and evaluation can be performed. It can also arouse students' interest in aeronautical informatics education and demonstrate flight with a vivid visual experience. To provide the basic flight training, we design a training syllabus and brief flight training



Figure 1: RATD Setup.

manual. A list of basic training requirements based on this syllabus should be fulfilled at the final evaluation. According to those training requirements, the corresponding pilot flying skills which can be learned from the brief flight manual are required to be obtained during the training process. Besides training scenarios for introducing basics of flight, it provides a hands-on tutorial to demonstrate how to conduct Man-in-the-Loop simulator testing of flight systems. The tutorial demonstrates model-based design and simulation-based verification of a simple auto-throttle controller using Scilab/Xcos¹. It includes an example of Model-In-the-Loop (MIL), Software-In-the-Loop (SIL) and Hardware-In-the-Loop (HIL) tests. Figure 1 shows the RATD which is located at the Aeronautical Informatics Laboratory in Clausthal University of Technology.

2 Flight Simulation in Aeronautical Education

Flight simulation is playing a significant role of training resources in aeronautical education such as in the aeronautical engineering discipline. The study of design and development of a most suitable flight simulation system itself has become a key factor to influence the quality of aeronautical education. It is quite hard to find relevant works which focus on aeronautical informatics education, however there are some existing works on aeronautical engineering education, which could be referenced to guide our design and develop-

¹<https://www.scilab.org/scilab/gallery/xcos>

ing processes.

As described in [3] in 1991, they already used "Microsoft Flight Simulator" as a "Flight-Test Platform" in introductory courses and flight performance laboratory courses for aerospace education at the University of Notre Dame and the University of California, they believed that the use of Microsoft Flight Simulator had been benefit to their programs. In 1997, the work [4] summarized ten years experiences of inclusion of flight simulation into the undergraduate aeronautical engineering curriculum, which had proved that the Genesis Engineering Flight Simulator helps aeronautical education in many ways, for example, enhancing student understanding of difficult concepts, giving students and faculty an enjoyable and educational experience. Another useful case was mentioned in [5] in 1998, the Flight Engineering Systems Trainer(FEST) was a reconfigurable PC-based device, which was designed for students to acquire flight test data and to demonstrate the aircraft performance and handling qualities. In 1999, the work [6] described a simulation system which gave studying flight control systems students a great opportunity to implement and test their designs in a realistic environment. With the advantage of using MATLAB, GENESIS 3000 was used in flight mechanics education, which had the ability to perform any types of flight control law which is supported through MATLAB toolboxes [7]. The work [8] in 2007 presented the use of flight simulator in flight dynamics and control education to bring the student higher motivation and better learning quality at West Virginia University.

3 RATD Architecture

The design of the RATD system architecture is as shown in Figure 2. The main part of the RATD system architecture is loosely coupled into five components namely Interface Computer (IC), Instructor Operator Station (IOS), Visual Simulation (VIS), Model, and Cockpit. There are also Extensible Interfaces capable of working with other platforms. For example, with this special design, The RATD can work with Raspberry PI ² and Scilab/Xcos platforms which provides a model-based flight system developing and testing.

2Simulate is a distributed real-time simulation framework capable of facilitating integrating a wide range of models and simulation components like data

²<https://www.raspberrypi.org/>

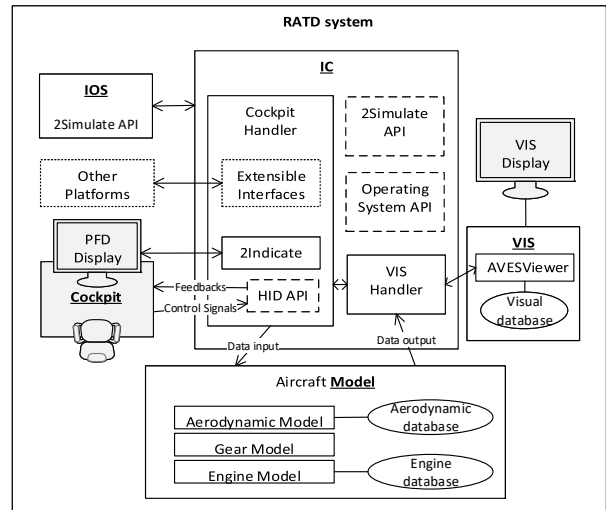


Figure 2: RATD System Architecture.

recorders or image generators to co-simulation of various subsystems with complex behaviors and interfaces [9]. In 2Simulate simulation architecture, there are three basic concept units called Control Center, Target and Model, three basic software components namely 2Simulate Real-Time Framework (2SimRT), 2Simulate Control Center (2SimCC) and 2Simulate Model Control (2SimMC). One Control Center controls a number of Targets which may co-simulate various Models. The RATD system is built based on the 2Simulate framework.

IOS is the control center capable of Run, Pause and Stop the entire system, which is based on 2SimCC. It connects and manages two other components named targets: IC and Model through UDP ports. It has the ability to access and edit targets data dictionaries at runtime.

IC is based on 2SimRT, which plays an indispensable role of interface throughout the whole system. It is connected with Cockpit through USB cable, VIS, Model and IOS through UDP ports. The IC handles the application logic and data communication among Cockpit, Model and VIS, like how to process the Cockpit input data from pilots before sending them to the aircraft Model and what should the aircraft Model react after touching the border of the terrain. The connection with IOS lets IOS to access and change the data dictionaries at runtime.

Model of aircraft that we used is a nonlinear aircraft VFW 614 provided by German Aerospace Center (DLR), Institute of Flight Systems. The model computes the flight dynamics and outputs flight states

back to IC after getting the Cockpit devices signals with pilot instructions as its inputs from IC.

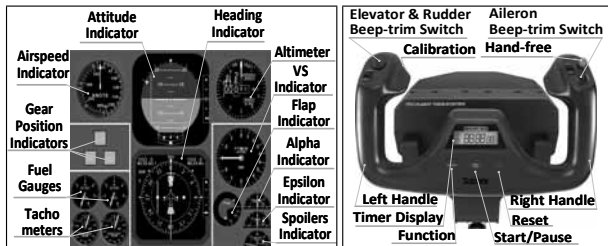


Figure 3: PFD ³.



Figure 4: Yoke ³.

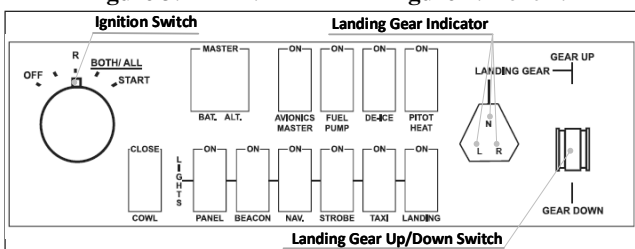


Figure 5: Switch Panel ³.

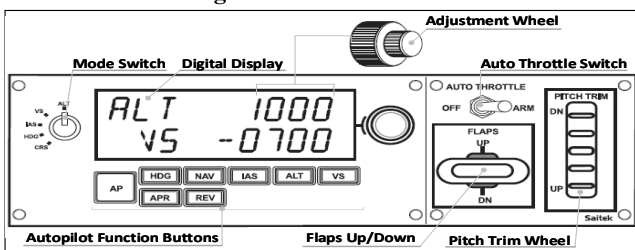


Figure 6: Multi Panel ³.

Cockpit includes Primary Flight Display (PFD), physical Human-machine Interfaces and Out-of-cockpit Display (VIS Display). To offer better user experiences with low-cost, we use the ground-based cockpit with operational instrumental bundles provided from Saitek ³ which is capable of meeting the basic flight operation instructions. These devices include Pro Flight Yoke with Throttle Quadrant, Pro Flight Pedals, Pro Flight Multi Panel, Pro Flight Switch Panel and Pro Flight Instrument Panel. With the help of a multi-platform library named HIDAPI⁴, we build the communication between these cockpit devices and the RATD via USB I/O interfaces on Windows. The design of PFD is as shown in Figure 3. the functional designs of Yoke, Switch Panel and Multi Panel are as shown in Figure 4, Figure 5 and Figure 6.

³Reproduced from <http://www.saitek.com/uk/down/manuals.html>

⁴See <http://www.signal11.us/oss/hidapi/>

4 RATD for Basic Flight Training

4.1 Training syllabus

The training syllabus is designed to have one scenario with multiple missions. Each mission requires several key skills to be accomplished by the pilots during the flight envelope. To accomplish this training syllabus, pilots should read the Brief Flight Training Manual to learn how to fly RATD simulation system with the VFW 614 flight model.

The Scenario

Start the airplane and take off at the Braunschweig airport, climb and maintain cruising flight, then turn back towards to the Braunschweig airport, finally descend and decelerate to land on the Braunschweig airport. This one-round flight scenario is divided into three missions: Takeoff, Climb and Cruise and Landing.

Missions

1. **Takeoff:** Start and take off the flight model at the Braunschweig airport;
2. **Climb and Cruise:** Start the flight model in the air, climb to the assigned cruising level, then perform cruising flight.
3. **Landing:** Start the flight model in the air and descend and decelerate to land on the runway at Braunschweig airport.

4.2 Brief Flight Training Manual

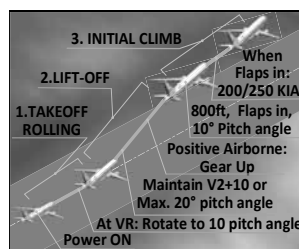


Figure 7: Takeoff.

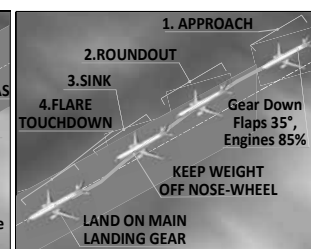


Figure 8: Landing.

The brief flight manual is specially designed for performing the RATD flight simulation training which contains the basic flight skills required in the training syllabus. It is based on the official Flight Training Manual of VFW 614 [10]. It presents the basic aircraft handling and system operation of VFW 614 model in RATD system including preflight preparations and

three types of basic flight scenarios: Preflight Preparation and Check; RATD Setup and Calibration; Cockpit Familiarization; Simulation Starting; Engine Starting; Rolling Takeoff as depicted in Figure 7; Climb and Cruise; Descent and Landing as depicted in Figure 8. It is designed to be used only for the VFW 614 model in the RATD system with special regard to give aeronautical informatics researchers and students some basic flight instructions. If you have changed another aircraft model, this manual maybe not suits your situation. However, the basic concept of flight operating and controls gained from this flight training manual should be helpful in any case.

5 RATD for Flight System Development

RATD is not only a device which is useful for basic flight training but also a flexible and easily configured platform where aeronautical informatics research can be conducted in varieties of fields such as automatic flight control law design and evaluation and pilot assistance system. The domain of aeronautical informatics research is mainly focusing on the intersection between informatics and aeronautics [1]. This section is going to demonstrate how to build a flight system with model-based design and verify it with Man-in-the-Loop simulator tests in RATD using a simple auto-throttle controller example.

5.1 Modeling and Simulation-based Development Environment

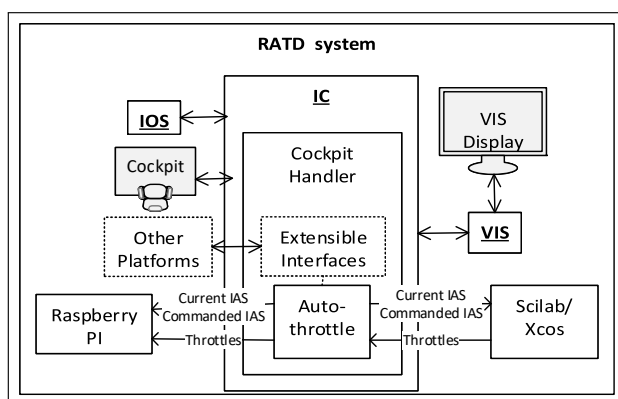


Figure 9: Modeling and Simulation-based Development Environment Architecture.

The modeling and simulation-based development environment architecture is as shown in Figure 9. The

connection with Scilab/Xcos provides us a modeling and simulation-based development environment. In the Cockpit Handler sub component of IC, we add an Auto-throttle sub component with the help of Extensible Interfaces. The Auto-throttle sub component is responsible for building the connection between the Cockpit subsystem and the auto-throttle controller. The communication protocol used in this tutorial among Auto-throttle sub component, Scilab/Xcos and Raspberry PI is UDP protocol. In the Scilab/Xcos, the adapted UDP receiving and sending blocks are used. In Raspberry PI, the normal UDP sockets are used. When the pilot sets the commanded airspeed in the Cockpit side. The Auto throttle sub component of Cockpit Handler takes the cockpit input controls over and sends the current airspeed and commanded airspeed to Scilab/Xcos or Raspberry PI. The controller in Scilab/Xcos or Raspberry PI calculates the corresponding values of throttles, then sends them back to the Auto-throttle sub component. The Auto-throttle sub component uses those values calculated by the auto-throttle controller to set the aircraft throttle volumes.

5.2 Modeling and Simulation-based Development of Auto-throttle Controller

After the environment is successfully built, the next step is to utilize it to design, develop and test an auto-throttle controller. The design of auto-throttle controller can be variant depending on the requirements.

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt} \quad (1)$$

In this tutorial, we are going to design a simple Proportional-Derivative controller (PD controller). First, we need design the mathematical model as shown in Equation 1. Then, we use the model-based design tool Scilab/Xcos to represent that equation with models as shown in Figure 10. Eventually, we should perform a Model-in-the-Loop (MIL) testing as shown in Figure 11 to verify that designed Auto-throttle model. If it passes the MIL testing, the next step we need generate code with the help of Xcos Re-useable and Customizable Code Generator⁵. Next, to verify the generated code, a Software-in-the-Loop (SIL) testing as shown in Figure 12 should be performed. Finally, if it passes the SIL testing, we can deploy the generated code into a target hardware like Raspberry PI 3 used

⁵<https://forge.scilab.org/index.php/p/xcos-code-generator/>

in this tutorial to make a Hardware-in-the-Loop (HIL) testing.

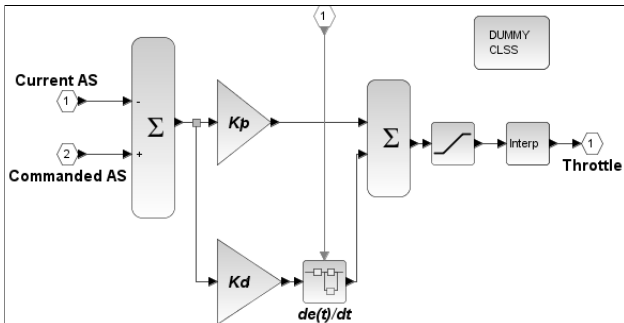


Figure 10: Auto-throttle Controller.

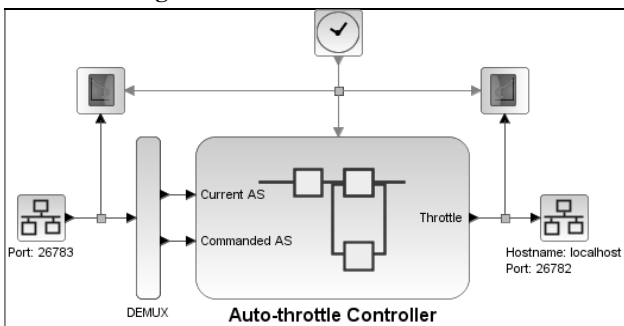


Figure 11: MIL Model.

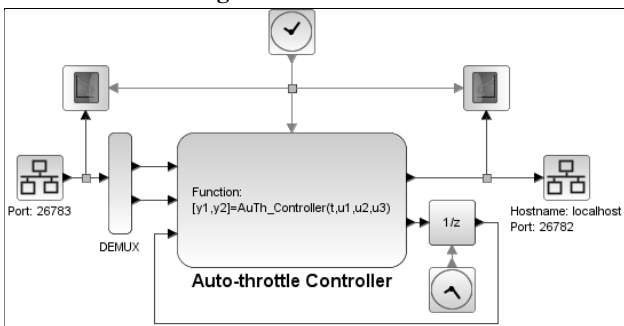


Figure 12: SIL Model.

6 Conclusion

In this paper, we analyzed the requirements for a flight simulation in aeronautical informatics research and education. Then we summarized the works about how flight simulation took positive effects on aeronautical education. We took some references from those works and developed a specially designed RATD for aeronautical informatics research and education purposes based on those requirements. We also designed the training syllabus and brief flight training manual for offering the basic flight training, and a hand-on tu-

torial to demonstrate how to build flight systems by implementing a modeling and simulation-based approach. After having RATD operated in the aeronautical informatics laboratory, we are now getting positive feedbacks from students and researchers.

References

- [1] Umut Durak, Jürgen Becker, Sven Hartmann, and Nikolaos S.Voros. *Advances in Aeronautical Informatics. Technologies Towards Flight 4.0*. Springer, 2018.
- [2] David Allerton. *Principles of flight simulation*. John Wiley & Sons, 2009.
- [3] EJ Jumper and JW Baughn. The use of Microsoft Flight Simulator in aerospace education. *American Institute of Aeronautics and Astronautics*, 127:875–883, 1991.
- [4] Thomas R Yechout, Steven L Morris, Thomas Yechout, and Steven Morris. Flight simulation and aeronautical education-Ten years of experience. In *1997 World Aviation Congress*, page 5635, 1997.
- [5] MG Kellett, A Robinson, J Bauer, and N Baker. Real-Time Flight Simulation for Graduate Aerospace Education. In *AIAA Conference Proceeding*, 1998.
- [6] Roger Beck and Wayne C Durham. Flight simulation in flight dynamics education. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, pages 9–11.
- [7] Thomas R Yechout. A MATLAB-Based Flight Simulator for Flight Mechanics Education. 2002.
- [8] Mario G Perhinschi and Marcello R Napolitano. Using flight simulation for flight dynamics and control education enhancement. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, page 6389, 2007.
- [9] Jürgen Gotschlich, Torsten Gerlach, and Umut Durak. 2Simulate: A distributed real-time simulation framework. *ASIM Mitteilung 149/AR-GESIM Report 42*, 2014.
- [10] Fokker Flugzeugwerke. The Flight Training Manual of VFW 614. <http://www.flight-manuals-online.com/product/vfw614/>, 2017.

Prädiktion des Reglerverhaltens in industriellen Zuluftanlagen im Lackierbetrieb durch maschinelle Lernverfahren

P. Oetinger¹, S. Alt¹, T. Schlotterer¹, M. Weickgenannt¹

¹Dürr Systems AG

Philipp.Oetinger@durr.com

Für die Automatisierung des Lackierprozesses in industriellen Lackieranlagen werden Steuerungs- und Regelungssysteme eingesetzt, um die relevanten Prozessparameter einzustellen und zu halten. Die Temperatur und Feuchte des Luftstroms in der Lackierkabine, die durch Belüftungsanlagen konditioniert werden, sind hierbei für die Lackqualität besonders relevante Prozessparameter. Eine besondere Herausforderung für die Regelung der Luftkonditionen ist das Anfahren neuer Arbeitspunkte bei einer Änderung der Prozessbedingungen. Für die Wirtschaftlichkeit des Lackierprozesses spielt hierbei die Dauer des Anfahrvorgangs eine wichtige Rolle, die sowohl vom Führungsverhalten der Regelung als auch von den klimatischen Bedingungen der eingezogenen Außenluft abhängig ist. In diesem Beitrag werden verschiedene Methoden des maschinellen Lernens auf der Basis der transienten Trajektorien der Prozessgrößen bei Anfahrvorgängen betrachtet. Dabei wird die komplette Vorgehensweise von der Datenaufbereitung über die Merkmalsextraktion bis zur geeigneten Modellwahl erläutert. Für die Merkmalsextraktion werden sowohl regelungstechnische Kenngrößen zur Charakterisierung des Führungsverhaltens als auch klassische Kenngrößen wie die Hauptkomponenten der Prozessdaten verwendet und gegenübergestellt. Diese werden durch statistische Testverfahren auf Signifikanz überprüft. Zum Abschluss werden die aktuellen Ergebnisse mit Realdaten validiert und bewertet.

1 Einführung

Um die relevanten Prozessparameter für die Automatisierung des Lackierprozesses in industriellen Lackieranlagen einzustellen und zu halten, werden Steuerungs- und Regelungssysteme verwendet. Zuluftanlagen, wie beispielhaft in Abbildung 1 abgebildet, werden hierbei eingesetzt, um konditionierte Luftströme zur Verfügung zu stellen. Dabei werden die Anforderungen an die Luftkonditionen durch den Lackierprozess bestimmt und betreffen neben dem Volumenstrom insbesondere die Temperatur und relative Feuchte.

Das Einstellen und Halten von definierten Sollwerten für die Luftkonditionen ist üblicherweise Aufgabe eines Regelungssystems. Dieses verändert die eingeogene und durch klimatische Wetterverhältnisse beeinflusste Luft in gewünschter Weise. Für das Regelsystem ist insbesondere das Anfahren der Sollwerte nach Produktionspausen und -störungen eine Herausforderung.

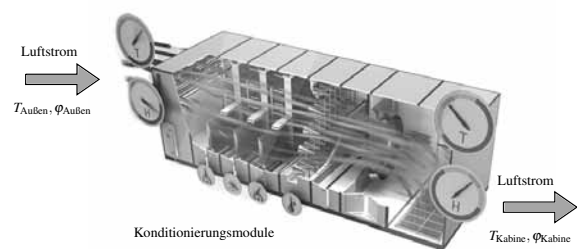


Abbildung 1: Darstellung einer industriellen Zuluftanlage mit den Konditionierungsmodulen Vorheizler, Kühler, Nachheizler und Befeuchter.

Da die Produktion erst gestartet werden kann, wenn die Prozessparameter der konditionierten Luft die Prozessgrenzen konstant einhalten, ist es für eine Optimierung der Produktionsplanung notwendig, die Dauer des Anfahrvorgangs und somit den möglichen Zeitpunkt des Produktionsstarts frühzeitig zu kennen.

Während des Betriebs der Zuluftanlage werden Prozesswerte kontinuierlich aufgezeichnet und in einer Datenbank gespeichert. Diese Daten sind die Basis für die datengetriebenen Modellierungsmethoden durch maschinelle Lernverfahren. In diesem Beitrag wird ein Modell vorgestellt, das den Zusammenhang zwischen den gemessenen Prozessparametern und der Dauer des Anfahrvorgangs herstellt. Hierfür werden Methoden des maschinellen Lernens verwendet, die aus gespeicherten Prozesswerten ein Modell ermitteln.

Im Folgenden wird zunächst ein Überblick über das betrachtete System der industriellen Zuluftanlage gegeben. Auf der Grundlage der Systembeschreibung werden anschließend die genaue Problemstellung und die daraus resultierenden Arbeitsschritte für die Modellbildung zur Prädiktion der Anfahrdauer abgeleitet. Abschließend werden die Inhalte des Beitrags zusammengefasst.

2 Systembeschreibung

Zuluftanlagen, wie in Abbildung 2 schematisch dargestellt, werden in Lackierereien dazu verwendet, um Prozessluft in definierte Prozessgrenzen bezüglich der Temperatur T_{Kabine} und der Feuchte φ_{Kabine} zu konditionieren. Die Konditionierung erfolgt dabei durch unterschiedliche Konditionierungsmodule, die den eingezogenen Luftstrom bezüglich der Temperatur und Feuchte gezielt verändern.

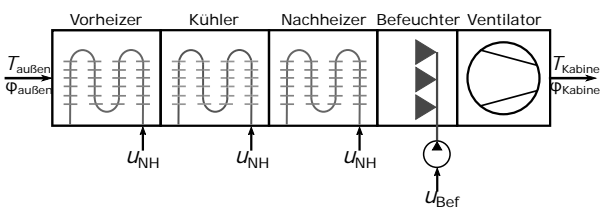


Abbildung 2: Schema einer Zuluftanlage mit Außenbedingungen $T_{\text{Außen}}$ und $\varphi_{\text{Außen}}$, Kabinenbedingungen T_{Kabine} und φ_{Kabine} und Stellgrößen u_{VH} , u_{K} , u_{NH} und u_{Bef} .

Die in Abbildung 2 dargestellte Zuluftanlage besteht aus einem Vorheizger, einem Kühlaggregat, einem Nachheizger und einem Befeuchter. Diese Modulkonfiguration erlaubt es, unabhängig von den gegebenen Außenbedingungen $T_{\text{Außen}}$ und $\varphi_{\text{Außen}}$, vom Prozess vorgegebene Sollwerte T_{soll} , φ_{soll} durch eine Regelung anzufahren und zu halten.

Aus regelungstechnischer Sicht handelt es sich bei dem System um ein nicht-quadratisches Multiple-Input Multiple-Output System [4]. Die Stellgrößen u_{VH} , u_{K} , u_{NH} und u_{Bef} der Module werden durch die Ventilstellungen der Heiz- und Kühlmodule und die Pumpenfrequenz des Befeuchters definiert. Die Luftkonditionen T_{Kabine} und φ_{Kabine} sind gemessene Regelgrößen, die durch die Stellgrößen gezielt beeinflusst werden. Die Außenbedingungen $T_{\text{Außen}}$ und $\varphi_{\text{Außen}}$ des Luftstroms am Eintritt in die Zuluftanlage sind gemessene Störgrößen.

Bei der Regelung zur Einstellung der Kabinenkonditionen T_{Kabine} und φ_{Kabine} werden unterschiedliche Regelungsstrategien, wie zum Beispiel PID-Regler [5], [6] oder modellbasierte Regler [1] eingesetzt. Bei der Realisierung eines Anfahrvorgangs verwendet die Regelung in Abhängigkeit der klimatischen Außenbedingungen unterschiedliche Module. Dies hat eine starke Arbeitspunktabhängigkeit des Führungsverhaltens der Anlage zufolge und erschwert die betrachtete Problemstellung zur Prädiktion der Anfahrdauer, die nachfolgend erläutert wird.

3 Problemstellung

Im Beitrag wird eine Prädiktion des Reglerverhaltens beim Anfahren der Anlage durch maschinelle Lernverfahren untersucht. Hierbei werden auf der Basis persistierender Messdaten Modelle entwickelt, die einen Zusammenhang zwischen den Trajektorien gemessener Sensorsignale und der Anfahrdauer herstellen.

Ziel der Modelle ist es, schon vor Erreichen der konstanten Sollwerte T_{soll} , φ_{soll} die Anfahrdauer $\Delta t = t_{\text{Ende}} - t_0$ zu präzisieren. Die Zeitpunkte t_0 und t_{Ende} beschreiben den Beginn- und Endzeitpunkt des Anfahrvorgangs. Durch die Prädiktion der Anfahrzeit Δt kann somit der Zeitpunkt t_{Ende} zur Produktionsfreigabe präzisiert werden. Ein beispielhafter Anfahrvorgang mit den relevanten Zeitpunkten für die Prädiktion ist in Abbildung 3 dargestellt.

Für die Prädiktion werden neben den Regelgrößen T_{Kabine} und φ_{Kabine} auch die Stellgrößen u_{VH} , u_{K} , u_{NH} und u_{Bef} , sowie die die Störgrößen $T_{\text{Außen}}$ und $\varphi_{\text{Außen}}$ des betrachteten Systems verwendet. Für die Prädiktion der Anfahrzeit Δt werden die Trajektorien der Sensorsignale vor und während des Anfahrvorgangs betrachtet. Der Zeitpunkt t_p , an welchem die Prädiktion berechnet wird, ist ein Entwurfsparameter für das Modell.

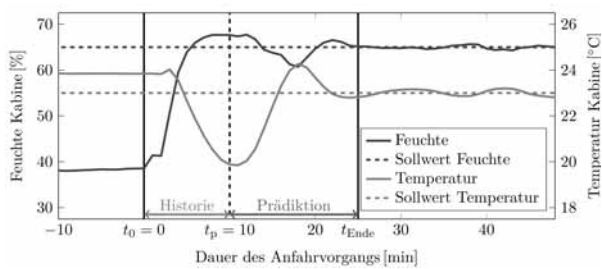


Abbildung 3: Beispielhafter Anfahrvorgang mit dem Startzeitpunkt t_0 , dem Prädiktionszeitpunkt t_p und dem Endzeitpunkt t_{Ende} .

Besondere Herausforderungen für die Prädiktion des Anfahrverhaltens stellen insbesondere der Einfluss der zeitlich veränderlichen Außenbedingungen, sowie das komplexe nichtlineare Systemverhalten der Anlage dar, was eine starke Arbeitspunktabhängigkeit des Anfahrverhaltens nach sich zieht und bei der Modellbildung zu berücksichtigen ist. Für die Modellierung werden sowohl Klassifikations- als auch Regressionsmodelle verwendet. Die Arbeitsschritte zur Modellbildung werden nachfolgend erläutert.

4 Modellbildung

In Abbildung 4 sind die notwendigen Schritte zur datengetriebenen Modellbildung schematisch in einem Flussdiagramm dargestellt, siehe zum Beispiel [2].

Als Grundlage der datengetriebenen Modellbildung dienen persistierte Prozessdaten (A), die im Allgemeinen durch eine Datenbank bereitgestellt werden. Für die betrachtete Aufgabenstellung zur Prädiktion der Anfahrdauer ist es notwendig, dass die zugrunde liegenden Daten möglichst viele Anfahrvorgänge beinhalten, die das Anfahrverhalten in Abhängigkeit unterschiedlicher Arbeitspunkte charakterisieren.

Anschließend werden die Daten in Hinblick auf die betrachtete Aufgabenstellung aufbereitet und bereinigt (B). Für die Prädiktion des Anfahrverhaltens bedeutet dies, dass aus den persistierten Daten alle Anfahrvorgänge herausgefiltert werden. Dadurch liegt zum jeweiligen Startvorgang und den dazu gehörigen Sensoren eine Menge von Zeitreihen vor, die einen Anfahrvorgang beinhalten. Ein weiterer Verarbeitungsschritt umfasst das Labeling der Anfahrvorgänge, was bedeutet, dass die Dauer des Anfahrvorgangs berechnet und dem entsprechenden Datensatz zugeordnet wird.

Als wesentlicher Schritt der Modellierung werden Merkmale, die das dynamische Verhalten des Anfahrvorgangs erfassen, aus den Zeitreihen extrahiert (C) und selektiert (D). Für die Merkmalsextraktion werden sowohl regelungstechnische Kenngrößen zur Charakterisierung des Führungsverhaltens als auch klassische Kenngrößen aus der Statistik [3], wie die Hauptkomponenten der Prozessdaten, verwendet und im Beitrag gegenüber gestellt. Für die Merkmalsselektion werden diese durch statistische Testverfahren auf Signifikanz überprüft.

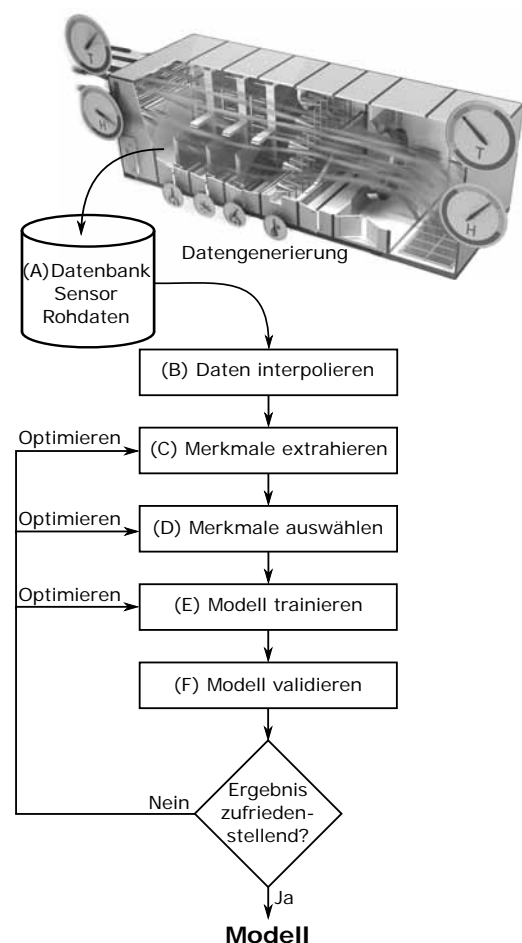


Abbildung 4: Flussdiagramm zur Darstellung der Arbeitsschritte zur datengetriebenen Modellbildung.

Abschließend werden die selektierten Merkmale als Grundlage für das Modelltraining (E) und die Validierung (F) verwendet. Für das Modelltraining kommen sowohl Klassifikations- als auch Regressionsmodelle in Frage. Im Beitrag werden unterschiedliche Modelle vorgestellt und in Hinblick auf ihre Prädiktionsgüte am Beispiel einer industriellen Zuluftanlage evaluiert.

5 Zusammenfassung

Im Beitrag wird ein Verfahren zur Prädiktion der Anfahrdauer von industriellen Zuluftanlagen vorgestellt. Das Verfahren basiert auf Methoden des maschinellen Lernens, durch welche Prädiktionsmodelle auf der Basis historischer Messdaten trainiert werden können. Die Kerninhalte des Beitrags lassen sich dabei wie folgt zusammenfassen:

- Die Funktionsweise und die relevanten Messdaten von industriellen Zuluftanlagen werden erläutert.
- Die Problemstellung zur Prädiktion der Anfahrdauer für die beschriebenen industriellen Zuluftanlagen wird detailliert beschrieben.
- Die Besonderheiten der datengetriebenen Modellbildung für die gegebene Problemstellung werden erklärt. Hierbei liegt der Fokus auf der Merkmalsextraktion und -selektion, in welcher sowohl klassische als auch regelungstechnische Kenngrößen verwendet und evaluiert werden.

Zusätzlich wird die Leistungsfähigkeit der Methodik durch die Vorstellung experimenteller Ergebnisse am Beispiel einer industriellen Zuluftanlage gezeigt.

Literatur

- [1] S. Alt und O. Sawodny, Model-based temperature and humidity control of paint booth HVAC systems, 2015 IEEE International Conference on Mechatronics (ICM), Nagoya, 2015, pp. 160-165.
- [2] T. Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Verlag, 2. Auflage, 2009.
- [3] G. James, D. Witten, T. Hastie und R. Tibshirani, An Introduction to Statistical Learning, Springer Verlag, 2013.
- [4] J. Lunze: Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen. Berlin Heidelberg: Springer Verlag, 2014.
- [5] E. Sadeghipour, E. R. Westervelt and S. Bhattacharya, Painting green: Design and analysis of an environmentally and energetically conscious paint booth HVAC control system, 2008 American Control Conference, Seattle, WA, 2008, pp. 3325-3330.
- [6] C. Underwood, HVAC Control Systems: Modeling Analysis and Design in Taylor and Francis, 1999.

Modellierung und Simulation hochviskoser, nicht-isothermer Fluide mit freier Oberfläche

Dimitri Harder¹, Edmond Skeli¹, Dirk Weidemann¹, Klaus Panreck¹

¹Institut für Systemdynamik und Mechatronik
FH Bielefeld

dimitri.harder@fh-bielefeld.de

edmond.skeli@fh-bielefeld.de

dirk.weidemann@fh-bielefeld.de

klaus.panreck@fh-bielefeld.de

Mit dem Ziel effiziente Regelungs- und/oder Diagnoseverfahren nutzen zu können, greift man auch in der verfahrenstechnischen Industrie vermehrt auf mathematische Modelle, die die zugrundeliegenden physikalischen Prozesse hinreichend genau beschreiben, zurück. Vor diesem Hintergrund wird im vorliegenden Beitrag auf die Modellierung und Simulation des Verhaltens einer nicht-isothermen, hochviskosen Fluidströmung eingegangen. Das Verhalten des Fluids genügt einem System partieller Differentialgleichungen, welches neben den Navier-Stokes-Gleichungen auch die Poisson-Gleichung für den Druck und die Energiegleichung umfasst. Hinsichtlich der numerischen Berechnung der einzelnen Prozessgrößen wird eine Kombination des Marker und Cell (MAC) Verfahrens mit dem Projektionsverfahren nach Chorin und einer Temperaturberechnung auf einem krummlinigen Gitter vorgestellt. Das MAC-Verfahren dient zur Identifikation der freien Oberfläche, indem masselose Partikel über das initialisierte Fluidgebiet eingefügt und mit den berechneten Geschwindigkeiten bewegt werden. Über das Projektionsverfahren nach Chorin werden darüber hinaus die Druck- und Geschwindigkeitsverläufe ermittelt. Charakteristisch für die typische Nutzung des MAC-Verfahrens ist, dass die partiellen Differentialgleichungen örtlich auf einem rechteckigen Gitter diskretisiert werden. Dies führt beim betrachteten Fall jedoch dazu, dass die Grenze zwischen Fluid und Hindernis im Inneren des Gitters verläuft. Im vorliegenden Modell wird demgegenüber ein krummliniges Gitter genutzt. Wesentlicher Vorteil dessen ist, dass die äußeren Gitterknoten direkt auf den umliegenden Hindernisse und Grenzen liegen, was auf ein reduziertes System von gewöhnlichen Differentialgleichungen führt.

1 Einleitung

Gegenstand des vorliegenden Beitrags ist die Modellierung und Simulation des Verhaltens eines nicht-isothermen, hochviskosen Fluids, welches zwischen zwei gegensinnig rotierenden Zylindern einläuft. Hierbei wird davon ausgegangen, dass sich im Raum zwischen den Zylindern zu Beginn kein Fluid befindet, sondern dieser Raum sich erst beim Einlaufen mit dem Fluid füllt. Das Einlaufen des Fluids stellt somit ein instationäres Prozessverhalten dar, bei dem sich die Fluidgrenze örtlich über der Zeit solange verändert, bis sich der stationäre Zustand einstellt. Hin-

sichtlich der Bestimmung des Fluidverhaltens ist insbesondere die Kenntnis über die zeitliche Veränderung des Verlaufs der Fluidgrenze notwendig. Vor diesem Hintergrund wird die Fluidgrenze, d.h. die an die umgebende Luft angrenzende Oberfläche des Fluids, die im Weiteren als freie Oberfläche bezeichnet wird, mit Hilfe der Marker und Cell (MAC) Methode ermittelt, vgl. [3]. Typischerweise werden bei Nutzung der MAC-Methode die zugrunde liegenden partiellen Differentialgleichungen örtlich auf einem rechteckigen Gitter diskretisiert. Demgegenüber findet im vorliegenden Fall ein krummliniges Gitter Verwendung. Der wesentliche Vorteil des krummlinigen Gitters besteht darin, dass die äußeren Gitterknoten sich auf

der Oberfläche der Zylinder befinden, während beim rechteckigen Gitter die äußeren Knoten innerhalb der Zylinder liegen. Somit weist das resultierende System von gewöhnlichen Differentialgleichungen bei Nutzung eines krummlinigen Gitter eine geringere Dimension als beim rechteckigen Gitter auf. Neben den Größen Druck und Geschwindigkeit wird ferner auch die Temperatur des Fluids bestimmt.

2 Modellierung

Die Dynamik des Fluids genügt den Navier-Stokes-Gleichungen, wobei zusätzlich die thermische Energiegleichung das Temperaturverhalten beschreibt. Da im vorliegenden Fall das Verhalten des als inkompressibel angenommenen Fluids nur für den zweidimensionalen Fall betrachtet wird und darüber hinaus der Einfluss der Gravitation aufgrund der hohen Viskosität vernachlässigbar ist, lässt sich die Bewegung des Fluids mittels

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\eta}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (1)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\eta}{\rho} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right), \quad (2)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3)$$

beschreiben. Hierbei repräsentiert u die Geschwindigkeit in x -Richtung, v die Geschwindigkeit in y -Richtung, p den Druck, ρ die Dichte und η die dynamische Viskosität.

Des Weiteren genügt die zeit- und örtliche Entwicklung der Temperatur T der partiellen DGL

$$\begin{aligned} \rho C_p \left(\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} \right) = \\ \lambda \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \\ + 2\eta \left(\frac{\partial u}{\partial x} \right)^2 + \eta \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + 2\eta \left(\frac{\partial v}{\partial y} \right)^2, \end{aligned} \quad (4)$$

wobei C_p die spezifische Wärmekapazität und λ die Wärmeleitfähigkeit ist.

3 Ortsdiskretisierung

Eine bekannte Vorgehensweise zur Bestimmung des örtlichen Verlaufs einer freien Oberfläche bei Fluiden stellt die MAC-Methode dar, vgl. [1, 5] und die Ausführungen in Abschnitt 5. Um die MAC-Methode nutzen zu können, ist die Berechnung der Geschwindigkeiten des Fluids und somit die Lösung der zugrunde liegenden Navier-Stokes-Gleichungen erforderlich. Hinsichtlich einer numerischen Lösung der Navier-Stokes-Gleichungen bedarf es der Überführungen der partiellen Differentialgleichungen in ein System von gewöhnlichen Differentialgleichungen, was mit einer geeigneten Ortsdiskretisierung gelingt. In der Regel findet eine Ortsdiskretisierung mit konstanten Schrittweiten je Raumrichtung Verwendung, wobei ein rechteckiges Gitter entsteht. Dabei werden die relevanten Prozessgrößen, die neben den für den MAC-Algorithmus benötigten Geschwindigkeiten auch den Druck und die Temperatur umfassen, örtlich von einander versetzt (staggered Grid) berechnet. Die Komponenten der Geschwindigkeit werden entsprechend der Darstellung in Fig. 1 in der Mitte der Zellenkanten und der Druck und die Temperatur jeweils in der Mitte der Zellen berechnet. Eine Zelle ist somit durch die Kanten, an denen sich die Geschwindigkeiten berechnen definiert.

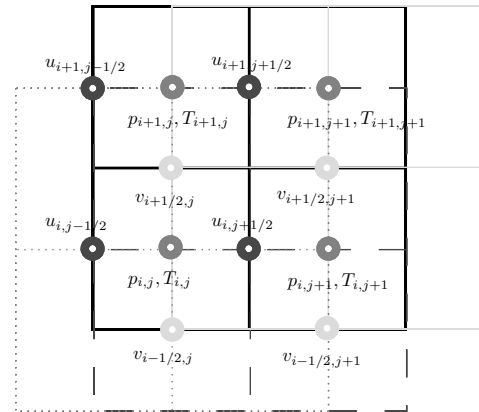


Fig. 1: Gitter mit den versetzten Berechnungsknoten

Fig. 2 ist zu entnehmen, dass bei den hier vorliegenden geometrischen Grenzen (Zylinder) die Nutzung eines rechteckigen Gitters dazu führt, dass sich einzelne Gitterpunkte außerhalb des Fluidgebiets befinden. Vor diesem Hintergrund wird im vorliegenden

Fall ein krummliniges Gitter verwendet, bei dem die Schrittweite Δx konstant ist, während Δy entlang der Strömungsrichtung variiert wird, s. Fig. 3. Wie bereits beim rechteckigen Gitter befinden sich auch beim krummlinigen Gitter die Berechnungspunkte für die Geschwindigkeiten auf der Mitte der Zellenkante und die Punkte für Druck und Temperatur in der Zellmitte. Durch die Verwendung eines krummlinigen Gitters reduziert sich die Anzahl der gewöhnlichen Differentialgleichungen, die aus der Ortsdiskretisierung hervorgehen, gegenüber dem Fall beim rechteckigen Gitter. Der wesentliche Vorteil eines krummlinigen Gitters besteht daher in einem geringeren Berechnungsaufwand. Allerdings ist eine sinnvolle Nutzung des krummlinigen Gitters nur bei hinreichend großen Radien möglich, da bei kleinen Radien das Risiko einer ungenaueren Berechnung besteht.

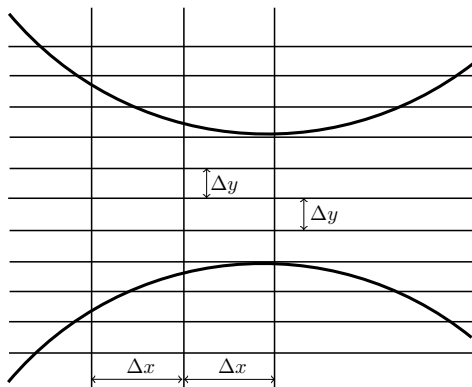


Fig. 2: Rechteckiges äquidistantes Gitter

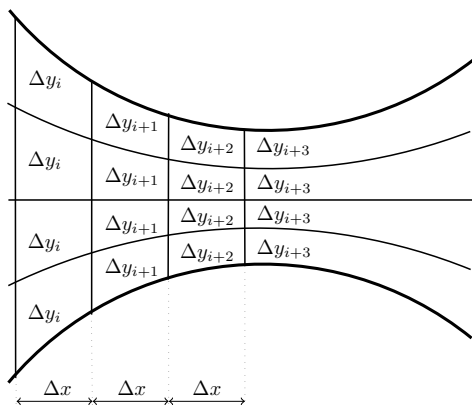


Fig. 3: Krummliniges Gitter

4 Lösungsvorgehen

Um die Navier-Stokes-Gleichungen zu berechnen, wird das Projektionsverfahren nach Chorin [2] angewandt. Das Verfahren von Chorin berechnet im ersten Schritt Pseudo-Geschwindigkeiten, indem der Term des Druckgradienten in den Navier-Stokes-Gleichungen zu null gesetzt wird. Anhand der in (7) gegebenen Poisson-Gleichung wird im zweiten Schritt der Druck berechnet und anschließend die zuvor berechneten Geschwindigkeiten korrigiert. Das Vorgehen gestaltet sich wie folgt:

1. Berechne die Pseudo-Geschwindigkeiten \tilde{u} und \tilde{v} aus den Gleichungen

$$\frac{\partial \tilde{u}}{\partial t} = -u \frac{\partial u}{\partial x} - v \frac{\partial u}{\partial y} + \frac{\eta}{\rho} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (5)$$

$$\frac{\partial \tilde{v}}{\partial t} = -u \frac{\partial v}{\partial x} - v \frac{\partial v}{\partial y} + \frac{\eta}{\rho} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right). \quad (6)$$

2. Berechne den Druck mit Hilfe der Poisson-Gleichung

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = \rho \left(\frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{v}}{\partial y} \right). \quad (7)$$

3. Korrigiere die Geschwindigkeiten indem die Druckgradienten entsprechend

$$u = \tilde{u} - \frac{1}{\rho} \frac{\partial p}{\partial x}, \quad (8)$$

$$v = \tilde{v} - \frac{1}{\rho} \frac{\partial p}{\partial y} \quad (9)$$

von den zuvor berechneten Pseudo-Geschwindigkeiten subtrahiert werden.

Die Temperatur berechnet sich aus (4) mit den korrigierten Geschwindigkeitswerten. Anschließend folgt die Bestimmung der freien Oberfläche.

5 Bestimmung der freien Oberfläche

Die hier verwendete MAC-Methode findet sich in den Aufsätzen [1, 5] und dient der Bestimmung der freien Oberfläche. Hierbei werden masselose Partikel zur

Markierung derjenigen Zellen genutzt, die vom Fluid teilweise oder vollständig gefüllt sind. Mit anderen Worten, jede Zelle, in der sich mindestens ein masseloses Partikel befindet, ist Teil des Gebietes in dem sich das Fluid befindet. Vor diesem Hintergrund bezeichnet man die masselosen Partikel auch als Marker. Grenzen nicht markierte, d.h. leere Zellen an eine Fluidzelle, dann ist diese Fluidzelle diejenige in der die Oberfläche verläuft. Solche Konstellationen sind in Fig. 4 und 5 dargestellt.

An einer freien Oberfläche für inkompressible Fluide gilt, dass Normal- und Tangentialspannungen identisch null sind, vgl. [4]. Für eine zweidimensionale Fläche gelten somit die beiden Bedingungen

$$\frac{p}{\rho} - 2\frac{\eta}{\rho} \left[n_x n_x \frac{\partial u}{\partial x} + n_x n_y \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + n_y n_y \frac{\partial v}{\partial y} \right] = 0, \quad (10)$$

$$\left[2n_x m_x \frac{\partial u}{\partial x} + (n_x m_y + n_y m_x) \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + 2n_y m_y \frac{\partial v}{\partial y} \right] = 0, \quad (11)$$

wobei $\mathbf{n} = (n_x, n_y)$ der nach außen gerichtete Normalen- und $\mathbf{m} = (m_x, m_y) = (n_y, -n_x)$ der Tangentialvektor ist. Die freien Oberflächenbedingungen

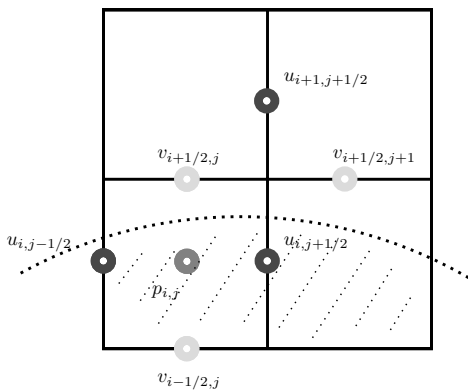


Fig. 4: Markierte Zelle mit einer freien angrenzenden Zelle

werden anhand der angrenzenden freien Zellen bestimmt, was im Folgenden an Hand von zwei Beispielen erläutert wird. In Fig. 4 hat die untere linke Zelle eine Seite, die an eine freie Zelle grenzt. In diesem Beispiel ist somit die Normalkomponente n_x sehr

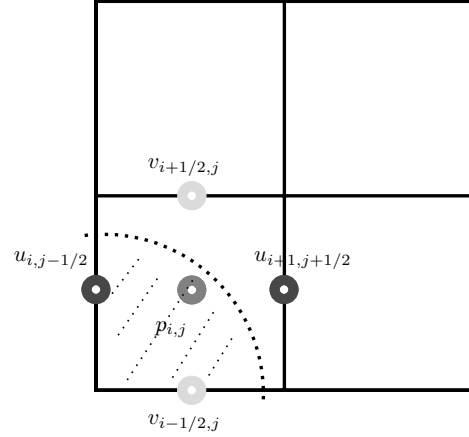


Fig. 5: Markierte Zelle mit zwei freien angrenzenden Zellen

klein und (10) sowie (11) vereinfachen sich zu

$$p - 2\eta \left(\frac{\partial v}{\partial y} \right) = 0, \quad (12)$$

und

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = 0. \quad (13)$$

Der Druck $p_{i,j}$ für die Zelle in Fig. 4 wird aus (12) und die Geschwindigkeit $u_{i+1,j+1/2}$ aus (13) berechnet, wodurch die freie Oberflächenbedingungen erfüllt werden. Andere Szenarien, bei denen die markierte Zelle ebenfalls nur an einer Seite an eine leere Zelle angrenzt, werden analog vereinfacht. In Fig. 5 grenzen zwei leere Zellen an die markierte Zelle, in einem derartigen Szenario wird angenommen, dass der Normalenvektor in einem Winkel von 45° zu den offenen Seiten zeigt. In diesem Fall vereinfachen sich (10) und (11) zu

$$p - \eta \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = 0 \quad (14)$$

und

$$\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} = 0. \quad (15)$$

In Fig. 5 wird der Druck nach (14) bestimmt und die Geschwindigkeiten $u_{i+1,j+1/2}$ und $v_{i+1/2,j}$ werden den gegenüberliegenden Geschwindigkeiten der Fluidzellen gleich gesetzt, sodass (15) erfüllt wird. Weitere Szenarien, die sich einstellen können, finden sich bspw. in [1, 5].

6 Ergebnisse

Obgleich die numerischen Ergebnisse den örtlichen und zeitlichen Verlauf der Geschwindigkeiten, Drücke und Temperaturen des Fluids umfassen, werden im Weiteren nur einzelne Temperaturprofile des Fluids entlang der Stömungsrichtung von drei unterschiedlichen Schichten zu jeweils vier verschiedenen Zeitpunkten dargestellt. Zum Zeitpunkt t_1 sind die Anfangstemperaturen des betrachteten Fluidgebietes mit einer Temperatur von 425 Kelvin und der Umgebungstemperatur von 400 Kelvin dargestellt. In Fig. 6 liegt das Fluid an einem Zylinder an, der eine konstante Temperatur von 435 Kelvin aufweist. Durch eine Fortbewegung des Fluids in Folge der Zylinderdrehung kommt es zu einer Scherung des Fluids, die in einer Temperaturerhöhung resultiert, sodass das Fluid teilweise Temperaturen annimmt, die höher als die Zylindertemperatur sind. Ab dem Ort an dem die Scherung abnimmt, trägt der Zylinder somit zur Abkühlung des heißeren Fluids bei. Mit dem Anstieg des Druckes durch die Verengung des Raumes zwischen den Zylindern wird das Fluid ferner sowohl beschleunigt als auch verzögert, was zu einem erneuten Anstieg der Temperatur führt. Das Temperaturprofil zum Zeitpunkt t_4 weist in Fig. 6 darüber hinaus eine Verschiebung nach rechts auf. Hieran erkennt man, dass das Fluid den Spalt, der zwischen den Zylindern vorliegt, durchströmt hat.

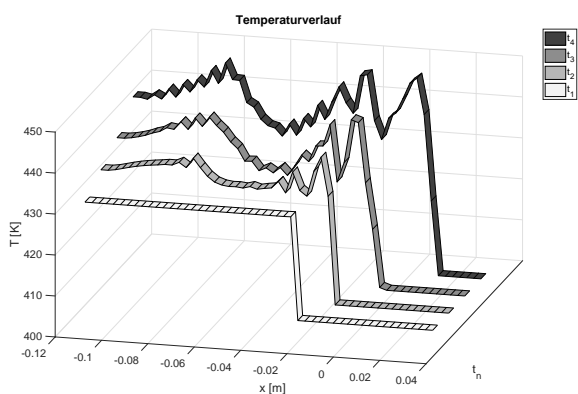


Fig. 6: Temperaturprofile der am oberen Zylinder angrenzenden Schicht

Die Änderung des Temperaturprofils sowohl der mittleren als auch der am unteren Zylinder angrenzenden

Schicht, s. Fig. 7 und 8, basiert auf denselben physikalischen Vorgängen (Scherung etc.), die zur Temperaturveränderung der am oberen Zylinder angrenzenden Schicht geführt haben. Anhand Fig. 8 erkennt man

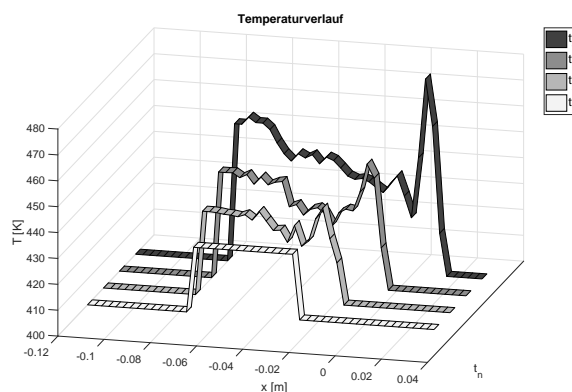


Fig. 7: Temperaturprofile der mittleren Schicht

ferner, dass sich das Fluid in einzelnen Gebieten nicht nur in Richtung des Zylinderspalts bewegt sondern an einzelnen Orten auch ansammelt.

7 Zusammenfassung

Im vorliegenden Beitrag wird ein mathematisches Modell eines nicht-isothermen, hochviskosen Fluids, welches zwischen zwei gegensinnig rotierenden Zylindern einläuft, vorgestellt. Das Modell bietet die Möglichkeit in Kombination mit der MAC-Methode den Verlauf der freien Oberfläche zu ermitteln. Hierfür wird das zugrunde liegende verteiltparametrische Modell hinsichtlich der numerischen Simulation auf einem krummlinigen Gitter diskretisiert und nicht, wie sonst üblich, auf einem rechteckigen Gitter. Da bei einem krummlinigen Gitter der Rand der Zylinder identisch mit den äußeren Linien des Diskretisierungsgitters ist, gelingt es das bei der Ortsdiskretisierung entstehende System von gewöhnlichen Differentialgleichungen gegenüber demjenigen bei Nutzung eines rechteckigen Gitters zu verringern. Allerdings ist eine sinnvolle Nutzung des krummlinigen Gitters nur bei hinreichend großen Radien möglich. Zu kleinen Radien führen aufgrund der gröberen Diskretisierung typischerweise zu ungenaueren Berechnungsergebnissen.

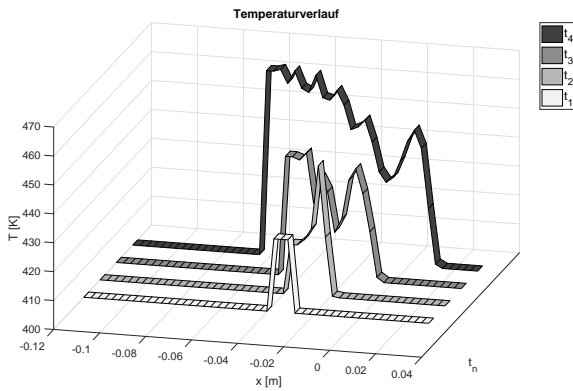


Fig. 8: Temperaturprofile der am unteren Zylinder angrenzenden Schicht

Literatur

- [1] A. A. Amsden and F. H. Harlow. The smac method: A numerical technique for calculation incompressible fluid flows. Technical report, Los Alamos Scientific Laboratory of the University of California, 1970.
- [2] A. J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104):745–762, October 1968.
- [3] F. H. Harlow and J. Eddie Welch. Numerical calculation of time-dependent viscous incompressible-flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182 – 2189, December 1965.
- [4] C. W. Hirt and J. P. Shannon. Free-surface stress conditions for incompressible-flow calculations. *Journal of Computational Physics*, 2:403 – 411, 1968.
- [5] M. F. Tome and S. Mckee. Gensmac: A computational marker and cell method for free surface flows in general domains. *Journal of Computational Physics*, 110:171–186, March 1994.

The RESIN climate change adaptation project and its simple modeling approach for risk-oriented vulnerability assessment

Daniel Lückerath, Manfred Bogen, Erich Rome, Betim Sojeva,

Oliver Ullrich, Rainer Worst, Jingquan Xie

Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS

Schloss Birlinghoven, 53757 Sankt Augustin, Germany

[first.last]@iais.fraunhofer.de

Urban population centers are especially vulnerable to extreme weather events and climate change. Local decision makers demand standardized processes, methods, and tools enabling them to design and implement climate adaptation strategies for their specific contexts. The project “Climate Resilient Cities and Infrastructures – RESIN” aims at providing such a set of methods and tools; it utilizes the impact chain modeling approach to capture and represent cause-effect relationships underlying risks and vulnerabilities in urban population centers, enabling a systematic analysis and evaluation. Highlighting on a number of concepts, such as hazard, exposure, stressors, coping capacity, and vulnerability, these impact chains constitute the base for further quantitative modeling steps.

1 Introduction

High concentrations of residents and economic assets render urban population centers especially vulnerable to the impact of extreme weather events and consequences of climate change (see [1]). Such disasters endanger large numbers of residents and critical infrastructure systems at the same time, thereby also impacting inter-regional and global economic networks (see [2]). The trend towards increasing urbanization in Europe – by 2050 82% of all Europeans are expected to live in urban population centers (see [3]) – and the increasing dependencies of infrastructure components make it necessary for municipalities to develop proactive strategies to increase their resilience against climate-related disasters.

Climate change adaptation measures, in contrast to climate protection efforts that are mainly aimed at reducing greenhouse gas emissions, are designed to reduce the impact of climate change on social and biological systems, such as urban population centers. Unfortunately, not many standardized methods and toolsets exist today that enable municipal decision makers to plan, assess, and implement adaptation measures, thereby helping them to consider, analyze, and evaluate risks and vulnerabilities under specific, climate change related scenarios. In the context of such a systematic scenario analysis, the application of a simulation model can be of considerable benefit. The foundation of such a model—and by extension of effective

adaptation measures – is a comprehensive understanding of the risks and vulnerabilities themselves.

This paper describes characteristics and generation of impact chains representing cause-effect relationships that form risks and vulnerabilities of urban population centers in the context of the ongoing EU project “Climate Resilient Cities and Infrastructures – RESIN” (see [6]). The project is aimed at developing practical and applicable methods and tools to support municipalities in designing and implementing climate adaptation strategies for their local contexts. RESIN also systematically compares and evaluates methods for climate change adaptation in order to move towards a formal standardization of adaptation strategies.

RESIN is one of several interdisciplinary, practice-based research projects investigating climate resilience in European cities. The EU project “Reconciling Adaptation, Mitigation and Sustainable development for citiES – RAMSES” (see [4]) that was concluded recently, developed methods and tools to quantify evidence of the impacts of climate change and the costs and benefits of adaptation measures to cities. The EU project “Smart Mature Resilience – SMR” (see [5]) aims at developing a resilience management guideline to support city decision-makers in developing and implementing resilience measures.

The paper continues with a short introduction to the RESIN project and its fundamental process (see Section 2). It then goes on to describe the impact chain

modeling approach used to capture and represent cause-effect relationships underlying risks and vulnerabilities in urban population centers, enabling a systematic analysis and evaluation (see Section 3). The paper concludes with a short description of the steps necessary to complete the modeling process based on the generated impact chains (see Section 4).

2 The RESIN project

RESIN investigates climate change adaptation practices in European cities in order to develop standardized methods and decision support tools that decision makers can use to develop local adaptation strategies. The project builds on previous research by combining existing approaches to climate change adaptation and disaster risk management while taking into account all of the core elements of the urban system and their interrelations. One of the central aims of RESIN is to provide standardized methods and tools for comprehensive risk-oriented vulnerability assessments of an urban population center. These efforts are based on the latest state of discussion and conceptual approaches of the Intergovernmental Panel on Climate Change (IPCC) Assessment Report 5 (AR5, see [7]). RESIN views a city as a “system of systems” comprised of

complex social, ecological, and technical sub-systems that overlap and interact with one another.

All RESIN methods and tools are being developed by means of co-creation with the cities of Bilbao (Spain), Greater Manchester (United Kingdom), Paris (France), and Bratislava (Slovakia). The first co-creation process with Bilbao, aiming at producing a vulnerability and risk mapping on a neighborhood (barrios) scale, started in July 2016 and concluded in September 2017.

Process, methods, and tools are standardized and can be applied to more European urban population centers, but at the same time support the tailoring to the specific needs of a municipality, depending on the varying degrees of maturity of their adaptation processes.

The application of the RESIN conceptual framework follows four principal stages (see [8]):

1. Assessing climate risks
2. Developing adaptation objectives
3. Prioritizing adaptation options
4. Developing an implementation plan

The impact chain modeling technique described in the following section is part of the modelling process during the first stage of this adaptation planning process.

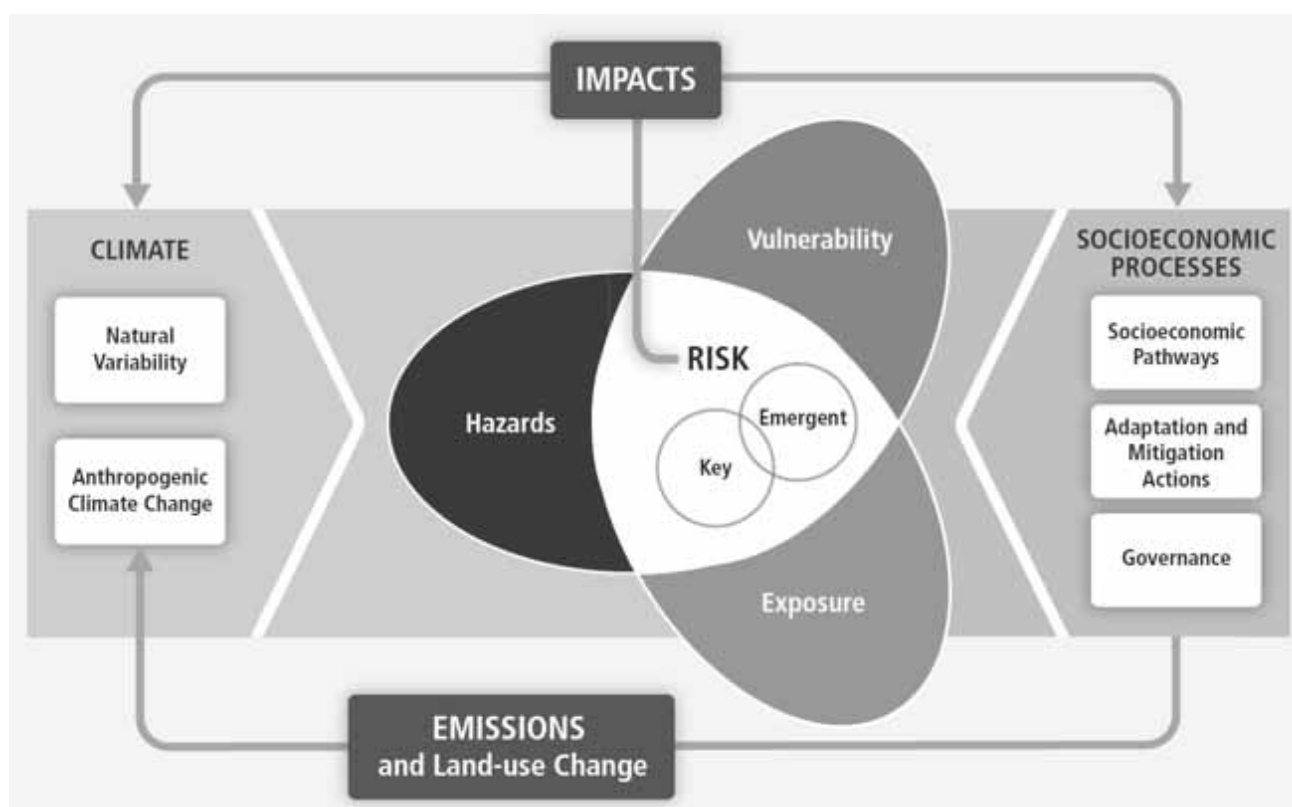


Figure 1. Risks as compositions of hazards, exposure, and vulnerability (source: [7])

In RESIN, the design chosen for the vulnerability assessment process is based on the indicator-based framework provided by “The Vulnerability Sourcebook” (see [9]). Since the sourcebook is based on IPCC AR4 (see [10]), the framework had to be adapted to account for the conceptual move to a risk-based approach by IPCC AR5 (see [7]). Compared to AR4, AR5 defines and uses terms like *vulnerability* and *exposure* in different ways: Where exposure to climate change hazards was once considered to be part of vulnerability—alongside sensitivity to hazards and capacity to adapt—the move to risk has separated out exposure. As a result, risk is now regarded by the IPCC as a function of climate hazard, exposure, and vulnerability (see Figure 1).

3 Using impact chains to model risks and vulnerabilities

Impact chains are tools (described in [9]) for capturing and structuring the components of a particular cause-effect relationship. They describe the basic connections between the elements, and prepare the selection of quantitative indicators for many of the components. Experiences during the co-creation processes with local experts from Bilbao, Bratislava, Manchester, and Paris show that impact chain diagrams are easy to grasp and apply, and that participants appreciate the structured modeling approach. As a result, impact chains are not exhaustive, but describe the common understanding of the stakeholders present at the workshops. Often, experts found that impact chain diagrams

gave them first clues towards potential adaptation measures.

As part of impact chain modeling, RESIN utilizes a number of concepts (see Figure 2) to derive overall risk estimations: drivers, hazard, exposure, stressors, sensitivity, coping capacity, vulnerability, and impacts.

A *hazard* is defined as “...the potential occurrence of a natural or human-induced physical event or trend, or physical impact that may cause loss of life, injury, or other health impacts, as well as damage and loss to property, infrastructure, livelihoods, service provision, and environmental resources” (see [7]). A climate-related hazard is a special case that is (at least partially) caused by climatic *drivers*. Examples include flooding, heatwave, drought, and water scarcity.

Exposure refers to the objects or systems that are exposed: The presence of people, livelihoods, species or ecosystems, environmental services and resources, infrastructure, or economic, social, or cultural assets in specific places that could be adversely affected.

Non-climatic trends and events, which are called *stressors*, can have an important effect on an exposed system. Examples are population growth or change of land-use; a larger percentage of sealed surface will in general increase the susceptibility to flooding events and thus the vulnerability of all exposed objects.

Different objects are more or less sensitive to a hazard. This is captured by the concept of *sensitivity*, defined as the degree to which an exposed object, species or

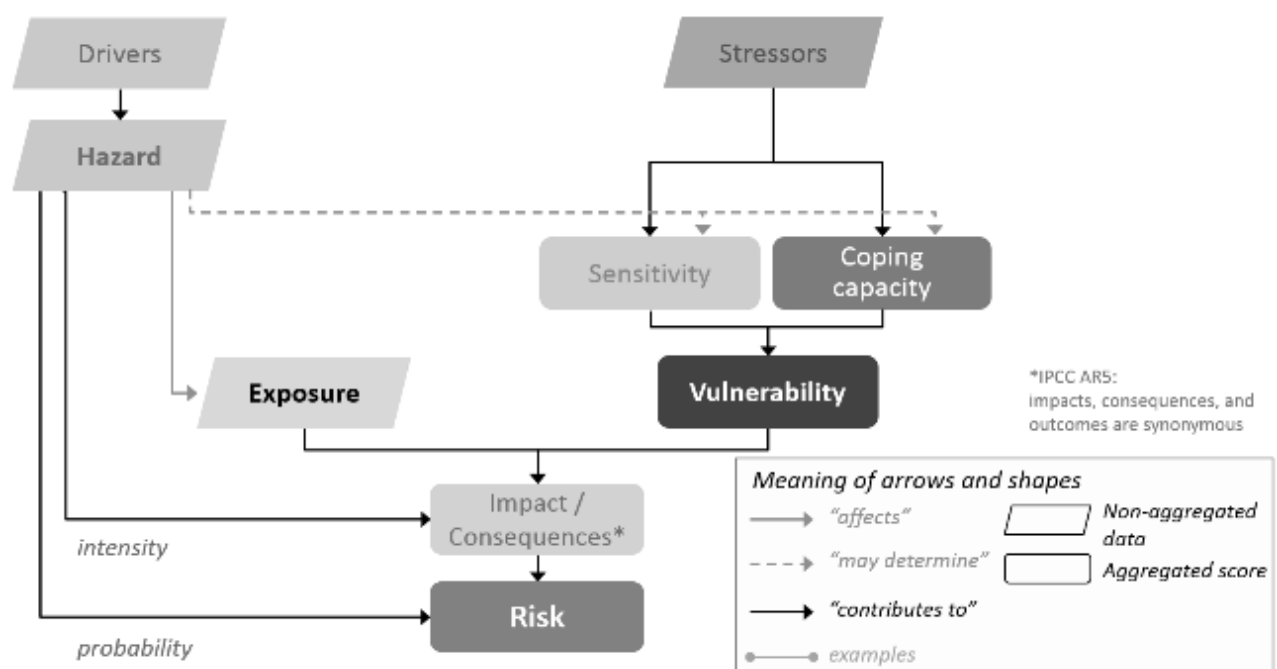


Figure 2. A risk-oriented vulnerability assessment schema (source: [11])

system could be affected by the considered hazard. As such, sensitivity towards a hazard can be perceived as a property of an exposed object in regard to a specific hazard. Examples for sensitivity include the degrees of surface sealing, age and density of a population, household-income, or elevation and density of buildings.

Coping capacity is defined as “the ability of people, institutions, organizations, and systems, using available skills, values, beliefs, resources, and opportunities, to address, manage, and overcome adverse conditions in the short to medium term” (see [12]). Examples include the draining capacity of sewer systems, a dike’s height, education and awareness of the population, and availability of early warning systems, while examples for adaptive capacity include diversity of economic activities, state of the city infrastructure, network redundancy, diversity of land-use, or availability of hospital beds.

Vulnerability is derived from the interplay of stressors, sensitivity, and coping capacity. It contributes directly to the impact or consequences that a hazard causes to the exposed objects.

Risk is classically computed by multiplying the probability of an adverse event with the magnitude of the expected consequences (see [12]). A risk assessment takes into account the characteristics and intensity of the considered hazard, as well as the set of objects exposed to it. The probability of a hazard affecting the set of objects may be estimated from extrapolating historical data or simulation results concerning the fre-

quency of the hazard and the development of the objects. The vulnerability of the objects exposed to the hazard then determines the consequences.

The RESIN vulnerability assessment process starts with a systematic analysis and selection of hazards, drivers, and stressors relevant to the urban area under examination. These results serve as a base for the detailed planning of the assessment and ensures that the – usually limited – resources available for the assessment are spent on the most pressing current and future hazards, and no other threats or possible dependencies between different hazards are overlooked. In addition, a thorough documentation of the rationale for selecting hazards, drivers, and stressors is recommended to ensure that future assessments or re-evaluations yield comparable results.

Once the vulnerability assessment is completed, impact chain diagrams (see Figure 3) are developed. The diagrams then visually illustrate cause-effect relationships between the elements contributing to the impact of a given combination of hazards and exposed objects.

RESIN impact chain diagrams explicitly differentiate between hazards on the one hand and impacts and consequences on the other (see Figure 4) – thus representing cause and effect. In addition, the diagrams model relationships between sensitivity and coping capacity, and exposed object. Each element of an impact chain may be described in a qualitative way by specifying attributes, such as “green infrastructure” for coping capacity, and later also in a quantitative way by assigning

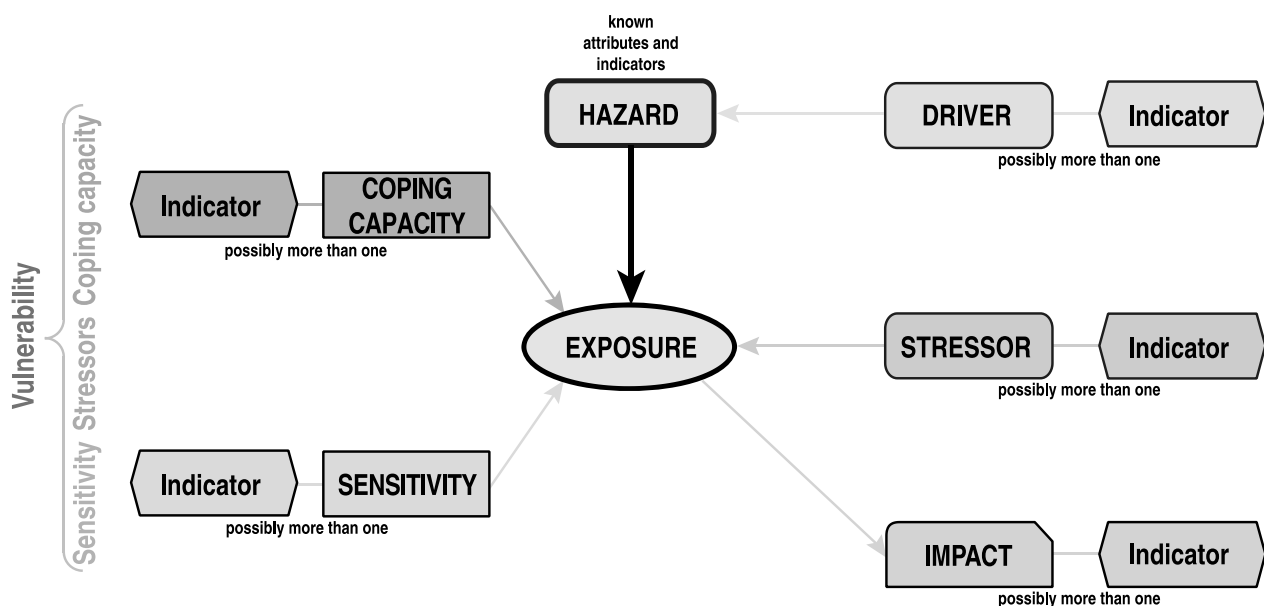


Figure 3. Generic impact chain diagram schema (source: [11])

measurable indicators, such as “percentage of green area per city district”. Usually, impact chain diagrams are developed during collaborative workshops with experts and stakeholders.

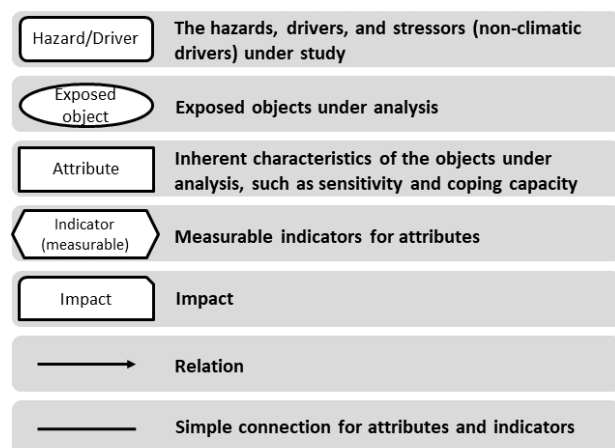


Figure 4. Elements used in impact chain diagrams (source: [11])

Figure 5, developed during a co-creation workshop with local experts, shows an example impact chain for the hazard-exposure combination heatwave on public health for the city of Bilbao. Here, a heatwave is defined as a day when the average temperature exceeds 32°C. Both the coping capacity and the sensitivity indicators contain a mixture of infrastructure related measures as well as social indicators, such as the amount of green infrastructure and the percentage of elderly people. On the right side of the diagram, the impact indicators cover mainly health related and economic consequences. The impact chain development (without the definition of measureable indicators) concludes the qualitative part of the assessment process and is a very valuable outcome itself. End-users without the necessary resources – both in terms of personnel and knowledge – may decide to end the vulnerability assessment at this point. Others may opt to go for the quantitative part outlined below.

4 Further steps

Based on the insights gained during this phase, impact chains are annotated by further identifying measurable indicators for all identified elements, and by gathering necessary data for their calculation. To ease the indicator selection process, established directories of standard indicators can be employed. Such directories can be found, for example, in the annex of the Vulner-

ability Sourcebook (see [9]), the annex of the Covenant of Mayors for Climate and Energy Reporting Guidelines (see [13]) or the indicator database of the European Union Framework Programme 7 project MOVE (Methods for the Improvement of Vulnerability Assessment in Europe, see [14]). These annotated impact chains are then utilized as a base for evaluation and simulation models of impacts, hazards, sensitivity, coping capacity, or stressors – which in turn are applied to move from risk assessments based on historic data to risk assessment of future (simulated) scenarios. Acknowledgments

The authors thank their partners in the RESIN consortium for their valuable contributions during the development and test process. This paper is based in part upon work in the framework of the European project “Climate Resilient Cities and Infrastructures – RESIN”. This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 653522.

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.

5 References

- [1] Coletti, A., De Nicola, A., Villani, M.L. *Building climate change into risk assessment*. In: Natural Hazards, Vol. 84, Issue 2, pp. 1307–1325, 2016.
- [2] Burke, M., Hsiang, S.M., Miguel, E. *Global non-linear effects of temperature on economic production*. In: Nature, Vol. 527, pp. 235–239, 2015.
- [3] United Nations, Department of Economic and Social Affairs, Population Division. *World Urbanization Prospects: The 2014 Revision, Highlights*. United Nations, 2014.
- [4] RAMSES – Reconciling adaptation, mitigation and sustainable development for cities. URL, visited on Dec 05, 2017, <http://www.ramses-cities.eu/>
- [5] SMR – Smart Mature Resilience. URL, visited on Dec 05, 2017. <http://smr-project.eu>

- [6] RESIN – Climate Resilient Cities and Infrastructures. URL, visited on Nov 11, 2017: <http://www.resin-cities.eu/>
- [7] Intergovernmental Panel on Climate Change (IPCC). *Summary for Policymakers*, in: Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part A: Global and Sectoral Aspects. Cambridge University Press, Cambridge, United Kingdom, and New York, NY, USA, pp. 1–32, 2014.
- [8] Carter, J., Connelly, A., Handley, J., Hincks, S., Karvonen, A. *Deliverable D1.3 Conceptual Framework*. EU H2020 Project RESIN, Manchester, UK, 2016.
- [9] German Federal Ministry for Economic Cooperation and Development. *The Vulnerability Sourcebook. Concept and guidelines for standardised vulnerability assessments*. Deutsche Gesellschaft für Internationale Zusammenarbeit, Bonn and Eschborn, Germany, 2014.
- [10] IPCC. *Climate Change 2007: Synthesis Report*. Pachauri, R.K and Reisinger, A. (eds.) IPCC, Geneva, Switzerland, 2007.
- [11] Rome, E., Bogen, M., Worst, R., Lückerath, D., Sojeva, B., Voss, H., Voss N., Xie, J. *Deliverable D2.3 Realisation and implementation IVA-VIA*. EU H2020 Project RESIN, Sankt Augustin, Germany, 2017.
- [12] IPCC. *Annex II: Glossary*, in: Climate Change: Synthesis Report. R.K. Pachauri and L.A. Meyer (eds.) IPCC, Geneva, Switzerland, pp. 117-130, 2014.
- [13] Neves, A., Blondel, L., Brand, K., Hendel Blackford, S., Rivas Calvete, S., Iancu, A., Melica, G., Koffi Lefeivre, B., Zancanella, P., and Kona, A.: *The Covenant of Mayors for Climate and Energy Reporting Guidelines*; EUR 28160 EN; doi:10.2790/586693, 2016.
- [14] MOVE Indicator Database Visualiser of the FP7 project MOVE. URL, visited on Oct 25, 2017: <http://www.gi4dr.org/move/>

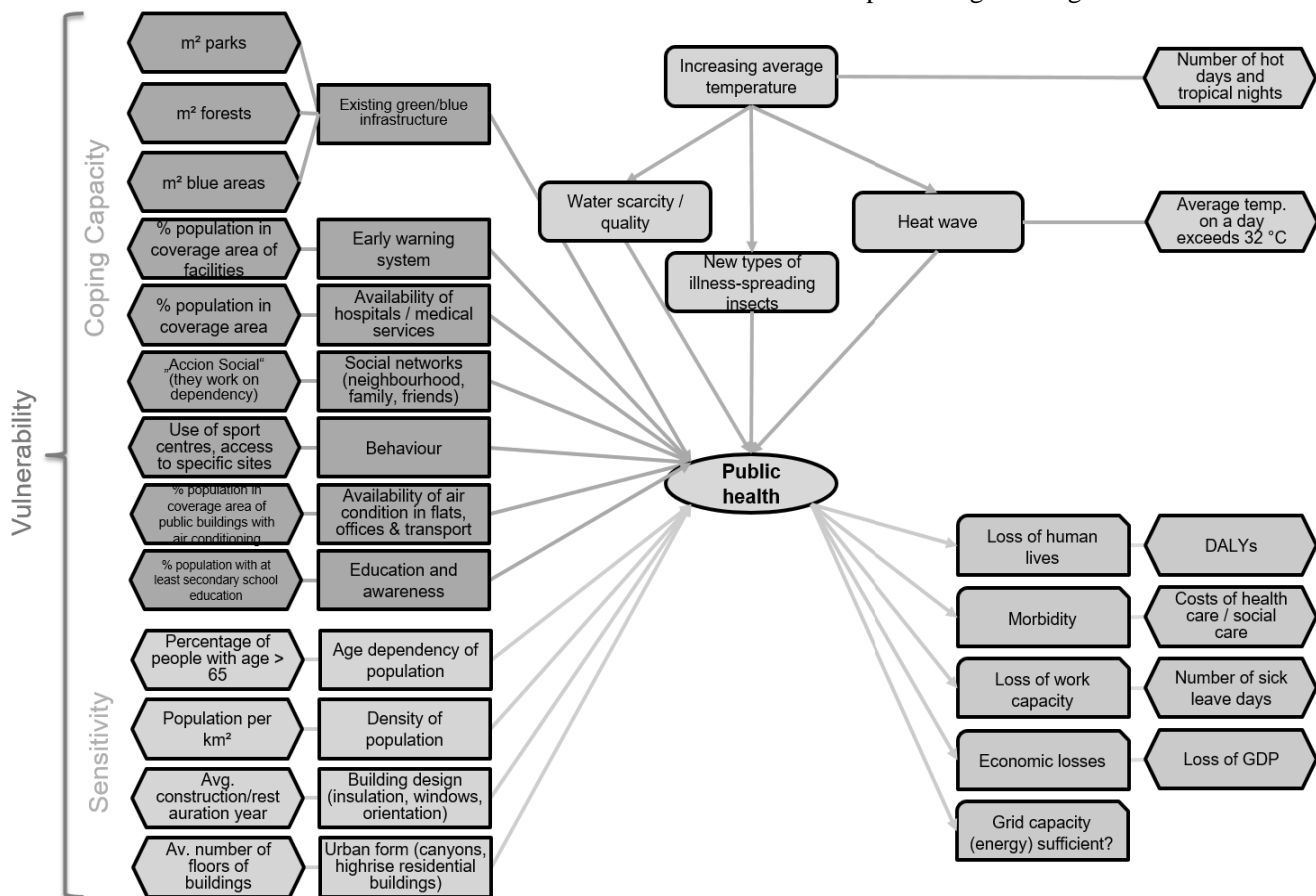


Figure 5. Example impact chain for the hazard-exposure combination heat wave on public health in the city of Bilbao (source: [11])

Integration eines Multi-Agenten-Systems in eine Ereignisdiskrete Materialflusssimulation

Michael Scholz¹, Toni Donhauser¹, Jörg Franke¹

¹Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik (FAPS)

Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)

Egerlandstraße 7-9, D-91058 Erlangen

Michael.Scholz@faps.fau.de

Der vorliegende Beitrag adressiert die Integration eines Multi-Agenten-Systems in eine ereignisdiskrete Materialflusssimulation um die Möglichkeit einer Vorhersagbarkeit des Gesamtsystems, bestehend aus planbaren Einzelentscheidungen der beteiligten Entitäten, auf diskrete Störeinflüsse zu gewährleisten. Aufgrund des zunehmenden Trends zu individualisierten und kundenspezifischen Produkten adressieren aktuelle Forschungsvorhaben im Bereich der intralogistischen Materialflusssysteme hochflexible Produktionsstrukturen, die von den klassischen Typen, des verrichtungsorientierten oder prozessfolgeorientierten Fertigungslayouts, abweichen. Diese Produktionsnetzwerke erzeugen auftragsindividuelle Arbeitspläne, welche anhand der aktuellen Rahmenbedingungen des Fertigungsnetzwerks erstellt werden. Die Auftragsallokation direkt durch den Auftrag selbst stellt daher eine Schlüsselfunktion dar. Diese Aufgabe kann mit Hilfe eines Multi-Agenten-Systems bestehend aus Auftrags-, Maschinen-, und Transportagenten gelöst werden. Im Rahmen des Beitrags wird eine Methode vorgestellt, welche die Allokation von Transportaufgaben auf intralogistische vielseitige Transportfahrzeuge behandelt. Die Lösungsfindung erfolgt durch eine Verhandlung des jeweiligen Auftragsagenten mit den Fahrzeugagenten anhand der im Auftrag hinterlegten Zielvorgaben und der auftragsindividuellen Logistik-KPIs der Fahrzeuge. Um eine Optimierung und Vorhersagbarkeit des Allokationsalgorithmus bei diskreten Ereignissen zu ermöglichen ist eine Integration des Multi-Agenten-Systems in eine Materialflusssimulation notwendig. Hierzu stellt der Beitrag unterschiedliche Möglichkeiten des Informationsaustauschs zwischen dem Multi-Agenten-Framework JADE und der Simulationsumgebung Plant Simulation vor.

1 Einleitung

Aktuell ist in der fertigen Industrie ein Trend zu individualisierten und kundenspezifischen Massenprodukten zu erkennen. Das bedeutet, dass sich die produzierten Losgrößen jedes einzelnen Derivats weiter reduzieren und schließlich in einer Produktion von Unikaten zu Kosten eines Massenprodukts münden. Dies stellt die etablierten Produktionsstrategien der Massenproduktion vor eine Herausforderung und bedingt ein Umdenken in der Fabrikgestaltung. Der Wandel von einer klassischen verketteten Fließfertigung hin zu sich auftragsindividuell vernetzenden Produktionsanlagen ist notwendig [1]. Antworten auf diesen Trend sind bereits in Ansätze sowohl in der Forschung z.B. durch das Projekt ARENA 2036, als auch in der fertigen Industrie, z.B. durch die Speed Factory von Adidas zu erkennen. In diesen Szenarien wird für jeden individuellen Auftrag ein spezifischer Arbeitsplan auf Basis der aktuellen Gegebenheiten des Fertigungsnetzwerks erstellt. Somit sind die einzelnen Arbeitsfolgen und somit auch die

Reihenfolge der notwendigen Fertigungsanlagen für jedes Produkt im Vorfeld nicht geplant. Dadurch ist die etablierte prozessfolgeorientierte Anordnung der Produktionsanlagen und deren feste Verkettung durch Materialflusssysteme nicht zielführend. Es entstehen einzelne Produktionsinseln und das Material wird auftragsindividuell von autonomen und hochflexiblen Transportsystemen zwischen den Anlagen transportiert. Im Rahmen des Forschungsprojekts E|Flow wird die Auftragsallokation der einzelnen Transportaufträge von einem Auftragsagenten übernommen, der mit den Fahrzeugagenten eines vielseitigen autonomen Transportsystems die Vergabe der Aufträge verhandelt. Es entsteht ein intralogistisches Multi-Agenten-System (MAS). Die Einzelentscheidungen dieses Systems bei der Vergabe sind anhand der Strategie der teilnehmenden Agenten planbar, die Lösung komplexer Materialflüsse bestehend aus vielen Aufträgen und Transportentitäten bei der Auftragsallokation und der Reaktion auf ereignisdiskrete Störeinflüsse wie z.B. Maschinenstillständen ist jedoch nicht

vorhersagbar. Abhilfe schafft hier eine ereignisdiskrete Simulation, die die Auftrags- und Transportagenten direkt integriert

2 Grundlagen von Ereignisdiskreten Materialflusssimulationen und Multi-Agenten-Systemen

Ein etablierter Vertreter für ereignisdiskrete Simulationsumgebungen stellt Plant Simulation dar. Die Applikationsdomäne liegt hier vor allem in der Abbildung etablierter Fertigungsstrukturen nach dem verrichtungsorientierten oder prozessfolgeorientierten Fertigungslayout.

Im Bereich der MAS bietet das Framework JADE (Java Agent DEvelopment) eine Middleware für agentenbasierte Peer-to-Peer Anwendungen nach dem Standard der FIPA (Foundation for Intelligent Physical Agents)

2.1 Ereignisdiskrete Materialflusssimulation in Plant Simulation

Während eines Simulationslaufs wertet die Simulationssoftware das Simulationsmodell nur zu bestimmten Zeitpunkten aus. Diese Punkte sind vordefinierte Ereignistypen und markieren den Beginn oder das Ende typischer Fertigungs- oder Intralogistikprozesse, z.B. die Beendigung eines Verarbeitungsschrittes oder den Übergang eines Teils von einem Förderer zum anderen. Es wird davon ausgegangen, dass das System nicht zwischen zwei Ereignissen wechselt und das Modell daher nur zum Zeitpunkt des Eintretens des Ereignisses berechnet werden muss. [2]

Ein Schlüsselement jeder ereignisdiskreten Simulationssoftware ist die Timeline, eine chronologische Liste von Ereignissen. Bei der Ausführung eines Simulationslaufs werden Ereignisse in diese Liste aufgenommen. Die Software wertet das Modell aus, erstellt Statistiken für relevante Parameter und generiert neue Ereignisse entsprechend der Logik des Modells und seines aktuellen Zustands. [3]

2.2 Multi-Agenten-Systeme

Der Begriff "Agent" stammt aus dem Bereich der verteilten künstlichen Intelligenz und existiert seit den 1980er Jahren [4]. Die Art der in diesem Beitrag verwendeten Agenten ist wie folgt definiert: *"Ein Agent ist ein Computersystem, das sich in einer bestimmten Umgebung befindet und in dieser Umgebung autonom agieren kann, um seine Designziele zu erreichen"*[5]. Abhängig vom Einsatzgebiet lassen

sich Agenten und MAS in drei Kategorien gruppieren: [6]

- Systeme mit Fokus auf Interaktion, Koordination und Kooperation
- Intelligente Agenten mit Fokus auf Planung und maschinelles Lernen
- Agenten für die Mensch-Maschine-Interaktion

Kernmerkmale der Agenten sind Autonomie, Bewusstsein, Proaktivität und Kommunikationsfähigkeit [7].

- Autonomie bedeutet, dass ein Agent seine Aktionen ausführen kann, ohne um Erlaubnis fragen zu müssen. Er hat die absolute Kontrolle über seinen inneren Zustand und sein Verhalten. Dies ist jedoch nur eine Pseudo-Autonomie. Das Verhalten eines Agenten und alle möglichen Aktionen sind in seinem Programmcode definiert.
- Bewusstsein bedeutet, dass ein Agent über die Mittel verfügt, seine Umgebung zu überwachen, entweder durch Sensoren, globale Wissensquellen oder durch andere Agenten. Folglich beinhaltet er ein Verständnis oder ein Modell der Umwelt.
- Proaktivität bedeutet, dass ein Agent Ziele hat und in der Lage ist, Pläne zu machen, um diese Ziele zu erreichen. Er kann seine Pläne an Veränderungen in seinem Umfeld anpassen.
- Die Fähigkeit zur Kommunikation bedeutet, dass Agenten mit anderen Agenten oder sogar mit anderer Software interagieren können. Durch das Senden und Empfangen von Nachrichten können sie Dienste bereitstellen oder nutzen, Informationen austauschen oder Verhandlungen aufnehmen.

Mehrere Agenten, die in derselben Umgebung existieren, bilden nicht automatisch ein MAS. Ein MAS definiert man, sobald Agenten zusammenarbeiten, um ein gemeinsames Ziel zu erreichen, z.B. die Organisation des Materialflusses in einer Produktionslinie. Die daraus resultierenden Beziehungen und Abhängigkeiten zwischen den Bearbeitern, die Organisationsstruktur, die Allokation von Aufgaben und der Informationsaustausch sind der Zweck des MAS. [7]

2.3 FIPA Architektur für Multi-Agenten-Systeme in JADE

Die *Foundation for Intelligent Physical Agents* (FIPA), hat mehrere Spezifikationen für die Entwicklung von MAS erstellt, wobei der Schwerpunkt auf der Interoperabilität agentenbasierter Technologien liegt. In der *Abstract Architecture Specification* definiert die FIPA alle Kernelemente eines MAS, von denen die beiden wichtigsten die Agentenkommunikation und das Agentenmanagement sind. [8]

Agenten kommunizieren durch den Austausch von Nachrichten, die in der *Agent-Communication-Language* (ACL) standardisiert sind [9]. Eine ACL-Message besteht aus folgenden Elementen:

- Der eigentliche Inhalt
- Ein Performativ, das die Absicht des Agenten angibt, der die Nachricht gesendet hat (z.B. eine Anfrage oder eine Ablehnung).
- Parameter, um den Inhalt zu beschreiben und wie er interpretiert werden kann.
- Parameter zur Steuerung eines laufenden Gesprächs

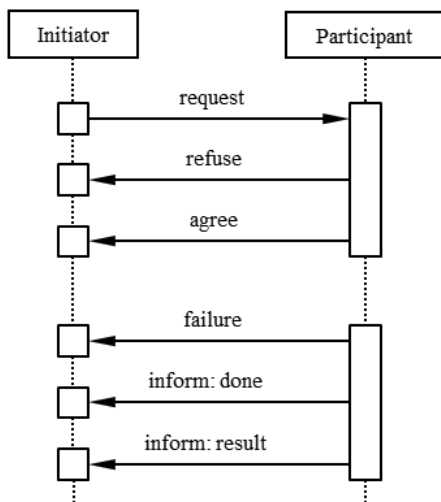


Abbildung 1. FIPA-Request-Protocol [9]

Um sinnvolle Gespräche zu führen, verwenden Agenten Ontologien, eine semantische Sprache und *Interaction Protocols* (IPs). Eine IP definiert eine Abfolge von Nachrichten, die ausgetauscht werden müssen, wenn zwei oder mehr Agenten an einer Konversation beteiligt sind. Ein einfaches Beispiel für eine IP ist das *FIPA-Request-Protocol* (Abb. 1). Der Initiator dieser Interaktion fordert die Ausführung einer Aktion vom Teilnehmer an. Der Teilnehmer entscheidet, ob

er diesem Antrag zustimmt oder nicht. Sobald die Aktion ausgeführt ist, informiert der Teilnehmer den Initiator über das Ergebnis.

Die Agentenverwaltung besteht aus mehreren Komponenten, die für das Funktionieren eines MAS erforderlich sind. Die *Agent Platform* (AP), bestehend aus Maschinen, Betriebssystemen und anderer Software, ist die physische Infrastruktur und das Fundament eines MAS. Jede AP verfügt über ein *Agent Management System* (AMS), das die Aufsicht über die AP ausübt. Ein *Directory Facilitator* (DF) stellt den sogenannten *Yellow Pages Service* zur Verfügung, bei dem Agenten ihre Dienste registrieren. Der *Message Transport Service* (MTS) sorgt für die korrekte Zustellung von ACL-Nachrichten (Abb. 2). [10]

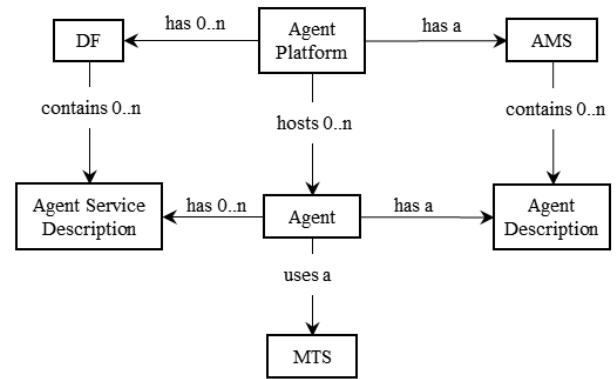


Abbildung 2. Komponenten eines AMS [10]

3 Schnittstellen zum Datenaustausch zwischen JADE, SPS und Plant Simulation

Die Forschungsprojekte KS-Sim II und E|Flow erforschen Schnittstellen, die einen Datenaustausch zwischen einer ereignisdiskreten Simulation in Plant Simulation und einer Speicherprogrammierbaren Steuerung (SPS) bzw. einem Multi-Agenten-System im Framework JADE ermöglichen. Somit ist es möglich, während der Planungsphase des Materialflusssystems die Ontologie der Agenten anhand der zu erwartenden Auftragslage zu testen und zu optimieren. Im realen Testbetrieb werden dann diese Agenten direkt auf die einzelnen erforschten Transportsysteme implementiert ohne eine Anpassung an der Ontologie vornehmen zu müssen. Damit entspricht das tatsächliche Verhalten der Transportsysteme bei der Auftragsallokation dem Verhalten der Agenten während der Simulation, wodurch die notwendige Vorhersagbarkeit des Materialflusssystems gegeben ist.

3.1 Informationsaustausch über eine Socket-Schnittstelle

Es gibt zwei grundsätzliche Möglichkeiten, Daten zwischen Softwareanwendungen auszutauschen, den speicherbasierten und den nachrichtenbasierten Datenaustausch. Die speicherbasierte Interprozesskommunikation (*memory-based inter-process communication* IPC) setzt voraus, dass die Anwendungen auf die gleiche physikalische Speichereinheit zugreifen können, während die nachrichtenbasierte IPC durch das Senden von Daten über ein Netzwerk durchgeführt wird.

Aufgrund des verteilten und dezentralen Charakters von MAS ist der nachrichtenbasierte IPC die einzig gangbare Option. Nachrichtenbasierte IPCs nutzen Netzwerk-Sockets zur Datenübertragung. Sowohl Plant Simulation als auch JADE unterstützen den Datenaustausch über Sockets.

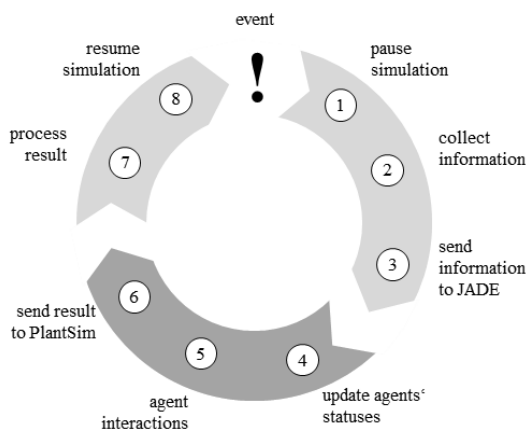


Abbildung 3. Methode des Datenaustauschs zwischen Plant Simulation und JADE über eine Socket-Schnittstelle

Der Datenaustausch zwischen beiden Programmen erfolgt in mehreren Schritten, wenn ein Ereignis (in der Simulation) eintritt, das eine Entscheidung der Agenten erfordert. Der Ablauf dieses Austausches erfolgt wie folgt (vgl. Abb. 3):

1. Der Simulationslauf in Plant Simulation wird angehalten.
2. Alle relevanten Informationen über den aktuellen Systemzustand werden gesammelt.
3. Diese Informationen werden in eine ACL-Nachricht verpackt und über die Socket-Verbindung an JADE gesendet.
4. Alle Agenten erhalten ein Update ihres Status.

5. Der direkt vom Ereignis betroffene Agent erhält eine spezielle Ereignisbenachrichtigung.
6. Die Agenten interagieren miteinander und treffen eine Entscheidung.
7. Dieses Ergebnis wird über die Socket-Verbindung an Plant Simulation zurückgesendet.
8. Der Simulationslauf wird fortgesetzt.

3.2 Informationsaustausch über XML

Soll der gegenwärtige Zustand in einem Produktionswerk als Ausgangszustand in einer Materialflusssimulation verwendet werden, ist ein so genannter digitaler Schatten [11] im Simulationsmodell abzubilden. Hierzu werden auf Knopfdruck alle Daten und Informationen, die Auskunft über den Materialfluss und Betriebszustände von Maschinen und Anlagen geben und aus den SPS ausgelesen werden können, in das Modell integriert.

Um eine pfadbasierte Einbindung in das Netzwerk des Simulationsmodells zu gewährleisten, bietet sich das XML-Format an. Da eine durchgängige, exakte Produktverfolgung in der Regel nie gegeben ist und SPSen Material nur zu bestimmten Zeitpunkten identifizieren, ist der gegenwärtige Produktionszustand aus einer gewissen zeitlichen Rückblende auf die erfassten SPS-Werte abzuleiten. Um mit der langandauernden Speicherung dieser Werte in der SPS deren Leistung aufgrund der Auslastung der Speicherkapazität nicht zu beeinflussen, empfiehlt sich, die Werte in kurzen Zeitintervallen aus der SPS zu exportieren und in XML-Dateien abzuspeichern. Aus diesen Dateien kann bei Bedarf der relevante Satz herangezogen werden, der für die Generierung des digitalen Schattens von Nöten ist. Die Umwandlung in eine für die Simulationsbausteine integrierbare Struktur kann entweder bereits in der SPS, oder auf einem zwischengelagerten Server, oder direkt im Simulationsmodell erfolgen. Die entsprechende Architektur ist in Abb. 4 dargestellt.

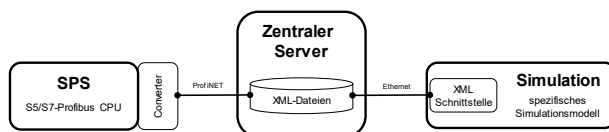


Abbildung 4. Architektur zur Integration von SPS-Werten über XML-Dateien in ein Simulationsmodell

3.3 Informationsaustausch über OPC UA

Möchte man sekundengenaue Auskünfte über den Produktionszustand in das Simulationsmodell integrieren oder Simulationsergebnisse durch Anpassung von Variablen in den SPSen unmittelbar in der Produktion umsetzen, stellt das Kommunikationsprotokoll OPC UA ein geeignetes Mittel dar. Soll beispielsweise ein Transportmittel exakt lokalisiert werden und ist der entsprechende Lokalisierungsparameter über eine Variable in dessen Steuerung enthalten, so kann dieser Wert über eine OPC UA-Schnittstelle direkt von Plant Simulation aus einem auf der SPS eingerichteten OPC UA-Server (auf neuen Steuerungsgenerationen vorhanden, z.B. Siemens S7-1500) abgerufen werden. Die entsprechende Architektur ist in Abb 5 dargestellt. Mit dem importierten Wert lässt sich die Position des Fahrzeugs im Simulationsmodell identisch zum realen Aufenthaltsort festlegen.

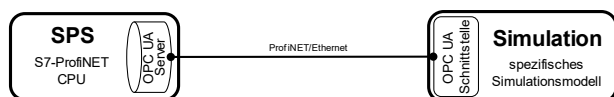


Abbildung 5. Architektur zur Verknüpfung von SPS-Werten über OPC UA mit Plant Simulation

Ein weiterer Vorteil dieser Architektur besteht darin, dass auch in Rückrichtung einfach Daten übermittelt werden können. So kann aus Plant Simulation ein Objekt auf dem OPC UA Server adressiert werden, das wiederum mit einer Variable in der SPS verknüpft ist. Greift nun die SPS in deren Ausführungszyklus auf eine Variable zu, die ihren Wert vom OPC UA Server abrufen, so können auf diesem Wege unmittelbar Anweisungen aus Plant Simulation an die SPS übertragen werden, indem der Wert in das entsprechende Objekt auf dem Server geschrieben wird. Damit ist die direkte Implementierung von Simulationsergebnissen in der Produktion aus technischer Sicht gewährleistet.

4 Einbindung des Multi-Agenten-Systems in die Ereignisdiskrete Simulation über eine Socket-Schnittstelle

Für eine initiale Umsetzung einer Pull-orientierten Produktionslinie wurde der Datenaustausch zwischen Plant Simulation und JADE mit einer Socket-Schnittstelle realisiert. Dazu wurde in Plant Simulation ein Modell erstellt, das aus Lagereinheiten, Maschinen und einer Senke, die das gesamte verarbeitete Material eliminiert. Das Materialhandling zwischen den einzelnen Stationen erfolgt durch vielseitige autonome Transportfahrzeuge (*Versatile Autonomous Vehicles VAV*). [12]

Alle Einheiten werden in JADE durch einen Agenten repräsentiert. Die Organisation des Materialflusses folgt diesem Verhalten:

- Wann immer eine Maschine stillsteht, fordert sie eine Lieferung von Rohmaterial an.
- Um Rohmaterial zu erhalten, muss die Maschine sowohl mit den Lagereinheiten als auch mit den VAVs interagieren.
- In einem ersten Schritt fordert die Maschine Informationen über den aktuellen Bestand und den Standort jeder Lagereinheit an.
- Die Maschine wählt dann eine Lagereinheit aus, basierend auf Bestand und Entfernung.
- In einem zweiten Schritt fordert die Maschine alle VAVs auf, zu signalisieren, ob sie für einen Transport zur Verfügung stehen, und ihren aktuellen Standort zu übermitteln.
- Die Maschine wählt dann das verfügbare VAV aus und hat die kürzeste Distanz zum zuvor gewählten Speicher.

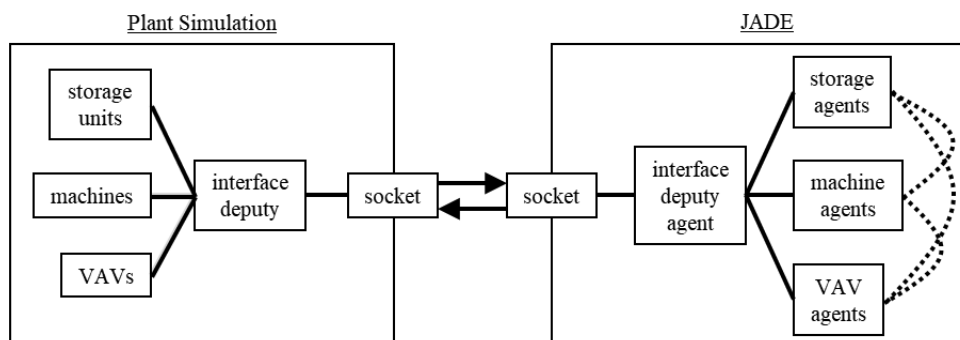


Abbildung 4. Methode des Datenaustauschs zwischen Plant Simulation und JADE über eine Socket-Schnittstelle

- Sobald ein VAV für einen Transportauftrag ausgewählt wurde, fährt es zur Lagereinheit, nimmt das Material auf und bringt es zur Maschine.

Um das Material abzutransportieren, fordert eine Maschine ein VAV an, dass das Material aufnimmt und zur Senke transportiert. Diese Organisation des Materialflusses wird ausschließlich durch den Austausch von Nachrichten zwischen den Agenten erreicht. Plant Simulation generiert nur Ereignisse und stellt den Agenten alle notwendigen Informationen zur Verfügung (z.B. aktuelle Position aller VAVs, Bestandsinformationen der Lagereinheiten). Die Agenten führen dann ihr Verhalten aus und melden das Ergebnis zurück. Schließlich verarbeitet Plant Simulation diese Informationen und setzt den Simulationslauf fort (Abb. 4).

5 Zusammenfassung

Der Beitrag beschreibt eine Methode, die Ontologie von Agenten eines MAS in eine ereignisdiskrete Simulation zu integrieren. Im konkreten Fall wurde eine Schnittstelle zwischen in JADE implementierten Agenten und Plant Simulation geschaffen. Dadurch ist es möglich, das Verhalten der einzelnen Agenten auf ereignisdiskrete Störeinflüsse vorherzusagen. Anhand historischer Produktionsdaten können daher zukünftig hochflexible Materialflusssysteme ausgelegt und die Strategie der Auftragsvergabe, sowie die Strategie der beteiligten Software-Agenten optimiert werden. Weiterhin kann durch den Informationsaustausch der realen Welt zum Simulationsmodell im laufenden Betrieb ein digitales Abbild der Intralogistik geschaffen werden. Dies stellt den Befähiger dar, bei zukünftigen selbstregelnden Produktionssystemen eine Vorhersagbarkeit der Systemlösung bei plötzlich auftretenden Störeinflüssen wie Maschinenstillständen zu gewährleisten, auch wenn die Summe der Einzelentscheidungen der beteiligten Software-Agenten nicht vorhersagbar ist.

6 Danksagung

Die Autoren bedanken sich herzlich beim Freistaat Bayern für die Förderung des Projektes *Green Factory Bavaria* [13] im Rahmen der Zukunftsinitiative *Aufbruch Bayern*.

7 References

- [1] P. Tenerowicz-Wirth. *Kommunikationskonzept für selbstlernende Fahrzeugkollektive in der Intralogistik*. Dissertation, Lehrstuhl für Förderertechnik und Materialfluss Logistik der TU München, Deutschland, 2013.
- [2] U. Bracht, D. Geckler und S. Wenzel. *Digitale Fabrik*. Springer-Verlag, Deutschland, 2011.
- [3] L. März, W. Krug, O. Rose und G. Weigert. *Simulation und Optimierung in der Produktion und Logistik*. Springer-Verlag, Deutschland, 2011.
- [4] T. Lüth. *Technische Multi-Agenten-Systeme*. Carl Hanser Verlag, Deutschland, 1998.
- [5] VDI Richtlinie 4499 Blatt 1. *Digitale Fabrik – Grundlagen*. 2008.
- [6] O. Shehory und A. Sturm. *Agent-Oriented Software Engineering*. Springer-Verlag, Deutschland, 2014.
- [7] P. Meier. *Agentenkomponenten*. Martin Meidenbauer Verlagsbuchhandlung, Deutschland, 2006.
- [8] FIPA Specification SC00001L. *FIPA Abstract Architecture Specification*. 2002
- [9] F. Bellifemine, G. Caire und D. Greenwood. *Developing Multi-Agent Systems with JADE*. John Wiley & Sons Ltd, England, 2007.
- [10] FIPA Specification SC00023K. *FIPA Agent Management Specification*. 2004
- [11] T. Bauernhansl, J. Krüger, G. Reinhart, et. al. *WGP-Standpunkt Industrie 4.0*. Wissenschaftliche Gesellschaft für Produktionstechnik – WGP, Deutschland, 2016.
- [12] C. Kästle, H. Fleischmann, M. Scholz, S. Härter und J. Franke. *Cyber-Physical Electronics Production*. In: Industrial Internet of Things, Springer International Publishing, Deutschland, 2017.
- [13] F. Karl, P. Schnellbach, G. Reinhart, J. Böhner, S. Freiberger, R. Steinhilper, S. Kreitlein, J. Franke, T. Maier, J. Pohl und M. F. Zäh. *Green Factory Bavaria Demonstrations-, Lehr- und Forschungsplattform zur Erhöhung der Energieeffizienz*. wt Werkstatttechnik online, S. 629-632, 2012.

E|Melt: Erweiterung einer unternehmensspezifischen Materialfluss- und Energiesimulation zur Abbildung variabler Betriebsstrukturen der Nichteisen- Schmelz- und Druckgussindustrie

Andreas Buswell¹, Prof. Dr. Wolfgang Schlüter¹

¹Hochschule Ansbach

a.buswell@hs-ansbach.de

Die Erweiterung eines bestehenden betriebsspezifischen Simulationsmodells für Nichteisen- Schmelz- und Druckgussbetriebe auf variable Betriebsstrukturen ermöglicht die Simulation verschiedener Betriebe. Eine Analyse des grundsätzlichen Aufbaus verschiedener Betriebe zeigt die Unterschiede und Gemeinsamkeiten sowie ihre Auswirkungen auf den innerbetrieblichen Materialfluss. Basierend auf der Beschreibung verschiedener Kernkomponenten der Materialfluss- und Energiesimulation ergeben sich die nötigen Anpassungen der beiden Simulationsebenen, um eine simulative Nachbildung unterschiedlicher Werke zu realisieren. In einem ersten Schritt kann die Anlagenkonfiguration und damit die Betriebsstruktur mit der Anzahl der produzierenden Druckgussmaschinen, der Schmelzöfen und der Transportstapler frei erzeugt werden. Das in der bestehenden betriebsspezifischen Simulation existierende Energiemodell der Schmelzöfen wird soweit verallgemeinert und abstrahiert, dass aufwändige messtechnische Untersuchungen zur Ermittlung ofenspezifischer Parameter entfallen. Die resultierenden Simulationen erlauben Aussagen über die betriebliche Energieeffizienz und Produktivität von verschiedenen Unternehmen und bieten die Möglichkeit, für einen spezifischen Betrieb intralogistische Verbesserungsmaßnahmen simulativ zu untersuchen.

1 Energiepotentiale der NE-Schmelz- und Druckgussindustrie

Die Bedeutung der Energieeffizienz hat speziell in Deutschland infolge der Energiewende und der steigenden Konkurrenz durch die Globalisierung stark zugenommen. Das Energie- und Kosteneinsparungspotenzial ist dabei in energieintensiven Branchen, wie z. B. der Nichteisen (NE)-Schmelz- und Druckgussindustrie besonders hoch. In dieser Branche liegt der Energieverbrauch pro Tonne gutem Guss in der Regel zwischen 2000 und 6000 kWh [1, 2, 3]. Dies führt zu einer hohen Energiekostenbelastung, die laut dem statistischen Bundesamt 25 % der Bruttowertschöpfung übersteigen kann [4]. Bis zu 60 % der aufgebrauchten Energie entfallen auf den Aufschmelzvorgang des Aluminiums, der in der Regel in gasbefeuerten Sekundäraluminiumschmelzöfen der werksinternen Schmelzerei abläuft. Die Energieeffizienz der verwendeten Schmelzöfen ist stark abhängig von deren Bauart, Alter, Auslastung und Betriebsweise [1, 5, 6, 7]. Die alterungs- und typbedingten Energieverluste eines Ofens können mittels Optimierungsmaßnahmen, wie bspw. Erneuerung der Ausmauerung oder einer verbesserten Isolierung reduziert werden [8]. Entsprechende Maßnahmen sind jedoch meist mit

hohen Investitionen und längerfristigen Anlagenstillständen verbunden. Weiteres Einsparpotenzial ergibt sich hingegen durch eine Optimierung der Auslastung und Fahrweise, deren Erprobung jedoch mit weitreichenden Eingriffen in den Produktionsablauf verbunden ist und damit die Planungssicherheit des Werks gefährdet. Simulationsgestützte Voruntersuchungen bieten Schmelz- und Druckgussbetrieben die Möglichkeit, Maßnahmen im Voraus zu beurteilen, ohne die Produktion zu beeinflussen.

2 Betriebsstruktur Druckgussbetriebe

Charakteristisch für den Fertigungsprozess von Schmelz- und Druckgussbetrieben ist eine Kombination kontinuierlicher (Schmelzen) und diskreter (Gießen) Prozessschritte. Um diesen hybriden Materialfluss in einer Simulation abbilden zu können, ist eine umfassende Analyse der betrieblichen Strukturen notwendig. Die Prozessschritte eines typischen Schmelz- und Druckgussbetriebs sind in Abbildung 1 dargestellt.

Das Energiemodell der Schmelzöfen erlaubt zudem eine präzise Untersuchung der Auswirkungen auf den Schmelzbetrieb.

Die kontinuierlichen und ereignisdiskreten Prozesse eines Referenzbetriebs wurden identifiziert und mit Matlab, Simulink und Stateflow in ein hybrides Simulationsmodell umgesetzt. Dabei übernehmen die drei genannten Werkzeuge folgende Funktionen:

- Matlab:
Simulationssteuerung und -auswertung,
Objektinstanzierung und -verwaltung
- Simulink:
Simulation kontinuierlicher Prozesse
- Stateflow:
Simulation ereignisdiskreter Prozesse

Der grundlegende Simulationsablauf ist in Abbildung 2 dargestellt. Das Energieflussmodell erfasst die thermodynamischen Vorgänge der Aluminiumschmelzöfen. Mithilfe des Materialflussmodells wird der komplette Materialfluss innerhalb des Werks erfasst. In einem von der Simulation entkoppelten Steuerungsmodul werden auf Basis der ermittelten Anlagen- und Prozessparameter sowie der definierten Steuerungsstrategien Aufträge für die verschiedenen Komponenten generiert. Die ermittelten Aufträge lösen Events in den Stateflow-Charts der zugeordneten Anlagen aus.

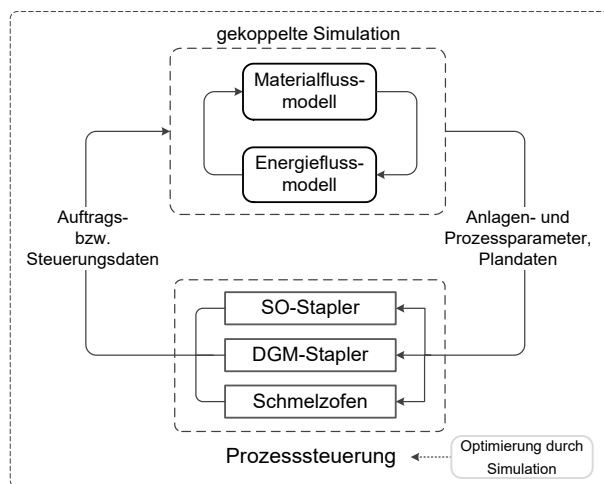


Abbildung 2: Bestandteile der Betriebssimulation

An zahlreichen Prozessen sind mehrere Komponenten beteiligt. So werden bei einem Beschickungsvorgang Daten (bspw. Füllstände) zwischen dem Schmelzofen und dem SO-Stapler ausgetauscht. Hierzu enthält die Simulation ein zentrales „Event-

Handling“-Objekt, das die Kommunikation der Komponenten untereinander gewährleistet.

Neben der gegenseitigen Beeinflussung der Komponente des Materialflussmodells untereinander stellt insbesondere die Wechselwirkung zwischen Materialfluss- und Energiemodell eine Herausforderung dar. Eine separate Simulation der beiden Ebenen wäre technisch deutlich einfacher zu realisieren, würde aber keine gegenseitige ständige Aktualisierung beider Modelle ermöglichen, was die Nutzung vordefinierter Simulationsparameter zur Folge hätte und beispielsweise die dynamische Auftragsermittlung aus aktuellen Betriebsdaten verhindern würde. Daher wurde zwischen Material- und Energieflussmodell eine bidirektionale Kopplung realisiert, die eine gegenseitige Aktualisierung der beiden Simulationsebenen ermöglicht (s. Abbildung 3).

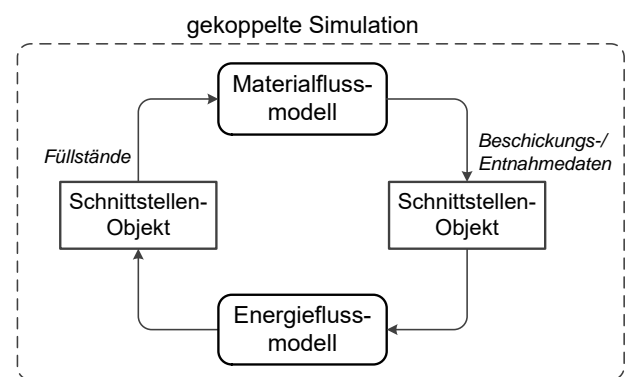


Abbildung 3: Kommunikation der Simulationsebenen

Die Kommunikation beider Ebenen erfolgt mithilfe eines Schnittstellen-Objekts, das den Austausch der relevanten Daten zwischen beiden Modellen ermöglicht. Durch diese Kopplung wird die Untersuchung des Einflusses intralogistischer Verbesserungsmaßnahmen auf den Betrieb der Schmelzöfen ermöglicht.

Mithilfe der beschriebenen Simulation konnten Einsparpotenziale im spezifischen Energieverbrauchs (kWh pro Tonne gutem Guss) von bis zu 20 % aufgezeigt werden, die mittels Maßnahmen wie Materialvorwärmung oder optimierter Beschickungsvorgänge erzielbar sind [10].

Die Anwendbarkeit des des im Projekt Smart Melting entwickelten Simulationsmodells beschränkt sich jedoch auf einen einzelnen Betrieb, also dessen Anlagenkonfiguration und Steuerungsstrategien. Das Projekt E|Melt zielt darauf ab, eine variable Simulationsumgebung zu erstellen, die die Abbildung einer möglichst großen Bandbreite an Betrieben

der NE-Schmelz- und Druckgussbranche ermöglicht. Hierzu ist eine Reihe von Modifikationen an der bestehenden Simulation nötig.

4 Erweiterung auf variable Betriebsstrukturen

Da die Betriebe, wie oben gezeigt, unterschiedlich strukturiert sind, wird im green Factory Bavaria-Nachfolgeprojekt E|Melt ein an verschiedene Industriepartner anpassbares Simulationsmodell entwickelt, das eine vergleichende Bewertung der Ergebnisse erlaubt. Hierzu sind weitreichende Modifizierungen des Ursprungsmodells nötig. Mithilfe einer variablen Modellgenerierung werden das Materialfluss- und das Energiemodell gemäß der betrieblichen Spezifikationen erzeugt. Zudem erfordert die Betrachtung weiterer Unternehmen die Implementation zusätzlicher Steuerungsstrategien und die Entwicklung eines Energiemodells für weite Anwendungsbereiche. Die Betriebsstrukturen erfordern zudem eine Erweiterung der Funktionalitäten des Modells, bspw. um eine Trennung verschiedener Materialflüsse (im Werk: unterschiedliche Legierungen).

Für eine Nachbildung variabler Betriebsstrukturen muss die Simulation um die folgenden Features erweitert werden:

- a) Konfigurierbarkeit der Betriebsstruktur (Anzahl und Parametrisierung der Anlagen)
- b) Konfigurierbarkeit des betrieblichen Materialflusses (Anzahl Legierungen, Prozesszeit, etc.)
- c) Auswahl verschiedener Steuerungsstrategien und Möglichkeit zur Definition neuer Strategien
- d) Adaptierbarkeit des Energiemodells auf möglichst viele Schmelzofentypen

In einem ersten Schritt wurde die Simulation daher in ein Baukasten-System überführt, das es ermöglicht, alle Simulations-Komponenten auf Basis einer betrieblichen Anlagenkonfiguration zu erzeugen und mit den übergeordneten Modulen (Schnittstellen, Steuerung) zu verknüpfen. Das Baukasten-System ermöglicht eine Erweiterung und Konfiguration der Simulation für beliebige Werksstrukturen auf Basis der gängigen Anlagenparameter und einer Analyse des Materialflusses.

Aus der Untersuchung des betrieblichen Materialflusses ergeben sich auch die werkspezifischen Steuerungsstrategien. Das in Abschnitt 0 beschriebene

Steuerungsmodul bündelt die Auftragsstellung der Simulation an einem zentralen Punkt. Somit können neue Strategien nach einer Analyse der Prozesse und Überführung in einen Algorithmus zügig implementiert und getestet werden.

Im ursprünglich abgebildeten Betrieb wurde lediglich eine Legierung verwendet. Bei der Analyse weiterer Betriebe zeigte sich jedoch, dass häufig mehrere Legierungen verwendet werden und somit auch mehrere voneinander getrennte Materialflüsse existieren. Die Verwendung verschiedener Legierungen ist im ursprünglichen Modell nicht vorgesehen. Durch den objektorientierten Ansatz der Simulation war es jedoch möglich, die einzelnen Komponenten in einem standardisierten Verfahren um eine entsprechende Eigenschaft zu erweitern und eine Abfrage des verwendeten Legierungstyps an den Übergabestellen zu ergänzen. Nach der Implementation verschiedener Legierungen ist die Simulation imstande, getrennte Materialflüsse und ihre Auswirkungen auf den Produktionsbetrieb sowie die Auslastung der Schmelzöfen zu erfassen.

Ein Kernelement der während des Projekts Smart Melting entwickelten Simulation stellt das Energiemodell der Schmelzöfen dar, das die thermodynamischen Vorgänge im Inneren der Schmelzöfen sehr detailliert wiedergeben kann. Seine Ermittlung ist jedoch mit einem komplexen Modellierungsprozess verbunden, der umfangreiche Informationen über Aufbau und Arbeitsweise der Anlagen voraussetzt. Weiterhin übersteigt die benötigte Messdatengrundlage (bspw. Rauchgastemperaturen und Füllstände) oftmals den Grad der Datenerfassung innerhalb eines Werks. Die Erarbeitung eines vereinfachten Modells, das mit Einschränkungen in der Funktionalität eine Berechnung der Versorgung mit Flüssigaluminium ermöglicht, ist daher für eine variable Betriebssimulation essenziell.

Ein ideales Modell sollte mit minimalen Genauigkeitseinbußen in den Materialfluss integriert werden können und weiterhin aussagekräftige Ergebnisse über den Energieverbrauch der Anlage liefern.

Die Entwicklung des vereinfachten Energiemodells erfolgt durch eine detaillierte Prozessanalyse anhand von umfangreichen Messdaten. Mithilfe dieser Informationen konnte das detaillierte thermodynamische Modell auf drei Parameter reduziert werden.

Diese beschreiben die Beeinflussung der Schmelzleistung durch Beschickungsvorgänge und den Füllstand des Ofens.

Damit kann die Berechnung der Schmelzleistung erfolgen und die Simulation des gesamten Materialflusses bleibt möglich. Zusätzlich bietet das Modell die Möglichkeit, den Gasverbrauch innerhalb des Simulationszeitraums nachzubilden.

Eine exakte Analyse der thermodynamischen Vorgänge ist auf Basis eines vereinfachten Energiemodells nicht möglich. Im Gegenzug erlaubt es aber eine Simulation von Schmelzöfen unter Kenntnis weniger Anlagenparameter.

5 Ergebnisse

Die unternehmensspezifische Materialflusssimulation mit dem komplexen thermodynamischen Energiemodell wurde während des Projekts Smart Melting anhand von realen Produktionsdaten einer Woche validiert, wobei nur geringfügige Abweichungen bei Materialfluss (1,5 % Unterschied der geschmolzenen Aluminium-Masse) und Energiemodell (0,5 % beim kumulierten Gasverbrauch) auftraten [10].

Um die Güte des vereinfachten Energiemodells zu bewerten, wird dieses in die ursprüngliche validierte Simulation implementiert. Es werden 17 Schichten simuliert, was einer vollständigen Arbeitswoche entspricht. Hierbei wurden die im Partnerbetrieb vorliegende Anlagenkonfiguration und die zugehörigen Steuerungsstrategien zur Erzeugung des Materialflussmodells verwendet. Das vereinfachte Energiemodells wurde gemäß der in Abschnitt 4 beschriebenen Reduzierungen und den benötigten Ofendaten erzeugt. Demgegenüber erfolgte die komplexe Modellierung unter Berücksichtigung aller während des Projekts Smart Melting ermittelten Informationen über die Wirkungsweise der Schmelzöfen.

Die Bewertung der Auswirkungen auf den Materialfluss erfolgte anhand des Material-Inputs durch die SO-Stapler in das System (zugeführte Masse durch Kippstationen ist modellunabhängig) sowie des Outputs der Druckgussmaschinen (Aluminiummasse der Produkte). Nach einer Simulation von 17 Schichten ergaben sich folgende Differenzen der beiden Modelltypen:

- Material-Input: 0 %
- Material-Output: 0,01 %

Zwischen beiden Systemen besteht also nahezu kein Unterschied hinsichtlich der dem System zugeführten Materialmenge und bei der Verarbeitung des flüssigen Aluminiums in den Druckgussmaschinen. Zur Verdeutlichung dieses Ergebnisses zeigt die Abbildung 4 den zeitlichen Verlauf des kumulierten Material-Inputs während der ersten Schicht.

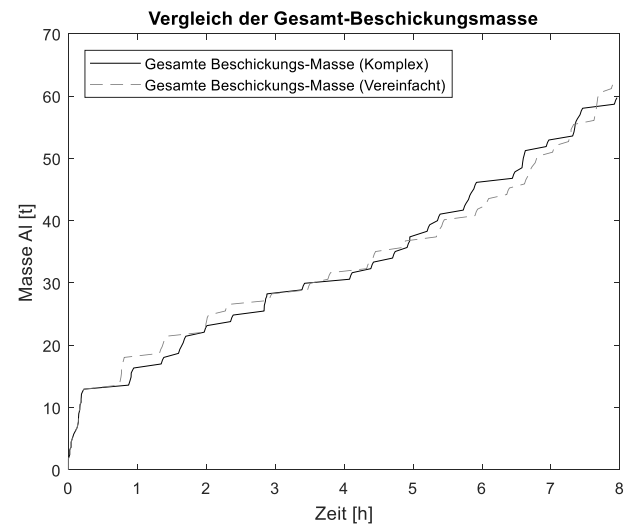


Abbildung 4: Kumulierter Verlauf der Beschickungsmasse

Der Verlauf der Input-Masse zeigt, dass die Vereinfachung des Energiemodells eine Beeinflussung des betrieblichen Materialflusses mit sich bringt. Allerdings ändert sich das grundsätzliche Verhalten des Systems nicht, sodass ein Ausgleich der Abweichungen erfolgt. Während am Ende der in Abbildung 4 dargestellten Schicht eine Differenz von 2,5 t vorliegt, wird in beiden Modellvarianten nach 17 Schichten exakt die gleiche Masse zugeführt.

Einen ähnlichen Verlauf zeigt auch der Materialfluss-Output. Die Produktion der Druckgussmaschinen wird durch die Vereinfachung des Energiemodells also nicht beeinflusst. Für die Druckgussmaschinen spielt ausschließlich eine ausreichende Versorgung mit Flüssialuminium eine Rolle, die aufgrund der betrachteten Anlagenkonfiguration bei beiden Modellvarianten gegeben ist. Erst bei Untersuchung von Versorgungsengpässen ist mit deutlicheren Auswirkungen, resultierend aus dem unterschiedlichen Energiemodell der Schmelzöfen zu rechnen.

Die Berechnung des Energieverbrauchs der Schmelzöfen wird durch die Vereinfachung des Energiemodells stark beeinträchtigt. Aufgrund der durchgeführten Vereinfachung ist die Berechnung von Momentanwerten des Gas-/Energieverbrauchs nicht mehr

möglich. Allerdings erlaubt auch die vereinfachte Form des Ofenmodells eine Bestimmung des kumulierten Gasverbrauchs innerhalb des simulierten Zeitraums. Hierbei kommt es zu einer Abweichung von 5 % gegenüber der komplexen Modellierung.

Die Ergebnisse zeigen, dass eine Vereinfachung des thermischen Modells der Schmelzöfen ohne Einbußen der Genauigkeit möglich ist. Allerdings sind weitere Untersuchungen des Verhaltens unter Betrachtung verschiedener Versorgungssituationen vonnöten.

6 Ausblick

Die Erweiterung einer unternehmensspezifischen Simulation auf variable Betriebsstrukturen in der Nichteisen- Schmelz- und Druckgussindustrie stellt eine simulationstechnische Herausforderung dar. Mithilfe der beschriebenen Anpassungen und Erweiterungen ist es möglich, das Materialfluss- und Energiemodell einer zunächst werkspezifischen Simulation auf abweichende Betriebsstrukturen anzuwenden.

In der nächsten Projektphase wird die Simulation durch reale Prozessdaten eines weiteren Werks mit einer abweichenden Betriebsstruktur validiert. Hierdurch wird erstmals eine vergleichende Untersuchung der Wirksamkeit innerbetrieblicher Verbesserungsmaßnahmen in unterschiedlichen Werksstrukturen ermöglicht. Als Folge können die simulativ ermittelten Ergebnisse und Handlungsempfehlungen deutlich genauer an die industriellen Projekt-Partner angepasst werden.

7 References

- [1] C. Belt. *Current State of Aluminum Melting and Holding Furnaces in Industry*. JOM 67 (2015) 11, S. 2690–2695, 2015.
- [2] M. Bosse, E. Frost, M. Hazrat, J.-M. Rhiemeier und H. Wolff. *Ermittlung von branchenspezifischen Potentialen zum Einsatz von erneuerbaren Energien in besonders energieintensiven Industriesektoren am Beispiel der Gießerei-Industrie*. IfG Institut für Gießereitechnik, Ecofys Germany GmbH, Düsseldorf, 2013.
- [3] C. Herrmann, H. Pries und G. Hartmann. *Energie- und ressourceneffiziente Produktion von Aluminium-Druckguss*. Springer Vieweg, Berlin, 2013.
- [4] C. Schimansky. *Energiepolitik*. <http://www.bdguss.de/themen/energie/#.WLAU5PJCMQM>. Letzter Zugriff am 22.12.2017.
- [5] P. Felder, U. Flückiger und T. Wisskopf. *Gießereibetrieb - Pinch-Analyse Prozessanlagen und Heizungsanlagen: Schlussbericht*. EnergieSchweiz, Bern, 2014.
- [6] W. Stephan, K. Zitzmann, G. Pröbstle, und J. Kapischke. *Effiziente Energieverwendung in der Industrie*. Bayerisches Landesamt für Umweltschutz (LfU), Augsburg, 2005.
- [7] K. Salonitis, B. Zeng, H. A. Mehrabi und M. Jolly. *The Challenges for Energy Efficient Casting Processes*. Procedia CIRP 40 (2016), S. 24–29, 2016.
- [8] W. Schlüter, A. Ringleb und S. Hirschberg. *ZFN01 - Energiepotentialanalyse des Schmelzbetriebes Werk Nürnberg*. Hochschule Ansbach, Ansbach, 2013.
- [9] A. Ringleb. *Green Factory Bavaria: Smart Melting*. http://greenfactorybavaria.de/gf/cms/front_content.php?idcat=90. Letzter Zugriff am 22.12.2017.
- [10] W. Schlüter, M. Henninger, A. Buswell und J. Schmidt. *Schwachstellenanalyse und Prozessverbesserung 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.

Numerische Untersuchung einer Diffusor-Kontraktion und Beeinflussung des Strömungszustandes durch Wandkonturierung

Konstantin Zacharias¹, Wolfgang Schlüter¹

¹Hochschule Ansbach, Fakultät Ingenieurwissenschaften

konstantin.zacharias@hs-ansbach.de

Die numerische Berechnung von abgelösten Strömungen mit positivem Druckgradienten, stellt heute noch eine Herausforderung für die Turbulenzmodelle dar. Der Strömungsabriss in einem Diffusor wird hauptsächlich aufgrund der Druckzunahme verursacht und führt bei einem zu großen Ablösegebiet zu einer Absenkung der Performance. Zur Bestimmung des Strömungszustandes in Diffusor-Kontraktionen werden CFD-Simulationen mit dem $k-\omega$ -SST-Modell durchgeführt. Zunächst wird der Übergang des Zustandes, anliegende zu abgelöste Strömung, anhand einer 2D "Flow-Regime-Map" validiert. Der gleiche Übergang wird anschließend für eine 3D Diffusor-Kontraktion, quadratisch zu rund, ermittelt. Mit der Konturierung der Diffusorwand wird die Performance eines Diffusors, der sich im Bereich der leicht abgelösten Strömung befindet, verbessert und die Ablöseblase verkleinert.

1 Einleitung

Der Diffusor ist ein zentrales Bauteil bei technischen Anwendungen mit Innenströmungen und wird in der industriellen Praxis zur Druckrückgewinnung oder zur Anpassung unterschiedlicher Querschnitte eingesetzt. Durch die divergente Geometrie des Diffusors wird die Strömung verzögert und der Druck nimmt zu, d.h. kinetische Energie wird in Druckenergie umgewandelt. Eine zu schnelle Querschnittserweiterung und ein daraus resultierender steiler Druckanstieg kann zur Ablösung des Fluids an der Wand führen. Die abgelöste Strömung "blockiert" den Fluidtransport und führt schließlich zu einem geringeren Druckrückgewinn, schlechteren Wirkungsgrad und gegebenenfalls zu Instationaritäten.

CFD-Simulationen haben sich in der Auslegung und Optimierung von strömungstechnischen Bauteilen etabliert. Trotz der intensiven Turbulenzforschung und den daraus resultierenden optimierten Turbulenzmodellen ist eine korrekte Berechnung von abgelösten Strömungen unter Einwirkung des positiven Druckgradienten schwierig. Während zwei Workshops der "European Research Community on Flow, Turbulence and Combustion" (ERCOFTAC)

wurden numerisch Diffusoren mit einer abgelösten Strömung untersucht. Im ersten Workshop [1] diente das Experiment von Obi [2] und Buice [3], ein asymmetrischer Diffusor mit einem hohen Aspect-Ratio, als 2D-Testfall. Es wurden unterschiedliche Turbulenzmodelle untersucht, von denen das $k-\omega$ -SST-Modell bei den linearen Modellen das beste Ergebnis lieferte. Im zweiten Workshop [4] wurden zwei 3D-Diffusoren mit einem kleinen Aspect-Ratio und einer starken Strömungablösung untersucht. Das Experiment wurde von Cherry [5] unter Verwendung der Magnetic-Resonance-Velocimetry Technik durchgeführt. Mit dieser Methode konnte eine Datenbasis einer hochauflösenden 3D-Strömung bereitgestellt werden. Alle RANS-Turbulenzmodelle scheiterten an der Berechnung der Position und Größe der Ablöseblase. Eine Ausnahme bildete das speziell entwickelte ANSYS EARSIM Modell. Mit den Large-Eddy-Simulationen (LES) und den LES-Verwandten Modellen konnte eine gute Übereinstimmung zum Experiment erzielt werden. In dieser Arbeit wird das $k-\omega$ -SST-Modell zur numerischen Berechnung der Diffusorströmungen verwendet. Bei der Auswertung der 3D-Strömungen werden nur Diffusoren betrachtet, bei denen die Ablösung gerade einsetzt.

Das Ziel der vorliegenden Arbeit ist es, den Strömungszustand in einer 3D Diffusor-Kontraktion zu bestimmen. Hierfür soll eine "Flow-Regime-Map" erstellt werden mit der der Strömungszustand der Diffuser-Kontraktion analytisch abgeschätzt werden kann. Zusätzlich soll numerisch der Effekt der Wandkonturierung auf den Strömungszustand und die Diffusor-Performance untersucht werden.

1.1 Strömungszustand im 2D Diffusor

Die Diffusor-Performance und der Strömungszustand in einem 2D Diffusor sind schon seit längerer Zeit Gegenstand von experimentellen Untersuchungen [6] [7] [8]. Der Strömungszustand im Diffusor wird nach der Art der Ablösung beschrieben. Die unterschiedlichen Strömungszustände sind in Abbildung 1 in Abhängigkeit ihrer geometrischen Parameter dargestellt. Die Einteilung erfolgte nach experimentellen Untersuchungen am rechteckigen Diffusor für das Medium Wasser [9] und wurden anschließend für das Medium Luft bestätigt.

Anliegende Strömung (No Appreciable Stall)

Dieses Strömungsgebiet liegt unterhalb der Linie a-a und ist charakterisiert durch die stationäre und an der Wandkontur anliegende Strömung. Diffusoren mit anliegender Strömung besitzen ein kleines Querschnittsflächenverhältnis $\frac{w_2}{w_1}$ und damit einen kleinen Öffnungswinkel θ .

Beginnende Ablösung (Appreciable Stall)

Im Zwischenbereich der Linien a-a und a'-a' beginnt die Strömung an vereinzelt Stellen abzulösen. Dies geschieht zunächst in den Ecken, da dort die Impulsverlustdicke größer ist als z.B. im Zentrum der Diffusorwand [5]. Die Strömung folgt noch überwiegend der Kontur und es entsteht keine oder nur eine geringe Rückströmung.

Instationäre Ablösung (Large Transitory Stall)

Die instationäre Diffusorströmung befindet sich zwischen den Linien a'-a' und b-b in Abbildung 1. Die Strömung in diesem Bereich ist unregelmäßig mit starken Fluktuationen. Im Diffusor werden Ablösegebiete gebildet und heraus befördert, was zu starken Druckfluktuationen führt. Die Position der instationären Strömungsbildung variiert über die gesamte Geometrie. Ashjaee und Johnston [7] untersuchten Diffusoren in diesem "Large Transitory Stall Regime" und beobachteten, dass das

Ablösegebiet von einer Diffusorwand zur anderen wandert. Dies war kein schlagartiger Umschlag, sondern die Intervalle erstreckten sich unterschiedlich von Minuten bis hin zu Stunden. Eine Vorhersage, wann und an welcher Wand die Ablösung stattfand, war nicht möglich.

Die Strömungszustände die sich oberhalb der Linie b-b befinden werden in dieser Arbeit nicht berücksichtigt, da diese ein hohes Querschnittsverhältnis besitzen und in der technischen Anwendung von untergeordneter Rolle sind.

Die Performance eines Diffusors lässt sich an Hand des Wirkungsgrades η oder des dimensionslosen Druckbeiwerts (pressure recovery coefficient) c_p beurteilen [6]. Der Druckbeiwert wird über die statische Druckdifferenz und dem dynamischen Druck am Einlass gebildet.

$$c_p = \frac{p_2 - p_1}{\frac{1}{2}\rho\bar{u}_1^2} \quad (1)$$

Diffusoren mit anliegender Strömung und leicht beginnender Ablösung besitzen die größte technische Relevanz. Der höchste Druckrückgewinn wird in einer leicht abgelösten Diffusorströmung erreicht.

2 CFD-Simulationen

Die numerischen Simulationen werden mit der kommerziellen CFD-Software StarCCM+ durchgeführt. Das Strömungsmedium ist Wasser mit einer Dichte $\rho=1000 \text{ kg/m}^3$ und einer dynamischen Viskosität von $\mu=0.001 \text{ Pas}$. Für alle Simulationen ist ein konstanter Massenstrom von $\dot{m}=1.111 \text{ kg/s}$ eingestellt.

2.1 Geometrie und Simulationseinstellungen

Die geometrischen Maße des rechteckigen 2D Diffusors und der 3D Diffusor-Kontraktion sind in Abbildung 2 dargestellt. Bei der Diffusor-Kontraktion wird der rechteckige Querschnitt am Inlet zu einem runden Querschnitt überführt.

Die numerischen Simulationen werden mit dem k- ω -SST-Modell ohne Wandfunktionen durchgeführt. Das unstrukturierte Polyedernetz ist bis in den wandnahen Bereich aufgelöst, sodass ein y^+ -Wert von $y^+<1$ erreicht wird. Um die Unabhängigkeit der Simulationsergebnisse von der Vernetzung

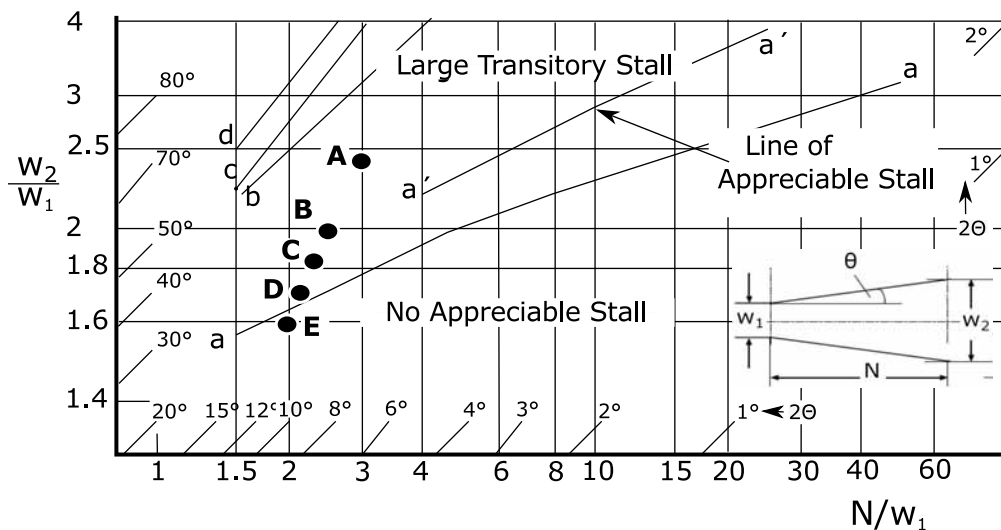


Abbildung 1: Einteilung des Strömungszustandes für ein geradlinigen rechteckigen 2D Diffusor in der "Flow-Regime-Map" nach [9]

sicherzustellen wird eine Netzstudie durchgeführt und die Elementgröße für alle untersuchten Geometrien verwendet. Die Diffusor-Performance und der Strömungszustand korrelieren sehr stark mit dem Geschwindigkeitsprofil (Grenzschichtdicke) [6]. Alle Simulationen werden daher mit einem voll ausgebildeten Inlet-Geschwindigkeitsprofil durchgeführt, welches über ein zusätzliches Modell berechnet und dem Simulationsmodell als Randbedingung aufgeprägt wird.

3 Simulationsergebnisse

3.1 2D Flow-Regime-Map

Die numerischen Untersuchungen zum Strömungszustand finden zunächst an rechteckigen 2D-Diffusoren statt. Dabei werden alle Geometrie Größen, die in Abbildung 2 aufgeführt sind, konstant gehalten und nur der Einströmquerschnitt w_1 variiert. Die fünf untersuchten Diffusorgeometrien sind in Tabelle 1 aufgelistet und in die Flow-Regime-Map in Abbildung 1 eingetragen.

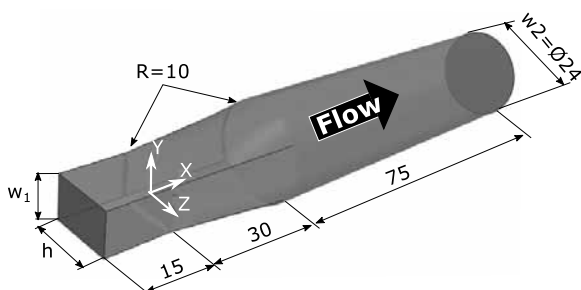
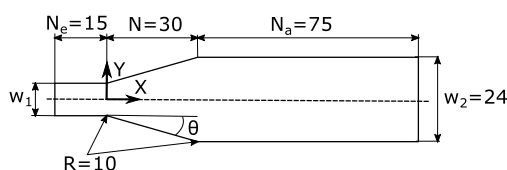


Abbildung 2: Geometrie des rechteckigen 2D Diffusors (oben) und der Diffusor-Kontraktion (unten)

2D Diffusor	A	B	C	D	E
w_1 [mm]	10	12	13	14	15
Ablösegebiet [%]	>2.5	>2.5	2.5	0.67	0

Tabelle 1: Untersuchte 2D Diffusoren

Die Simulationsergebnisse zeigen eine gute Übereinstimmung mit der experimentell ermittelten Flow-Regime-Map. Die Diffusoren A und B liegen im Bereich der instationären Diffusorströmung und weisen ein großes Ablösegebiet auf. Diese wurden nicht weiter ausgewertet, da die 2-Gleichungs-Turbulenz-Modelle nicht für diesen Bereich geeignet sind [4]. Die Diffusoren C und D befinden sich im Bereich der beginnenden Ablösung. Die Strömung folgt überwiegend der Diffusorkontur und das Ablösegebiet nimmt nur einen kleinen Anteil des Gesamtvolumens ein. Die Strömung des Diffusors E folgt vollständig der Kontur und weist keine Ablösung auf. Die experimentell ermit-

telte Linie a-a wird somit gut über CFD-Simulationen wiedergegeben.

3.2 Wandkonturierung des 2D Diffusors

Im Folgenden wird der Einfluss der Wandkonturierung auf den Strömungszustand und den Druckbeiwert c_p im rechteckigen 2D Diffusor numerisch untersucht. Abbildung 3 zeigt zwei mögliche Konturformen. Ausgehend von einem geradlinigen Diffusor wird der Punkt P in y -Richtung verschoben. Dabei entsteht eine Glockenform wenn der Punkt in positive y -Richtung verschoben wird und eine Trompetenform bei einer Verschiebung in negativer Richtung.

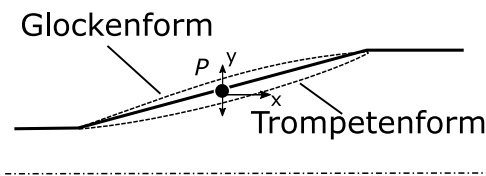


Abbildung 3: Schematische Darstellung der Wandkonturierung

Die Abbildungen 4 und 5 zeigen die Auswirkung der Konturierung auf den Druckbeiwert c_p und auf die Größe des Ablösegebiets. Die Wandkonturierung des Diffusors D hat nur einen geringen Einfluss auf den Druckbeiwert c_p und die Größe des Ablösegebiets (Abb. 4). Bei einer Verschiebung des Punktes P in negative y -Richtung steigt der Druckbeiwert c_p geringfügig an und das Ablösegebiet vergrößert sich. Wird der Punkt P in positive y -Richtung verschoben, sodass eine Glockenform entsteht, wird das Ablösegebiet zunächst etwas verkleinert und steigt anschließend stark an. Die Konturierung der Wand des Diffusors C hat einen größeren Effekt auf das Ablösegebiet und auf den Druckbeiwert c_p . Bei einer Glockenform wird der Ablösebereich unmittelbar vergrößert und der Druckbeiwert c_p verschlechtert sich. Für die Strömungsbedingungen im Diffusor C ist die Trompetenform die ideale Konturierung (Abb. 5). Bei einer Verschiebung des Punktes P um 0.5 mm kann das Ablösegebiet um 1.5% verkleinert werden und der Druckbeiwert auf $c_p = 0.73$ gesteigert werden. In beiden Fällen war es nicht möglich einen Diffusor, der sich im abgelösten Strömungsbereich befindet, durch Konturierung der Wand in den Bereich der anliegenden Strömung zu verschieben. Dies deckt sich auch mit den numerischen Simulationen von Härtel [10], der wandkonturierte 2D Diffusoren auf Basis des Grenzschichtkonzeptes untersuchte. Jedoch

kommt Härtel zu dem Ergebnis, dass für den Diffusor C die Glockenform die ideale Geometrie ist. Härtel verwendete jedoch ein vereinfachtes numerisches Verfahren zur Berechnung der Diffusorströmung. Es ist davon auszugehen, dass die numerische Berechnung mit dem $k-\omega$ -SST-Modell genauere Ergebnisse liefert.

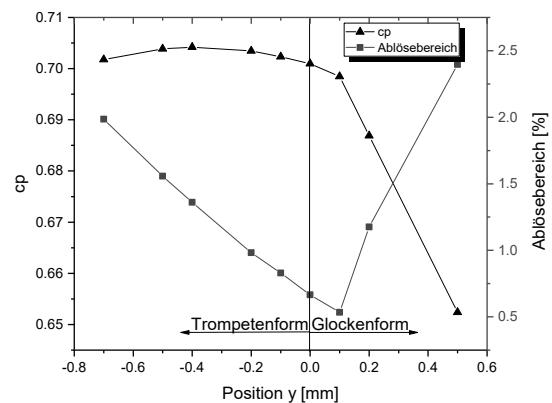


Abbildung 4: Darstellung des Druckbeiwerts c_p und der prozentualen Größe des Ablösegebietes bei unterschiedlicher Wandkonturierung des Diffusors D

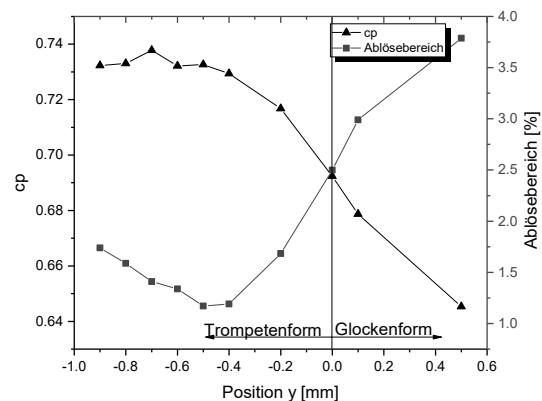


Abbildung 5: Darstellung des Druckbeiwerts c_p und der prozentualen Größe des Ablösegebietes bei unterschiedlicher Wandkonturierung des Diffusors C

Um eine Aussage treffen zu können, ab welchem Flächenverhältnis welche Konturform zu bevorzugen ist, ist eine größere Datenbasis notwendig.

3.3 3D Diffusor-Kontraktion

Zur numerischen Untersuchung des Strömungszustandes in der 3D Diffusor-Kontraktion werden zunächst die Einströmquerschnitte (w_1) aus der 2D Simulation kombiniert zu einem rechteckigen Einströmquerschnitte. Anschließend werden weitere quadratische Einströmquerschnitte untersucht. Alle untersuchten Geometrien sind in der Tabelle 2 mit der Größe des Ablösegebiets aufgelistet. Bei einer zu starken Strömungsablösung wird das Ablösegebiet nicht weiter untersucht. Für die Kombination der Diffusor-Kontraktion wird der Diffusor A und D mit dem Diffusor E gewählt. Der 2D Diffusor E liegt im Gebiet der anliegenden Strömung und bei den 2D Diffusoren D und A reißt die Strömung von der Wand ab. Da bei der Kontraktion vom rechteckigen Profil auf den runden Querschnitt die Ecken in Strömungsrichtung verschwinden, wird ein Strömungsabriss bei höheren Querschnittsverhältnissen (w_2/w_1) erwartet. Die CFD-Simulationen zeigen bei beiden Diffusor-Kontraktionen ein großes Ablösegebiet. Ausgehend von der quadratischen Diffusor-Kontraktion EE wird der Einströmquerschnitt vergrößert bis eine Geometrie mit anliegender Strömung erreicht wird.

Diff.-Kontraktion	AE	DE	EE	FF	GG
w_1 [mm]	10	14	15	17	18
h [mm]	15	15	15	17	18
Ablösegebiet [%]	>2.5	>2.5	>2.5	1.5	0

Tabelle 2: Untersuchte Diffusor-Kontraktionen

Abbildung 6 zeigt den Vergleich der Diffusor-Kontraktion DE und FF als Konturplot der Axialgeschwindigkeit für drei Querschnitte. Die breitere Linie kennzeichnet die Axialströmungsgeschwindigkeit von $u = 0$ m/s und der Abstand der Konturlinien beträgt 0.5 m/s. In beiden Fällen reißt die Strömung bereits am Diffusoreintritt ($x=0$) ab. Der Strömungsabriss wird in den Ecken initiiert bevor es zum Übergang auf den kreisförmigen Querschnitt kommt. Von der Initiierung breitet sich die Ablöseblase an den Diffusorwänden weiter aus und legt sich stromab wieder an die Diffusorwände an. Bei der Kontraktion FF findet das Wiederanlegen bereits zwischen der Position $x=15$ und $x=60$ statt. In der untersuchten Diffusor-Kontraktion GG bleibt die Strömung an der Kontur anliegen. Für die quadratische Diffusor-Kontraktion liegt somit die Linie a_k - a_k , an der ein Strömungsabriss initiiert wird, zwi-

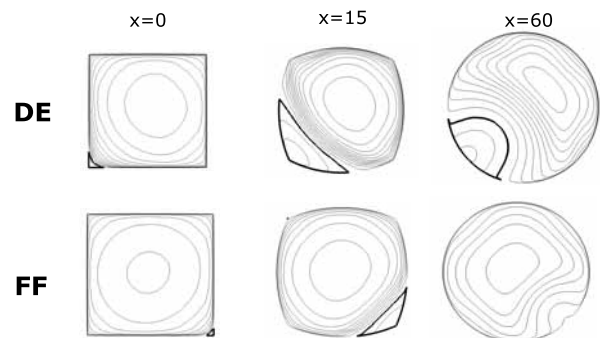


Abbildung 6: Konturplot der Axialgeschwindigkeit für unterschiedliche Querschnittsflächen der Diffusor-Kontraktion DE und FF

chen den Diffusoren FF und GG. In Abbildung 7 ist diese Linie qualitativ in die Diffusor Flow-Regime-Map eingetragen. Die Linie a_k - a_k gilt es für weitere Querschnittsverhältnisse (w_2/w_1) zu bestätigen und für weitere N/w_1 fortzuführen. Mit dieser Flow-Regime-Map kann dann der Strömungszustand in einer Diffusor-Kontraktion von einem quadratischen auf einen runden Querschnitt analytisch abgeschätzt werden.

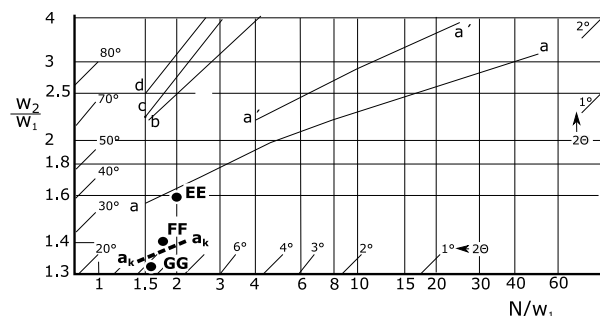


Abbildung 7: Flow-Regime-Map mit Erweiterung der quadratisch zu runden Diffusor-Kontraktion

4 Zusammenfassung und Ausblick

Das Ziel der Arbeit war es, numerisch den Strömungszustand in Diffusor-Kontraktionen zu untersuchen und den Einfluss der Wandkonturierung auf den Druckbeiwert c_p und das Ablösegebiet zu ermitteln. Hierfür wurden numerisch fünf geradlinige, rechteckige 2D Diffusoren untersucht und der

Strömungszustand mit der "Flow-Regime-Map" verglichen. Der Übergang von dem anliegenden Strömungszustand zur leicht abgelösten Strömung konnte mit dem $k-\omega$ -SST-Modell validiert werden. Anschließend wurden die 2D Diffusoren zu rechteckigen/quadratischen Einlaufquerschnitten kombiniert und der Strömungszustand in der Diffusor-Kontraktion untersucht. Es ist zunächst erwartet worden, dass aufgrund der Kontraktion, rechteckig/quadratisch auf rund, die Strömung erst bei höheren Querschnittsverhältnissen (w_2/w_1) abreist. Die Simulationsergebnisse zeigten, dass genau der entgegengesetzte Fall auftritt. Der Strömungsabriss wird bereits in den Ecken am Diffusoreintritt initiiert bevor die Ecken weiter stromab in das runde Profil überführt werden. Um eine anliegende Strömung in der Diffusor-Kontraktion zu erhalten, muss im Vergleich zum 2D ein wesentlich kleineres Querschnittsverhältnis (w_2/w_1) gewählt werden. Es konnte eine Linie a_k-a_k für eine Diffusor-Kontraktion ermittelt werden, die den Übergang von der anliegenden zur abgelösten Strömung definiert. Diese Linie gilt es in weiteren Simulationen zu bestätigen und zu vervollständigen.

Der Einfluss der Wandkonturierung wurde an zwei 2D Diffusoren numerisch untersucht. Ein Diffusor der sich bereits im abgelösten Bereich befindetet kann durch die Konturierung der Wände nicht in der Bereich der anliegenden Strömung verschoben werden. Der Diffusor D, der bereits im Bereich des optimalen Druckbeiwerts c_p liegt, konnte durch die Wandkonturierung nicht wesentlich verbessert werden. Der geradlinige Diffusor C wies im Vergleich zum Diffusor D ein größeres Ablösegebiet auf. Durch die Konturierung zur Trompetenform konnte das Ablösegebiet verkleinert und der Druckbeiwert c_p verbessert werden. Diese Erkenntnis ist in weiteren CFD-Simulationen auf die Diffusor-Kontraktion zu übertragen.

References

- Flow in an Asymmetric Plane Diffuser*, Ninth Symposium on Turbulent Shear Flows, 1993
- [3] C. Buice, J.K. Eaton *Experimental investigation of flow through an asymmetric plane diffuser*, Center for Turbulence Research, 1995
- [4] European Research Community on Flow, Turbulence and Combustion, *13th ERCOFTAC Workshop on Refined Turbulence Modelling*, Report, 2008
- [5] E. Cherry, C. Elkins, J. Eaton, *Geometric sensitivity of three-dimensional separated flows*, International Journal of Heat and Fluid Flow, 29, 803-811, 2008
- [6] L.R. Reneau, J.P. Johnston, S.J Kline, *Performance and Design of Straight, Two-Dimensional Diffusers*, Journal of Basic Engineering, 89, 141-150, 1967
- [7] J. Ashjaee, J.P. Johnston, *Straight-Walled, Two-Dimensional Diffusers Transitory Stall and Peak Pressure Recovery*, Journal of Fluid Engineering, 102, 275-282, 1980
- [8] Robert D. Blevins, *Applied Fluid Dynamics Handbook*, Van Nostrand Reinhold, 1984
- [9] R.W. Fox, S.J. Kline, *Flow Regimes in Curved Subsonic Diffusers*, Journal of Basic Engineering, 84, 303-316, 1962
- [10] A.P. Härtl, *Optimierung von Diffusoren durch Konturierung der Wände auf der Basis des Grenzschichtkonzeptes*, VDI-Verlag, Dissertation Ruhr-Universität Bochum, 1989

[1] European Research Community on Flow, Turbulence and Combustion, *8th ERCOFTAC Workshop on Refined Turbulence Modelling*, Report, 1999

[2] S. Obi, K. Aoki, S. Masuda, *Experimental and Computational Study of Turbulenz Separating*

Simulationsbasierte Bestimmung optimaler Betriebsparameter von Biogasanlagen

David Wagner¹, Wolfgang Schlüter¹

¹Hochschule für angewandte Wissenschaften Ansbach

david.wagner@hs-ansbach.de

Um den Einsatz fossiler Brennstoffe zur Energiegewinnung und deren Auswirkung auf die Umwelt reduzieren zu können, kommen verstärkt erneuerbare Energien zum Einsatz [1]. Eine Möglichkeit der Energiegewinnung aus landwirtschaftlichen Abfällen stellt die Biogasproduktion dar. Dabei werden durch bakterielle Fermentation Substrate unterschiedlicher Herkunft in Methan und Kohlendioxid gespalten. Dieser Prozess unterliegt starken Schwankungen hinsichtlich der Methanerausbeute [2]. Eine sinnvolle Optimierung lässt sich aufgrund der Vielzahl der möglichen Betriebsparameter nur über Modellierung und Simulation erreichen. Obwohl eine Vielzahl an Fermentationsmodellen in der Literatur veröffentlicht sind [3, 4, 5] unterscheiden sich diese stark in puncto Vorhersagekraft, Komplexität und Anwendbarkeit. Für die Beschreibung des kompletten Biogasreaktors werden stets dynamische Modelle hergeleitet, welche aus mehreren gekoppelten Differentialgleichungen bestehen. Eine Parameterreduktion der Modelle auf ein Minimum ist anzustreben, um die stabile Lösbarkeit des gesamten Systems in einem angemessenen zeitlichen Rahmen zu ermöglichen. Es muss daher ein Kompromiss zwischen Genauigkeit und Simplifizierung gefunden werden. Eine Möglichkeit die Komplexität von Modellen und damit die Parameterungenauigkeit zu reduzieren bietet die Sensitivitätsanalyse [6]. Nachfolgend werden zwei Modelle der anaeroben Methanproduktion unterschiedlicher Komplexität hinsichtlich ihrer Parametersensitivität, Betriebsgrenzen und Schätzbarkeit verglichen. Im Falle fehlender Sensitivitäten wird eine Parameterreduktion vorgenommen. Es lassen sich mit diesem Vorgehen bereits durch Simulationsstudien Modelle vergleichen und hinsichtlich ihrer Genauigkeit und Vorhersagekraft beurteilen und somit Empfehlungen für die Abbildung des Realprozesses treffen.

1 Einleitung

Während fossile Brennstoffe einer weltweiten Verknappung unterworfen sind [7], hängen Biogasanlagen lediglich von einem kontinuierlichen Zustrom von landwirtschaftlichen Abfällen ab. Diese Abfälle können tierischen (Gülle) oder pflanzlichen Ursprungs (Gräser, Maissilage etc.) sein. In Biogasanlagen werden diese von verschiedenen mikrobiellen Konsortien unter Ausschluss von Sauerstoff in Methan umgesetzt. Eine Herausforderung innerhalb dieses Prozesses ist die Prozessführung, da zum einen die Zusammensetzung der Substrate, als auch die Zustromrate (Dilution) starken mengenmäßigen, oft jahreszeitlich bedingten, Schwankungen unterliegt. Der Prozess im Reaktor selbst gliedert sich in fünf sequentielle Schritte [8]

1. Desintegration: Aufschluss der komplexen Sub-

strate (physikochemisch)

2. Hydrolyse: Kohlenhydrate, Eiweiße und Fette werden gespalten
3. Acidogenese: Produktion organischer Säuren
4. Acetogenese: Produktion von Essigsäure
5. Methanogenese: Produktion von Methan aus Essigsäure sowie Wasserstoff und CO₂

Obwohl verschiedene Szenarien, wie etwa das Absterben der Biomasse, zu einem Ausfall der Methanproduktion führen, ist die Kontrolle des Prozessverhaltens entscheidend [9]. Ein stabiler Prozess ist nicht zuletzt aufgrund der großen Zeitspannen, in denen die Biogasanlage betrieben wird, von großer Bedeutung. Die leicht steuerbaren Parameter umfassen die Zusammensetzung und Menge des Zustroms. Eine hohe Produktivität wird aufgrund

höherer Ausbeuten bevorzugt. Die Produktivität ist das Produkt aus Zustrom-, oder Dilutionsrate d und Biomassekonzentration $x(t)$ im Gleichgewicht, daher führt eine Erhöhung von d zu einer höheren Produktivität. Dieser Effekt kehrt sich allerdings ab der kritischen Dilution um. Je genauer kritische Betriebspunkte bekannt sind, umso besser kann der Prozess eingestellt werden. Eine Identifizierung und Optimierung ist notwendig, um hohe Ausbeuten zu erreichen [10]. Weiterhin ist ein stabiler und robuster Betrieb vonnöten, um konstant hohe Umsatzraten zu erzielen.

Durch die Menge an möglichen Einflussgrößen, ist der Einsatz von Modellen unumgänglich, mit denen durch Simulation Optimierungen durchgeführt werden können. Aus diesem Grund wurden diverse Modelle hergeleitet, die die Methangärung möglichst genau abbilden. Das bekannteste ist das Anaerobic Digestion Model 1 (ADM1). Dieses wurde im Rahmen einer Modellierungs-Initiative 2002 entwickelt mit dem Ziel, ein generalisiertes Modell für die Beschreibung der anaeroben Umsetzung zur Verfügung zu haben. Ein Nachteil ist die hohe Komplexität des Modells, weitere werden in [11] beschrieben. Diese Nachteile erschweren die Einbindung in ein Prozessmodell ebenso wie die Anwendung des Modells auf den realen Prozess. Aus diesen Gründen werden im Folgenden zwei vereinfachte Modelle herangezogen und per Simulation miteinander verglichen. Modell A beruht auf drei Differenzialgleichungen und ist in Dunn *et al.* 2003 dargestellt. Es beinhaltet die Annahme, dass Essigsäure direkt vorliegt, welche lediglich in Methan umgesetzt wird. Das detailliertere Modell B beruht auf fünf Differentialgleichungen und unterscheidet zwischen acetogener und methanogener Phase [5]. Es enthält daher zwei Substratstufen und zwei Biomassekonzentrationen. Die Produktion von Methan wird durch Essigsäure inhibiert ([12]) und es wird eine Sterberate für die Mikroorganismen inkludiert. Beide Modelle werden hinsichtlich ihrer kritischen Betriebsgrenzen miteinander verglichen. Weiterhin wird aufgrund der Sensitivitäten, eine Parameterreduktion vorgenommen.

2 Material & Methoden

Alle hier gezeigten Simulationen werden mithilfe von Matlab in der Version 9.2 entwickelt. Dabei wird, falls nicht anders beschrieben, der solver *ode45*-verwendet.

Für die Herleitung der kritischen Dilution wird die symbolische Toolbox in Matlab verwendet und die analytische Lösung über den Befehl *solve* gefunden. Dasselbe Vorgehen wird für die Bestimmung der Gleichgewichtspunkte verwendet. Die Lösungen werden über Simulationen, bei dem die Prozessvariablen variiert werden, bestätigt. Für die Sensitivitäten wird jeder Parameter um 25% einer vorgegebenen Basis-konfiguration (Tabelle 2) variiert und die Auswirkung auf die Produktkonzentration im stationären Zustand als Kennzahl für den Einfluss auf den Modellausgang verwendet.

2.1 Modellgleichungen

Modell A beinhaltet für das Bakterienwachstum $\frac{dx}{dt}$ eine Monodkinetik, sowie Ratenausdrücke erster Ordnung für die Substratänderung $\frac{ds}{dt}$ und Produktbildung $\frac{dp}{dt}$. Die genutzten Gleichungen sind im Folgenden gezeigt.

2.1.1 Modell A

$$\frac{dx}{dt} = -d \cdot x + r_x \quad (1)$$

$$\frac{ds}{dt} = d \cdot (s_f - s) + r_s \quad (2)$$

$$\frac{dp}{dt} = -d \cdot p + r_p \quad (3)$$

$$\mu = \frac{\mu_{\max} \cdot s}{K_s + s} \quad (4)$$

$$r_x = \mu \cdot x$$

$$r_s = \frac{-r_x}{Y_{x/s}} \quad (5)$$

$$r_p = (k_1 + k_2 \cdot \mu) \cdot x$$

2.1.2 Modell B

$$\frac{dx_1}{dt} = -d \cdot x_1 + r_x - k_{d1} \cdot x_1 \quad (6)$$

$$\frac{dx_2}{dt} = -d \cdot x_2 + r_{ax} - k_{d2} \cdot x_2 \quad (7)$$

$$\frac{ds}{dt} = d \cdot (s_f - s) + r_s - r_{ap} \quad (8)$$

$$\frac{da}{dt} = d \cdot (a_f - a) + r_{as} + r_a + r_{am} \quad (9)$$

$$\frac{dp}{dt} = -d \cdot p + r_p \quad (10)$$

$$\mu_1 = \frac{\mu_{\max 1} \cdot s}{k_{x1} + s} \quad (11)$$

$$\mu_2 = \frac{\mu_{\max 2} \cdot a}{k_{x2} + a} \quad (12)$$

$$\begin{aligned} r_x &= \mu_1 \cdot x_1 \\ r_s &= \frac{-r_x}{Y_{x/s}} \\ r_{ap} &= (k_{sx1} + k_{nx1} \cdot s \cdot (\frac{1}{k_{sn} + s})) \cdot x_1 \\ r_{as} &= -Y_{as} \cdot r_{ap} \\ r_{ax} &= \mu_2 \cdot x_2 \\ r_a &= \frac{-r_{ax}}{Y_a} \\ r_p &= v_{\max} \cdot x_2 \cdot (\frac{a}{a + k_m}) \cdot (\frac{k_{im}}{a + k_{im}}) \\ r_{am} &= \frac{-r_p}{Y_{ma}} \end{aligned} \quad (13)$$

Parameter, [Einheit]	Wert	Bedeutung
$\mu_{\max, \max 1, \max 2}, [h^{-1}]$	3e-1	max. Wachstumsraten
$k_{s, sn}, [kg/m^3]$	1e-1	Sättigungskonstanten
$k_{x1}, [kg/m^3]$	2.6e-1	
$k_{x2}, [kg/m^3]$	3e-3	Michaeliskonstante
$k_m, [kg/m^3]$	2.08e-2	
$k_{im}, [kg/m^3]$	5.9e-2	Inhibierungskonstante
$k_{d1, d2}, [h^{-1}]$	1e-6	Sterbekonstante
$k_{1, sx1}, [h^{-1}]$	3e-2	Bildungskonstante p 1
$k_2, [-]$	0.08	Bildungskonstante p 2
$k_{nx1}, [h^{-1}]$	8e-2	Energieerhaltkonstante
$v_{\max}, [h^{-1}]$	2.08e-2	max. Umsatzrate
$Y_{x/s}, [kg/kg]$	0.8	Ausbeutekoeffizienten
$Y_{as}, [kg/kg]$	0.83	
$Y_a, [kg/kg]$	0.82	
$Y_{ma}, [kg/kg]$	0.26	

Tabelle 2: Biologische Parameter - Basiskonfiguration

Parameter, [Einheit]	Wert	Bedeutung
$s_f, [kg/m^3]$	10	Substratkonz. Zustrom
$a_f, [kg/m^3]$	0	Acetatkonz. Zustrom
$d, [h^{-1}]$	2.1e-1	Zustromrate

Tabelle 3: Betriebsparameter - Basiskonfiguration

2.2 Modellparameter

Die Prozessvariablen werden im gezeigten Rahmen für die Simulationen variiert (in Tabelle 1 dargestellt), um Gleichgewichte bestimmen zu können und die Ergebnisse der analytischen Betrachtung zu verifizieren.

Prozessvariable, [Einheit]	Grenzen	Bedeutung
$x, [kg/m^3]$	0 - 5e-1	Biomasse
$s, [kg/m^3]$	0 - 20	Substrat
$p, [kg/m^3]$	0	Produkt

Tabelle 1: Für Simulation genutzte Prozessvariablen

Die im Modell enthaltenen Parameter werden standardmäßig auf die in Tabellen 2 und 3 genannten Werte gesetzt. Tabelle 2 zeigt die unveränderlichen biologischen Parameter, während die in Tabelle 3 gezeigten Werte die Basis für Betriebsoptimierungen darstellen.

Sowohl die biologischen, als auch technischen Parameter werden in der Sensitivitätsanalyse um 25% variiert, um die Parameter zu finden, die für eine Modellreduktion in Frage kommen.

3 Ergebnisse & Diskussion

3.1 Modellvergleich kritische Dilution

Die kritische Dilution d_{crit} wird in der Literatur meist über die maximale Produktivität bestimmt [13]. Sie gibt an, ab welcher Zustromrate ein Auswaschen der Zellen eintritt (Verdünnungseffekt). Dabei gilt, dass die Produktivität bis zu einem Maximalwert mit steigender Zustromrate ansteigt, ab d_{crit} fällt sie hingegen rapide ab. Neben der numerischen Bestimmung lässt sich d_{crit} auch analytisch bestimmen, wenn die Gleichungssysteme dies erlauben. Sowohl bei Modell A als auch bei Modell B ist eine analytische Bestimmung von d_{crit} möglich. Die Herleitung ist im Folgenden für Modell A gezeigt und ergibt sich äquivalent für Modell B. Im Unterschied zu Modell A ergeben sich für B zwei d_{crit} , jeweils ein Wert für eine Bakterienart.

Aus der Biomassebilanz ergibt sich im stationären Zustand:

$$\frac{dx}{dt} = 0 \quad (14)$$

$$0 = -d \cdot x + r_x \quad (15)$$

Zusammen mit Gleichung 5 folgt:

$$d = \mu \quad (16)$$

Eingesetzt in Gleichung 4 kann man direkt einen Ausdruck für die Substratkonzentration herleiten und mit dieser Information ergibt sich aus der Gleichung 2 im stationären Zustand folgende Gleichung:

$$x = Y_{x/s} \cdot \left(s_f - \frac{d \cdot k_s}{\mu_{max} - d} \right) \quad (17)$$

Aus den vorliegenden Größen und den gegebenen Gleichungen lässt sich direkt die Produktivität $P(d) = d \cdot x$ (Gleichung 18) herleiten.

$$P(d) = d \cdot Y_{x/s} \cdot \left(s_f - \frac{d \cdot k_s}{\mu_{max} - d} \right) \quad (18)$$

Einsetzen der Werte liefert ein d_{crit} von $0,27 \text{ h}^{-1}$. Bestätigen lässt sich dieser Sachverhalt durch eine Reihensimulation, in dem die Produktivitäten bei d von 0,1 bis 0,3 berechnet werden, wie in Abbildung 1 zu sehen ist.

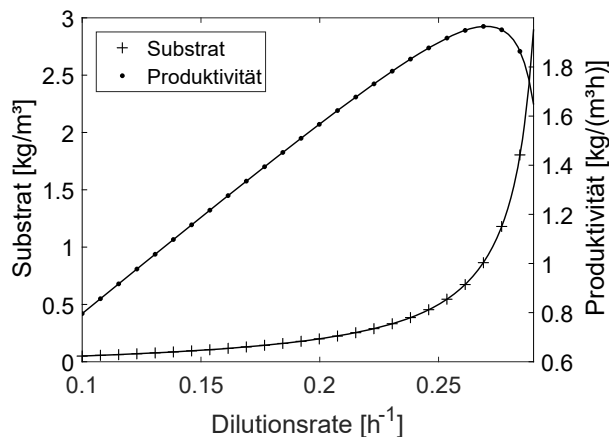


Abbildung 1: Kritische Dilution Modell A

Im Modell B wird nicht das komplette Substrat in Bakterienwachstum umgesetzt. Ein Teil des Substrates wird dafür genutzt die Bakterienpopulation energetisch aufrecht zu erhalten. Daher ist der Wert für d_{crit} geringer (siehe Abbildung 2). Während sich bei den gegebenen Parametern ein Maximalwert für die Produktivität 1 bei $d_{crit} = 0,26 \text{ h}^{-1}$ einstellt, liegt er bei der Produktivität 2 mit $0,23 \text{ h}^{-1}$ deutlich niedriger.

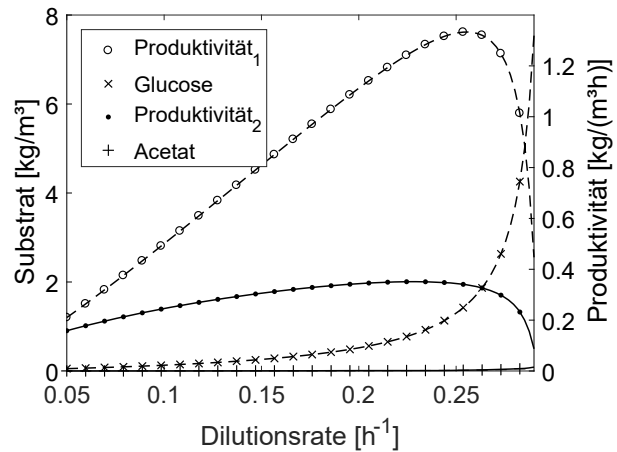


Abbildung 2: Kritische Dilution Modell B

Nicht zuletzt liegt das daran, dass die Methanogenen auf das von den Acetogenen gebildete Acetat zum Wachsen angewiesen sind, welches, wie zuvor bereits beschrieben, nicht äquimolar aus Glucose gebildet wird. Die stationäre Biomassekonzentration liegt daher bei den Methanogenen niedriger. Die Zustromrate muss immer unter d_{crit} liegen, da es ansonsten zu Instabilitäten im Prozess kommen kann.

3.2 Sensitivitätsanalyse & Modellreduktion

Wie bereits eingangs erwähnt, hat ein komplexes Modell Vor- und Nachteile. Während der Vorteil darin liegt, dass meist mehr Effekte mit einbezogen werden und daher der Genauigkeitsgrad erhöht wird, besteht die Gefahr der Überbestimmung. Mithilfe der Fisher-Information lässt sich erkennen, ob Parameter in Modellen nicht schätzbar sind bzw. eine zu hohe Varianz vorliegt [14]. Dafür verwendet man die Sensitivitäten der Parameter. Eine einfache Möglichkeit die Sensitivität von Parametern zu bestimmen ist die 'one factor at a time'-Analyse. Dabei werden die Modelle simuliert und jeweils ein Parameter variiert und dessen Auswirkung auf den Modellausgang graphisch analysiert. Je sensibler ein Parameter, desto höher ist sein Einfluss auf den Modellausgang und desto genauer lässt sich dieser bestimmen. Umgekehrt führen niedrige Sensitivitäten zu hohen Ungenauigkeiten der betroffenen Parameter. Die Auswertung der durchgeführten Analyse ist für Modell A in Abbildung 3 und für Modell B in Abbildung 4 gezeigt. Es ist ersichtlich, dass in Modell A alle Parameter einen Einfluss auf den Modellausgang zeigen, während in Modell B

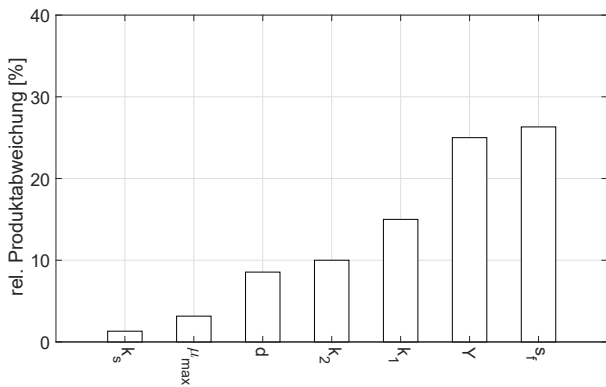


Abbildung 3: Betrag der relativen Produktabweichung der einzelnen Parameter in Modell A

die Sterbekonstanten der Bakterien keinerlei Einfluss haben. In der Literatur wird die Sterberate auch häufig als vernachlässigbar eingestuft. Dies deckt sich mit den Ergebnissen der vorliegenden Sensitivitätsanalyse. In Anbetracht der Größenordnung von $k_d = 10^{-6}$

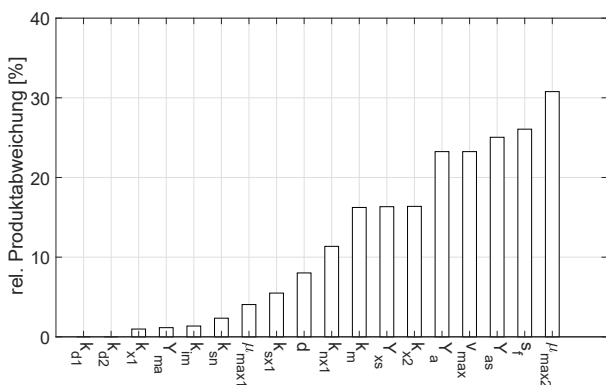


Abbildung 4: Betrag der relativen Produktabweichung der einzelnen Parameter in Modell B

h^{-1} für beide Spezies können die Parameter auf Null gesetzt werden. In Abbildung 5 ist das Ergebnis einer solchen Modellreduktion für 16 verschiedene Konzentrationsszenarien gezeigt, bei denen sowohl k_{d1} als auch k_{d2} auf Null gesetzt wurden. Es ist ersichtlich, dass der Modellausgang (Produktkonzentration) nicht durch die Zerfallsrate beeinflusst wird. Somit kann das Modell von 18 auf 16 Parameter reduziert werden, ohne Einbußen in der Modellgüte zu machen.

4 Zusammenfassung & Ausblick

Bereits mit einfachen vergleichenden Methoden können Aussagen darüber getroffen werden, welches Mo-

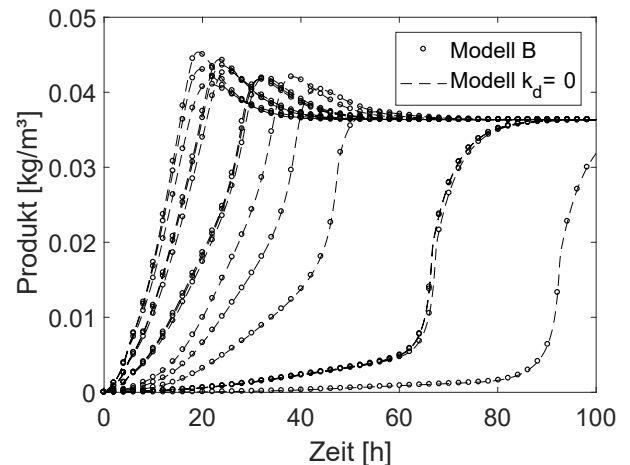


Abbildung 5: Dynamische Produktkonzentration in Modell B (Punkte) und dem reduzierten Modell B (gestrichelte Linie)

dell für die Simulation von Biogasreaktoren geeigneter ist. Zwar bietet ein einfaches Modell sowohl diverse Berechnungsvorteile, die in der Dauer der Simulation (Berechnungszeit ist bei Modell B 14-fach höher als bei Modell A) und der Stabilität der Lösung liegen, jedoch ist ein komplexeres Modell notwendig, um Limitationen im Detail ausfindig zu machen. Nach erfolgter Validierung kann ein geeignetes Fermentationsmodell in ein generalisiertes Prozessmodell eingebunden werden, welches physikochemische (Massenübergang und Strömungsdynamik) als auch kinetische Phänomene koppelt. Damit wird es möglich sein gezielte Vorhersagen für den realen Biogasbetrieb zu treffen. Durch ein geeignetes Prozessmodell kann die Zustromrate optimal gewählt werden, Rührergeschwindigkeiten können auf das Substrat angepasst und für die jeweiligen Substratzusammensetzungen optimiert werden. Gleichzeitig werden Flaschenhälse erkannt, die in der Realanlage zu einer verminderten Methanproduktion führen würden. Damit wird die Basis für eine modellprädiktive Steuerung von Biogasanlagen gelegt.

Literatur

- [1] N. L. Panwar, S. C. Kaushik, and Surendra Kothari. Role of renewable energy sources in environmental protection: A review. *Renewable and Sustainable Energy Reviews*, 15(3):1513–1524, 2011.

- [2] Peter Weiland. Biogas production: Current state and perspectives. *Applied Microbiology and Biotechnology*, 85(4):849–860, 2010.
- [3] D. J. Batstone, J. Keller, R. B. Newell, and M. Newland. Modelling anaerobic degradation of complex wastewater. I: Model development. *Bioresource Technology*, 75(1):67–74, 2000.
- [4] G. Kiely, G. Tayfur, C. Dolan, and K. Tanji. Physical and mathematical modelling of anaerobic digestion of organic wastes. *Water Research*, 31(3):534–540, 1997.
- [5] R. Moletta, D. Verrier, and G. Albagnac. Dynamic modelling of anaerobic digestion. *Water Research*, 20(4):427–434, 1986.
- [6] Andrea Saltelli, Marco Ratto, S Tarantola, and F Campolongo. Sensitivity analysis for chemical models. *Chemical Reviews*, 105(7):2811, 2005.
- [7] Shahriar Shafiee and Erkan Topal. When will fossil fuel reserves be diminished? *Energy Policy*, 37(1):181–189, 2009.
- [8] Lise Appels, Jan Baeyens, Jan Degève, and Raf Dewil. Principles and potential of the anaerobic digestion of waste-activated sludge. *Progress in Energy and Combustion Science*, 34(6):755–781, 2008.
- [9] Henrik Bangsø Nielsen, Hinrich Uellendahl, and Birgitte Kiær Ahring. Regulation and optimization of the biogas process: Propionate as a key parameter. *Biomass and Bioenergy*, 31(11-12):820–830, 2007.
- [10] Yadvika, Santosh, T. R. Sreekrishnan, Sangeeta Kohli, and Vineet Rana. Enhancement of biogas production from solid substrates using different techniques - A review. *Bioresource Technology*, 95(1):1–10, 2004.
- [11] F. Blumensaat and J. Keller. Modelling of two-stage anaerobic digestion using the IWA Anaerobic Digestion Model No. 1 (ADM1). *Water Research*, 39(1):171–183, 2005.
- [12] S. Fukuzaki, N. Nishio, M. Shobayashi, and S. Nagai. Inhibition of the fermentation of propionate to methane by hydrogen, acetate, and propionate. *Applied and Environmental Microbiology*, 56(3):719–723, 1990.
- [13] A. Normandin, J. Thibault, and B. P.A Grandjean. Optimizing Control of a Continuous Stirred Tank Fermenter Using Neural Networks. *Bioprocess Engineering*, 10(May 1993):109–113, 1994.
- [14] Jorma J. Rissanen. Fisher information and stochastic complexity. *IEEE Transactions on Information Theory*, 42(1):40–47, 1996.

Strategien zur Reorganisation von Mischstapeln in automatischen Blocklagern

Markus Fittinghoff¹,

¹Hochschule Heilbronn

markus.fittinghoff@hs-heilbronn.de

In diesem Beitrag soll gezeigt werden, wie mit Hilfe verschiedener Strategien der Reorganisation in einem automatischen Kranlager für Papierrollen die Leistungen beeinflusst werden können. Insbesondere wird ein Bewertungsverfahren vorgestellt, welches eine Priorisierung von umzulagernden Rollen aus Mischstapeln durchführt. Ziel ist es den Effekt auf den Durchsatz der Krananlagen zu untersuchen, wenn Papierrollen mit vergleichsweise geringem Umlageraufwand bevorzugt umgelagert werden. Dieses Verfahren verspricht Vorteile, wenn die Zeitfenster zur Durchführung einer vollständigen Reorganisation nicht ausreichend groß sind.

1 Einleitung

Blocklager stellen einerseits die kompakteste Lagerform bei andererseits beschränkten Zugriffsmöglichkeiten auf das Lagergut dar [1, 2, 3]. In einigen Branchen ist aufgrund der vorteilhaften Flächen- bzw. Raumnutzung bei stapelbarem Lagergut und einer vollständigen Automatisierbarkeit diese Form der Lagerung sehr häufig zu finden. Exemplarisch zu nennen sind die papier- und metallverarbeitende Industrie bis hin zur Stapelung von Containern in Häfen. Gerade bei Einsatzfällen mit geringen Artikelanzahlen können artikel- oder chargenreine Stapel gebildet werden. Immer wenn es gelingt gemäß einem Ordnungskriterium, wie der Artikel- oder Auftragsnummer, sortenreine Stapel zu bilden, entfällt der Nachteil hinsichtlich der Einschränkungen in den Zugriffsmöglichkeiten bzw. das damit verbundene last-in-first-out (lifo) Prinzip. Dann ist es für die Betreiber der Lager unerheblich, welches Gut innerhalb eines Stapels zuerst ausgelagert wird. Durch Vermeidung von Umlagerungen kommt es zu einer vergleichsweise hohen Leistungsfähigkeit. Letztere hängt dann im Wesentlichen von den kinematischen Eigenschaften der eingesetzten Fördermittel, den Dimensionen des Lagers und den Positionen der Ein- und Auslagerplätze ab. Berechnungsverfahren zur Bestimmung des Durchsatzes bietet für diesen Fall die VDI Richtlinie 4480 Blatt 4, sofern auf „Artikel- und Auftragsstrukturen, Lagerstrategien, Umlagerungen, Kommissioniervorgänge etc.“ verzichtet wird [4].

In der betrieblichen Praxis liegt der geschilderte Idealfall häufig nicht vor. Vielmehr treten im dynamischen Betrieb der Blocklager Situationen auf, zum Beispiel aufgrund von Restriktionen in der Produktion und nicht vorhersehbarem Bestellverhalten von Kunden,

die zu einer Einlagerung verschiedener Artikel in einen Stapel, einem sogenannten Mischstapel, zwingen. Sofern - zu späteren Zeitpunkten – bspw. Informationen über Kundenaufträge bekannt werden oder auch Kapazität durch Phasen erhöhter Auslagerungen verfügbar geworden ist, kann eine Reorganisation von Mischstapeln durchgeführt werden.

1.1 Reorganisation

Unter der Reorganisation werden im Kontext der Lagerverwaltung die Überprüfung des Lagerguts hinsichtlich seines optimalen Lagerplatzes und eine möglicherweise sinnvolle Umlagerung des Gutes verstanden [5]. Solche Umlagerungen verursachen Transporte und belasten damit die Fördermittel zusätzlich. Grundsätzlich ist zu hinterfragen, ob die positiven Effekte einer Reorganisation überwiegen oder evtl. die Gesamtleistung des betrachteten Systems gar reduzieren. Einen Überblick über die Literatur, die Komplexität der Fragestellung samt mathematischer Modelle zur Lösung, Heuristiken und experimentell ermittelte Ergebnisse geben Pazour und Carlo [6]. Weitere Ansätze geben auch Korte, Wehking und Neuhäuser [7, 8].

1.2 Bewertung von Mischstapeln

Der hier angedachte Beitrag befasst sich speziell mit einer Bewertung und der Auswahl von Stapeln in einem automatischen Blocklager für Papierrollen, die priorisiert einer Reorganisation unterzogen werden sollen. Es werden mit Hilfe der Simulation die Auswirkungen zweier Reorganisationsstrategien untersucht. Zum einen wird überprüft, wie sich der Durchsatz in dem Lager ändert, wenn nicht vollständig gefüllte Sta-

pel, sogenannte Anbruchstapel, sortenrein zusammengestellt und verdichtet werden. Ziel dieser Strategie ist es, möglichst wenige artikelreine Stapel zu erzeugen.

Zum anderen wird die Strategie untersucht, Mischstapel aufzulösen und zu sortenreinen Stapeln umzulagern. Hierzu werden sämtliche Mischstapel bewertet und die Stapel mit dem geringsten Umlageraufwand zunächst bearbeitet. Zur Verdeutlichung der Problematik dienen folgende Beispiele mit Stapeln, bei denen nur ein Zugriff von oben möglich sei:

Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711

Tabelle 1. Sortenreiner Stapel

Lagergut Artikelnummer 6430
Lagergut Artikelnummer 6430
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711

Tabelle 2. Mischstapel mit zwei Artikeln, Umlageraufwand = 2

Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 4711
Lagergut Artikelnummer 6430
Lagergut Artikelnummer 6430

Tabelle 3. Mischstapel mit zwei Artikeln, Umlageraufwand = 4

Materialflusssysteme: Förder- und Lagertechnik. 3., völlig neu bearbeitete Auflage. Berlin, Heidelberg: Springer Verlag, S. 62.

[3] Pfohl, Hans-Christian (2010): Logistiksysteme. (Betriebswirtschaftliche Grundlagen) Berlin, Heidelberg: Springer Berlin Heidelberg, S. 124.

[4] VDI 4480 Blatt 4 Durchsatz von automatischen Lagern mit mehrfachtiefer Lagerung. (VDI-Richtlinie) Berlin: Beuth Verlag. 2002. S. 3

[5] VDI 3601 Warehouse-Management-Systeme. (VDI-Richtlinie) Berlin: Beuth Verlag GmbH. 2015. S. 37

[6] Pazour, Jennifer A.; Carlo, Héctor J.: „Warehouse reshuffling: Insights and optimization“. In: Transportation Research Part E: Logistics and Transportation Review. 73, 2015, S. 207–226, DOI: 10.1016/j.tre.2014.11.002.

[7] Korte, David; Wehking, Karl-Heinz: „Effiziente Reorganisationsmaßnahmen in manuellen Kommissioniersystemen. In.: Logistics Journal: Proceedings, Vol. 2016. DOI: http://dx.doi.org/10.2195/lj_Proc_korte_de_201610_01

[8] Neuhäuser, Daniel; Wehking, Karl-Heinz: „Der Lagerorganisationsgrad als Steuerungsgröße für optimale Reorganisationszyklen in Kommissioniersystemen“. In: Logistics Journal: Proceedings, Vol. 07. 2011.

2 Referenzen

[1] Arnold, Dieter; Isermann, Heinz; Kuhn, Axel; u. a. (2008): Handbuch Logistik. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 648.

[2] ten Hompel, Michael; Schmidt, Thorsten; Nagel, Lars; Jünemann, Reinhardt (Hrsg.) (2007):

Alterssimulation anhand eines sensorgesteuerten Transportwagens

Prof. Dr.-Ing. Annett Großmann, Dipl.-Ing. Forough Lang

Hochschule Heilbronn

annett.grossmann@hs-heilbronn.de

Diese Arbeit stellt die Ergebnisse eines Forschungsprojektes mit der Firma Blickle Räder+Rollen vor. Thema des Projektes war die Einbeziehung älterer Arbeitnehmer in die ergonomische Gestaltung der Mensch-Maschine-Schnittstelle eines handbetätigten Transportwagens. Es werden drei unterschiedliche Versionen dieses elektrisch angetriebenen Transportwagens miteinander verglichen, wobei die altersabhängigen Einschränkungen durch einen gerontologischen Alterssimulationsanzug (GERT) simuliert werden.

1 Aufgabe/Thema

„In Anbetracht des steigenden Anteils älterer Arbeitnehmer sind die Unternehmen gefordert, mit dieser alternden Belegschaft produktiv und innovativ zu sein, um im globalen Wettbewerb bestehen zu können. [...] Unternehmen [sind] dazu angehalten, durch eine alter(n)sgerechte Arbeitsorganisation und Arbeitszeitgestaltung entscheidend entgegenzuwirken, um die Leistungsfähigkeit älterer Beschäftigter zu erhalten [...]. Zu eruieren ist, welchen Einfluss das altersbedingt nachlassende Leistungsvermögen auf die Produkt- und Prozessgestaltung hat.“ ([3], S. 2)

Dabei sind diese älteren Arbeitnehmer „nicht wie vielfach angenommen weniger leistungsfähig als jüngere Arbeitnehmer. Auch wenn die physischen Fähigkeiten einem Alterungsprozess unterliegen, so ist es immer ein individueller Prozess, der entscheidend durch die Person selbst sowie eine ergonomische Arbeitsplatzgestaltung positiv beeinflusst werden kann. Unternehmen, die erkannt haben, dass der Alterungsprozess kein Abbau- sondern ein Umbauprozess ist, haben in der Regel eine positivere Einstellung gegenüber älteren Mitarbeitern.“ ([3], S. 36)

„Mit einer Simulation [z.B. des Alters] können Arbeitsprozesse [...] bereits vor der Umsetzung eines bestimmten Systementwurfs mit Bezug auf arbeitswissenschaftliche Kriterien bewertet, beurteilt und optimiert werden. Hierfür ist ein quantitatives Modell der Arbeitsprozesse notwendig. Ein wesentlicher Mehrwert der Simulation liegt darin, dass der Modellbenutzer nicht den im realen Arbeitssystem geltenden Einschränkungen unterliegt. Primärer Einsatzbereich [der Alterssimulation] ist die Prognose von Leistung, Belastung, Beanspruchung, Zuverlässigkeit o.ä. in Abhängigkeit verschiedener Einflussgrößen. Dies kann bereits in der Konzeptions- und Entwicklungsphase oder nachgelagert zur Optimierung eines bereits in Benutzung befindlichen Systems stattfinden. Die Simulation bietet sich immer dann an, wenn

die Systemvarianten in der Realität nur mit unverhältnismäßig hohem Aufwand gebildet werden können und gewinnt daher als Unterstützungswerkzeug für die Planung, Steuerung und Optimierung von komplexen soziotechnischen Systemen ständig an Bedeutung“ ([4], S. 472).

Ziel des diesem Bericht zugrundeliegenden Forschungsprojektes war es, einen handbetätigten Transportwagen unter Berücksichtigung der Vorgehensweise nach ISO 9241-210 (mit den notwendigen Phasen und Verifikationsschritten) zu gestalten, der dafür sorgt, dass ältere Arbeitnehmer bei der ergonomischen Gestaltung einbezogen und berücksichtigt werden. „Als probates Mittel zum Nachempfinden körperlicher Leistungseinschränkungen älterer Menschen hat sich die Nutzung sogenannter Alterssimulationsanzüge etabliert. Diese bestehen zumeist aus einem Overall, verschiedenen Gewichten und Bandagen sowie sinneseinschränkenden Brillen und Kopfhörern. Mit Hilfe dieser Komponenten wird versucht, dem Träger ein realistisches Bild von den Einschränkungen im Alter zu vermitteln.“ ([3], S. 3) Deshalb erfolgte der Test des Ausgangsmodells des Transportwagens und der ergonomisch gestalteten Prototypen mit und ohne den gerontologischen Alterssimulationsanzug (GERT), der in Abbildung 3 zu sehen ist. Hersteller dieses Anzuges ist die Firma Produkt + Projekt Wolfgang Moll (www.produktundprojekt.de).

Getestet wurde zunächst ein Transportwagen mit einem elektrischen Antriebssystem und Drehfahrschaltern. Dieser ist modular aufgebaut und eignet sich zum Anbau an unterschiedliche Transport- und Trailersysteme verschiedener Wagenbauer. Durch Einbau des Antriebssystems ist der Kraftaufwand beim innerbetrieblichen Transport wesentlich geringer als beim manuellen Schieben. Insbesondere Belastungen der Bediener beim Beschleunigen und Verzögern der Transportfahrzeuge werden wesentlich verringert und somit die Ergonomie verbessert.

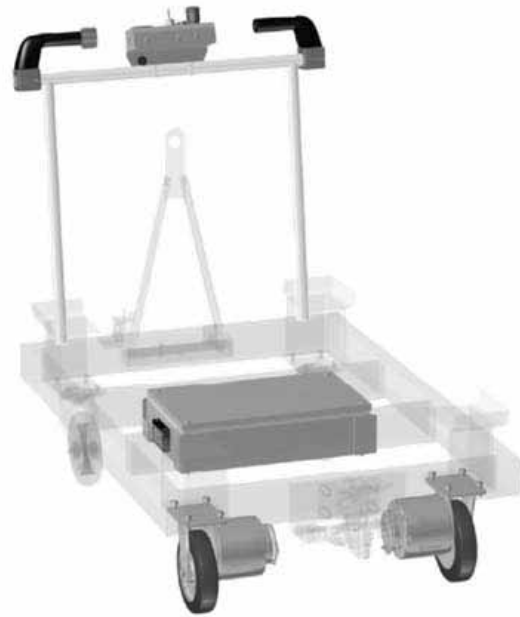


Abbildung 1. Herkömmliche Lösung eines Transportwagens mit elektrischem Antriebssystem (V01) mit Dreh-Fahr-schaltern, hier mit einer Beladung von 1000 kg

Das herkömmliche Antriebssystem besteht aus folgenden Komponenten:

1. Zwei Bockrollen mit zwei Elektromotoren
2. Steuerbox resp. Steuerkonsole mit Batterie
3. Mensch-Maschine-Schnittstelle (Cockpit)

Alle Elemente sind modular aufgebaut und nachrüstbar. Die Steuerung eines solchen Wagens erfolgt bisher durch beidhändige Bedienung mittels Zug/Druck auf die beiden Haltegriffe und Betätigung des Fahrgebers.

Im Falle des hier zu gestaltenden, handbetätigten Transportwagens, welcher von der Fa. Blickle Räder+Rollen gestellt wurde, erfolgte folgendes Vorgehen:

- 1) Anforderungen an ergonomische Transportgeräte formulieren,
- 2) Prototypen einer Version V02 mit Kraftsensor-Lösung bauen und durch Alterssimulationstests mit dem Ausgangsmodell V01 vergleichen,
- 3) Gestaltungshinweise basierend auf den Tests erarbeiten,
- 4) Prototypen einer optimierten Lösung (ErgoMove 1000) bauen und durch Alterssimulationstests mit der am Markt vorhandenen Lösung V01 sowie V02 vergleichen,
- 5) Auswertung der Tests und Erarbeitung von Gestaltungshinweisen.

2 Anforderungen an ergonomische Transportgeräte

„Richtig angewandt, optimiert die Ergonomie die Leistung und Effizienz des Arbeitssystems einschließlich der Arbeitenden ohne nachteilige Wirkungen für deren Gesundheit, Wohlbefinden oder Sicherheit.“ (EN ISO 6385: 2004, S. 13).

Bei der Gestaltung von Arbeitssystemen – hier die ergonomische Gestaltung des Transportwagens mit elektrischem Antriebssystem V01 – sollte der Gestaltungsprozess so weit wie möglich systematisiert ablaufen. Gemäß EN ISO 6385: 2004, S. 6 bedeutet dies „im Gestaltungsprozess müssen die hauptsächlichen Wechselwirkungen zwischen einer oder mehreren Personen und den Bestandteilen des Arbeitssystems [...] berücksichtigt werden.“

„Ziel einer ergonomischen Gestaltung von Arbeitssystemen ist die Optimierung der Arbeitsbeanspruchung, die Vermeidung beeinträchtigender Auswirkungen und die Förderung erleichternder Auswirkungen. Eine nicht beeinträchtigte menschliche Leistung wird gleichzeitig oft die Effektivität und Effizienz des Systems verbessern und so zur Erleichterung eines weiteren wichtigen Ziels der ergonomischen Gestaltung von Arbeitssystemen beitragen.“ (EN ISO 6385: 2004, S. 6).

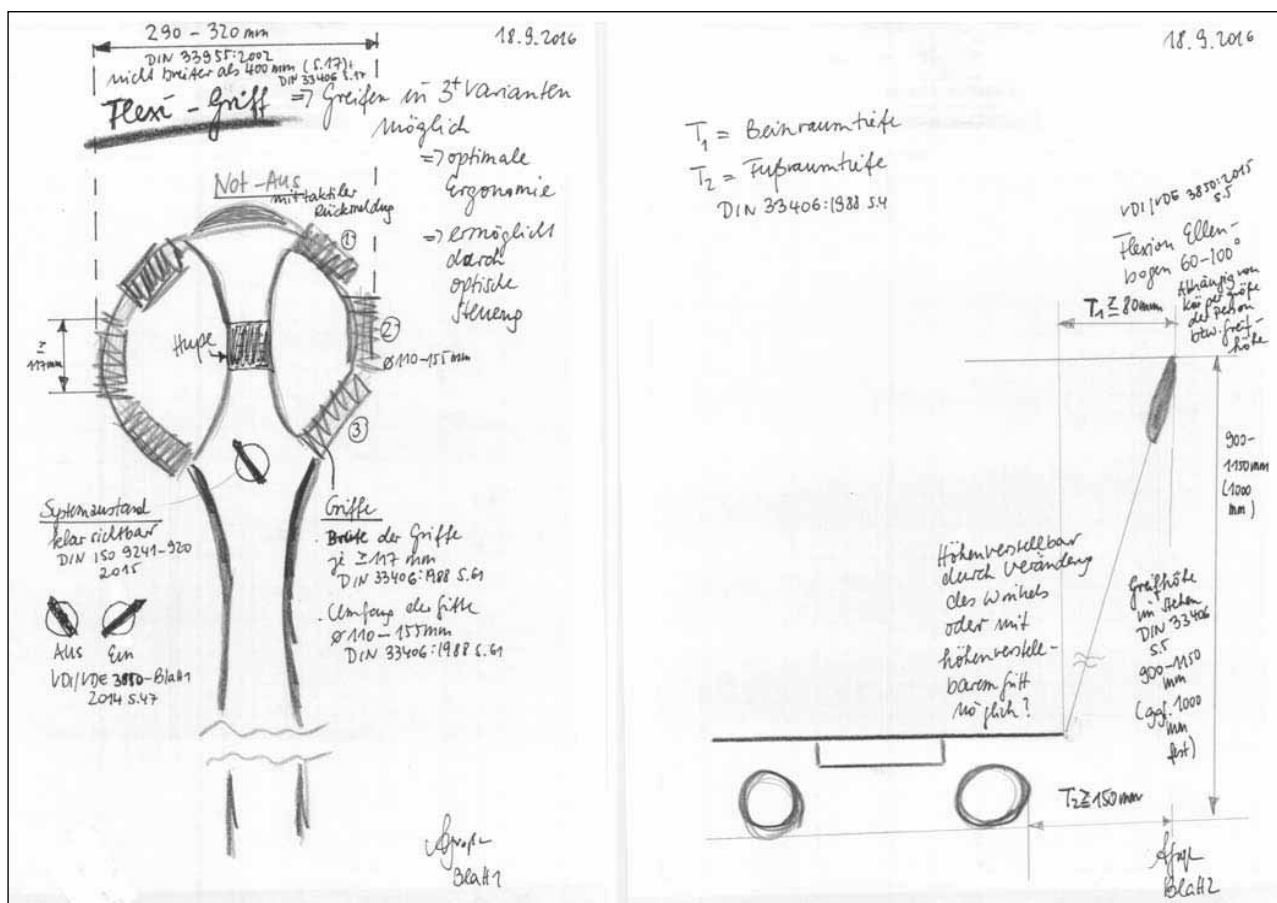


Abbildung 2. Gestaltungsideen des ergonomischen Transportwagens mit Flexi-Griff in der Ansicht von vorne und von der Seite. [5]

3 Testversuche mit dem Alterssimulationsanzug GERT

Es standen drei Versionen des elektrisch angetriebenen Transportwagens (V01, V02 und ErgoMove 1000) der Fa. Blickle Räder+Rollen innerhalb des Projektes zur Verfügung, die jeweils mit und ohne Alterssimulationsanzug GERT unter Laborbedingungen getestet wurden. Testpersonen waren Studierende des Studiengangs Technisches Logistikmanagement der Hochschule Heilbronn. Dabei stellt die Version V01 den am Markt verfügbaren Ausgangszustand mit Dreh-Fahrschalter dar. Die Version V02 wurde anhand der Vorgaben zur ergonomischen Gestaltung als Studie zu einer Kraftsensor-Lösung entwickelt und dann getestet. Die Version ErgoMove 1000 entstand als vorläufige Endversion unter Berücksichtigung der Anforderungen aus den Tests von V01 und V02. Das ErgoMove 1000 der Fa. Blickle Räder+Rollen ist ein modulares Baukastensystem für bestehende Wagensysteme, bestehend aus zwei angetriebenen Bockrollen, einem zentralen Cockpit mit integrierter Steuerung und Akkus. Im Vergleich zu herkömmlich am Markt gängigen Antriebslösungen mit Dreh-

Fahrschaltern verfügt das Blickle ErgoMove 1000 über einen Ergonomiegriff mit integrierter Kraftsensorik. Dieser erfasst die an ihm einwirkenden Kräfte des Bedieners in drei Dimensionen: vor / zurück, links / recht und Torsion. Beim Loslassen der Haltegriffe bremst der Wagen automatisch ab.

4 Auswertung und Ergebnisse

Für die Auswertung wurden die Ergebnisse der Tests der verschiedenen Transportwagen bzw. Prototypen anhand ausgewählter Eigenschaften miteinander verglichen und in Kategorien zusammengefasst. Die Diagramme in den Abbildungen 12 bis 16 geben einen Überblick über die Ergebnisse. Dabei veranschaulichen die orangefarbenen Balken die Bewertungen für die Transportwagen bzw. Prototypen V01, V02 und ErgoMove 1000 ohne GERT und die blauen Balken die Bewertungen für Transportwagen mit GERT. Auf der x-Achse werden die Eigenschaften in Punkten dargestellt. Dabei repräsentiert der Wert 0 „Eigenschaft trifft gar nicht zu“, Wert 1 „Eigenschaft trifft nicht zu“, Wert 2 „Eigenschaft trifft teilweise zu“, Wert 3 „Eigenschaft trifft zu“ und Wert 4 „Eigenschaft trifft vollkommen zu“.



Abbildung 3. Alterssimulationsanzug GERT



Abbildung 4. Ausgangsmodell V01



Abbildung 5. Testumgebung Labor



Abbildung 6. Prototyp V02



Abbildung 7. Prototyp V02



Abbildung 8. 3D-Druck Griff Prototyp V02



Abbildung 9-11. Test Prototyp ErgoMove 1000 der Fa. Blickle Räder+Rollen mit Alterssimulationsanzug GERT

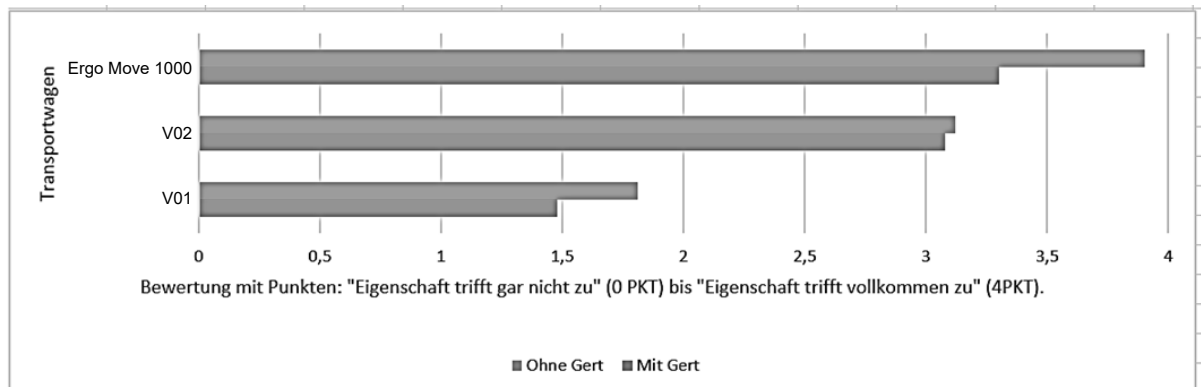


Abbildung 12. Vergleich der drei Transportwagen (mit/ohne GERT) anhand der Eigenschaft „Die Höhe und Ausrichtung der Griffe sind aus ergonomischer Sicht passabel“.

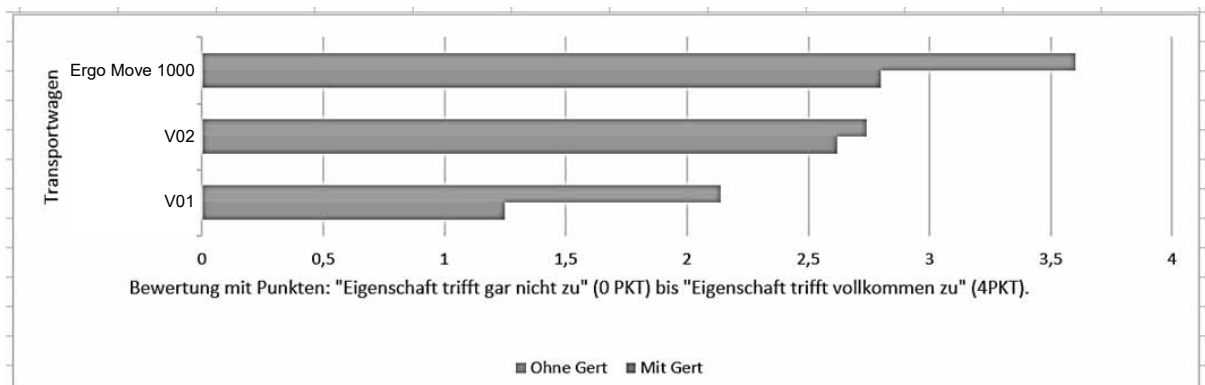


Abbildung 13. Vergleich der drei Transportwagen (mit/ohne GERT) anhand der Eigenschaft „Die Geschwindigkeitsregelung ist einfach“.

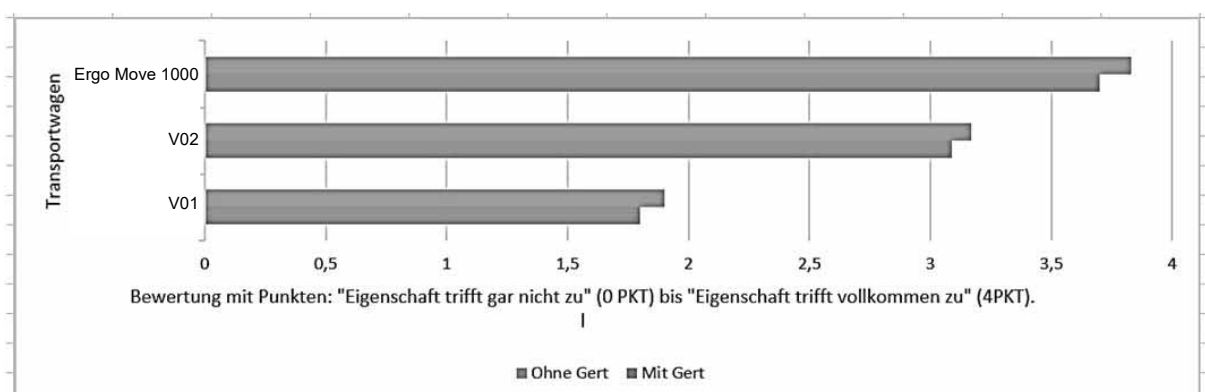


Abbildung 14. Vergleich der drei Transportwagen (mit/ohne GERT) anhand der Eigenschaft „Das Ein- und Abschalten des Wagens ist für die Anwender problemlos“.

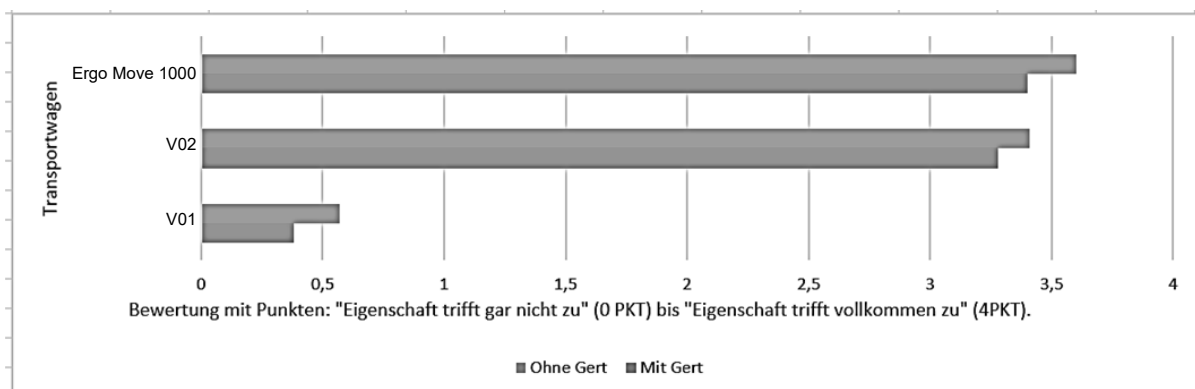


Abbildung 15. Vergleich der drei Transportwagen (mit/ohne GERT) anhand der Eigenschaft „Die Farbe und Platzierung der Bedienelemente sind aus Sicht der Tester optimal“.

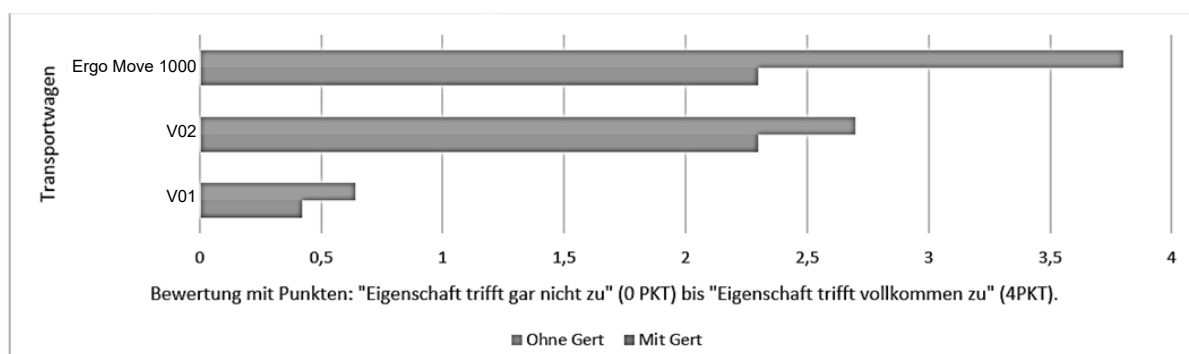


Abbildung 16. Vergleich der drei Transportwagen (mit/ohne GERT) anhand der Eigenschaft „Der Notausknopf ist für die Anwender schnell und gut erreichbar“.

Die Ergebnisse zeigen, dass die Gestaltung des handbetriebenen Transportwagens in nahezu allen Kategorien alter(n)sgerecht ist und dass die Benutzer – mit oder ohne Alterssimulationsanzug GERT – mit der Gestaltung und Bedienung des Transportwagens ErgoMove 1000 zufriedener waren gegenüber der am Markt verfügbaren herkömmlichen Lösung mit Dreh-Fahrschalter V01 sowie dem Prototypen zur Kraftsensor-Lösung V02.

5 Zusammenfassung und Schlussfolgerungen

Die Aufgabe bestand darin, einen am Markt verfügbaren Transportwagen mit elektrischem Antriebssystem und Dreh-Fahrschaltern ergonomisch und alter(n)sgerecht zu gestalten. Im ersten Schritt wurden Anforderungen an ergonomische Transportgeräte formuliert, es wurden auf dieser Basis die Prototypen V02 und ErgoMove 1000 gebaut und mit Hilfe des Alterssimulationsanzuges getestet. Ziel der Gestaltung war es, eine Alter(n)sgerechtigkeit zu realisieren. Dies konnte erreicht werden, wie die Tests ergeben haben. Die Ergebnisse der Tests zeigen dabei eine deutliche Verbesserung der Version ErgoMove 1000

gegenüber der herkömmlichen Lösung V01 und der Kraftsensor-Studie V02. Sowohl ohne Alterssimulationsanzug GERT und insbesondere mit GERT ist der alter(n)sgerecht gestaltete Wagen ErgoMove 1000 eindeutig die bessere Variante.

6 References

- [1] DIN EN ISO 9241-210: 2004 Ergonomie der Mensch-System-Interaktion – Teil 210: Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme (ISO 9241-210:2010); Deutsche Fassung EN ISO 9241-210:2010.
- [2] EN ISO 6385: 2014: Grundsätze der Ergonomie für die Gestaltung von Arbeitssystemen (ISO/DIS 6385:2014); Deutsche Fassung prEN ISO 6385:2014.
- [3] Scherf, Chr. 2014: Entwicklung, Herstellung und Evaluation des Modularen AlterssimulationsanzugesExtra (MAX); Universitätsverlag Chemnitz, 2014.
- [4] Schlick, Chr., Bruder, R., Luzcak, H. 2010: Arbeitswissenschaft, 3. Auflage, Springer: 2010.
- [5] Skizzen Großmann, A. (unveröffentlicht) 2016
- [6] Bilder Großmann, A.; Bozkurt, B.; 2017/2018

WILLE: wissenschaftliches Lehren und Lernen zur Förderung der Gründungskultur an Hochschulen

Jana Heimel¹, Joachim Allhoff², Martin Vural³

¹Hochschule Heilbronn, Max-Planck-Str. 39, 74081 Heilbronn

jana.heimel@hs-heilbronn.de

²Institut für angewandte Marktforschung (H-Infam) an der Hochschule Heilbronn,
Max-Planck-Str. 39, 74081 Heilbronn

joachim.allhoff@hs-heilbronn.de

³Hochschule Heilbronn, Max-Planck-Str. 39, 74081 Heilbronn

martin.vural@hs-heilbronn.de

Der Beitrag behandelt die Konzeptionierung innovativer Lehr- und Lernformate im Gründungskontext, welche im Rahmen des vom Ministerium für Wissenschaft, Forschung und Kunst in Baden-Württemberg geförderten Projekts WILLE an der Fakultät für International Business entwickelt und pilotiert werden. Für die Weiterentwicklung steht die Integration von Ansätzen des forschungsbasierten Lernens sowie Blended Learning im Kontext der Unternehmensgründung im Fokus. Zunächst werden wesentliche Ergebnisse einer großflächig angelegten empirischen Studie an der Hochschule Heilbronn dargelegt, dann das theoretische Konzept (FUB-Modell) entwickelt, welches die drei Dimensionen forschungsbezogenes Lernen (Forschen), Unternehmensgründung (Start-ups, Inkubatoren) sowie Blended Learning (Lehre) integriert und anschließend in studentischen Projektgruppen pilotiert. Die Projektergebnisse werden schließlich technisch in einer IT-Lösung umgesetzt, welche auch über der Projektlaufzeit hinaus von Studierenden angewendet werden soll.

1 Einleitung

1.1 Problemstellung

Forschung und Innovationen zählen neben Bildung zu den wichtigsten Säulen unseres gesellschaftlichen Fortschritts und Wohlstands. Hochschulen als Ort der Wissenschaft tragen deshalb eine große Verantwortung gegenüber der Gesellschaft. Zur Sicherung unserer Wettbewerbsfähigkeit bilden neue innovative Ideen und daraus entstehende marktfähige Produkte und Dienstleistungen den Schlüssel für Innovation. An Hochschulen wurde die rechtliche Sicherung und wirtschaftliche Verwertung von Ideen lange Zeit ignoriert. Oft profitierten auch andere Akteure davon. Fast alle Hochschulen haben mittlerweile Beratungsstellen eingerichtet und Fakultäten der Wirtschaftswissenschaften haben das Thema Unternehmensgründung in ihr Curriculum aufgenommen. Allerdings fehlt oft noch die Verknüpfung

zwischen Forschung, Lehre und Gründung. Die wenigsten wissen, wie man innovative Ideen oder Forschungsergebnisse wirtschaftlich verwerten kann. Für die Realisierung einer lebendigen Gründerkultur muss das Miteinander methodischer Kompetenzen sowie Fachwissen gefördert werden.

Die Gründungsquote von Hochschulabsolventen liegt bei unter einem Prozent und das obwohl mindestens jeder Dritte sich (sehr) gut vorstellen kann, ein eigenes Unternehmen zu gründen. Dabei hat die Anzahl der Studierenden ihr Allzeithoch erreicht. Mit Blick auf eine beinahe Vollbeschäftigung in Deutschland ist die Lehre herausgefordert, neue Lernformate im Gründungskontext (weiter) zu entwickeln. Unternehmensgründungen sind ein wichtiger Beitrag für die Entwicklung innovativer Produkte und Dienstleistungen sowie für die wirtschaftliche Entwicklung allgemein. Es mangelt folglich an einem adäquaten Lehrmodell für die Förderung von Gründungen. Darüber hinaus ist das forschungsbasierte Lernen auch ein zentraler Bestandteil an Hochschulen

für angewandte Wissenschaften (für wissenschaftlich fundierte Ausbildung, kreatives, kritisches und vernetztes Denken). Zwar hat die Wissenschaft diese Problematik seit geraumer Zeit erkannt und bringt hier stetig neue Konzepte hervor, jedoch nicht vor dem Gründungskontext. Dieser bedarf der Vermittlung von Wissen nicht nur auf eine spezielle Art, sondern auch anderer Inhalte. Unser Teilprojekt von WILLE greift diese Problematik auf, indem die drei Dimensionen forschungsbezogenes Lernen (Forschen), Unternehmensgründung (Start-ups, Inkubatoren) sowie Blended Learning (Lehre) integriert betrachtet werden.

1.2 Zielsetzung

Ziel des Projekts ist es, neue innovative Lehr- und Lernformate im Gründungskontext zu erproben und weiterzuentwickeln. Für die didaktische Weiterentwicklung steht die Integration von Ansätzen des forschungsbasierten Lernens sowie Blended Learning im Fokus. Bei der Verzahnung von Forschung und Lehre werden durch den Einsatz forschungsbasierter und innovativer Lehrmethoden Studierende zum selbstständigen Lernen motiviert, mit dem Ziel, das Potenzial und die Anzahl von Unternehmensgründungen im Studienbetrieb langfristig zu fördern. Im Vordergrund steht der Ausbau erfolgreicher Lehrformate für forschungsbasiertes Lernen sowie Blended Learning. Fachlich sollen diese auf ingenieur-, informations- und wirtschaftswissenschaftliche Studiengänge zugeschnitten sein. Als Ergebnis soll eine WILLE-App vorliegen, welche auf Game-based und Blended Learning Tools aufbaut und der Wissensvermittlung, Sensibilisierung und Qualifizierung von Studenten für potenzielle Gründungsvorhaben dient.

2 Projekt WILLE

2.1 Projektzielsetzung und -struktur

Das WILLE-Projekt „Nah an Praxis und Forschung – innovative Lernmodelle an der Hochschule Heilbronn“ strebt mit neuen Lehrkonzepten einen Beitrag zur Qualitätssteigerung der Lehre sowie eine verbesserte Wissensvermittlung in einem interdisziplinären Umfeld an. Das vom Ministerium für Wissenschaft, Forschung und Kunst in Baden-

Württemberg bis 2019 geförderte Projekt beschäftigt sich mit der Entwicklung eines innovativen auf Forschung basierten Lehr- und Lernmodells.

Anknüpfend an die Eingangsüberlegungen möchte das Projekt neue Erkenntnisse zur didaktisch praktischen Umsetzung von Lehrveranstaltungen generieren, Lehr- und Lernmodelle weiterentwickeln und in die Lehre integrieren. Im Mittelpunkt stehen dabei die Studierenden. Die Wissensvermittlung sollte optimal auf die Bedürfnisse von Studierenden zugeschnitten sein. Ferner gilt es, Studierende in Ihrem Lernen bestmöglich zu unterstützen und das eigenständige Lernen zu ermöglichen. Das vorrangige Projektziel lässt sich in zwei Teilbereiche untergliedern: „Forschungsbezogenes Lernen“ sowie „Problemorientiertes und projektorientiertes Lernen“.

2.2 Teilprojekt „Forschungsbezogenes Lernen“

Das Vorhaben wird an der Fakultät International Business durchgeführt. Für das Gelingen baut das Vorhaben nicht nur auf die Inhalte, sondern auch auf die Kompetenzen und das Erfahrungswissen der Lehrenden aus den bereits bestehenden Vorlesungen auf.

Im Teilprojekt „Forschungsbezogenes Lernen“ steht die engere Verbindung von Forschung und Lehre im Kontext der Unternehmensgründung im Vordergrund. Das bereits im Studium angeeignete Wissen soll durch die Ausbildung von Urteilsfähigkeit mit Forschungsfragestellungen erweitert werden. Dies soll die Studierende nicht nur auf die Abschlussarbeit besser vorbereiten, sondern auch den hochschulinternen „akademischen Fahrstuhl“ befördern. Studierende sollen ein kritisch reflektierendes akademisches Denken und Handeln annehmen. Die Herausforderungen liegen hier in der optimalen Gestaltung von Lehrveranstaltungen und der Aktivierung sowie Befähigung von Studierenden für eigenständige Forschungsvorhaben.

3 Methodisches Vorgehen

Das methodische Vorgehen umfasst folgende sechs Phasen (s. Abb. 1): In Phase 1 (Bestandsaufnahme) erfolgte im Rahmen einer empirischen Erhebung mittels einer Online-Umfrage unter allen Studierenden der Hochschule Heilbronn die Bestandsaufnahme zu

den drei (FUB) Dimensionen. Im zweiten Schritt (Recherche und Mobilisierung) wurde in Unterstützung mit studentischen Forschungsgruppen eine fundierte Recherche zum Thema Forschungsbasiertes Lernen, Blended Learning und Unternehmensgründung durchgeführt. Anschließend wurden die Studierenden durch Teilnahme an forschungs- sowie gründungsbezogenen Workshops (u.a. Design Thinking) für die Verprobungs-/Gründungsphase mobilisiert. In der aktuellen Phase 3 (konzeptionelle Entwicklung) wird ein generisches forschungsbezogenes Lehr-/Lernmodell entwickelt. Ausgehend von aktuellen Forschungsergebnissen soll dabei das Modell die drei Dimensionen Forschen, Unternehmensgründung und Blended Learning integrieren. In Phase 4 (Verprobung) soll das Modell zur Anwendung kommen. Hierbei werden forschungsbasiertes Lernen und Blended Learning-Ansätze an einem konkreten Beispiel, der Gründung (bzw. Planung) eines Unternehmens gemeinsam mit den WILLE-Probanden angewandt. In Phase 5, der (technischen) Implementierung, wird das theoretische Modell in eine technische Applikation überführt. In der abschließenden Phase 6 (Evaluation) erfolgt die Bewertung des Umsetzungserfolgs. Der vorliegende Beitrag fokussiert

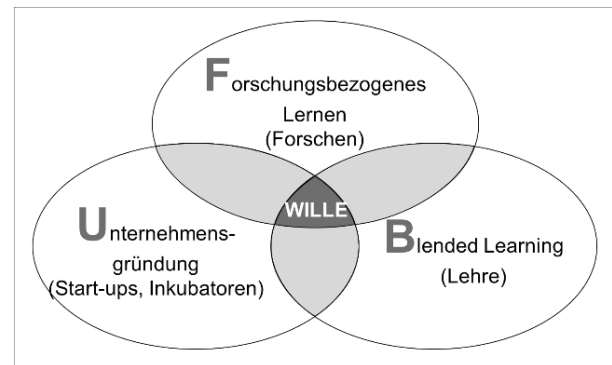


Abbildung 2: Das WILLE FUB-Modell

der Hochschule Heilbronn (HHN) eine Bestandsaufnahme zu den drei FUB-Dimensionen Forschungsbezogenes Lernen (Forschen), Unternehmensgründung (Startups, Inkubatoren) sowie Blended Learning (Lehre). Die Online-Umfrage wurde im Juni/Juli 2017 an allen vier Standorten der HHN durchgeführt. Der Befragungslink wurde per E-Mail über den Studierenden-Verteiler an alle Studierenden versandt (N=8329 Gesamt (Stand: WS 2015/16), 836 Teilnehmende, Teilnahmequote: 10 %).

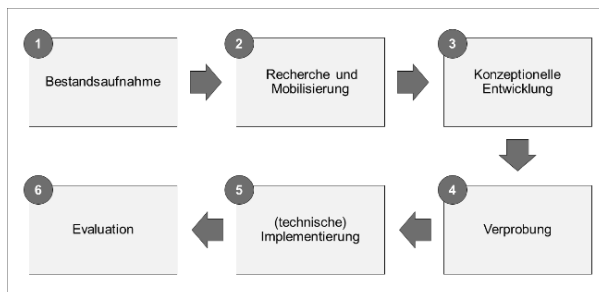


Abbildung 1: Methodisches Vorgehen

auf die Darstellung der ersten drei Projektphasen. Zudem wird ein Modell zur Diskussion gestellt, welches die Themen Gründung, forschungsbasiertes Lernen und Blended Learning integriert (s. Abb. 2).

4 Entwicklung eines forschungs-basierten Lernmodells

4.1 Empirie-basierte Bestandsaufnahme

In Phase 1 des WILLE-Projektes erfolgte im Rahmen einer Online-Umfrage unter allen Studierenden

Studienmotivation

Zunächst wurden die Studierenden nach der intrinsischen und extrinsischen Motivation für die Wahl eines Studiengangs gefragt. Intrinsische Motive umfassen Motive zur Selbstverwirklichung, interessante Berufstätigkeiten usw. Extrinsische Motive hingegen umfassen Motive bzgl. gehobener beruflicher Position und Verdienstmöglichkeiten. Zudem standen hierbei auch die Bedeutung des Wissenserwerbs bzgl. Selbständigkeit sowie die Arbeit in der angewandten Forschung und Entwicklung bei der Entscheidung für einen Studiengang im Fokus. Insbesondere intrinsische Motive werden bei der Studiengangwahl als wichtig erachtet (s. Abb. 3), also „... einen Beruf mit interessanten Inhalten auszuüben“ (Antwortoptionen wichtig/sehr wichtig zusammengefasst: 92 %), oder „... eine Stelle zu bekommen, in der ich aufgehen kann“ (Antwortoptionen wichtig/sehr wichtig zusammengefasst: 91 %). Aber auch extrinsische Motive spielen noch eine wichtige Rolle, d.h. Motive wie „... immer eine Stelle zu finden“ (82 %), „eine gut bezahlte Position zu bekommen“ (80 %) oder auch eine „... angesehene berufliche Position zu erhalten“ (77 %), siehe [1]. Deutlich geringer ausgeprägt ist die Motivation für Studierende, das notwendige Wissen

für eine Selbständigkeit zu erhalten (Antwortoptionen wichtig/sehr wichtig zusammengefasst: 39 %). Einen Einstieg in die angewandte Forschung und Entwicklung zu bekommen ist für 28 % (sehr) wichtig. Bei diesem Statement zeigen sich einige deutliche Unterschiede zwischen Studierenden in den verschiedenen Fakultäten. So ergibt sich für Studierende in der Fakultät Mechanik und Elektronik (n=77) eine Wichtigkeit von 76 % und bei der Studierenden der Fakultät Informatik (n=29) eine Wichtigkeit von 55 %. Bei Studierenden technischer Fächer besteht entsprechend eine höhere Bereitschaft, in der angewandten Forschung und Entwicklung zu arbeiten.

Motivation zur Wahl des Studiengangs

Ich habe mich bewusst für meinen Studiengang entschieden, weil ich mit einem Studienabschluss in meinem Studiengang...



Abbildung 3: Studienmotivation

Selbsteingeschätzte Fähigkeiten

Neben der Studienmotivation wurde in der Online-Umfrage nach der Selbsteinschätzung von studienrelevanten Fähigkeiten der Studierenden in den Themengebieten Unternehmensgründung und Forschungsorientierung gefragt (mit einer Skala von 1 „In sehr hohem Maße“ bis 5 „Gar nicht“, s. Abb. 4). Über die Fähigkeiten, „sich selbst und seinen Arbeitsprozess effektiv zu organisieren“ (Werte 1 „In sehr hohem Maße“ und 2 zusammengefasst, 70 %) sowie „fächerübergreifend zu denken“ (69 %), aber auch „wirtschaftlich zu denken und zu handeln“ (66 %) sowie „analytisch zu denken“ (62 %) verfügen die Studierenden laut Selbsteinschätzung in hohem Maße.

Die Fähigkeit, wirtschaftlich zu denken und zu handeln wird insbesondere von Studierenden der Fakultät

Management und Vertrieb mit 81 %, der Fakultät Wirtschaft und Verkehr (78 %) sowie der Fakultät International Business (71 %) hoch eingestuft.

Vergleichsweise niedrig hingegen werden die Fähigkeiten, „wissenschaftliche Methoden anzuwenden“ (42 %) sowie „eine Geschäftsidee zu entwickeln“ (31 %) von den Studierenden eingeschätzt.

Selbsteingeschätzte Fähigkeiten

In welchem Maße verfügen Sie Ihrer Einschätzung nach zum jetzigen Zeitpunkt Ihres Studiums über die folgenden Fähigkeiten?



Abbildung 4: Selbsteingeschätzte Fähigkeiten

Einstellungen zum Studium

Auch bei den Einstellungen zum Studium an der Hochschule Heilbronn standen einerseits die Forschungsorientierung und auch das Thema Unternehmensgründung/Selbständigkeit im Fokus. Hier zeigt sich ein starkes Interesse an einem praktisch orientierten Studium. So wird bei dem Item „Ich finde es wichtig, dass die Lehrenden an der Hochschule Heilbronn aus der Praxis kommen“ ein Zustimmungswert von 88 % erreicht (Antwortoptionen „stimme voll und ganz zu“ und „stimme zu“ zusammengefasst). Auch bei dem Statement „Im Studium interessiere ich mich vor allem für praxisorientierte Projekte“ ergibt sich ein Zustimmungswert von 82 %. Geringere Zustimmung, aber noch auf hohem Niveau von 70 % erreicht das Item „Ich finde es wichtig, dass die Lehrenden an der Hochschule Heilbronn Studierende in wissenschaftliche Projekte einbinden“. Auch die Zustimmung zur Aussage „Im Studium interessiere ich mich besonders für praxisbezogene Forschungsprojekte“ erreicht mit 65 % noch eine hohe Zustimmung.

Im Gegensatz dazu wird dem Item „Ich wende gerne

wissenschaftliche Methoden zur Lösung praktischer Probleme an“ nur zu 34 % zugestimmt. Auch „Ich habe großes Interesse am wissenschaftlichen Arbeiten „erzielt nur eine Zustimmung von 34 %. Die geringsten Zustimmungswerte von 25 % erreicht das Statement „Ich lese gerne wissenschaftliche Texte“. Dabei zeigen sich nur geringe Zusammenhänge zwischen diesen Items und der Anzahl der Semester.

Einstellung zum Studium – Forschungs-/Praxisorientierung
Welchen der folgenden Aussagen zu Ihrem Studium können Sie zustimmen bzw. nicht zustimmen?



Abbildung 5: Einstellungen zum Studium

Unternehmensgründung

Ein weiterer Themenkomplex in der empirischen Bestandsaufnahme war die Frage nach der Absicht einer Unternehmensgründung. Dazu wurde zunächst gefragt, ob sich Studierende der HHN mit den Themen Selbständigkeit und Gründung beschäftigt haben (s. Abb. 6). Etwa ein Viertel der Befragten gaben an, sich noch nie mit diesen Themen beschäftigt zu haben (24 %). Aber etwa die Hälfte gab an, sich zumindest ein wenig damit auseinandergesetzt zu haben (48 %). Eine regelmäßige Beschäftigung mit diesen Themen gaben 11 % und eine intensive Beschäftigung gaben 6 % an. Und 2 % der Befragten sind bereits selbständig. In einem weiteren Schritt wurden die Studierenden nach ihrer Gründungsabsicht gefragt. Hierbei geben 15 % der Befragten Studierenden an, sich sehr gut vorstellen zu können, ein Unternehmen zu gründen und selbständig zu arbeiten (s. Abb. 7). Dies stellt das Kernpotenzial für die Unternehmensgründung und Selbständigkeit dar. Weitere 19 % der Befragten geben an, sich dies zumindest eher vorstellen zu können. Das erweiterte Potenzial für eine Unternehmensgründung lässt sich

Beschäftigung mit Unternehmensgründung

Haben Sie sich schon einmal mit den Themen Selbständigkeit und Gründung beschäftigt?

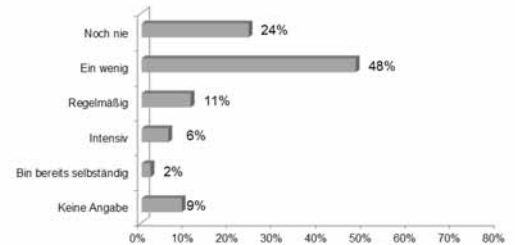


Abbildung 6: Beschäftigung mit Unternehmensgründung

somit auf 34 % beziffern. Dieses Potenzial ist innerhalb der verschiedenen Fakultäten unterschiedlich ausgeprägt (siehe Abb. 7 unten). Das größte Potenzial zeigt sich in der Fakultät International Business (37 %), die Fakultät Technik und Wirtschaft (19 %) sowie die Fakultät Wirtschaft und Verkehr (16 %). Das geringste Potenzial zeigt sich in den Fakultäten Informatik (6 %) und Technische Prozesse (3 %). Zudem

Unternehmensgründung

Können Sie sich vorstellen, ein Unternehmen zu gründen u. selbständig zu arbeiten?



Ergebnisse nach Fakultäten:
Befragte, die sich vorstellen können, ein Unternehmen zu gründen und selbständig zu arbeiten, n=283
(Zusammengefasst: „Ja, das kann ich mir sehr gut vorstellen“ und „Ja, das kann ich mir eher vorstellen“)

Gesamt	Mechanik u. Elektronik (TE)	Technische Prozesse (TP)	Informatik (IT)	Wirtschaft und Verkehr (WV)	International Business (IB)	Technik und Wirtschaft (TW)	Management und Vertrieb (MV)
34%	11%	3%	6%	16%	37%	19%	9%

Abbildung 7: Gründungsabsicht

ergeben sich einige interessante Zusammenhänge zwischen der Absicht, ein Unternehmen zu gründen sowie einigen Items aus den oben genannten Themenkomplexen zu Motivation zur Studiengangwahl (s. Abb. 3). So besteht zwischen der Motivation, das notwendige Wissen im Studiengang zu erhalten und der Absicht ein Unternehmen zu gründen eine

positive hohe Korrelation (Koeffizient nach Pearson) von .66 (Signifikanzniveau von .01). Auch die Einschätzung der eigenen Fähigkeit, eine Geschäftsidee zu entwickeln (s. Abb. 4) korreliert positiv mit der Selbsteinschätzung, eine Geschäftsidee entwickeln zu können (positive mittlere Korrelation nach Pearson von .36, Signifikanzniveau von .01).

Blended Learning

Als dritten Themenblock wurde in der Umfrage die Nutzung und Wichtigkeit von E-Learning-Plattformen ermittelt. 60 % der befragten Studierenden nutzen mindestens wöchentlich E-Learning-Plattformen wie z.B. Ilias für Lehrveranstaltungen. Auch hochschulexterne Online-Plattformen wie YouTube werden von 49 % der Studierenden mindestens wöchentlich genutzt (s. Abb. 8).

Nutzung von E-Learning-Plattformen

Wie häufig nutzen Sie E-Learning-Plattformen (wie z.B. Ilias) oder hochschulexterne Online-Plattformen (wie z.B. Youtube.com) für Ihre Lehrveranstaltungen?

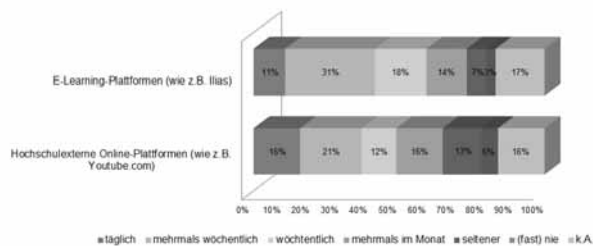


Abbildung 8: Nutzung von E-Learning-Plattformen

Ausgeprägter als die Nutzungsfrequenz von E-Learning- bzw. Online-Plattformen wird aber deren Wichtigkeit eingestuft (s. Abb. 9). 67 % der Befragten geben an, dass E-Learning-Plattformen wie Ilias für Lehrveranstaltungen wichtig oder sehr wichtig sind. Und immerhin noch 54 % geben an, dass hochschulexterne Plattformen für Lehrveranstaltungen (sehr) wichtig sind.

4.2 Recherche und Mobilisierung

Im Rahmen eines Semesterseminars, der Lernphase 1, wurde das FUB-Modell pilotiert. Hierfür wurden drei studentische Gruppen formiert. Die Proban-

Wichtigkeit von E-Learning-Plattformen

Und wie wichtig sind Ihnen E-Learning-Plattformen (wie z.B. Ilias) oder hochschulexterne Online-Plattformen (wie z.B. Youtube.com) für Ihre Lehrveranstaltungen?

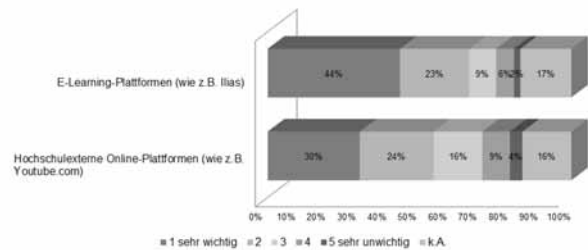


Abbildung 9: Wichtigkeit von E-Learning-Plattformen

dennguppe aus Bachelor-Studierenden soll bereits in der Frühphase ihres Studiums mit Qualifizierung und Wissensvermittlung für den Gründergedanken sensibilisiert werden. In der Phase geht es zum einen darum, didaktische Lehrformate weiterzuentwickeln, um wissenschaftliches Arbeiten und zum zweiten darum betriebswirtschaftlichen Fragestellungen mit Blick auf potenzielle Gründungen für die Studierenden konkret erfahrbar zu machen.

In einem ersten Schritt wurden sie in das Forschungsprojekt instruiert, für das Thema sensibilisiert und schließlich mit konkreten Frage-/Problemstellungen die drei FUB-Dimensionen betreffend konfrontiert, die sie in der ersten Semesterhälfte zu lösen hatten. Jede Gruppe hat sich dabei auf eine Dimension ihrer Wahl spezialisiert. Im Fokus stand die Dimension „Forschen“, bei der die Studierenden mittels einer fundierten Literaturrecherche Wissen in den drei Dimensionen erwerben sollten.

Bei der Verzahnung von Forschung und Lehre ist das übergeordnete Ziel, die Studierenden nach einem vorangegangenen Wissenserwerb und der Ausbildung von Urteilsfähigkeit mit Forschungsfragestellungen zu konfrontieren, damit sich ein kritisch reflektierendes akademisches Denken bei den Studierenden ausbilden kann. Professor/innen und wiss. Mitarbeiter/innen leiten Studierende in der Beteiligungsrunde an. Durch die Projekt- bzw. Teamarbeit erwerben Studierende in der Verzahnung von Forschung und Lehre ein ganzheitliches Wissen, das die eigene wissenschaftliche Karriere befördert und somit Erfahrungen über Lehrbuchwissen hinaus möglich macht. Um dieses Ziel zu erreichen, müssen

Studierende, Mitarbeiter und Forscher Teil einer ganzheitlichen und abgestimmten Weiterentwicklungsstrategie sein.

In Übereinstimmung mit den Studienergebnissen zeigte sich, dass Studierende großes Potenzial bezüglich der Dimension Forschen haben. Aufgrund dieses Mankos wurde Studierenden zunächst einmal durch intensive Schulungsmaßnahmen die Durchführung eines Forschungsprozesses vermittelt. Dieses iterative Vorgehen, welches im Sinne eines Trial and Error Verfahrens exerziert wurde, siehe [5], wird auch im FUB-Modell aufgegriffen (siehe Abschn. 4.3). In einem nächsten Schritt wurde gemeinsam mit den Studierenden ein ihren Vorstellungen entsprechender WILLE-Lern-/Lehrraum aufgesetzt. Im WILLE-Raum können Lern- und Lehrkonzepte über das Inter-/Intranet in Verbindung mit ‚klassischen‘ Lernmethoden und -medien optimal genutzt werden. Allein das strukturelle Aufbaukonzept des Raumes wirkte sich stark auf die Lernbereitschaft und –motivation sowie aktive Teilnahme der Studierenden am Unterricht aus.

Der WILLE-Raum wurde noch in Lernphase 1 für Workshops zur Mobilisierung der Studierenden im Hinblick auf Unternehmensgründung genutzt. Die Workshops dienten der Praktizierung von Kreativitätstechniken wie bspw. Design Thinking, um erforderliches Wissen sowie Ideen für potenzielle Gründungsvorhaben zu vermitteln bzw. zusammenzutragen. In der anschließenden Lern-/Lehr- und Orientierungsphase wird das FUB-Modell von Lehrenden entwickelt und danach verprobt.

4.3 Das FUB-Modell

Die Schwierigkeit bei dieser Diskussion besteht darin, ein geeignetes Konzept zu entwickeln, das sowohl den Lehrenden als auch den Lernenden ausreichend in den Themenkontext integriert. Letztlich geht es aber um neue Lehrformate für forschungsbasiertes Lernen im Gründungskontext.

Rückblickend kann gesagt werden, dass Lehrinhalte an Hochschulen schon immer geprägt waren von Forschungsergebnissen und der Teilhabe von Studierenden an Forschungsaktivitäten. Beide Prozesse, das Forschen und das Lernen, haben viele Gemeinsamkeiten und sind miteinander vergleichbar, siehe [6] und [9]. Für Forschende als auch Studierende ist das Lernen der Schlüssel zu

noch unbekanntem Wissen. Ausgangspunkt bei beiden Prozessen sind Fragestellungen, die mit Handlungs- oder Erkenntnisproblematiken verbunden sind. Folglich zielen beide auf die Erweiterung der Handlungs-, Begründungs- oder Erklärungsfähigkeit ab. Dabei unterscheiden sich beide Prozesse durch ihre Reichweite: Lernprozesse konzentrieren sich auf das Individuum und die Erweiterung ihrer Handlungsfähigkeit. Hingegen konzentrieren sich Forschungsprozesse auf die Erweiterung der kollektiven Handlungsfähigkeiten mittels verbreedeter Verfügbarkeit des Wissens für die Gesellschaft, siehe [8].

Im Gegensatz zur Forschung spielt das Thema Unternehmensgründung an Hochschulen nur eine untergeordnete Rolle und ist vom Ziel „Entrepreneurial University“ weit entfernt, auch wenn bspw. die TU München oder die Universität Mannheim sich sehr bemühen, siehe [2], [7] und [10]. Insbesondere reicht das bisherige Einbinden des Themas in die Curricula nicht aus und ganze Programme zur akademischen Gründungsausbildung fehlen bzw. es besteht hier noch ein erheblicher Verbesserungsbedarf, siehe [10] und [14]. Und das, obwohl Lernen im Prozess der Unternehmensgründung das Fundament bildet und integraler Teil ist, siehe [11]. In seinem Model (s. Abb. 10) unterteilt Rae [11] Entrepreneurial Learning in persönliche und soziale Emergenz, kooperative Unternehmung und kontextabhängiges Lernen. In der persönlichen und sozialen Emergenz geht es vor allem um die Entwicklung einer unternehmerischen Identität. Diese ist sehr stark von den bisherigen Lebenserfahrungen sowie vom familiären Umfeld geprägt. Neben Bildung, beruflicher Entwicklung und zwischenmenschlicher Beziehungen, beeinflusst die Entwicklung der Selbstwahrnehmung und die eigenen Ziele für die Zukunft, die Entwicklung der unternehmerischen Identität. Die Idee, die hinter dem Konzept kooperative Unternehmung steckt, ist, dass durch kooperative Beziehungen mit anderen ein Lernprozess angestoßen wird. An diesem interaktiven Austauschprozess können sich sowohl Mitarbeiter als auch Kunden oder Investoren beteiligen. Ziel für den Unternehmensgründer ist eine gemeinsame Vision, an die alle Beteiligten glauben und daran arbeiten können. Das kontextabhängige Lernen zielt auf das Lernen innerhalb von Netzwerken (u.a. mit Stakeholder etc.) ab. So können (ähnliche) persönliche Erfahrungen ausgetauscht und analysiert werden, aber auch intuitives Handeln geübt und die Fähigkeit, neue Möglichkeiten zu erkennen und Neues zu kreieren, verbessert werden.



Abbildung 10: Tiradic model of entrepreneurial learning [11]

Grundvoraussetzung für forschungsbasiertes und unternehmerisches Lernen ist die aktive und selbstständige Teilhabe der lernenden Person, siehe [11] und [13]. Aus diesem Grund folgt das Projekt dem Verständnis der subjektwissenschaftlichen Lerntheorie von [5].

Bei der Entwicklung eines didaktischen Designs greifen wir auf bestehende Modelle zurück und berücksichtigen neue Lehrmethoden mit Einsatz von E-Learning und Blended Learning.

Für Lehre im Format der Forschung sind nicht immer Lehrforschungsprojekte oder eine großflächige Einbindung der Studierenden in reale Forschungsprojekte notwendig, siehe [9]. Forschungsbasierte Lehre beginnt schon durch das Explizieren und damit Nachvollziehbar-Machen von Handlungsproblematiken, Fragestellungen und Erkenntnisinteressen sowie deren Zusammenhänge. Im Kontext von forschungsbasierter Lehre stellt Ludwig das für unser Projekt geeignete Format, nämlich den Typ „Forschungsprozess“, näher vor (s. Abb. 11). In diesem Typus finden Studierende nicht nur die Möglichkeit, sich mit ihren eigenen Lern- und Studieninteressen auseinander zu setzen und zu reflektieren, sondern auch aktiv mit der Gestaltung des Forschungsprozesses zu beschäftigen. Beispielhaft bieten sich dafür abgeschlossene Forschungsprojekte oder gemeinsam in einer Lehrveranstaltung entwickelte Lehrforschungsprojekte an. Die Aufgaben der Lehrenden bei der forschungsbasierten Lehre besteht darin, die Ideen der Studierenden aufzugreifen, die Machbarkeit der Forschungsfrage im Blick zu behalten und die Studierenden im Forschungsprozess zu beraten und zu unterstützen, siehe [13]. Der Einsatz von digitalen Medien in der Hochschullehre

schreitet immer weiter voran. Wannapiroon hat unter Berücksichtigung von forschungsbasiertem Lernen und Blended Learning ein Modell entwickelt (s. Abb. 12). Ziel ist es, mit diesem Modell, Studierende verstärkt kritisches und reflexives Denken, Problemlösungsansätze sowie Forschungskompetenzen zu vermitteln.



Abbildung 11: Typ Forschungsprozess [9]

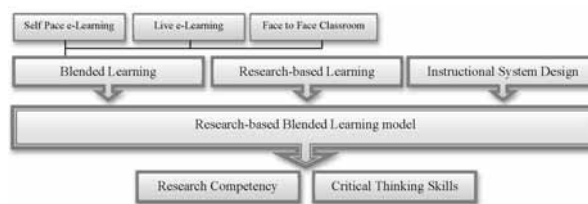


Abbildung 12: Conceptual framework of the Research-based Blended Learning model [16]

Blended Learning wird im Modell in selbst bestimmtes E-Lernen, Live E-Lernen und Präsenzveranstaltungen unterteilt. Neben dem Blended Learning sind noch forschungsbasiertes Lernen und ein Leitfaden für Lehrende Bestandteil des Modells.

4.4 Verprobung

In der zweiten Lehrphase wird der Fokus noch stärker auf dem unternehmerischen Lernen liegen. Begleitet wird diese Phase durch Gastvorträge und Themenworkshops. In den Vorträgen und Workshops werden praxisrelevante Schlüsselqualifikationen vermittelt. Erfahrene Partner, wie z. B. die IHK oder die Universität Mannheim, sollen dafür gewonnen werden.

Der Einstieg in das unternehmerische Lernen wird durch ein cloudbasiertes Start-up-Simulationsspiel erfolgen. Studierende aus unterschiedlichen Fakultäten (Technik, Wirtschaft und Informatik) werden in Gruppen eingeteilt und müssen sich mit der Existenzgrün-

derung eines Unternehmens beschäftigen. Folgende Lernziele lassen sich zusammenfassen [15]:

- Vermittlung von BWL-Grundlagen
- Erkennen und Berücksichtigen von Rahmenbedingungen für wirtschaftlichen Erfolg in einem dynamischen Wettbewerbsumfeld
- Auswählen und Umsetzen von Strategien zur Erreichung von Zielen
- Evaluieren von Unternehmensdaten
- Effektives Treffen von Entscheidungen im Team

Ab dem Sommersemester 2018 sollen Lehrende und Studierende gemeinsam an einem App-Game arbeiten. Dabei sollen die Erfahrungen und Erkenntnisse aus beiden Lehrphasen einfließen. Die neue App soll den Anforderungen der drei Hauptkategorien von Schwan entsprechen (Schwan fasste die 13 Prinzipien von Gee (2003) in drei Hauptkategorien zusammen)[12]. Die Hauptkategorien beschreiben was „gute“ Spiele kennzeichnen:

- **Kategorie 1:** Lernenden Handlungsspielräume eröffnen
- **Kategorie 2:** Prinzipien, die Kompetenzen zum Lösen von Problemen fördern
- **Kategorie 3:** Prinzipien, die das Verständnis fördern

5 Fazit

Der vorliegende Beitrag zeigt, dass bei der Verzahnung von Forschung und Lehre durch den Einsatz forschungsbasierter und innovativer Lehrmethoden Studierende zum selbstständigen Lernen und Gründen motiviert werden können und so das Potenzial an Unternehmensgründungen im Studienbetrieb langfristig gesteigert werden kann. Das FUB-Modell liefert ein forschungsbasiertes Lern- und Lehrkonzept zur Förderung von Gründungsinitiativen an Hochschulen. Es gibt Lösungsvorschläge an die Hand, wie Lehrende Lernende unterstützen und dazu befähigen können, eine eigene Gründungsidee zu entwickeln und umzusetzen. Der Fokus liegt auf dem forschungsbasierten Identifikationsprozess von Handlungs- und Erkenntnisproblematiken. Ohne eine Motivation gibt es keinen Grund für den Lernenden seine Handlungsfähigkeit zu erweitern. Sowohl die empirischen Studienergebnisse als auch die Lernphase 1 zeigen, dass es sogenannter Trigger bedarf, damit Lernende überhaupt auf die Idee kommen, zu gründen. Die erarbeiteten Erkenntnisse sowie die damit einhergehen-

den forschungsbasierten Lern- und Lehrmodelle werden direkten und fortlaufenden Eingang in die Lehre finden.

References

- [1] M. Albert, K. Hurrelmann, G. Henzel. *Jugend 2015. 17. Shell Jugendstudie*. Fischer, Frankfurt, 2015.
- [2] Bundesministerium für Wirtschaft und Energie (BMWi). *Das ist Exist 2016*. Druck- und Verlagshaus Zarbock, Frankfurt, 2016.
- [3] J. Gee. *What video games have to teach us about learning and literacy*. Palgrave Macmillan, Basingstoke, 2003.
- [4] S. Golla, H. Bergmann. *Unternehmertum an Hochschulen in Deutschland. Ergebnisse des Global University Entrepreneurial Spirit Students' Survey*. St. Gallen/Fulda, 2016.
- [5] K. Holzkamp. *Lernen. Subjektwissenschaftliche Grundlegung*. Campus, Frankfurt, New York, 1993.
- [6] L. Huber. *Warum Forschendes Lernen nötig und möglich ist*. In: L. Huber, J. Hellmer & F. Schneider (Hg.). *Forschendes Lernen im Studium. Aktuelle Konzepte und Erfahrungen*. Universitätsverlag Weber, Bielefeld, 2009.
- [7] M. Kulicke. *15 Jahre EXIST „Existenzgründungen aus der Wissenschaft“*. *Entwicklung des Förderprogramms von 1998 bis 2013*. Arbeitspapier der wissenschaftlichen Begleitforschung zu „EXIST – Existenzgründungen aus der Wissenschaft“. Fraunhofer-Institut für System- und Innovationsforschung, Karlsruhe, 2014.
- [8] J. Ludwig. *Forschungsbasierte Lehre als Lehre im Format der Forschung*. *Brandenburger Beiträge zur Hochschuldidaktik*, Heft 3, p. 7-16, 2011.
- [9] J. Ludwig. *Lehre im Format der Forschung*. *Brandenburger Beiträge zur Hochschuldidaktik*, Heft 7, p. 7-19, 2014.
- [10] N. Nieuwenhuizen, D. Groenewald, J. Davids, L. Janse van Rensburg, C. Schachtebeck. *Best*

practice in entrepreneurship education. Problem and Perspectives in Management (open-access), Vol. 14 No. 3, p. 528-536, 2016.

- [11] D. Rae. *Entrepreneurial learning: a narrative-based conceptual model*. *Journal of Small Business and Enterprise Development*, Vol. 12 No. 3, p. 323-335, 2005.
- [12] S. Schwan. *Game Based Learning – Computerspiele in der Hochschullehre*. Veröffentlicht auf <http://e-teaching.org>, http://e-teaching.org/didaktik/konzeption/methoden/lernspiele/game_based_learning/gamebasedlearning.pdf, 2006.
- [13] M. Sonntag, J. Rueß, C. Ebert, K. Friederici & W. Deicke. *Forschendes Lernen im Seminar. Ein Leitfaden für Lehrende*. Humboldt-Universität zu Berlin. http://researchgate.net/publication/308047837_Forschendes_Lernen_im_Seminar_Ein_Leitfaden_fur_Lehrende (Abgerufen am 14.04.2017), 2016.
- [14] C. Schultz, D. Mietzner. *Gründungsausbildung an Hochschulen in Deutschland*. Wissenschaftliche Beiträge, Technische Hochschule Wildau, 18, p. 103-106, 2014.
- [15] TOPSIM-Startup. *Kurzbeschreibung des Onlineplanspiels Topsisim-Startup*. TATA Interactive Systems GmbH, Tübingen.
- [16] P. Wannapiroon. *Development Of Research-Based Blended Learning Model To Enhance Graduate Students' Research Competency And Critical Thinking Skills*. *Procedia – Social and Behavioral Sciences*, 136, p. 486-490, 2014.

Dynamische Simulation von Mini-Fahrzeugen mit Modern C++, Boost-Odeint und Robot-Operating-System

Niclas Heining, Uwe Ingelfinger, Ansgar Meroth, Frank Tränkle¹

Automotive Systems Engineering, Hochschule Heilbronn

Max-Planck-Str. 39, D-74081 Heilbronn

¹*frank.traenkle@hs-heilbronn.de*

Mini-Auto-Drive (MAD) ist ein neues Miniplant im Maßstab 1:24 zur Entwicklung von Lokalisierungs-, Bahnplanungs-, Bahnregelungs- und Sicherheitsfunktionen für hoch- und vollautomatisiertes Fahren. Die Entwicklung der Funktionen erfolgt in Modern C++14 oder mit MATLAB/Simulink/Stateflow. Im Falle von C++ wird die Middleware MODBAS-Safe zur Implementierung von Echtzeit-Tasks, nachrichtenbasierter Kommunikation und zur Überwachung des Systems hinsichtlich der funktionalen Sicherheit eingesetzt [1]. Die Simulations- und Experimentierumgebung von MAD basiert auf Robot-Operating-System (ROS) und der Game-Engine Unreal. Sie ermöglicht die Durchgängigkeit von Embedded-Software-Tests von Model-in-the-Loop über Software-in-the-Loop bis hin zum realen Fahrversuch [2].

Für diese Tests wird ein dynamisches, nichtlineares Einspurmodell inklusive Magic-Formula-Reifenmodell der verwendeten Mini-Z-Fahrzeuge entwickelt. Die Parameteridentifikation des Magic-Formula-Modells erfolgt anhand stationärer Kreisfahrten bei niedrigen und hohen Geschwindigkeiten mithilfe von Least-Squares-Verfahren unter Verwendung linearer und nichtlinearer Regressionsmodelle. Weiterhin stellt dieser Beitrag eine neuartige Implementierung dieses Fahrdynamikmodells als ROS-Knoten unter Einsatz von Modern C++ und Boost-Odeint vor.

1 Einleitung

Mini-Auto-Drive ist ein neues Miniplant zur Entwicklung von Softwarefunktionen für hoch- und vollautomatisiertes Fahren vergleichbar mit LIU-Racetrack [3] und ORCA [4]. Softwarefunktionen fürs automatisierte Fahren werden zunächst auf diesem Miniplant in Betrieb genommen, bevor sie auf dem Versuchsträger Audi A3etron der Hochschule Heilbronn appliziert werden. Bei MAD kommen Fahrzeuge der Mini-Z-Rennserie im Maßstab 1:24 im Gegensatz zum 1:43-Maßstab bei ORCA zum Einsatz.

Abschnitt 2 stellt die System- und Softwarearchitektur von MAD vor. Das System besteht aus einer konfigurierbaren Fahrbahn, einer Computervision und einem Echtzeit-Linux-Rechner. Die Softwarearchitektur ermöglicht die Entwicklung von Funktionen entweder mit MATLAB/Simulink/Stateflow oder in Modern C++. Die Steuerung und Visualisierung der realen oder simulierten Fahrzeuge erfolgt mit Robot-Operating-System (ROS) und der Game-Engine Unreal (siehe Abschnitt 6). Dabei können reale und si-

mulierte Fahrzeuge gleichzeitig betrieben und in einer virtuellen 3D-Umgebung visualisiert werden, wodurch eine Realisierung komplexer Fahrsituationen in Augmented-Reality ermöglicht wird.

Für die Simulation der Fahrzeuge und die Durchführung von Model-in-the-Loop- (MiL) und Software-in-the-Loop-Tests (SiL) wird ein genaues, realitätsnahes Fahrdynamikmodell benötigt. Dieses Modell wird in Closed-Loop mit den Softwarefunktionen des automatisierten Fahrens in der MAD-Experimentier- und Simulationsumgebung simuliert. In Abschnitt 3 wird dieses Fahrdynamikmodell auf Basis des Modells aus [4] hergeleitet. Entscheidend für die Modellgüte ist die Parametrierung der Lenkwinkelkennlinie und Pacejkas Magic-Formula-Reifenmodell [5]. Zur Parameteridentifikation werden die Position, der Gierwinkel, die Geschwindigkeit und der Schwimmwinkel des Fahrzeuges durch die Computervision von MAD erfasst. Die Schätzung der Parameter erfolgt mithilfe von Least-Squares-Verfahren unter Verwendung linearer und nichtlinearer Regressionsmodelle im Abschnitt 4.

Als weiteres wesentliches Ergebnis stellt Abschnitt 5 eine neuartige Implementierung des Fahrdynamikmodells als ROS-Knoten unter Einsatz von Modern C++ und Boost-Odeint vor. Das beschriebene Vorgehen kann allgemein zur Implementierung und Simulation nichtlinearer, dynamischer Zustandsraummodelle in expliziter Form verwendet werden.

2 Miniplant

Die Fahrzeuge der Mini-Z-Rennserie fahren auf einer horizontalen, ebenen Fahrbahnfläche der Größe 3x2 m. Ein Linux-PC steuert die Fahrzeuge über eine 2,4 GHz-Funkfernsteuerung. Die Online-Lokalisierung der Fahrzeuge erfolgt durch eine Infrarot-Kamera, die in 3 m Höhe mittig über der Fahrbahnfläche montiert ist. Zur Identifikation sind auf der Fahrzeugkarosserie Infrarot-Marker geklebt.

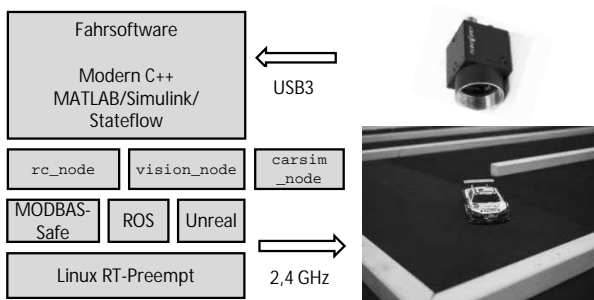


Abbildung 1: MAD-System- und Softwarekomponenten

MAD unterstützt den simultanen Betrieb von realen und simulierten Fahrzeugen. Dies wird durch den Einsatz von Softwarekomponenten ermöglicht, die einheitliche Schnittstellen aufweisen. Das Betriebssystem des Linux-PCs ist ein Ubuntu Linux und manuell installiertem Linux-Kernel mit RT-Preempt-Patch. Als Mittelschicht-Plattformen für die Fahrsoftware werden Robot-Operating-System (ROS) und MODBAS-Safe [1] eingesetzt. MODBAS-Safe verwendet u. a. POSIX-Echtzeit-Funktionen. Reale und simulierte Fahrzeuge werden in einer 3D-Umgebung mit der Game-Engine Unreal online visualisiert. Die System-Steuerung, die Kalibrierung von Softwareparametern und die Aufzeichnung der Fahrzeugsignale erfolgt mit der Experimentierumgebung ROS-rqt.

Die Fahrsoftware jedes einzelnen Fahrzeuges, egal ob real oder simulativ, ist in einem Ebenenmodell strukturiert mit den Komponenten: Master-Control, Navi-

gation, Path-Planning, Path- und Speed-Control, Perception und Localization. Alle Ebenen werden von der Komponente Supervision überwacht, die Fehlerreaktionen auslösen kann. Handelt es sich bei dem Fahrzeug um ein reales Fahrzeug, greift die Fahrsoftware auf die Softwarekomponenten `rc_node` und `vision_node` zur Funkfernsteuerung und Computervision zu. Im Falle eines simulierten Fahrzeuges werden diese beiden Softwarekomponenten durch die Komponente `carsim_node` ersetzt, die das Fahrdynamikmodell aus Abschnitt 3 simuliert und dessen Implementierung in Abschnitt 5 beschrieben wird.

Die Ebenen der Fahrsoftware können flexibel mit Modern C++ als ROS-Knoten oder mit MODBAS-Safe implementiert werden. Zur modellbasierten Entwicklung der ROS-Knoten wird alternativ MATLAB/Simulink/Stateflow verwendet. Wegen der einheitlichen Schnittstellen ist eine flexible, rapide Entwicklung verschiedener Algorithmen fürs automatisierte Fahren möglich. Einzelne Fahrzeuge können mit unterschiedlichen Algorithmen ausgestattet werden. Insbesondere durch die Überwachungsfunktionen von MODBAS-Safe und die Experimentierumgebungen von ROS und MATLAB/Simulink wird ein Performance-Vergleich der Algorithmen ermöglicht. So können verschiedene Algorithmen gegeneinander Rennen fahren, sowohl real als auch simulativ oder gemischt.

3 Fahrdynamikmodell

Alle MAD-Fahrzeuge sind identisch aufgebaut mit Vorderachslenkung und Hinterachsantrieben. Das Fahrdynamikmodell basiert auf dem in [4] beschriebenen Einspurmodell für die Minifahrzeug-Plattform ORCA, wobei ebenfalls das Magic-Formula-Reifenmodell [5] verwendet wird. In Ergänzung zu [4] werden Totzeiten eingeführt für die Ansteuerung der Lenkung und des Antriebsmotors.

3.1 Einspurmodell

Dem Einspurmodell liegen folgende Modellannahmen zugrunde:

- Das Fahrzeug wird als starrer Körper angenommen, dessen Schwerpunkt sich auf der Fahrzeuglängsachse befindet.
- Das Fahrzeug bewegt sich auf einer horizontalen Ebene.

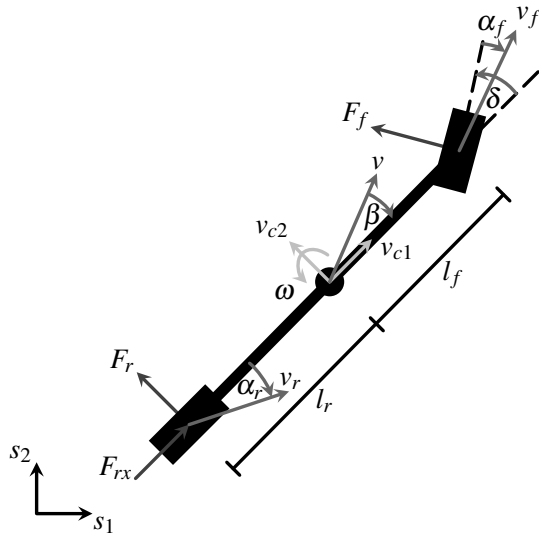


Abbildung 2: Einspurmodell

- Hub-, Wank- und Nickbewegungen werden vernachlässigt.
- Die Räder beider Achsen sind zusammengefasst.
- Die Längsdynamik ist eine totzeitbehaftete PT₁-Dynamik.

Der Zustandsvektor \boldsymbol{x} für das in Abbildung 2 dargestellte Ersatzschaltbild enthält folgende Elemente: Längsgeschwindigkeit $x_1 = v_{c1}$, kartesische, ortsfeste Koordinaten $x_2 = s_1$ und $x_3 = s_2$ des Schwerpunktes, Gierwinkel $x_4 = \psi$, Gierwinkelgeschwindigkeit $x_5 = \dot{\psi}$, Quergeschwindigkeit $x_6 = v_{c2}$.

Die normierten Stellsignale u_1 und $u_2 \in [-1; 1]$ für Motor und Lenkung werden über die 2,4 GHz-Funkfernsteuerung an das Fahrzeug übertragen. Beide Stellsignale weisen eine Totzeit T_t auf.

In Abhängigkeit der wirkenden Reifenlängs- und querkräfte wird das Fahrdynamikmodell als nichtlineares, dynamisches Zustandsraummodell in expliziter Form für Zeit $t > 0$ formuliert:

$$\dot{x}_1 = \dot{v}_{c1} = \frac{1}{m} (-F_f \sin \delta + F_{rx} + m v_{c2} \omega) \quad (1)$$

$$\dot{x}_2 = \dot{s}_1 = v_{c1} \cos \psi - v_{c2} \sin \psi \quad (2)$$

$$\dot{x}_3 = \dot{s}_2 = v_{c1} \sin \psi + v_{c2} \cos \psi \quad (3)$$

$$\dot{x}_4 = \dot{\psi} = \omega \quad (4)$$

$$\dot{x}_5 = \dot{\omega} = \frac{1}{J} (F_f l_f \cos \delta - F_r l_r) \quad (5)$$

$$\dot{x}_6 = \dot{v}_{c2} = \frac{1}{m} (F_f \cos \delta + F_r - m v_{c1} \omega) \quad (6)$$

Gleichung (1) ist der Impulserhaltungssatz in Längsrichtung, (2) und (3) transformieren die Fahrzeuggeschwindigkeit vom körperfesten ins ortsfeste Koordinatensystem, (5) definiert die Gierwinkelgeschwindigkeit, (4) ist der Drallerhaltungssatz bzgl. des Schwerpunktes, (6) ist der Impulserhaltungssatz in Querrichtung. Für den Lenkwinkel δ wird eine geradenförmige Kennlinie in Abhängigkeit des zeitverzögerten Stellsignals u_2 angenommen:

$$\delta(t) = \delta_m u_2(t - T_t) + \delta_b \quad (7)$$

Die Querkräfte F_f und F_r des Vorder- und des Hinterrads werden durch das Magic-Formula-Modell in Abschnitt 3.2 beschrieben.

Die Längskraft F_{rx} ergibt sich aus der Längsdynamik des Fahrzeuges, für welche eine totzeitbehaftete PT₁-Dynamik angenommen wird. Deren Parameter k , T und T_t werden durch Sprung- und Rampenantworten im Fahrversuch identifiziert. Die Geschwindigkeit v_r des Fahrzeuges bei Geradeausfahrt wird demnach durch folgende Differentialgleichung beschrieben:

$$\dot{v}_r(t) = -\frac{1}{T} v_r(t) + \frac{k}{T} u_1(t - T_t) \quad (8)$$

Daraus ergibt sich die folgende Beziehung für die Längskraft F_{rx} :

$$F_{rx}(t) = m \dot{v}_r(t) = m \left[-\frac{1}{T} v_{c1}(t) + \frac{k}{T} u_1(t - T_t) \right] \quad (9)$$

3.2 Magic-Formula-Reifenmodell

Das semiempirische Magic-Formula-Reifenmodell definiert die Reifenkräfte $F_{f,r}$ nichtlinear in Abhängigkeit der in Abbildung 2 dargestellten Schräglaufwinkel $\alpha_{f,r}$ [5]:

$$F_{f,r} = D_{f,r} \sin \left\{ C_{f,r} \arctan \left[B_{f,r} (1 - E_{f,r}) \alpha_{f,r} \right] + E_{f,r} \arctan (B_{f,r} \alpha_{f,r}) \right\} \quad (10)$$

Für die Parameteridentifikation des Reifenmodells in Abschnitt 4 werden verschwindende Biegefaktoren $E_{f,r} = 0$ angenommen, wodurch sich diese Gleichungen vereinfachen zu:

$$F_{f,r} = D_{f,r} \sin \left[C_{f,r} \arctan (B_{f,r} \alpha_{f,r}) \right] \quad (11)$$

Die Schräglaufwinkel $\alpha_{f,r}$ lassen sich nach Abbildung 2 wie folgt aus den Zustandsvariablen und dem Lenkwinkel δ berechnen:

$$\alpha_f = -\arctan \left(\frac{v_{c2} + l_f \dot{\psi}}{v_{c1}} \right) + \delta \quad (12)$$

$$\alpha_r = -\arctan\left(\frac{v_{c2} - l_r \dot{\psi}}{v_{c1}}\right) \quad (13)$$

4 Modellparameter

Das Fahrdynamikmodell aus Abschnitt 3 ist allgemein für Fahrzeuge mit Vorderachslenkung und Hinterachsantrieb gültig. Durch die Parameteridentifikation wird dieses Modell an die Fahrdynamik der Mini-Z-Fahrzeuge angepasst.

4.1 Lenkwinkel

Zur Bestimmung der Parameter δ_m und δ_b der Lenkwinkel-Kennlinie (7) werden stationäre Kreisfahrten ohne Schlupf mit unterschiedlichen, konstanten Stellsignalen u_2 bei einer niedrigen Geschwindigkeit von 0,5 m/s aufgezeichnet.

Über die Online-Lokalisierung durch die Computer-Vision wird die Fahrzeugposition $(s_{i1}, s_{i2})^T$ sowie der Gierwinkel ψ_i des Fahrzeuges mit einer Abtastzeit von 20 ms aufgezeichnet. Weiterhin berechnet die Online-Lokalisierung durch ein Erweitertes Kalman-Filter die Geschwindigkeit v_i und den Schwimmwinkel β_i . Bei einer stationären Kreisfahrt mit konstantem Stellsignal u_2 erfüllt jeder Messpunkt $i = 1, \dots, n$ die folgende Kreisgleichung mit noch unbekanntem Kreisradius r und Kreismittelpunkt $(s_{c1}, s_{c2})^T$:

$$(s_{i1} - s_{c1})^2 + (s_{i2} - s_{c2})^2 = r^2 \quad ; i = 1, \dots, n \quad (14)$$

Durch Umstellung von (14) ergibt sich das folgende Regressionsmodell für r , s_{c1} und s_{c2} :

$$\begin{aligned} & \begin{bmatrix} -2s_{i1} & -2s_{i2} & 1 \end{bmatrix} \cdot \begin{bmatrix} s_{c1} \\ s_{c2} \\ -r^2 + s_{c1}^2 + s_{c2}^2 \end{bmatrix} \\ & = -(s_{i1}^2 + s_{i2}^2) \quad ; i = 1, \dots, n \end{aligned} \quad (15)$$

Dieses überbestimmte lineare Gleichungssystem wird in MATLAB mit `mldivide` durch Least-Squares-Verfahren gelöst. Der Lenkwinkel δ berechnet sich bei langsamen stationären Kreisfahrten nach [6] dann aus dem durch diese Regression ermittelten Kreisradius r als:

$$\delta = \pm \arctan \sqrt{\frac{l^2}{r^2 - l_r^2}} \quad (16)$$

Das Stellsignal u_2 der Lenkung wird stufenweise von $\pm 0,3$ bis $\pm 1,0$ erhöht. Für jede einzelne Kreisfahrt

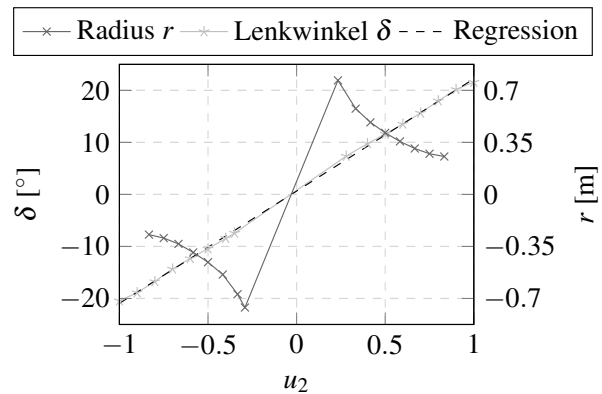


Abbildung 3: Ermittlung des Lenkwinkel-Parameter

wird der Lenkwinkel mithilfe der obigen Regression und Gleichung (16) berechnet. Aus den gewonnenen Informationen in Abbildung 3 werden durch eine lineare Regression die Parameter $\delta_m = 21,58^\circ$ und $\delta_b = 0,64^\circ$ der Lenkwinkel-Kennlinie (7) ermittelt. Die vorhandene Achsverschiebung um δ_b resultiert aus einem Linksdrall aufgrund des Lenkungs-spiels des Mini-Z-Fahrzeuges.

4.2 Magic-Formula-Parameter

Zur Bestimmung der Parameter der Reifenkennlinien (11) dienen ebenfalls stationäre Kreisfahrten. Jedoch werden jetzt der bekannte Lenkwinkel δ und die Geschwindigkeit v gleichzeitig verändert. Die Stell-signale des Lenkwinkels sind dieselben wie zuvor, jedoch wird die Geschwindigkeit bis zu 2,75 m/s schrittweise erhöht. Erneut erfolgt eine Messung der Geschwindigkeit v und des Schwimmwinkels β durch das Erweiterte Kalman-Filter.

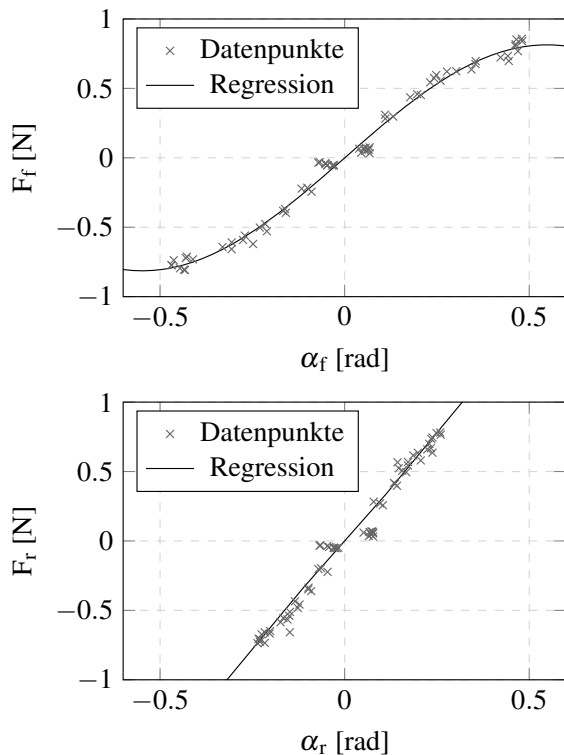
Im Sonderfall stationärer Kreisfahrten gilt: $\dot{\delta} = 0, \dot{v} = 0, \dot{\psi} = 0, \dot{\omega} = \dot{\psi} = 0, \dot{\beta} = 0$. Dies vereinfacht Gleichungen (5) und (6), sodass diese nach F_f und F_r unter Beachtung von $v_{c1} = v \cos \beta$ umgeformt werden können:

$$F_f = \frac{l_r}{(l_f + l_r) \cos \delta} \dot{\psi} m v \cos \beta \quad (17)$$

$$F_r = \frac{l_f}{l_f + l_r} \dot{\psi} m v \cos \beta \quad (18)$$

Für die Schräglaufwinkel α_f und α_r gelten Gleichungen (12) und (13).

Für jede Kreisfahrt berechnet sich jeweils ein Messwert für die Seitenkräfte $F_{f,r}$ mit dem dazugehörigen Schräglaufwinkel $\alpha_{f,r}$ aus diesen Gleichungen. Aus


Abbildung 4: Reifenkennlinie des Vorder- und Hinterrads

insgesamt 50 Kreisfahrten mit jeweils verschiedenen Geschwindigkeiten und Lenkwinkeln ergeben sich die Datenpunkte nach Abbildung 4.

Mithilfe des MATLAB-Befehls `lsqnonlin` erfolgt eine nichtlineare Regression der Reifenkennlinien (11) durch das Least-Squares-Verfahren mit der Levenberg-Marquardt-Methode. Die resultierenden Reifenparameter sind in Tabelle 1 aufgeführt.

Vorderrad	Wert	Hinterrad	Wert
B_f	0,025	B_r	0,0014
C_f	114,3442	C_r	16,398
D_f	0,8136	D_r	135,9653

Tabelle 1: Magic-Formula-Parameter

Die verbleibenden Parameter des Fahrdynamikmodells sind in folgender Tabelle 2 zusammengefasst. Der Radstand l sowie die Abstände l_f und l_r des Schwerpunktes zu den jeweiligen Achsen werden geometrisch vermessen. Die Masse m wird gewogen. Das Massenträgheitsmoment J um die Gierachse im Schwerpunkt wird an einem Schwingungsprüfstand der Hochschule Heilbronn gemessen.

	Parameter	Wert
l_f	Vorderachse zu COG	49 mm
l_r	Hinterachse zu COG	50 mm
l	Radstand	99 mm
m	Masse	132 g
J	Massenträgheitsmoment	$192 \cdot 10^{-6} \text{ kg m}^2$
T_t	Totzeit der Stellsignale	40 ms
T	Zeitkonstante Längsdynamik	800 ms
k	Verstärkungsfaktor Längsdynamik	4,5 m/s

Tabelle 2: Modellparameter des Mini-Z-Fahrzeuges

5 Modell in Modern C++

In MAD ist der ROS-Knoten `carsim_node` zur Simulation der Fahrdynamik in C++14 unter Verwendung von Boost-Odeint implementiert (siehe Abbildung 1). Der folgende Codeausschnitt zeigt den wesentlichen Teil der C++-Klasse zur Simulation des Fahrdynamikmodells aus Abschnitt 3. Die Methode `step` liest die Eingangssignale ein, berechnet einen Simulationsschritt mit Runge-Kutta 4. Ordnung und gibt die Ausgangssignale aus. Das Runge-Kutta-Verfahren von Boost-Odeint ruft unterlagert den C++-Funktorkonzept (`>`) auf, der die rechte Seite des Fahrdynamikmodells (1)-(6) implementiert. Diese Repräsentation eines zeitkontinuierlichen Zustandsraummodells in C++ erfolgt auf vergleichbar hohem Abstraktionsniveau wie eine entsprechende Modellierung in MATLAB/Simulink.

```
using StatesType = std::array<float, 6>;
CarParameters const * const p {
    CarParameters::p() };
float t { 0.0F };
StatesType x { {} };
boost::numeric::odeint::runge_kutta4<
    StatesType> solver;

void step(const InputsType& u,
          OutputsType& y) noexcept
{ ...
    solver.do_step(boost::ref(*this), x, t,
                  p->Ta);
... }

void operator()(const StatesType& x,
               StatesType& xd, const float t) noexcept
{ ...
    xd.at(0) = -Ff * std::sin(delta) / p->m
              + x.at(5) * x.at(4)
              + (-x.at(0) + p->k * u) / p->T;
```

```

xd.at(1) = x.at(0) * std::cos(x.at(3))
          - x.at(5) * std::sin(x.at(3));
xd.at(2) = x.at(0) * std::sin(x.at(3))
          + x.at(5) * std::cos(x.at(3));
xd.at(3) = x.at(4);
xd.at(4) = (Ff * p->lf * std::cos(delta)
          - Fr * p->lr) / p->J;
xd.at(5) = (Ff * std::cos(delta) + Fr
          - p->m * x.at(0) * x.at(4)) / p->m;
... }

```

6 Visualisierung in 3D

Eine der Zielsetzungen des Projekts besteht in der Visualisierung in 3D. Diese geht über die Visualisierung mit MATLAB oder ROS deutlich hinaus, denn sie ermöglicht es dem Betrachter, sich virtuell in eines oder mehrere der Fahrzeuge zu setzen, über ihnen mitzufiegen oder an jedem beliebigen Punkt in der Szene eine Kamera zu fixieren. Zudem erlaubt sie die Einblendung von Berechnungsdaten, z. B. Trajektorien in einer für die Betrachter ästhetisch hochwertigen Weise. Schließlich kann das gesamte visualisierte Modell in eine virtuelle Welt eingebettet werden. Die Visualisierung erfüllt damit pädagogische Zwecke, z. B. auf Messen. Unreal ist eine Game-Engine, die bereits seit 1998 auf dem Markt ist. Sie wurde für Ego-Shooter-Spiele entwickelt und liegt derzeit in Version 4.18 vor. Im Zusammenhang mit der Visualisierung von Verkehrssituationen wird sie bereits an anderer Stelle eingesetzt (z. B. bei [7]). Die Engine unterstützt u. a. eine C++-Schnittstelle unter Windows und Linux. Um live auf die Daten der Fahrzeuge (Position und Orientierung) sowie Trajektorien zugreifen zu können, wurde innerhalb eines Masterkurses ein ROS-Knoten implementiert, der direkt die Engine steuert. Aufgrund des Publisher-Subscriber-Mechanismus von ROS ist es nun möglich, mehrere Anzeigen parallel aus verschiedenen Blickwinkeln zu betreiben, auf denen Features der Szene ein- oder ausgeblendet werden können.

7 Zusammenfassung

Das Fahrdynamikmodell mit den identifizierten Parametern weist eine hohe Genauigkeit auf, sodass wiederholbare MiL- und SiL-Tests der Softwarefunktionen fürs automatisierte Fahren eine hohe Aussagekraft haben. Die Softwarearchitektur von Mini-Auto-Drive (MAD) ermöglicht weiterhin einen simultanen

Betrieb von simulierten und realen Fahrzeugen, womit ein flexibler Aufbau komplexer Fahrzenarien möglich ist. In Verbindung mit der objektorientierten Repräsentation der einzelnen Fahrzeuge und der Fahrzeugumgebung steht mit MAD und MODBAS-Safe [1] eine modellbasierte Softwareentwicklung für automatisiertes Fahren in Modern C++ [8] zur Verfügung.

Literatur

- [1] Daniel Tuchscherer and Frank Tränkle. A Multi-Platform Modern C++ Framework for Safety-Critical Embedded Software. To appear in embedded world Conference 2018.
- [2] Ansgar M. Meroth, Frank Tränkle, Bastian F. Richter, Marco Wagner, Michael Neher, and Jochen Luling. Functional Safety and Development Process Capability for Intelligent Transportation Systems. *IEEE Intelligent Transportation Systems Magazine*, 7(4):12–23, 2015.
- [3] Marcus Almén. LIU Racetrack. <http://www.isy.liu.se/edu/projekt/tsrt10/2016/racetrack/index.html>, 2016. Zugriff am 18. Nov. 2017.
- [4] Alexander Liniger, Alexander Domahidi, and Manfred Morari. Optimization-Based Autonomous Racing of 1:43 Scale RC Cars. *Optimal Control Applications and Methods*, 36(5):628–647, 2015.
- [5] Hans B. Pacejka. *Tire and Vehicle Dynamics*. Elsevier, 2012.
- [6] Bardini R. Schramm D., Hiller M. *Modellbildung und Simulation der Dynamik von Kraftfahrzeugen*. Springer Vieweg, 2 edition, 2013.
- [7] M. Tschentscher, B. Pruß, and D. Horn. A simulated car-park environment for the evaluation of video-based on-site parking guidance systems. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1571–1576, June 2017.
- [8] Daniel Tuchscherer, Alexander Weibert, and Frank Tränkle. Modern C++ as a Modeling Language for Automated Driving and Human-Robot Collaboration. In *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*, pages 136–142. ACM, 2016.

Towards State of the Art in open-set Face Identification

Oliver Huettner¹, Carsten Lanquillon², Sigurd Schacht

Heilbronn University of Applied Sciences, Business Information Systems

¹*huettner@stud.hs-heilbronn.de*

²*carsten.lanquillon@hs-heilbronn.de*

Face recognition methods have been existed since decades while there are still many challenges that have to be mastered. The standard benchmark database for unconstrained face verification (1:1 matching), namely LFW provides an overview of the best performing methods, which have reached almost perfect recognition results. However, open-set face identification (1:N with unknown subjects) turned out being a great challenge for today's methods. Our work provides an overview of methods that have been established since the Face in Video Evaluation (FIVE) report which was carried out in 2015. We found only few algorithms that consider open-set face identification, all of them seem to not be able handling critical real-world scenarios like detecting criminal suspects as that requires a small False Alarm Rate. The provided methods consider a high False Alarm Rate and / or deliver insufficient results regarding the accuracy. A comparison is possible only in a limited way, that's why we propose for future work an establishment of a new standard benchmark database.

1 Introduction

Face (or facial) recognition systems have been important to research as well as practitioners for several decades. Early approaches to localize and recognize faces in images date back more than 25 years [1]. However, there are still max open challenges to master. Face recognition based on frontal views has improved significantly within the last decades, but views from different angles as well as aging or other factors that influence a person's looks can be arduous and error-prone [2]. In this paper, we focus on face recognition for uncontrolled environments like it is often needed in practical implementations.

Based on a literature review on the state of the art in this field, we provide an overview of approaches which are suitable for face recognition in uncontrolled environments.

The introduction will be followed by research challenges and criteria revealing which requirements exist for face recognition in uncontrolled environments. A classification of the approaches identified will point out those approaches that meet these requirements to a certain extent. The results of the classification will be shown in a concept matrix to provide a good overview. Finally, a critical assessment of the approaches and a proposal for future work conclude this paper.

The results of this work are especially addressed to practitioners who want to enable face recognition in real scenarios with the right approach.

2 Research Challenges & Criteria

2.1 Research Challenges

Face recognition in uncontrolled environments meets different challenges that have to be mastered. According to [3] one of the greatest challenges is illumination variation. It affects the rate of identifying significantly by influencing attributes such as color and shape. The article names several approaches to alleviate this problem. A current method (Multi-Scale Retinex) for re-illuminating the images reaches a high success for recognizing faces, but cannot handle real-time-processing and sufficient color enhancement [4].

Additionally to illumination variation, [5] declares occlusion as a great challenge in real world applications. Further challenges are: age, hair, facial expression, gender, ethnic origin [6], image resolution, pose [7].

During our research we found that the LFW Database is a standard benchmark for face verification (matching of two given faces 1:1) and covers each of these challenges at least partially [8]. The LFW Homepage¹ features a list where all documented results of tested approaches are revealed. It shows that the accuracy in face verification already has reached top performance. To date the list includes ten published approaches that

¹ <http://vis-www.cs.umass.edu/lfw/>

reach an accuracy better than 0.95 while FaceNet [9] is the best performing approach with an accuracy of 0.9963.

Further investigation let us come across a study named FIVE (Face In Video Evaluation) [10] which revealed the actual challenge in face recognition for practical usage: open-set (including unknown identities) face identification (1:N allocating) in an uncontrolled environment featuring non-cooperative subjects. The study shows that submitted prototypes can't provide high accuracy within these constraints.

We then focused on looking for approaches that consider an open-set face identification in their work. We limited our search to literature that is not older than 2015 as we assume the FIVE study covers state of the art of the year it was undertaken.

2.2 Criteria

The following criteria are the basis for the classification of algorithms.

Approach

An algorithm can either be holistic, structural or hybrid [11]. The holistic approach identifies a face due to its whole representation while the structural approach locates relevant regions within the face. A hybrid approach uses both inputs.

Database

This Criterion reveals which database (under open-set evaluation) was used for evaluating the algorithm(s). Besides open available databases there also could have been used own (publicly unavailable) images or videos for evaluating. Within the classified methods there always have been used open available databases.

Used Algorithm

A face recognition method is often based on existing algorithms that enhance the existing solution. Usually these algorithms are named within the literature. This category shall reveal which commonly known algorithms have been used.

Similarity Measure

The similarity of two classes can be acquired by different measurement methods, e.g. Chi Square, Borda Count etc. [12]. Within the classified methods, only cosine similarity and an unnamed own approach have

been found. If neither is allocated, i.e., the similarity measurement was not explicitly revealed.

Performance

Regarding performance information, we considered accuracy and false alarm rate (FAR). In some literature, the FAR was not revealed. Without the FAR the accuracy is not meaningful, that's why we have left it off in that case.

3 Classification of Methods

Figure 1 shows the results of the classification. It is remarkable that none of the methods follows a pure structural approach, but there are many holistic approaches.

Method	Approach		Database					Used Algorithm								Similarity Measure		Performance	
	holistic	hybrid	LFW	MPIE	MegaFace	UCCS	CNN	PCA	Joint Bayesian	LDA	UDP	LPP	EVM	LBP	Cosine Similarity	Own Method	Documented Accuracy	FAR	
DeepID3	X	X					X	X	X								0.814	0.01	
OSSR + CRC	X	X													X		-	-	
SphereFace	X			X			X								X		0.757	0.000001	
VGG (EVM)	X	X					X					X					≈ 0.65	0.01	
VGG (LDA)	X	X					X	X	X								≈ 0.55	0.01	
VGG (Cos)	X	X					X								X		≈ 0.2	0.01	
Baseline		X			X			X	X				X				-	-	
LqfNet	X				X	X	X		X								-	-	
UCCS	X				X	X						X					-	-	
CVSSP	X				X	X									X		-	-	
FKDA (LDA)	X		X						x						X		0.693	0.001	
FKDA (UDP)	X		X							x					X		0.737	0.001	
FKDA (LPP)	X		X								x			X			0.733	0.001	

Figure 1. Classification of Methods

Even though LFW recommends following the given verification evaluation protocol, the provided methods (which chose LFW for evaluation) test it with an open-set identification evaluation protocol. We can't declare which database fits best to real world applications and therefore practical usage. But we want to reveal the number of images and individual subjects that are included. LFW contains 13,233 images from 5,749 people as a whole from whom 1,680 people provide two or more images. The CMU Multi-PIE Face database² contains more than 750,000 images from 337 people. The MegaFace (Challenge 1)³ database provides more than one million images from 690,572 people and therefore is the database with the greatest amount of people and images out of the methods we classified. The UCCS dataset contains more than 70,000 images of (hand cropped) face regions with more than 4,000

² <http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/MultiPie/Home.html>

³ <http://megaface.cs.washington.edu>

people [13]. This dataset is the only one out of these that mainly includes images of non-cooperative subjects and therefore features the greatest challenge.

Most of the given methods used convolutional neural networks (CNN) [13]–[16]. Four methods included Linear Discriminant Analysis (LDA) [13], [16], [17] and three methods considered Principal Component Analysis (PCA) within their approach [13], [14], [16]. Others are Joint Bayesian [14], Unsupervised Discriminant Projection (UDP), Locality Preserving Projections (LPP) [17], Extreme Value Machine (EVM) [13], [16] and Local Binary Pattern (LBP) [13]. CNNs as well as PCA and LDA has existed for more than 20 years. The broad usage of these algorithms reveals that they can still perform very well and are often the base for a successful approach. The challenging task of face identification in an open-set evaluation seems to be mastered in a more or less sufficient way by these algorithms. It may also represent that there are too few alternatives that have better performance.

The cosine similarity measurement seems to be the standard measurement method when it comes to open-set face identification. Caused by missing alternatives within the given methods we can't show up if other measurements would yield better results.

When it comes to accuracy it is inconclusive which method performs best as the considered False Alarm Rates (FAR) differ. A FAR of 0.01 means, one out of 100 unknown persons is erroneously assigned to a known person. Needless to say, if the FAR is higher, the accuracy usually rises, if the FAR is lower, the accuracy decreases. In [16] an example with a surveillance system reveals the critical aspect of a low FAR: If the gallery contains criminal suspects and a surveillance of a public place detects 100 persons a minute, one false alarm would occur per minute.

Depending on the intended use of the system it is essential to determine the FAR and if necessary (due to missing documented results with the determined FAR) test the method.

The SphereFace method [15] is the only one that considers a very small FAR with a false alarm in one out of one million unknown persons. At the same time, it provides sufficient accuracy compared to the other methods. The worst results are delivered by the three methods of [16] at a high FAR (0.01). All these methods are based on the VGG face network which uses CNN [18] and then are enhanced by EVM, LDA or simply cosine similarity for measurement.

The Multi-class Fukunaga Koontz Discriminant Analysis (FDKA) approach has an accuracy between 0.693 and 0.733 (at a FAR of 0.001) depending on the joined algorithm [17]. The best result is given with UDP.

The method with the highest documented accuracy is DeepID3 [14]. The FAR is at 0.01 which is like we earlier found too high for some real-world applications with practical usage. Hence it may not be the best choice and should be tested further.

Open-Set Sparse Representation and Collaborative Representation Classification (OSSR + CRC) is proposed by [19]. There is no exact accuracy at a given FAR revealed. The authors only provide a graphical overview showing the accuracy depending on the percentage of used images in gallery. With rising number of used images the accuracy gets better. At 100% (which makes it a closed-set identification) the accuracy is about 94%. With only 10% used images the mean accuracy is about 71%. The FAR is not mentioned in their work.

The performance of the four compared methods of [13], namely Baseline, LqfNet, UCCS and CVSSP is represented in a graphical overview. The FAR is not revealed, only an absolute number of false identifications on x-axis is given. The FAR cannot be calculated as they have not determined the amount of unknown people. Hence the results are not comparable to other methods as long as the number of unknown identities is unclear.

4 Conclusion

We analyzed the state of the art in face recognition for scenarios that require the evaluation task in form of open-set face identification. The 2015 FIVE reports already revealed a huge gap in performance of face recognition algorithms that are confronted with open-set identification and additionally non-cooperating subjects. Our findings underline their results, revealing that current methods for face recognition cannot handle open-set identification with a sufficient performance regarding critical scenarios like detecting criminal suspects. Sufficient accuracy can only be achieved at a high False Alarm Rate which leads to huge costs (e.g. hours of work by arresting wrong persons) and unacceptable distress of non-criminal people.

Furthermore, within our work we could not fully compare the provided methods. As the recommendation of an approach needs to be founded on comparison using consistent measurements, they have to be tested at the

same False Alarm Rate and in best case on the same dataset. However, while none of the methods with the given results is suitable for critical real world applications we find that this may not have first prioritization.

The literature search was done on IEEE, SpringerLink, Google Scholar and Emerald Insight. We searched in title or (when provided) metadata. We didn't search full-text and may have missed some works with significant findings. However, other works also confirm the lack of methods for open-set face identification [10], [16].

For future work, we suggest establishing a standard benchmark database for open-set face identification with non-cooperative subjects as the next generation of LFW. If a standard benchmark is given, a fair comparison of existing algorithms will lead to transparent performance. Hopefully this will provide faster research results as researchers can focus on the best performing algorithms to enhance or outperform them.

5 References

- [1] M. a. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces," *Journal of Cognitive Neuroscience*, vol. 3, no. 1. pp. 72–86, 1991.
- [2] K. S. Devi, "GBU Based Face Recognition Techniques : A Review," pp. 4–7, 2017.
- [3] K. S. Prado, N. T. Roman, V. F. Silva, J. L. Bernardes, L. A. Digiampietri, E. M. Ortega, C. A. M. Lima, L. M. V. Cura, and M. M. Antunes, "Automatic facial recognition: A systematic review on the problem of light variation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9732, pp. 211–221, 2016.
- [4] A. A. S. Gunawan and H. Setiadi, "Handling illumination variation in face recognition using multiscale retinex," *2016 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2016*, pp. 470–475, 2017.
- [5] A. S. Huwedi and H. M. Selem, "Face Recognition using Regularized Linear Discriminant Analysis under Occlusions and Illumination Variations," *Proc. 2016 4th Int Conf Control Eng. Inf. Technol. (CEIT-2016)*, no. December 2016, 2016.
- [6] C. Guzel Turhan and H. S. Bilge, "Class-wise two-dimensional PCA method for face recognition," *IET Comput. Vis.*, vol. 11, no. 4, pp. 286–300, 2017.
- [7] S. P. Mudunuri and S. Biswas, "Low Resolution Face Recognition Across Variations in Pose and Illumination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 5, pp. 1034–1040, 2016.
- [8] F. Moya Rueda, R. Grzeszick, and G. A. Fink, "Neuron pruning for compressing deep networks using maxout architectures," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10496 LNCS, pp. 177–188, 2017.
- [9] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07–12–June, pp. 815–823, 2015.
- [10] P. Grother, G. Quinn, M. Ngan, Information Access Division, and Information Technology Laboratory, "Face In Video Evaluation (FIVE) Face Recognition of Non-Cooperative Subjects," *Natl. Inst. Stand. Technol.*, 2017.
- [11] W. Zhao, R. Chellappa, P. J. Phillips, and a Rosenfeld, "Face recognition: A literature survey," *Acm Comput. Surv.*, vol. 35, no. 4, pp. 399–458, 2003.
- [12] J. Ruiz-Del-Solar, R. Verschae, and M. Correa, "Recognition of faces in unconstrained environments: A comparative study," *EURASIP J. Adv. Signal Process.*, vol. 2009, no. iv, 2009.
- [13] M. Günther, P. Hu, C. Herrmann, C. H. Chan, M. Jiang, S. Yang, A. R. Dhamija, D. Ramanan, J. Beyerer, J. Kittler, M. Al Jazaery, M. I. Nouyed, G. Guo, C. Stankiewicz, and T. E. Boult, "Unconstrained Face Detection and Open-Set Face Recognition Challenge," 2017.
- [14] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: Face Recognition with Very Deep Neural Networks," pp. 2–6, 2015.
- [15] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep Hypersphere Embedding for Face Recognition," pp. 6738–6746, 2017.
- [16] M. Gunther, S. Cruz, E. M. Rudd, and T. E. Boult, "Toward Open-Set Face Recognition," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2017–July, pp.

- 573–582, 2017.
- [17] F. Juefei-Xu and M. Savvides, “Multi-class Fukunaga Koontz discriminant analysis for enhanced face recognition,” *Pattern Recognit.*, vol. 52, pp. 186–205, 2016.
- [18] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep Face Recognition,” *Proceedings Br. Mach. Vis. Conf. 2015*, no. Section 3, p. 41.1-41.12, 2015.
- [19] A. Moeini, K. Faez, H. Moeini, and A. M. Safai, “Open-set face recognition across look-alike faces in real-world scenarios,” *Image Vis. Comput.*, vol. 57, pp. 1–14, 2017.

Mechanische CAD-Simulationsverfahren; Anlagen-Engineering mit DMU

Prof. Dr. W. Kalliwoda

Hochschule Heilbronn

werner.kalliwoda@hochschule-heilbronn.de

Der Beitrag beinhaltet einen anwendungsorientierten Überblick zu ausgewählten CAD-Simulationsverfahren, die auch in der Lehre Anwendung finden können.

1 Einleitung

Die realistische Darstellung eines komplexen Produktes im Rechner, wie z. B. der von Großanlagen in der chemischen Industrie, wird als „Digital Mock Up“ - wörtlich übersetzt „Nachbildung“ - bezeichnet.

Seit einigen Jahren entwickeln Hersteller ihre Anlagensysteme und deren Komponenten mit Hilfe von Gewerke übergreifender 3D-CAD-Software, beispielsweise mit dem Programmsystem CATIA V5. Die dabei entstehenden CAD-Modelle werden als virtuelles Produkt bezeichnet. Das virtuelle Produkt kann mithilfe von Simulationswerkzeugen in vielerlei Hinsicht visualisiert und analysiert werden [1].

2. DMU im Apparatebau

2.1 Montagesimulation

Applikation: CATIA V5-DMU Fitting

Die Anwendung ermöglicht eine Definition und Simulation von Montage- und Demontageprozessen. Damit kann in der Entwicklungsphase eines Produkts die Passgenauigkeit der Komponenten zueinander geprüft werden, um Aussagen zur Funktionsfähigkeit zu treffen. Gleichzeitig lässt sich mit der Methode eine Kollisionsanalyse erstellen um unerwünschte Bauteildurchdringungen zu visualisieren.

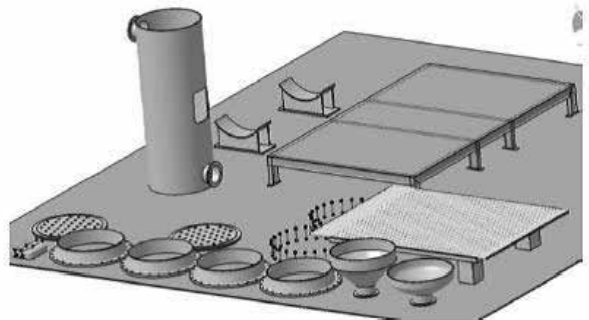
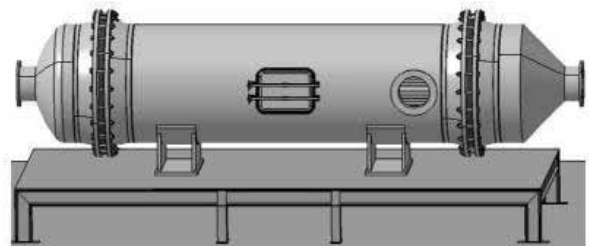


Abbildung 1. Wärmetauscher im Zusammenbau (o.) und in seine Einzelteile zerlegt (u.)

2.2 Kinematiksimulation

Applikation: CATIA V5-DMU Kinematik

Die Anwendung ermöglicht eine Definition und Simulation mechanisch bewegter Massen. Durch die Bewegungssimulation können mögliche Bauteilkollisionen analysiert und im laufenden Konstruktionsprozess behoben werden, um auftretende Störfälle im Betrieb zu vermeiden und die Anlagensicherheit zu gewährleisten.

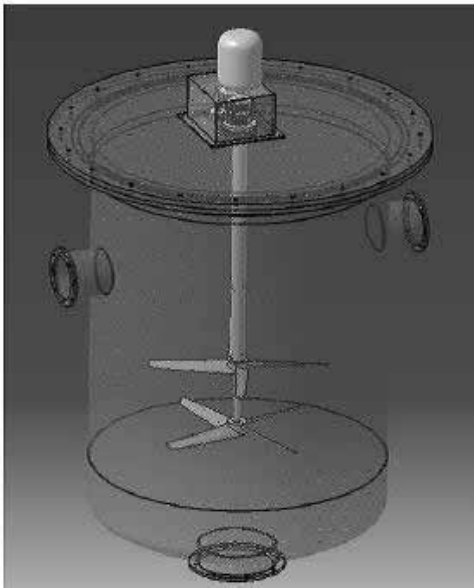


Abbildung 2. Rührwerksreaktor

3. Simulation der Bauteilfestigkeit

Applikation: (CATIA V5) Generative Structure Analysis [2]

Die Finite-Elemente-Methode (FEM) hat sich zu einem Standardverfahren der numerischen Simulation entwickelt. Anspruchsvolle und aufwändige Berechnungen die vor Jahren noch mit dem Großrechner durchgeführt wurden, sind heute mit einem gut ausgestatteten PC möglich. Die Methode beinhaltet die Erstellung eines Simulationsmodells (Pre-Processing), die Durchführung der Berechnung und die Auswertung und Darstellung der Ergebnisse (Post Processing). Die Plausibilität von berechneter Spannung und Verformung wird durch elementaren Festigkeitsnachweis an einem Beispiel aus der Statik nachgewiesen.

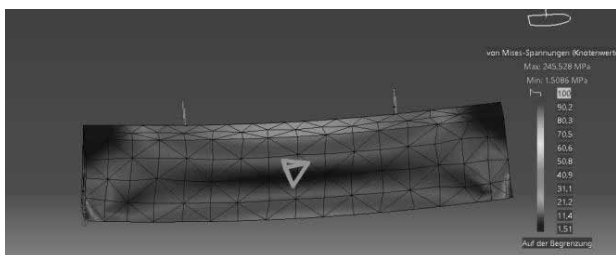


Abbildung 3. Stützträger unter Streckenlast

4. Simulation ressourcengerechter Fertigung

Applikation: SolidWorks Sustainability

Umweltverträglichkeit steht für Hersteller immer mehr im Fokus bei der Entwicklung und Herstellung von Maschinen und Anlagen. Ziel der umweltgerechten Produktentwicklung ist es, in der Entwicklungsphase des rechnergestützten Konstruierens eine Umweltverträglichkeitsprüfung in Echtzeit unter Zuhilfenahme einer umfassenden Lebenszyklusdatenbank durchzuführen.

Die Suche nach alternativen Werkstoffen und den daraus resultierenden Angaben zur Umweltauswirkung, wie z.B. der CO₂ Emission und des Energieverbrauchs werden in Abhängigkeit des Herstellverfahrens und den Einflussgrößen bei der Produktnutzung berechnet und protokolliert.

Für das im Folgenden dargestellte Lüfterrad eines Axialventilators wurde der Laufradwerkstoff von PA6 auf POM geändert. Die Einsatzzeit des Laufrads ist auf 5 Jahre und die tat. Betriebszeit auf 1 Jahr festgelegt.

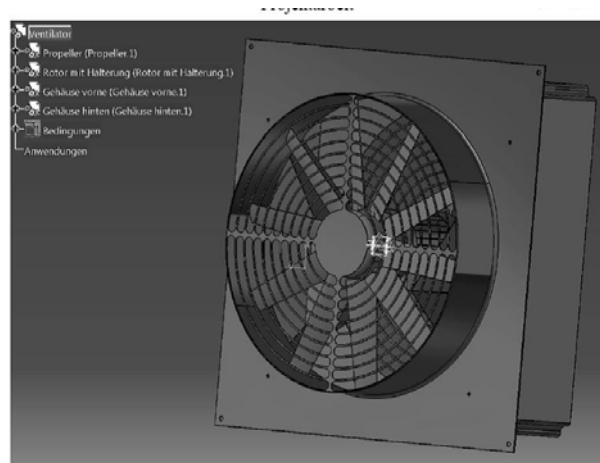


Abbildung 4. Lüfter-Gebläse mit substituiertem Laufradwerkstoff: CO₂ Reduzierung von 34%, Energieverbrauch: -28%

Literaturquellen

- [1] O. Abeln: Die CAX-Techniken in der industriellen Praxis, Carl Hanser Verlag
- [2] W. Koehldorfer: CATIA V5 Volumenmodellierung, Carl Hanser Verlag

Predictive Convolutional Autoencoding: Anomalieerkennung für eine automatisierte bildgestützte Qualitätsprüfung

Carsten Lanquillon¹, Jerome Tagliaferri, Marina Früh, Jananika Ragunathan, Inna Nenja
¹Hochschule Heilbronn, Business Information Systems

carsten.lanquillon@hs-heilbronn.de

Um eine bildgestützte Qualitätsprüfung stärker zu automatisieren, werden Fehler als Anomalien durch einen Predictive Convolutional Autoencoder erkannt. Der Autoencoder stellt ein Modell der Normalität dar, das ausschließlich auf fehlerfreien Bildern erlernt wird. Während fehlerfreie Bilder sehr gut reproduziert werden können, bekommt der Autoencoder Probleme, sobald Anomalien vorliegen. Fehler werden als somit als Abweichungen zwischen den Originalbildern und deren Rekonstruktionen erkannt. Die Funktionsweise dieses Ansatzes der Anomalieerkennung wird anhand von zwei Datensätzen demonstriert.

1 Einleitung

Die automatisierte Qualitätsprüfung mit Hilfe von Bildern [1] hat bereits eine lange Historie [2–6]. Aus Sicht des maschinellen Lernens lässt sich die Qualitätsprüfung als überwachtes Klassifikationsproblem mit einer IO-Klasse und einer oder mehreren Fehlerklassen auffassen [7–11], oder aber als Anomalieerkennung mit zumeist unüberwachten Lernverfahren [6]. Dabei gilt die IO-Klasse als „normal“ und die NIO-Klasse als Anomalie, die es als Abweichung von der zu spezifizierenden bzw. zu erlernenden Normalität zu erkennen gilt [12].

2 State-of-the-Art

Mit hinreichend vielen positiven und negativen Lernbeispielen sind überwachte Lernverfahren den unüberwachten hinsichtlich der Erkennungsleistung auf Grund der zusätzlichen Information durch die Kennzeichnung bekannter Klassen überlegen [5].

Bei der Qualitätsprüfung liegt meist eine sehr schiefe Klassenverteilung vor. Auch können einige Fehlerarten während der Lernphase noch unbekannt sein und erst in der Anwendungsphase erstmalig auftreten. Daher ist die Qualitätsprüfung mit unüberwachter Anomalieerkennung sinnvoll. Dabei ist es dennoch

möglich, bekannte Fehlerfälle zur Optimierung der Entscheidungsfindung zu verwenden [13].

Ein Nachteil der meisten bekannten Ansätze zur automatisierten Qualitätsprüfung ist die meist sehr aufwendige manuelle Merkmalerzeugung aus den Rohdaten der Bilder als Eingabe für die maschinellen Lernverfahren. Dies begrenzt den möglichen Grad der Automatisierung erheblich. Eine Abhilfe stellen Deep-Learning-Verfahren dar, die eigenständig eine geeignete Repräsentation erlernen können. Bei der Bildverarbeitung zeichnen sich dabei insbesondere Convolutional Neural Networks (CNNs) aus, die eine translationsinvariante Erkennung von Mustern bei begrenztem Parametersuchraum ermöglichen [14]. CNNs kommen in der bildgestützten Qualitätsprüfung bislang überwiegend in Form überwachter Klassifikation zum Einsatz [15–19].

3 Vorgehensweise

Die Idee der automatischen Merkmalerzeugung wird hier in Form eines Convolutional Autoencoders (CAEs) als unüberwachtes oder vielmehr *selbstüberwachtes* Verfahren aufgegriffen und als predictive Autoencoder [13] zur Fehlererkennung erweitert.

Ein CAE kann durch die automatische Extraktion relevanter Merkmale Bilder kompakt darstellen (Kom-

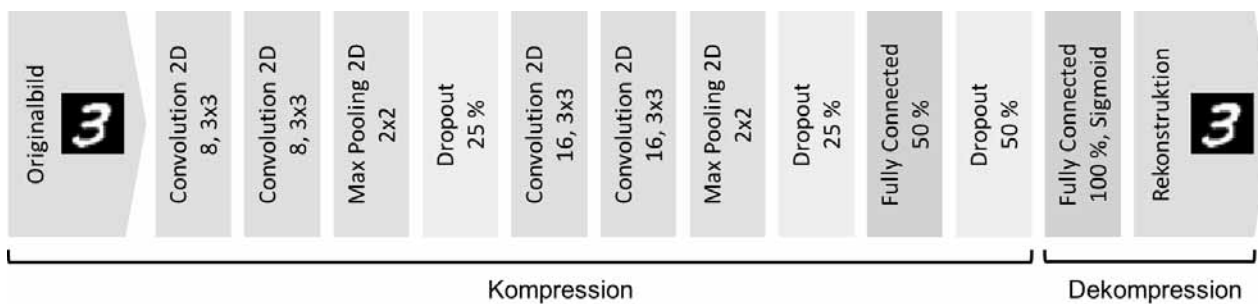


Abbildung 1: Architektur des Convolutional Autoencoders

pression) und rekonstruieren (Dekompression) wie in Abbildung 1 dargestellt. Das unüberwachte bzw. eher selbst-überwachte Lernverfahren basiert dabei nur auf Bildern, die keine Fehler enthalten und versucht die Normalität möglichst gut zu rekonstruieren.

Es wird angenommen, dass in der späteren Anwendung ein fehlerfreies Bild besser und genauer rekonstruiert werden kann als ein Bild mit Fehlern. Im Idealfall fehlt der fehlerhafte Bildausschnitt beim rekonstruierten Bild komplett und führt folglich zu einer größeren Abweichung der Rekonstruktion vom Originalbild. Mit bekannten Fehlerfällen kann deshalb in einem zweiten überwachten Schritt die Klassifikation in fehlerfrei und fehlerhaft erfolgen, indem ein optimaler Schwellwert für die Güte der Bildrekonstruktion definiert als Summe der quadrierten punktwweisen Abweichungen der Rekonstruktion vom Originalbild bestimmt wird. Durch den letzten Schritt wird der klassische CAE zum *predictive CAE*.

4 Architektur

Beim Aufbau eines geeigneten neuronalen Netzes gibt es unzählige Gestaltungs- und Wahlmöglichkeiten und gilt daher immer noch als *Kunst*. Die Architektur unseres Convolutional Autoencoders (CAE) orientiert sich stark am VGGNet [17] und ist mit seinen verschiedenen Schichten in Abbildung 1 dargestellt.

Es werden insgesamt vier *2D-Convolutional*-Schichten mit Filtern der Größe 3x3 kombiniert. In den ersten beiden Convolutional-Schichten werden je acht Filtern eingesetzt und in der dritten und vierten Schicht je 16 Filter. Nach jeweils zwei Convolutional-Schichten wird eine *2D-Max-Pooling*-Schicht mit nicht überlappenden 2x2-Pooling-Fenstern [20] und eine *Dropout*-Schicht mit einer Quote von 25% zur Regularisierung des Netzes eingesetzt [21].

Mit Ausnahme der letzten Schicht wird stets eine Rectified-Linear-Unit (*ReLU*) als Aktivierungsfunktion verwendet [22]. In der letzten Schicht, in der das Originalbild rekonstruiert wird, kommt die *sigmoid*-Aktivierungsfunktion zum Einsatz.

Nach den beiden Convolutional-Blöcken folgen zwei vollständig verbundene Neuronen-Schichten (*Fully-Connected*-Schichten), die durch eine Dropout-"Schicht" mit einer Quote von 50% getrennt sind. Die letzte Fully-Connected-Schicht mit sigmoid-Aktivierungsfunktion enthält für die Rekonstruktion der Originalbilder genau ein Neuron für jedes Pixel. Die erste Fully-Connected-Schicht enthält dagegen nur 50% der durch die Bildgröße vorgegebenen Neuronenzahl und erzwingt zusammen mit der letzten Dropout-"Schicht" eine Kompression der Originalbilder.

Als Loss-Funktion wird der mittlere quadratische Fehler (*mean squared error*) verwendet, der die Abweichungen zwischen den Originalbildern und ihren Rekonstruktionen zweckmäßig berechnet. Optimiert wird nach der Methode *ADADELTA* mit adaptiver Lernrate [23] mit Mini-Batches der Größe 32. Eine für die jeweilige Aufgabenstellung in Abhängigkeit der Größe und Art der Bilder angemessene Anzahl an Epochen wird interaktiv festgelegt.

5 Evaluierung

Die Überprüfung des Ansatzes zur Erkennung von Anomalien mit Convolutional Autoencoding mit der oben beschriebenen Architektur erfolgt basierend auf zwei Datensätzen. Zum einen wird eine erweiterte Version der MNIST-Datenbank für handgeschriebene Ziffern verwendet [24] und zum anderen kommt der vierte Datensatz für die Erkennung von Defekten in Bildern des Heidelberg Collaboratory for Image Processing (*HCI*) zum Einsatz [25].

5.1 Erweiterte MNIST-Datenbank

Die bekannte MNIST-Datenbank enthält normierte graustufen Bilder mit 28x28 Pixeln von handgeschriebenen Ziffern und entsprechende Klassenlabel für die reguläre Klassifikationsaufgabe.

Wir verwenden die 60,000 Bilder der Trainingsmenge ohne Klassenlabel, um das CAE-Normalitätsmodell zu erlernen. 50 Epochen genügen bereits, dass die Originalziffern gut rekonstruiert werden können. Dies wurde anhand der vorher nicht verwendeten Testdaten überprüft.

Um die Fähigkeit des CAE zur Anomalieerkennung zu überprüfen, werden jedoch Bilder mit Anomalien benötigt. Dazu haben wir Ziffern mit Anomalien wie in Abbildung 2 dargestellt künstlich erzeugt und die Testdaten damit ergänzt.



Abbildung 2: Ziffern mit Anomalien (rot umrandet).

Abbildung 3 zeigt die ursprünglichen Ziffern mit Anomalien und deren Rekonstruktion durch den CAE nach einer Lernphase mit 50 Epochen. Bei einer Lernphase mit 200 Epochen kann die Qualität der Rekonstruktion der normalen Ziffern weiter erhöht werden bei etwa vergleichbarer gewünschter Schwäche bei Ziffern mit Anomalien.

Eine genauere Analyse der Rekonstruktionsgüte zeigt, dass die Summen der quadrierten punktweisen Differenzen zwischen Originalen und Rekonstruktionen bei den normalen Ziffern sämtlich im Intervall $[0;1]$ liegen (Mittelwert 0.02 ± 0.07). Bei den Ziffern mit Anomalien sind die entsprechenden Summen um mehr als Faktor 10^3 größer und somit leicht als Anomalien identifizierbar.

5.2 HCI-Datensatz 4

Das Heidelberg Collaboratory for Image Processing (HCI) stellt verschiedene Datensätze bereit, die zur Anomalieerkennung verwendet werden können. Der ausgewählte Datensatz 4 besteht für Training und Test jeweils aus 575 Graustufenbildern mit 512x512 Pixeln. Sowohl Trainingsdaten als auch Testdaten enthalten Bilder mit Anomalien.



Abbildung 3: Originale mit Anomalien (obere Zeile) und deren Rekonstruktionen (untere Zeile) nach einer Lernphase mit 50 Epochen. Der CAE hat wie vermutet Schwierigkeiten bei der Rekonstruktion von Anomalien. Mit Ausnahme des Überlappungsbereichs bei der zweiten Anomalie von rechts und der daraus resultierenden starken Ähnlichkeit zu einer 8 sind die Anomalien kaum noch erkennbar.

In unserem Szenario verwenden wir zum Erlernen des CAE-Normalitätsmodells ausschließlich die 493 Bilder aus den Trainingsdaten ohne Anomalien. Zur effektiveren Nutzung der begrenzten Ressourcen, wurden die Bilder auf 128x128 Pixel skaliert. Abbildung 4 zeigt zwei Bilder des HCI-Datensatzes 4 mit Anomalien.

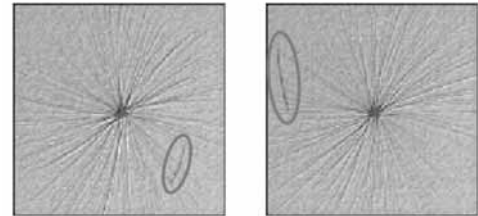


Abbildung 4: HCI-Datensatz 4: Bilder mit Anomalien.

In einem Testlauf mit nur 12 Bildern und 500 Epochen zeigt sich, dass der CAE prinzipiell die aufgrund der Textur deutlich komplexeren Bilder zufriedenstellend rekonstruieren kann und etwaige Bildausschnitte mit Anomalien dabei auch entfernt wie in Abbildung 5 zu erkennen.

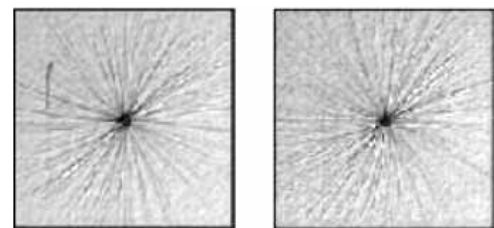


Abbildung 5: Original mit Anomalie (links) und dessen Rekonstruktion (rechts) nach einem Testlauf mit 12 Bildern und 500 Epochen.

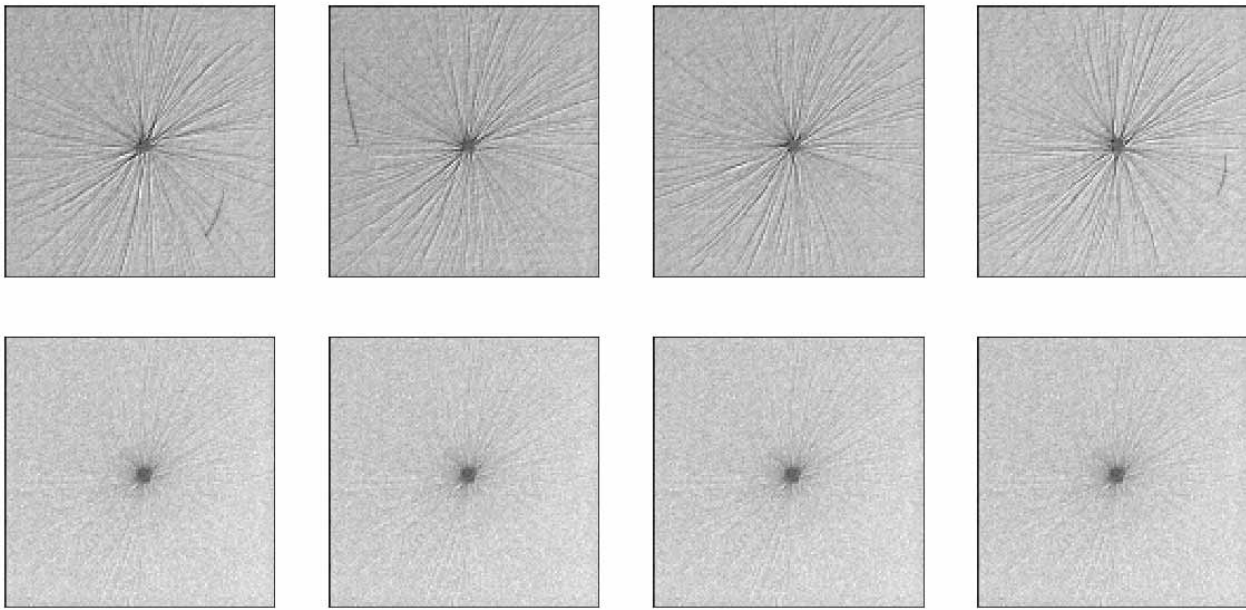


Abbildung 6: Originale des HCI-Datensatzes 4 zum Teil mit Anomalien (obere Zeile) und deren Rekonstruktionen (untere Zeile) nach einer Lernphase mit 200 Epochen. Der CAE hat wie vermutet Schwierigkeiten bei der Rekonstruktion von Anomalien. Allerdings ist die Rekonstruktion der Originale im Detail nicht zufriedenstellend.

Abschließend wird ein CAE-Normalitätsmodell mit allen 493 Trainingsdaten ohne Anomalien mit 200 Epochen erstellt. Mit diesem wurden Bilder mit und ohne Anomalien aus den Testdaten rekonstruiert. Das Ergebnis ist beispielhaft anhand von vier Bildern in Abbildung 6 dargestellt. Bildausschnitte mit Anomalien sind nach der Rekonstruktion wie erhofft nicht mehr erkennbar.

Allerdings ist die Qualität der Rekonstruktionen nicht zufriedenstellend. Die komplexe Textur in den Bildern wird nicht genau genug rekonstruiert, so dass die Abweichung zwischen Originalen und deren Rekonstruktionen auch ohne vorliegende Anomalien sehr groß ist. Dies führt zu erheblichen Problemen bei der automatischen Anomalieerkennung über die Summen der quadrierten punktwweisen Abweichungen.

6 Fazit

Es wurde exemplarisch gezeigt, dass eine Automatisierung der Qualitätsprüfung durch Anomalieerkennung mit Convolutional Autoencoding möglich ist.

Erwartungsgemäß unterscheiden sich die Ergebnisse je nach Anwendungsfall und der Beschaffenheit der vorliegenden Bilder. Die Rekonstruktion von Bildern mit komplexen Texturen ist besonders problematisch.

Weiterführende Untersuchungen sollen zeigen, ob mit einer verfeinerten Architektur insbesondere mit mehr Filtern und mit einer weiteren Erhöhung der Lerndauer auch in diesen Anwendungsfällen zufriedenstellende Ergebnisse erzielt werden können.

References

- [1] J. Beyerer, F. Puente Leon, and C. Frese. *Automatische Sichtprüfung: Grundlagen, Methoden und Praxis der Bildgewinnung und Bildauswertung*. Springer Vieweg, Berlin, Heidelberg, 2. edition, 2016.
- [2] R. T. Chin. Automated visual inspection: 1981 to 1987. *Computer Vision, Graphics, and Image Processing*, 41(3):346–381, 1988.
- [3] T. S. Newman and A. K. Jain. A survey of automated visual inspection. *Computer Vision and Image Understanding*, 61(2):231–262, 1995.
- [4] E. N. Malamas, E. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*, 21:171–188, 02 2003.
- [5] Y. Li and P. Gu. Free-form surface inspection techniques state of the art review. *Computer-Aided Design*, 36:1395–1417, 11 2004.

- [6] X. Xie. A review of recent advances in surface defect detection using texture analysis techniques. *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, 7(3), 2008.
- [7] D. Pham and E. J. Bayro-Corrochano. Neural classifiers for automated visual inspection. In *Proceedings of The Institution of Mechanical Engineers Part D-journal of Automobile Engineering*, volume 208, pages 83–89, 06 1994.
- [8] C. Sanby and L. Norton-Wayne. Machine vision inspection of lace using a neural network. In *Proceedings Symposium on Electronic Imaging: Science & Technology (SPIE)*, volume 2423, pages 314–322, 1995.
- [9] N. Guglielmi, R. Guerrieri, and G. Baccarani. Neural networks techniques for the optical inspection of machined parts. In Cornelius T. Leondes, editor, *Computer Aided and Integrated Manufacturing Systems*, volume 2, pages 77–106. World Scientific Publishing Company, 2003.
- [10] A. Kumar. Neural network based detection of local textile defects. *Pattern Recognition*, 36(7): 1645–1659, 2003.
- [11] A. Tellaeche and B. Robles. 3d machine vision and artificial neural networks for quality inspection in mass production pieces. In *2010 IEEE 15th Conference on Emerging Technologies Factory Automation (ETFA 2010)*, pages 1–4, Sept 2010.
- [12] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3):1–58, Jul 2009.
- [13] M. A. Gluck and C. E. Myers. Hippocampal mediation of stimulus representation: A computational theory. *Hippocampus*, 3(4):491–516, 1993.
- [14] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time-series. In M.A. Arbib, editor, *The handbook of brain theory and neural networks*, pages 255–258. MIT Press, Cambridge, MA, 1995.
- [15] J. Masci, U. Meier, D. C. Ciresan, J. Schmidhuber, and G. Fricout. Steel defect classification with max-pooling convolutional neural networks. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, 2012.
- [16] D. Soukup and R. Huber-Mörk. Convolutional neural networks for steel surface defect detection from photometric stereo images. In *Intelligent Systems and Computer Vision (ISVC)*, pages 668–677, 2014.
- [17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.
- [18] D. Weimer, A. Yoga Benggolo, and M. Freitag. Context-aware deep convolutional neural networks for industrial inspection. In *Proceedings Australasian Conference on Artificial Intelligence*, Canberra, Australia, 12 2015.
- [19] D. Weimer, B. Scholz-Reiter, and M. Shpitalni. Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Annals*, 65(1):417–420, 2016.
- [20] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. The MIT Press, Cambridge, Massachusetts ; London, England, 2016.
- [21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January 2014.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.
- [23] M. D. Zeiler. ADADELTA: An Adaptive Learning Rate Method. *Computing Research Repository*, 2012.
- [24] Y. LeCun and C. Cortes. MNIST handwritten digit database. 2010. URL <http://yann.lecun.com/exdb/mnist/>.
- [25] T. Hahn and M. Wieler. Weakly Supervised Learning for Industrial Optical Inspection. <https://hciweb.iwr.uni-heidelberg.de/node/3616>, Heidelberg Collaboratory for Image Processing, 2007.

Einbindung von Labormodellen mit industriepraktischen Fragestellungen in der Hochschullehre

Tobias Loose¹, Thomas Pospiech¹

¹Hochschule Heilbronn, Max-Planck-Straße 39, 74081 Heilbronn
tobias.loose@hs-heilbronn.de, thomas.pospiech@hs-heilbronn.de

Dieser Beitrag zeigt eine Reihe von industriepraktischen Anwendungen, wie sie in einer praxisorientierten Hochschullehre als Labormodelle eingebunden werden können. Hierbei werden sowohl kleine, physische Modelle als auch Simulationsmodelle exemplarisch mit industriepraktischen Fragestellungen gezeigt, z. B. Schwingungsdämpfung von Regalbediengeräten, schwappfreies Positionieren von Flüssigkeiten bei Abfüllanlagen. Die Anwendung unterschiedlicher Methoden der Regelungstechnik werden ebenso gezeigt, wie z. B. Zustandsregelung, Entwürfe von Reglern mit klassischen Methoden wie die Auslegung mit dem Bode-Diagramm, Sollwertanpassung. Ein Vergleich der Methoden aus den unterschiedlichen Anwendungen lässt sich damit ebenso gut demonstrieren.

1 Einleitung

Anspruchsvolle mathematisch-technische Fachgebiete wie Regelungstechnik begegnen Studierende aber auch erfahrene Ingenieure mit einem gewissen Respekt. Durch die Verknüpfung vieler Disziplinen zu einer funktionierenden Anlage wird die Regelungstechnik daher auch gerne als Königsdisziplin gesehen, siehe auch [2, 10, 11]. Um Studierenden darin den Weg zu ebnet, eignen sich physische Modelle, um theoretische Erkenntnisse in der Praxis anzuwenden. In diesem Beitrag sind eine Reihe industriepraktischer Fragestellungen und Fallbeispiele gezeigt, die mit kleinen physischen Modellen nachgebildet werden, z.B. Schwingungsreduzierung von Regalbediengeräten, schwappfreies Positionieren von Flüssigkeiten. Dabei werden theoretische Modellbildungen mit physischen Modellen verknüpft inkl. dem darauf aufbauenden Reglerentwurf. Es werden also industriepraktische Anlagen als Labormodelle nachgebaut.

2 Varianten der Modellbildung

Die theoretische Modellbildung ist essentiell, um ein Verständnis für die Realität aufzubauen. Daher ist in der klassischen Hochschullehre die theoretische Modellbildung ein fester Bestandteil, um damit verschiedene Entwurfsverfahren von Reglern zu behandeln.

In Lehrbüchern ist die theoretische Modellbildung naturgemäß die einzige Möglichkeit, um die Regelungstechnik zu beschreiben, siehe Abbildung 1 unten und [2, 10, 11]. Damit können und werden in der

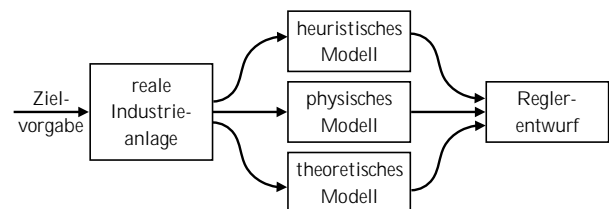


Abbildung 1: Modellbildung für den Entwurf von Reglern

Industriepraxis Anlagen regelungstechnisch ausgelegt und in Betrieb genommen. Bei „einfachen Standard-“ Industrieanlagen hat sich teilweise ein heuristisch-pragmatischer Reglerentwurf vor Ort an der Anlage etabliert, z. B. in Anlehnung an Verfahren nach Ziegler-Nichols, siehe [10]. Dem liegt aber ein heuristisches Modellverständnis zu Grunde, basierend auf theoretischen Erkenntnissen, siehe Abbildung 1 oben.

Aufgrund der teilweisen abstrakt-komplexen mathematischen Beschreibung bei der theoretischen Modellbildung kann mitunter die praxisnahe Anschauung verloren gehen. Um dem entgegen zu wirken, lassen sich in der Hochschullehre physische Modelle einbinden, siehe Abbildung 1 Mitte und z. B. [12]. In diesem Beitrag werden physische Modelle mit Anwendungsbezug zur Industriepraxis als weitere Ergänzung

in der Hochschullehre gezeigt. Damit können Studierende einen ersten Bezug zu industriellen Anwendungen und typischen Fragestellungen herstellen, z. B. wie Schwingungen bei Regalbediengeräten reduziert werden können. Lernerfahrungen und Anwendungen aus Sicht von Studierenden sind exemplarisch in [15] beschrieben.

3 Physische Modelle mit Bezug zur Industriepraxis

Die Schwingungsdämpfung einer **Krananlage** ist ein typisches Beispiel für eine sehr anschauliche regelungstechnische Anwendung, siehe Abbildung 2. Die theoretische Modellbildung und der Entwurf eines Zustandsreglers ist ausführlich in [2] gezeigt. Als physisches Modell lässt sich eine Krananlage gut in der Hochschullehre als Laborübung integrieren, z. B. als Modellbau Anlage wie in Abbildung 3 oder als LEGO® MINDSTORMS® wie in [8] gezeigt.

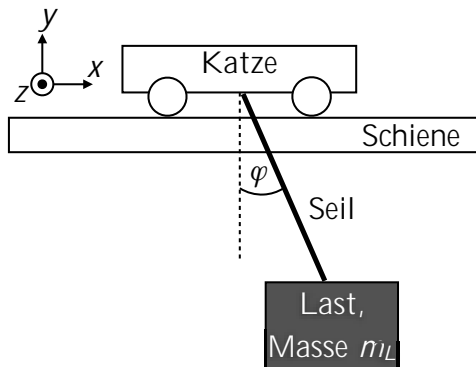


Abbildung 2: Prinzip eines Krans zur Positionierung der Katze in x bei gleichzeitiger Schwingungsdämpfung in φ

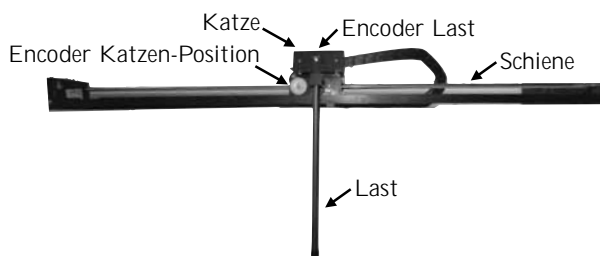


Abbildung 3: Physisches Modell eines Krans

Ein **Regalbediengerät** (RBG) ist ein weiteres Beispiel für eine schwach gedämpfte Industrieanlage, siehe Prinzip in Abbildung 4 und als physisches Vor-

stellungsmodell in Abbildung 5. Hierbei haben sich mehrere regelungstechnischen Methoden etabliert, um die Schwingung zu reduzieren, z. B. Vorgabe bestimmter Eingangssignale, Zustandsregler, Aufschaltungen, ausführlich in [5] beschrieben. Daneben gibt es weitere Industriepraxis-Methoden, die die Regelstrecke optimieren, z. B. zusätzlicher Antrieb an der Mastspitze, der sich an einer weiteren Schiene abstützt.

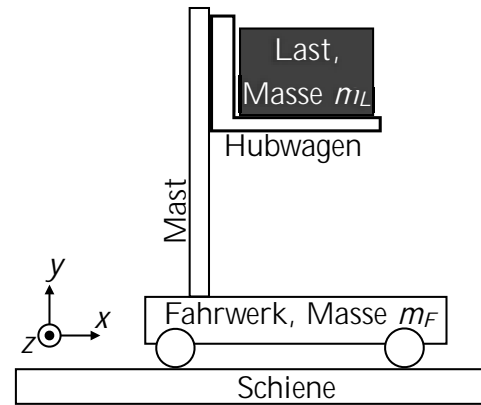


Abbildung 4: Prinzip eines Regalbediengerätes [5]

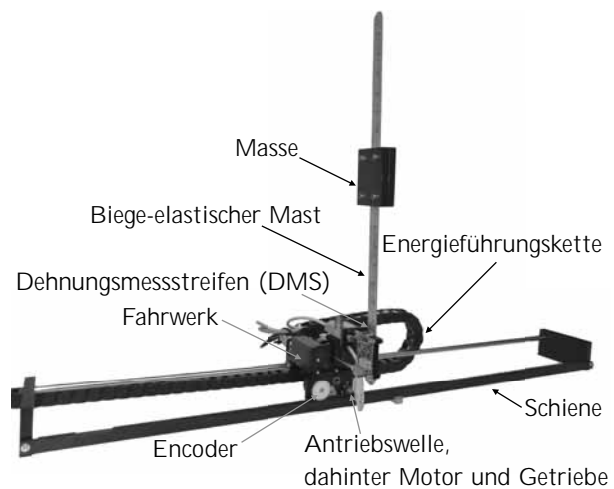


Abbildung 5: Physisches Modell eines Regalbediengeräts [5]

Schwappfreies Positionieren von Flüssigkeiten kommt in der Lebensmittel- und Pharmaindustrie zum Einsatz, siehe Abbildung 6. Diese Aufgabenstellung lässt sich u. a. mit LEGO® MINDSTORMS® oder mit Modellbau-Fahrzeugen nachgestalten, siehe Abbildung 7. Regelungstechnische Lösungen sind in [1, 4, 9] beschrieben, z. B. Input-Shaper oder Beschleunigungsrückführung anhand eines Schätzmodells.

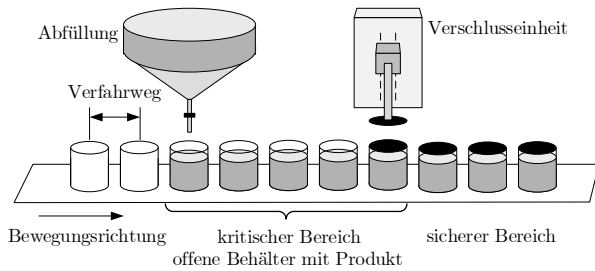


Abbildung 6: Prinzip von Abfüllanlagen [9]

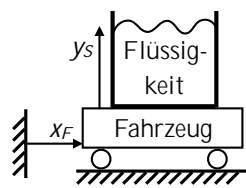


Abbildung 7: Prinzip eines physischen Labormodells für das schwappfreie ($y_S = konst.$) Positionieren x_F

4 Theoretische Modellbildung

Die in Kapitel 3 gezeigten Anwendungen lassen sich pragmatisch mit der Übertragungsfunktion

$$G(s) = \frac{K \cdot \omega_0^2}{s^2 + 2 \cdot D \cdot \omega_0 \cdot s + \omega_0^2} \quad (1)$$

und dem Verstärkungsfaktor K , dem Dämpfungsgrad D sowie der Eigenkreisfrequenz des ungedämpften Systems ω_0 beschreiben. Die Parameteridentifikation kann nun durch

- physikalische Grundgesetze und Materialeigenschaften (anhand Ersatzmodellen wie in Abbildung 8 und 9) oder
- Anregung des physischen Modells und anhand der Sensor-Messwerte (siehe z.B. Abbildung 12)

ermittelt werden. Beide Varianten eignen sich hervorragend als Vergleich für die Implementierung in einer Laborübung.

Darauf aufbauend lassen sich neben den physischen Modellen auch Simulationsmodelle als Laborübung entwerfen, siehe beispielhaft Abbildung 10.

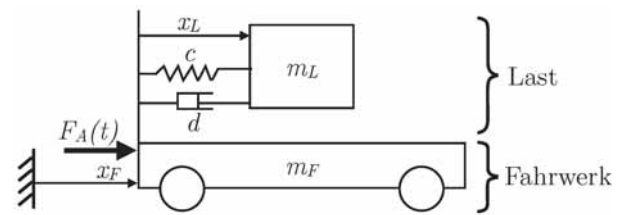


Abbildung 8: Vereinfachtes Modell eines Regalbediengerätes zur Positionierung des Fahrzeugs in x_F bei gleichzeitiger Schwingungsdämpfung des Mastes in x_L [5]

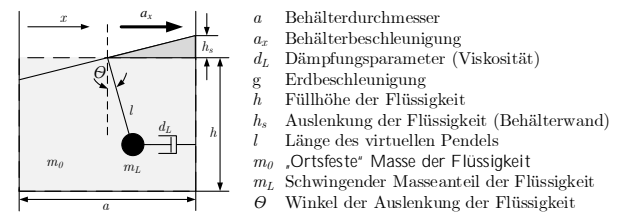


Abbildung 9: Modell einer schwappenden Flüssigkeit [9]

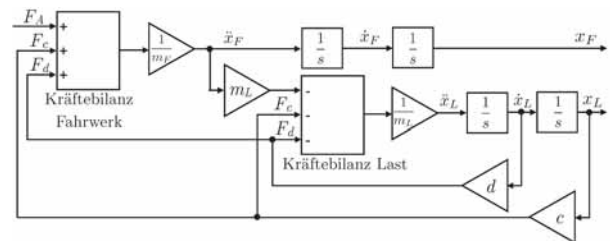


Abbildung 10: Blockschnittbild eines Regalbediengerätes [5]

5 Exemplarische Fragestellungen und Lösungsansätze

Bei den in Kapitel 3 vorgestellten Industrieanwendungen handelt es sich jeweils um stabile, schwach gedämpfte Regelstrecken, die grundlegend durch ein Verzögerungsglied 2. Ordnung beschrieben werden können, siehe Gleichung 1. Das regelungstechnische Ziel dieser Anlagen ist eine einfache Realisierung einer Positionsregelung bei gleichzeitiger Schwingungsreduzierung der Anwendung. Lösungsansätze sind u. a.

- spezielle Eingangssignale wie Input-Shaper oder angepasste Rampenzeiten an die Eigenfrequenz, siehe [3, 4, 5, 9, 13, 16],
- klassischer PID-Regler, z. B. zunächst jeweils als Eingrößenregelung der Positionierung bzw. Schwingungsreduzierung und dann die Kombi-

nation von beiden als Stellgrößen-Aufschaltung bei der Positionsregelung, siehe [5, 9, 10],

- Rückführung von Zuständen, z. B. die Rückführung der Beschleunigung, siehe [9],
- Zustandsregelung, siehe [2, 5].

Beispielhaft ist mit einem Labormodell das Ergebnis zur schwappfreien Positionierung in Abbildung 11 gezeigt. Dabei konnte die Auslenkung der Flüssigkeit deutlich reduziert werden, siehe [9].

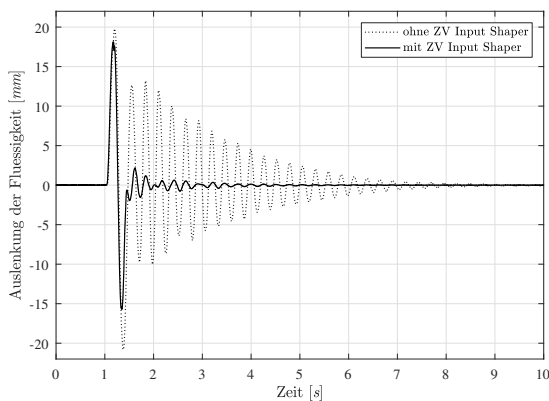


Abbildung 11: Ergebnisse mit einem Input Shaper [9]

Im o. g. regelungstechnischen Ziel der Anlagen wurde bereits die *einfache* Realisierung erwähnt. Damit verbunden ist eine typische Fragestellungen aus der Industriepaxis mit möglichst kostengünstigen Ausrüstungen eine ausreichende Regelgüte zu erzielen. Gerne wird dabei das Weglassen von Sensoren diskutiert, z. B. das Weglassen von Sensoren, die beim Regalbediengerät die Mastschwingungen messen, wie Beschleunigungssensoren oder Dehnungsmessstreifen. Natürlich gibt es auch hier regelungstechnische Lösungen, z. B. Messwertschätzung durch Beobachter oder modellbasierte Messwert-Schätzung, siehe [2, 9, 10]. Allerdings müssen hierbei Ungenauigkeiten, Komplexität oder teilweise sicherheitskritische Aspekte in Kauf genommen bzw. berücksichtigt werden. Daher wird in der Industriepaxis immer wieder gerne über alternative Messwerterfassung mit bereits vorhandenen Sensoren diskutiert. Ein Beispiel dafür ist mit der Strommessung des Elektromotors Rückschlüsse auf das Anlagenverhalten zu ziehen. Im Falle des Regalbediengerätes ist das also durch den Strom des Fahrwerkmotors die Mastschwingungen zu bestimmen. Allerdings ist das auch in realen Industrieanlagen oftmals schwierig, weil durch Reibungseinflüsse die Messung zu ungenau wird. Das kann an-

hand eines Labormodells gut gezeigt werden. In Abbildung 12 oben ist das Schwingungsverhalten mit einem Dehnungsmessstreifen (DMS) aufgenommen und unten die gleiche Schwingung durch den Motorstrom. Es ist deutlich zu erkennen, dass die Messwerte für regelungstechnische Aufgaben kaum zu gebrauchen sind. Es lassen sich allenfalls Anlagenparameter wie die Eigenfrequenz oder mit Abstrichen auch der Dämpfungsgrad ablesen. In diesem Beispiel ist die Reibung im Getriebe zu groß, um verlässliche Ergebnisse zu erhalten.

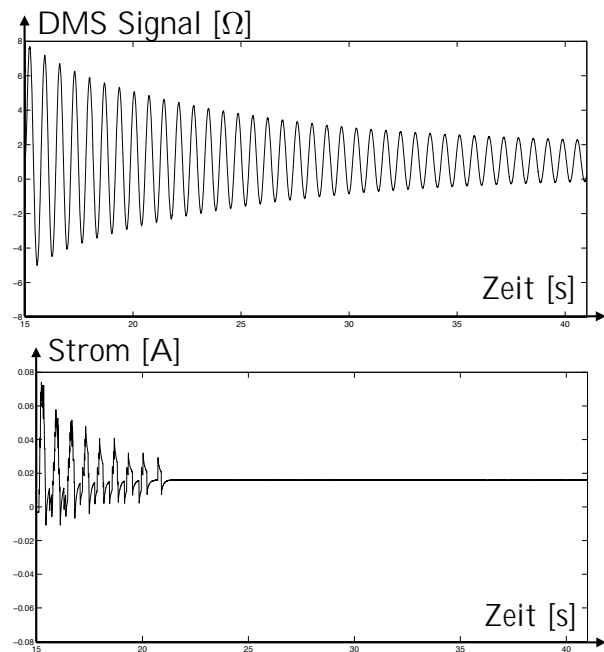


Abbildung 12: Sensormesswerte

Mit den hier dargestellten Industrieanwendungen lassen sich somit sehr anschaulich die Praxisbezüge in der Regelungstechnik im Rahmen der Hochschullehre zeigen. Darüber hinaus lassen sich wichtige Erkenntnisse zur Bewertung von Lösungen bei Industrieanlagen schlussfolgern, wie z. B. die Gewinnung von Informationen durch die Strommessung oder die spezifischen Vor- und Nachteile von Zustandsregelungen bei Industrieanwendungen gegenüber klassischen PID-Reglern, siehe auch [5, 7, 15].

6 Zusammenfassung

In diesem Beitrag ist ein didaktisches Konzept als Ergänzung für die Hochschullehre beschrieben, bei dem regelungstechnische Industrieanwendungen als physi-

sche Labormodelle integriert werden. Damit werden typische Fragestellungen der Industrieanwendungen aufgegriffen und gelöst, z. B. die Schwingungsreduzierung bei der Positionierung von Regalbediengeräten oder von Flüssigkeiten in Abfüllanlagen. Die spezifischen Vor- und Nachteile der einzelnen Lösungsmethoden lassen sich dabei schön anschaulich demonstrieren und durch (mathematische und / oder computergestützte) Modellbildung belegen. Die Labormodelle eignen sich als Demonstratoren in Vorlesungen, aber auch als eigenständige Laborarbeit für Studierende, z. B. mit LEGO® MINDSTORMS®, siehe auch [8, 15].

Literatur

- [1] F. Dodge. *The New Dynamic Behavior of Liquids in Moving Containers*. Research Institute San Antonio, Texas. 2000.
- [2] O. Föllinger. *Regelungstechnik*. VDE Verlag. 2016.
- [3] P. Hubinský, T. Pospiech. *Input Shaping for Slosh-Free Moving Containers with Liquid*. International Journal of Mechanics and Control, Vol. 09, No. 02. S. 27-34. 2008.
- [4] P. Hubinský, T. Pospiech. *Slosh-free Positioning of Containers with Liquids and Flexible Conveyor Belt*. Journal of Electrical Engineering Vol. 61, No. 2. S. 65-74. 2010.
- [5] T. Loose. *Benchmark-Anwendung zur Schwingungsanalyse und -dämpfung von Regalbediengeräten am Beispiel eines Labormodells*. In: *Proc., 25. Workshop Computational Intelligence* (Hoffmann, F.; Hüllermeier, E., Hg.), S. 127–144. Universitätsverlag Karlsruhe. 2015.
- [6] T. Loose. *Projektierung mechatronischer Anlagen als Laborarbeit in der Hochschullehre am Beispiel LEGO® MINDSTORMS®*. In: *Proc., 2. IFToMM D-A-CH Konferenz*, S. 1–8 (Tagungsband auf CD-ROM). Universität Innsbruck. 2016.
- [7] T. Loose: *Reglerentwurf mit klassischen und Computational Intelligence Methoden in der Hochschullehre am Beispiel von Modellen*. In: *Proc., 26. Workshop Computational Intelligence* (Hoffmann, F.; Hüllermeier, E.; Mikut, R., Hg.), S. 261–280. Universitätsverlag Karlsruhe. 2016.
- [8] T. Loose. *Projektierung mechatronischer Anlagen in der Hochschullehre am Beispiel von Labormodellen*. In: *Proc., Workshop 2017 ASIM/GI-Fachgruppen* (Commerell, W.; Pawletta, T., Hg.), S. 162–167. ARGESIM Verlag Wi- en, Hochschule Ulm. 2017.
- [9] T. Loose, Th. Pospiech. *Benchmark-Untersuchung zur Regelung schwach gedämpfter Systeme bei industriepraktischen Anwendungen*. In: *Proc., 27. Workshop Computational Intelligence* (Hoffmann, F.; Hüllermeier, E., Mikut, R., Hg.), S. 175–194. Universitätsverlag Karlsruhe. 2017.
- [10] H. Lutz, W. Wendt. *Taschenbuch der Regelungstechnik*. Europa-Lehrmittel. 2014.
- [11] W. Oppelt. *Kleines Handbuch technischer Regelvorgänge*. Weinheim / Bergstr.: Verlag Chemie. 1972.
- [12] T. Pospiech. *Ball in Balance*. <https://youtu.be/Wkb6QSBYkZwv>. 2017.
- [13] R. Ibrahim. *Liquid Sloshing Dynamics*. University Press, Cambridge. 2005.
- [14] D. Schröder. *Elektrische Antriebe – Regelung von Antriebssystemen*. Springer Vieweg. 2015.
- [15] C. Siegle, Th. Pospiech, T. Loose. *Regelungstechnische Projektierung und Realisierung mechatronischer Systeme am Beispiel von Hochschul-Laboranlagen*. In: *Proc., Workshop 2018 ASIM/GI-Fachgruppen* (Loose, T., Hg.), Hochschule Heilbronn. 2018.
- [16] W. Singhose, L. Pao. *A comparison of input shaping and time-optimal flexible-body control*. In: *Control Engineering Practice*, Volume 5, Issue 4, S. 459-467. 1997.

Lernerfahrung bei der Realisierung eines autonomen Transportfahrzeugs als Regelungstechnik-Laborarbeit

Christoph Obasohan¹, Thomas Pospiech¹, Tobias Loose¹

¹Hochschule Heilbronn, Max-Planck-Straße 39, 74081 Heilbronn

christoph.obasohan@web.de, thomas.pospiech@hs-heilbronn.de, tobias.loose@hs-heilbronn.de

Die Regelungstechnik ist eine sehr anwendungsorientierte ingenieurwissenschaftliche Disziplin, weil letztendlich eine funktionierende, automatisierungstechnische Anlage als Ergebnis herauskommen muss. Daher bietet es sich in der Hochschullehre an, auch komplexe, regelungstechnische Zusammenhänge anhand von Labormodellen in der Praxis umzusetzen. Dadurch haben Studierende die Möglichkeit theoretische Grundlagen in einer praktischen Anwendung zu beobachten und anzuwenden. Dieser Beitrag zeigt, welche Lernerfahrungen durch relativ einfache Labormodelle erzielt werden können. Zum Einsatz kommt dabei LEGO® MINDSTORMS®. Die Erkenntnisse aus Sicht von Studierenden wird hier anhand eines autonomen Transportfahrzeugs gezeigt.

1 Einleitung und Realisierung

Regelungstechnische Laborarbeiten bieten sich hervorragend an, um Vorlesungsinhalte zu vertiefen. Dazu wurde ein Konzept entwickelt, wie mit relativ einfachen Ausrüstungen ein Labor als *Ergänzung* zur Hochschullehre aufgebaut werden kann, siehe [4, 5, 6]. In diesem Kurzbeitrag wird die Lenkungs- und Geschwindigkeitsregelung eines autonomen Transportsystems als Beispiel einer Laborarbeit aus Sicht von Studierenden beschrieben, siehe Abbildung 1. Da-

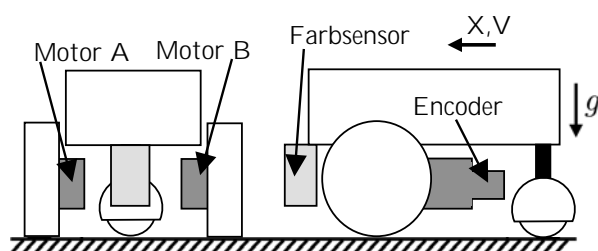


Abbildung 1: Schematischer Modellbau eines Fahrzeug

bei lassen sich mit relativ einfachen Modellausrüstungen regelungstechnische Fragestellungen sinnvoll beantworten. Das Modell des Fahrzeugs besteht aus zwei Elektromotoren A und B mit integrierten Encodern und einem Farbsensor. Fahrzeugwinkel sowie Geschwindigkeit werden über die zwei Motoren beeinflusst, siehe Abbildung 2. Das digitale Sensorsignal

wurde mit einem Tiefpassfilter geglättet, um die Regelung zu verbessern. Der Farbsensor wird für die Lageposition des Fahrzeugs zu einer Fahrbahn genutzt. Dabei ist eine geeignete Sensorposition gewählt worden, um z. B. äußere Störeinflüsse zu reduzieren oder möglichst große Messwertauflösungen und -bereiche zu erhalten, welche die Regelung positiv beeinflusst. Das Ziel dieser exemplarischen Laborarbeit ist es, eine hohe Spurtreue bei geregelter Geschwindigkeit zu erreichen. Dazu ist beispielsweise ein geschwindigkeitsabhängiger P-Anteil zur weitestgehenden Vermeidung „extremer“ Lenkungsbewegungen eingesetzt. Diese Idee ist aus der Automobilbranche entnommen, z. B. als aktive Lenkung bekannt (geschwindigkeitsabhängiger Lenkwinkelschlag). Im Rahmen der Laborarbeit lassen sich die Ergebnisse derartige Lösungen schön anschaulich demonstrieren.

2 Lernerfahrung

Durch derartige Laborarbeiten sind Studierende „gezwungen“ Theoriegrundlagen aus Vorlesungen und Literatur praktisch anzuwenden. Damit steigt die Bereitschaft und die Motivation Bücher und Beiträge zu dem Thema zu lesen, wie beispielsweise [1, 2, 3, 4, 7, 8]. Darüber hinaus lernen sie mit praxisnahen Problemstellungen umzugehen, z. B. Einwirkung

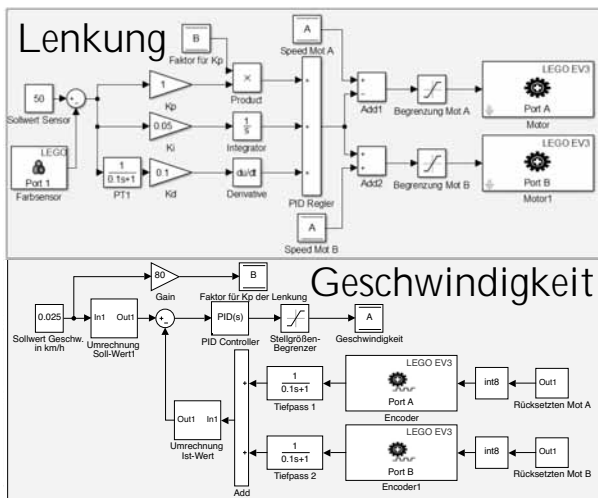


Abbildung 2: Beispielhafte Regelung der Lenkung und Geschwindigkeit

von Störeinflüssen aus der Umgebung, Verbesserung der Regelung durch das Filtern von Signalen oder die Optimierung der Konstruktion. Das *eigenständige* Erarbeiten einer Lösung und die damit verbundenen Herausforderungen einer ingenieurwissenschaftlichen Arbeit werden mit derartigen Laborarbeiten praxisnah erlebt, trotz relativ einfacher Laborausstattung mit LEGO® MINDSTORMS®. Sicherlich gibt es zu derartigen Aufgaben bereits einige Lösungen, siehe Internetrecherche z. B. mit den Suchwörtern „smooth line follower“. In einer derartigen Laborarbeit geht es aber auch darum, *eigene, neue* und bessere Lösungswege zu finden. Das ist eine sehr praxisrelevante Fragestellung für Ingenieure, weil oftmals bereits existierende, sehr gute Lösungen weiter optimiert werden müssen, siehe z. B. die kontinuierlichen Verbesserungen von Automobilen. Ebenso lässt sich die Erkenntnis aus der Laborarbeit mühelos auf andere Anwendungen übertragen. Ein Beispiel hierfür ist die Frequenzanpassung einer Schwinganker-Förderanlage. Nach einer Optimierung der Regelstrecke durch Reduzierung von Massenträgheiten, wurde die Fördergeschwindigkeit gesteigert. Die gesamte Regelung und damit die Versorgung einer Roboterzelle konnte damit verbessert werden.

3 Zusammenfassung

Dieser Beitrag zeigt an einem Beispiel wie mit relativ einfachen Labormodellen praxisnahe, regelungstechnische Projekte mit in die Hochschullehre einbezogen

werden können. Der Transfer auf industriepraktische Lösungen und Anwendungen ist ebenso skizziert.

Literatur

- [1] P. Beater. *Regelungstechnik mit Papier und Bleistift*. Norderstedt: Books on Demand. 2011.
- [2] O. Föllinger. *Regelungstechnik*. Berlin, Offenbach: VDE Verlag. 2016.
- [3] H. Jaschek, H. Voos. *Grundkurs der Regelungstechnik*. München: Oldenbourg. 2010.
- [4] T. Loose: *Reglerentwurf mit klassischen und Computational Intelligence Methoden in der Hochschullehre am Beispiel von Modellen*. In: *Proc., 26. Workshop Computational Intelligence* (Hoffmann, F.; Hüllermeier, E.; Mikut, R., Hg.), S. 261–280. Universitätsverlag Karlsruhe. 2016.
- [5] T. Loose. *Projektierung mechatronischer Anlagen in der Hochschullehre am Beispiel von Labormodellen*. In: *Proc., Workshop 2017 ASIM/GI-Fachgruppen* (Commerell, W.; Pawletta, T., Hg.), S. 162–167. ARGESIM Verlag Wien, Hochschule Ulm. 2017.
- [6] T. Loose, Th. Pospiech. *Einbindung von Labormodellen mit industriepraktischen Fragestellungen in der Hochschullehre*. In: *Proc., Workshop 2018 ASIM/GI-Fachgruppen* (Loose, T., Hg.), Hochschule Heilbronn. 2018.
- [7] H. Lutz, W. Wendt. *Taschenbuch der Regelungstechnik*. Haan-Gruiten: Europa-Lehrmittel. 2014.
- [8] G. Schulz, K. Graf. *Regelungstechnik 1*. De Gruyter Oldenbourg. 2015.

Das Zentrum für Maschinelles Lernen (ZML) an der Hochschule Heilbronn

Alexandra Reichenbach^{1,2*}, Wendelin Schramm^{1,2*}, Nicolaj Stache^{1,3*}

¹Zentrum für Maschinelles Lernen, Hochschule Heilbronn

²Fakultät für Informatik, Hochschule Heilbronn

³Fakultät für Mechanik und Elektronik, Hochschule Heilbronn

*A.R., W.S. und N.S. haben in gleichen Teilen zu dieser Arbeit beigetragen

zml@hs-heilbronn.de

Maschinelles Lernen (ML) ist nicht nur ein Zukunftsthema in der angewandten Forschung, es ist auch ein großes Thema in der Hochschul-Ausbildung. Die Problematik gliedert sich in drei Facetten auf:

1. Es gibt nicht genug qualifizierten Nachwuchs für die eigene Forschung. Es sei denn die Hochschule bildet die jungen Fachkräfte selbst zielgerichtet aus.
2. Es besteht ein großes Interesse der regionalen Industrie an ML-Verfahren und damit einhergehend ein ungenutztes Potenzial für Wissenschaftstransfer und Auftragsforschung.
3. In verschiedenen Fachgebieten wie Automobilindustrie, Medizin und Informatik werden zwar unterschiedliche Fragestellungen bearbeitet, es werden jedoch ähnliche, häufig sogar dieselben Methoden eingesetzt.

Dies führte zur Gründung eines interdisziplinären Lehr- und Forschungszentrums für Maschinelles Lernen (ZML) an der Hochschule Heilbronn (HHN). Seit dem 1. November 2017 wird das ZML vom Bundesministerium für Bildung und Forschung (BMBF) für zwei Jahre gefördert.

Das Zentrum wird sich im Rahmen dieses Artikels mit seiner Zielsetzung sowie seinen Kollaborations- und Ausbildungsangeboten vorstellen. Des Weiteren werden die aktuellen Forschungsschwerpunkte mit ihren spezifischen Herausforderungen bezüglich maschineller Lernverfahren diskutiert.

1 Aktuelle Situation und Zielsetzungen des ZML

Künstliche Intelligenz (KI) und Maschinelles Lernen (ML) sind ein Thema, das praktisch alle Branchen umfaßt. Während für Firmen wie Apple, Google oder Microsoft – Firmen die seit Jahren auf den großen wissenschaftlichen Fachtagungen wie der „NIPS“ im Programmkomitee vertreten sind – diese Techniken längst zur Kernkompetenz gehören, wird das Thema inzwischen für viele anderen Firmen zu einem der wichtigsten aktuellen Themen bezüglich ihrer IT-Infrastruktur. Damit ist ein Bedarf an Spezialisten vorhanden, der aus dem aktuellen akademischen Umfeld nicht mehr gedeckt werden kann. Denn die Arbeit in diesem Umfeld verlangt nach hochqualifiziertem Personal, welches in dem sich schnell entwickelnden Themenkomplex ständig auf dem neusten Stand bleiben muß. Diese Problemstellung verlangt danach, dass Academia und Industrie sparten- und fächerübergreifend an einem Strang ziehen.

1.1 Interdisziplinäre Ausbildung des wissenschaftlich qualifizierten Nachwuchses

Problem: Den an ML interessierten und aktiven Professorinnen und Professoren mangelt es an geeignetem Nachwuchs. Es fehlt weniger an Interessierten, sondern an den adäquaten Kenntnissen. Der Übergang vom Masterstudium in eine wissenschaftliche Tätigkeit oder in die Industrieentwicklung gelingt zu selten nahtlos.

Ziel: Das wichtigste Ziel ist die Ausbildung von wissenschaftlich qualifiziertem Nachwuchs. Mehr Studierende sollen den nahtlosen Übergang vom Masterstudium in die wissenschaftliche Forschung oder in die Entwicklungsabteilungen der Industrie meistern.

1.2 Plattform für interdisziplinäre Steuerung der vorhandenen Ressourcen schaffen

Problem: Unkoordinierte Lehre und Forschung führen zu unnötigen Doppelveranstaltungen und -

beschaffungen, aber auch zu Lücken in Wissen und Ausstattung. Know-How zerfasert und wichtige Techniken werden nicht von einer kritischen Masse an Mitarbeitenden beherrscht. Mangelnde Kommunikation führt zu Informationsverlust und unkoordinierten Alleingängen.

Ziel: Das ZML soll der Koordination der an der Hochschule eingesetzten Technologien und dem interdisziplinären Austausch dienen.

1.3 Entwicklung eines generischen Toolkits für Maschinelles Lernen

Problem: Alle Fachgebiete beschäftigen sich mit den gleichen oder ähnlichen Problemen und erfinden doch ständig das Rad neu.

Ziel: Die Hardware und Softwareausstattung der HHN soll typische Problemstellungen des ML fachübergreifend lösen helfen. Dazu wird ein generisches Toolkit entwickelt, das sich auf vielfältige Klassen von Problemen aus beispielsweise Ingenieurwissenschaften, Medizin und Informatik anwenden lässt.

1.4 Konsolidieren und Ausweitung der bestehenden Datenbasen

Problem: ML benötigt nicht nur viele Daten, sondern auch immer wieder neue Daten für wichtige Lern- oder Arbeitsschritte. Hierzu zählen z.B. die Vorverarbeitung von Daten für ML, Daten-Transformation, Entwicklung und Untersuchung von Methoden zur Daten-Augmentierung für lernende Verfahren etc.

Ziel: Bereits heute zur Verfügung stehende Datenquellen, wie z. B. klinisches Krebsregister der

SLK-Kliniken, neurologische Patientendatenbanken oder Fahrzeugdaten für das automatisierte Fahren, sollen robust, im Falle von Gesundheitsdaten auch datenschutzkonform, d.h. ausreichend pseudonymisiert und/oder anonymisiert, zur Verfügung stehen. Es sollen eigene Datenerhebungen durchgeführt werden, um die Datenbasis auszuweiten.

1.5 Anwendungsnahe Forschung und Wissenschaftstransfer

Problem: Der hohe Bedarf an qualifizierten ML-Fachkräften in der Wirtschaft ist bisher ungestillt. Fachkräfte können ohne Einschalten von Headhuntern praktisch nicht mehr gewonnen werden.

Ziel: Das ZML ist u.a. durch den interdisziplinär besetzten Beirat (Abb. 1) eng mit regionaler Industrie in der sehr wirtschaftsstarken Region Heilbronn-Franken – der „Region der Weltmarktführer“ verzahnt und unterstützt die Industrie durch angewandte Forschung und Wissenschaftstransfer.

1.6 Gesamtkonzept des ZML

Das ZML bildet einen physischen Anlaufpunkt (Labor) für die Ausbildung und Forschung im Bereich ML an der Hochschule Heilbronn. Das Zentrum wird initial von den Fakultäten für Informatik und für Mechanik und Elektronik unterstützt, diese betreiben das ZML gemeinsam mit dem Ziel die Lehre mit Forschungsinhalten anzureichern und forschungsrelevante Kenntnisse und Fertigkeiten (Abb. 1) den Masterstudierenden zu vermitteln. Ein aus Wissenschaftlern und Industrievertretern gemeinsam besetzter Beirat

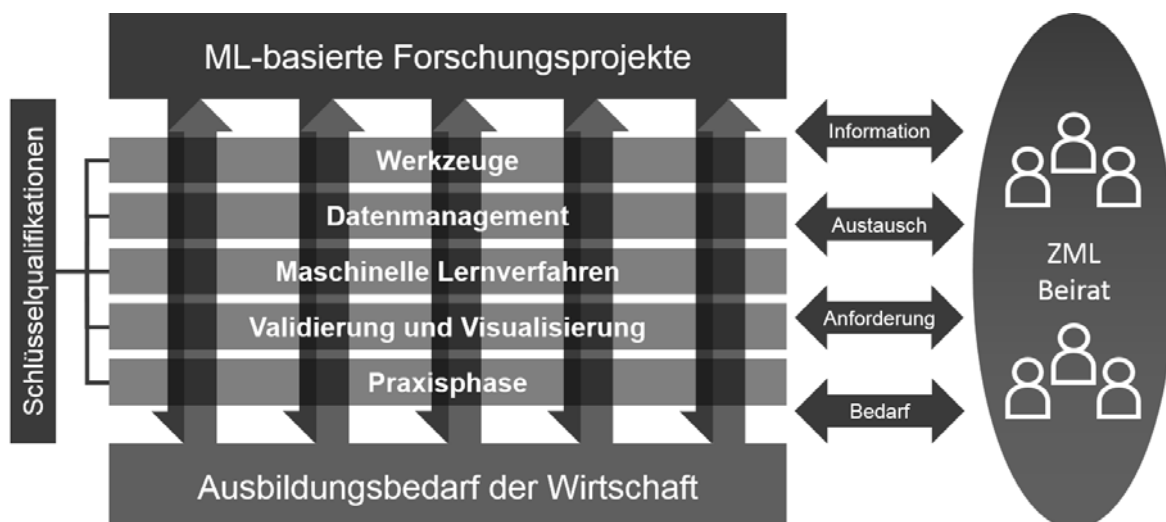


Abbildung 1. Konzept des ZML. Die Qualifizierungsmaßnahmen sind in fünf Schlüsselqualifikationen untergliedert und orientieren sich am Bedarf der Wirtschaft sowie an den ML-basierten Forschungsprojekten der HHN. Der mit Wissenschaftlern und Industrievertretern besetzte Beirat stellt den hohen Praxisbezug in Lehre und Forschung sicher.

fördert den wechselseitigen Wissenschaftstransfer in dem er über die Ausbildung und Forschung an der HHN informiert und in diesem Gremium der Bedarf der Wirtschaft gegengekoppelt wird (Abb. 1). Drei initial benannte Forschungsprojekte definieren den Qualifizierungsbedarf forschungsnah. Die Zentrumsausstattung wird fakultätsübergreifend abgestimmt.

2 Maschinelle Lernverfahren am ZML

Das ZML vermittelt fünf Schlüsselqualifikationen (Abb. 1):

1. **Werkzeuge:** wie Datenbanksysteme, Software und Softwarebibliotheken (z.B. TensorFlow, scikit-learn), Programmiersprachen (z.B. Python)
2. **Datenmanagement:** Vorbehandlung und Prüfung von Daten zur Eignung auf ML
3. **Maschinelle Lernverfahren:** die eigentlichen Algorithmen des ML
4. **Validierung und Visualisierung:** Prüfen, Erklären, Darstellen und Interpretieren von automatisch generierten Ergebnissen
5. **Praktische Anwendung:** praktische Beherrschung aller zuvor genannten Methoden mit Einbezug des ZML Toolkits

Das ZML führt in Abstimmung mit dem Beirat Standard-Werkzeuge für ML ein, die sich in anwendungsnaher Forschung und Industrie (z. B. Alphabet (vormals Google), ...) etabliert haben und noch nicht angemessen in der Lehre an der Hochschule vertreten sind. Als wichtiges Werkzeug kommt die Programmiersprache Python inkl. spezieller Bibliotheken für Datenverarbeitung und ML wie scikit-learn, pandas, TensorFlow, etc. zum Einsatz. Mit diesen Werkzeugen entwickelt das ZML ein generisches Toolkit, welches auch die Anbindung von Sensorik und Datenverarbeitungsroutinen umfasst. Der ML-Fokus liegt auf:

- **Künstliche Neuronale Netzwerke:** Deep Learning, Convolutional Neural Networks
- **andere überwachte Klassifikationsverfahren:** z.B. Support Vector Machine (SVM), Random Forest, Lineare Diskriminanzanalyse (LDA)
- **unüberwachte Lernverfahren:** z.B. Clustering, Hauptkomponentenanalyse (PCA), Unabhängigkeitsanalyse (ICA), Regressionsverfahren

- **Modell-Validierung:** Hierzu werden aus den vorhandenen Daten Trainingsdaten, Validierungsdaten und Testdaten abgespalten.
- **Modell-Visualisierung:** Entscheidungsgrenzen und Datenmerkmale visuell darstellen, damit Lernverfahren von den Trainingsdaten ausreichend abstrahieren und auch auf unbekanntensätzen eine nachvollziehbare Entscheidung treffen können.

Das ZML beschränkt sich nicht auf das Verwenden fertiger oder frei verfügbarer Datenbanken, sondern erhebt auch selbst Daten und speist diese in Lehre und Forschung mit ein. Hierfür steht beispielsweise das Testfeld Autonomes Fahren BW zur Verfügung.

3 Qualifizierungskonzept des ZML

Fakultätsübergreifend sollen wichtige Lerninhalte abgestimmt werden und ML-Veranstaltungen sollen für Studierende aller Fachrichtungen geöffnet werden.

Das ZML etabliert zusätzliche praktisch ausgerichtete Veranstaltungen, die fakultätsübergreifend für Masterstudierende geöffnet sind. Hochschulintern wird es ein neues **ML-Basisseminar** und eine forschungsnah **ZML-Projektarbeit** geben (zusammen 16 ECTS). Für Studierende, aber auch im Rahmen des Studiums Generale und für die interessierte Öffentlichkeit, gibt es eine **ML-Ringvorlesung** mit dem Ziel die Bedeutung des ML vor dem gesellschaftlichen Kontext zu beleuchten. Die Technologien und Einsatzzwecke des ML berühren Fragen des Datenschutzes und der Ethik und haben das Potenzial, gravierende gesellschaftliche Veränderungen zu bewirken; beispielsweise durch den Ersatz von Arbeitsplätzen in Industrie 4.0 Umgebungen.

Um die Industrie einzubinden und Studierenden einen alternativen Zugang zu den fünf ML-Schlüsselqualifikationen zu bieten, wird es eine einwöchige **ML Winter-School geben**, die im Sinne eines Crashkurses Studierenden und Firmenmitarbeitern die wichtigsten Kenntnisse vermitteln wird. Eine zusätzliche **Summer-School** im Umfang von zusammen zwei Wochen mit zusätzlich zu lösenden Projektaufgaben gibt die Möglichkeit an einem ausgewählten Forschungsthema praktisch mitzuarbeiten (learning-by-doing).

Das erfolgreiche Absolvieren von mindestens Ringvorlesung und beider Schools, alternativ aller

Masterveranstaltungen, soll mit einem *Zertifikat Maschinelles Lernen* honoriert werden. Ziel des ZML ist es weiter, ein Konzept zu einem berufsbegleitend erwerbbaaren anerkannten Hochschulabschluss zu entwickeln.

Generell produziert das ZML als Ergebnis der praktischen Übungen und Seminare eine Vielzahl von Datensätzen, Lehrmaterialien, Tools in Form von Programmcode, Video-Tutorials und vieles mehr. Diese werden zukünftigen Studierenden, aber auch anderen Wissenschaftlern und der Industrie zur Verfügung stehen. Damit haben die Studierenden, aber auch am ZML interessierte Industriemitarbeiter, die Möglichkeit, außerhalb der Veranstaltungen den Lehrstoff selbständig zu lernen und zu rekapitulieren.

4 Forschungsschwerpunkte des ZML

Die initialen Forschungsschwerpunkte des ZML ergeben sich aus den Forschungsgebieten der drei Gründungsmitglieder des Zentrums. Aufgrund des integrativen Konzepts des ZML werden sich die Forschungsfelder erweitern und es werden neue Themen aus der HHN und der Industrie hinzukommen. Im Folgenden werden die drei aktuellen Forschungsschwerpunkte mit ihren spezifischen Herausforderungen vorgestellt.

4.1 Autonomes Fahren (N. Stache)

Die Hochschule Heilbronn ist beteiligter Partner am Testfeld Autonomes Fahren Baden-Württemberg und somit in einem Bereich sehr engagiert, in dem ML-Verfahren regelmäßig eingesetzt werden. Kritische Fahrsituationen sind für die Weiterentwicklung automatisierter Fahrzeuge erforderlich. Ein manuelles Aufspüren dieser Situationen ist in den mehrere Millionen an Fahrkilometern umfassenden Datenbeständen zu aufwändig. In kritischen Situationen zeigt der Mensch Überlastungsreaktionen, die messtechnisch erfasst werden und ein erstes Indiz für eine kritische Verkehrssituation sein können [1]. ML soll dazu verwendet werden, den Zusammenhang zwischen Verkehrssituation und Überlastsituation eines Durchschnittsfahrers herzustellen. So ist es bei Abschluss dieses Themas möglich, mit aufgezeichneten Daten der Verkehrssituation [2] (also ohne menschlichen Fahrer) eine Aussage zu treffen, ob kritische Verkehrssituationen vorliegen oder nicht. Herausforderungen liegen hier unter anderem bei der reliablen automatisierten Erkennung einer Stress-

Situation beim Menschen sowie der Extraktion der relevanten Faktoren aus den komplexen Umfelddaten.

4.2 Gesundheitsökonomische Modellierung (W. Schramm)

Bei lebenslangen Erkrankungen wie Diabetes, Asthma und zunehmend auch bei Krebs können medizinische Ergebnisse und die damit verbundenen Behandlungskosten im Rahmen von klinischen Studien nicht ausreichend lange beobachtet werden. Für die gesundheitsökonomische Bewertung von Nutzen und damit verbundenen Kosten ist man auf Simulationsmodelle angewiesen. Die zunehmende Datenverfügbarkeit in der Medizin öffnet neue Möglichkeiten in der Bewertung von Gesundheitstechnologien (engl. Health Technology Assessment - HTA). Seit 2012 besteht eine vertraglich abgesicherte Kooperation mit dem Tumorzentrum der SLK-Kliniken Heilbronn mit Zugang zu Behandlungsdaten von mehr als 60.000 Patientenverläufen. Nach Klärung von Plausibilitätsfragen und der Repräsentativität dieses klinischen Registers konnte 2016 der Beleg geführt werden, dass mit Methoden des Data Minings semi-automatisch Erkrankungsmodelle für Brustkrebs erstellt werden können [3]. Hierfür ist derzeit noch die Bedienung eines kompetenten Arztes notwendig, als nächstes Ziel soll die Modellerstellung mit Hilfe von ML vollautomatisiert werden. Herausforderungen liegen hierbei unter anderem bei der Erschließung weiterer klinischer Datenquellen (*Real World Data*) und der Verbesserung und Validierung der Prognosemodelle.

4.3 Biomarker für Gehirnerkrankungen (A. Reichenbach)

Komplexe Erkrankungen des Zentralnervensystems wie Schizophrenie und Autismus, für welche die biologische Validität der Diagnose noch nicht etabliert ist, stellen eine große Herausforderung für Früherkennung, Individualdiagnostik, Therapieentwicklung und -verfolgung dar. Verhaltensbezogene und bildgebende Biomarker bieten die Möglichkeit nicht-invasiver Erfassung und Quantifizierung spezifischer Phänotypen [4]. Inzwischen stehen sehr große Patienten-Datenbanken mit bildgebenden, kognitiven und genetischen Datensätzen öffentlich zur Verfügung [5]. Diese Datensätze liegen überwiegend in sehr frühen Vorverarbeitungsstadien vor, nicht zuletzt da sie besondere Herausforderungen an Vorverarbeitung und

ML stellen. In diesem Projekt liegt ein starker Fokus auf der (semi-)automatischen Vorverarbeitung der verschiedenartigen Datensätze mit dem Ziel, eine generische und konfigurierbare Vorverarbeitungs-pipeline mit strengen Prüfverfahren bzgl. der Datenqualität zu implementieren. Im zweiten Schritt werden die in dem Gebiet gängigen Verfahren des ML implementiert. Das übergeordnete Ziel des Projektes ist es, der medizinischen Forschungscommunity einen vielfältig einsetzbaren Werkzeugkasten anzubieten sowie ein robustes Framework für die Lehre und weitere Forschungsvorhaben zu etablieren.

5 References

- [1] Healey, J.A. und R.W. Picard. *Detecting stress during real-world driving tasks using physiological sensors*. IEEE Transactions on intelligent transportation systems, 2005. 6(2): S. 156-166.
- [2] Punke, M., et al., *Kamera-Hardware*, in Handbuch Fahrerassistenzsysteme. 2015, Springer. S. 347-368.
- [3] Pobiruchin, M., et al., *A method for using real world data in breast cancer modeling*. Journal of biomedical informatics, 2016. 60: S. 385-394.
- [4] Woo, C.W., et al., *Building better biomarkers: brain models in translational neuroimaging*. Nat Neurosci, 2017. 20(3): S. 365-377.
- [5] Eickhoff, S., et al., *Sharing the wealth: Neuroimaging data repositories*. Neuroimage, 2016. 124(Pt B): S. 1065-8.

Predictive IT-Service Operation

Sigurd Schacht¹, Carsten Lanquillon¹, Simon Vollert¹, Naftali Mishiev¹,
Artjom Fink¹, Roland Licht¹, Bianca Parusel¹

¹ Heilbronn University of Applied Sciences, Business Information Systems
sigurd.schacht@hs-heilbronn.de

IT service operation aims to efficiently and effectively provide IT services. We attempt to predict future incidents and problems based on historic help desk data in order to optimize the trade-off between response time and costs. A major steps towards this goal is to discover semantically meaningful ticket clusters. In this report, we evaluate several approaches to find these clusters. Word2Vec find some useful clusters while poor data quality seems to be the major challenge in the help desk domain.

1 Introduction

IT service operation aims to efficiently and effectively provide IT services. In doing so, incident management describes the activities in an organization to identify and handle any disturbances of IT services, whereas problem management (PM) identifies the root cause of service issues identified by incident management to prevent future recurrences. The main issue of service operation is to manage the tradeoff between short response times to incidents and problems, which are mostly handled by manpower, an economic approach to reduce the cost per incident or problem, and service quality. In this report, we attempt to predict future incidents and problems based on historic help desk data. This allows resources to be planned more effectively and, thus, the trade-off between response time and costs to be optimized [1].

2 Goal and Scope

Predicting future issues based on historic data is a well-studied task in the field of predictive analytics. Undoubtedly, prediction performance will increase with more data being available, ever increasing computing power and methodological advances.

Working with real help desk data requires intensive and careful data preprocessing to make all available data fields of various forms and types amenable to common learning algorithms [2].

The goal of this paper is to empirically explore how to predict IT incidents based on historic help desk data and other data sources such as news feeds and employee statistics in order to develop an early warning system for IT service operation [3].

The analysis refers to the IT service operation stated by ITILV3 focusing on incident and problem management. Cultural specialties, time zones and further user-information will not be considered. The data set contains about 40,000 incidents and problems collected over a period of six years by a medium-sized enterprise using a central service desk [4].

3 Approach

We will apply different approaches to appropriately preprocess the raw help desk data. In particular, we perform cluster analysis for the incidents and problems using text mining approaches such as natural language processing and topic modeling. The first goal is to discover relevant categories which are often not recorded by the employees. An additional major challenge is rather poor data quality. The second goal is to use these categories to interactively gather supplemental news data so as to help increase prediction performance. In this report, only the first goal will be addressed.

4 Analysis

We follow the cross-industry standard process for data mining (*CRISP-DM*) as a well established process model for structuring and conducting data analysis tasks based on six proven phases covering the whole project [5].

4.1 Business Understanding

This work focuses on a local, medium-sized business which already uses a central service desk. Cultural specialties, time zones and further user information will not be considered. The analysis refers to the IT service management stated by ITILv3. The analysis goal is to identify clusters in help desk data of rather poor data quality.

4.2 Data Understanding

We have a database of about 15,000 tickets from 2013 up to the present. The tickets are primarily used in the IT service operations and do not follow any defined format. Even categories or priority flags are not used at all or in a consistent manner. Hence, the data requires intensive cleaning and enrichment to improve the clustering process which will primarily be based on the ticket text bodies.

A first exploratory analysis revealed that 14,189 tickets went through 148,122 state changes. Considering that 2,066 tickets were closed immediately without being opened, the each of the remaining tickets changed its state about 12 times. 1,073 tickets were opened five times and 837 tickets were opened six times without ever being closed. In contrast, 640 were closed successfully twice without ever being opened. Moreover, there are tickets that have been opened 27 times and, finally, were closed successfully without any further steps. The largest number of events for one ticket is 61 state changes. 13,329 tickets were closed within 14 days and 2,622 of those were open for 0 milliseconds. The mean ticket duration is 6 days. The longest open ticket was open for 4 years and 20 days. The largest number of state changes per ticket and day is 337 on October 5th, 2014.

To get a better understanding of the ticket text bodies, we analyzed the word being used. We identified 60,994 unique words with total word count of 1,639,566. Abb. 1 shows the word counts for the ten

most frequently used words. According to Zipf's law, many words were rarely used. For example, "aabbf", "aabeccdaf" and "aazvnblevwdk" occurred only once.

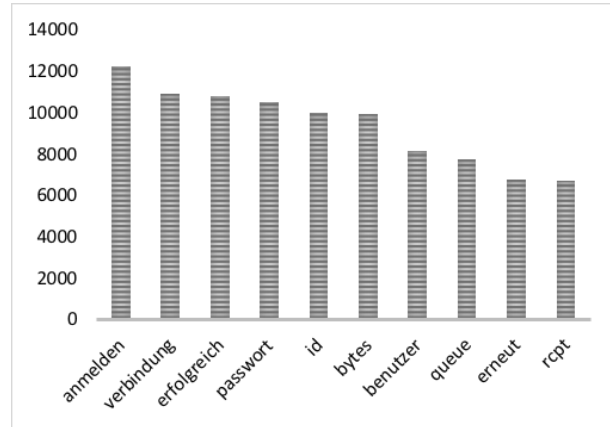


Figure 1: Word counts of the ten most common words.

4.3 Data Preparation

In the following, we applied some common text pre-processing steps to reduce complexity, i.e. to fight the curse of dimensionality: conversion to lower case, stop word removal, removal of multiple white spaces, removal of html tags, quotes and numbers and substitution of special German characters. Next, we manually constructed a domain specific stop word list which is used in addition to the common stop words.

Finally, the remaining words are converted to vectors based on the tf-idf weighting scheme which considers not only the frequency of a term but also its frequency with respect to the other ticket text bodies (document frequency). In addition, terms with extreme document frequency values were filtered out. Thus, the ticket texts are converted into a matrix that contains all the features from the ticket text bodies [6].

4.4 Modeling

In the modeling phase, we explore the following clustering approaches to find useful clusters: k-means, DBSCAN and Word2Vec.

4.4.1 K-Means

The k-means cluster algorithm iteratively partitions the available data points into k groups where k is a pa-

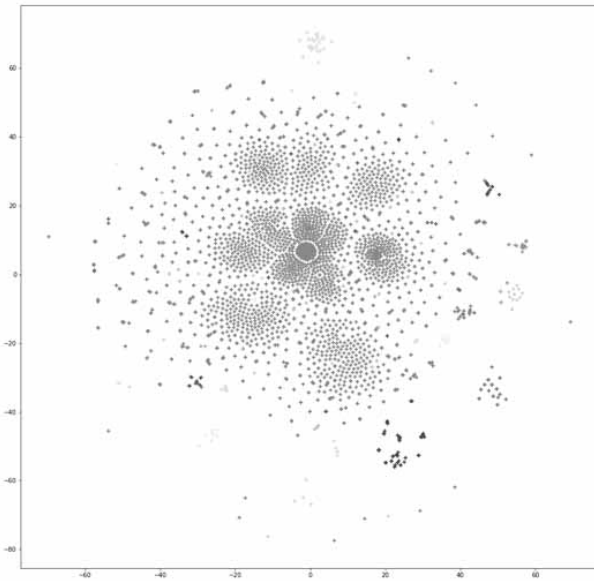


Figure 2: Clustering based on k-means with $k = 30$

parameter given by the user. Each cluster is represented by the centroid (mean) of all data points assigned to it. The k-means algorithm is rather fast. The major drawback is its dependency on the parameter k . Another problem is the initial definition of centroids which is typically made randomly and has a strong impact on the resulting clusters [7]. Approaches like *k-means+* try to improve clustering results by finding better cluster initializations.

We run the k-means+ algorithm with $k = 30$ clusters and 100 different initializations for at most 300 iterations per clustering run and report the best result. To better visualize the resulting clusters, as in Figure 2 the tfidf vectors need to be converted to 2d vectors [8].

4.4.2 DBSCAN

Density-Based Spatial Clustering of Applications with Noise (*DBSCAN*) is a density based clustering algorithm. This algorithm does not need the number of clusters as parameter, instead it discovers clusters based on the density of the data points. The two fundamental parameters of the DBSCAN algorithm are the radius of the neighborhood (epsilon) and the minimal number of points which must be in a neighborhood to define a cluster (minPts).

Based on these parameters, the data points are divided in three categories. Core points are points, where the number of data points in the range of epsilon matches

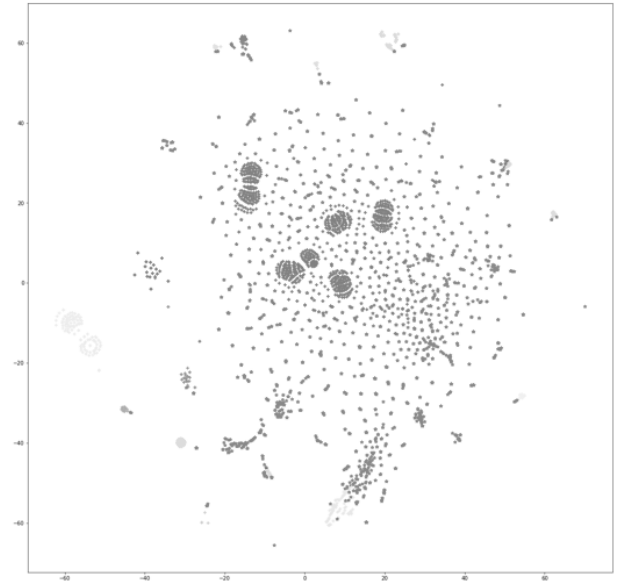


Figure 3: Clustering based on DBSCAN

at least the minPts. Border points are points which are not core points but are in the neighborhood of another core point. Outliers are neither core points nor border points which are converted into a distance matrix [12]. Figure 3 shows the result of DBSCAN.

4.4.3 Word2Vec

Word2Vec (W2V) is a group of models used to produce word embeddings [13]. Word embedding is a technique in which words or sentences of a vocabulary are translated into vectors consisting of real numbers [15]. As depicted in Figure 4, W2V consists of a 2-layer flat neural network that is used to understand the context of words [17].

As an input it gets a text corpus (here the a-body) and produces from the words of the text corpus a vector space with several hundred dimensions (here 100-300 dimensions) [18].

In this case, every single word of the text corpus (meaning the entire vocabulary) is assigned a vector in the vector space [19]. We use W2V based on the continuous skip-gram architecture which uses the current word to open the surrounding context (context window). As shown in Figure 5, the W2V skip-gram model consists of an input vector, a hidden layer and an output layer with softmax classification. The hidden layer is used to translate the weights of the hidden layer into the word vectors [16].

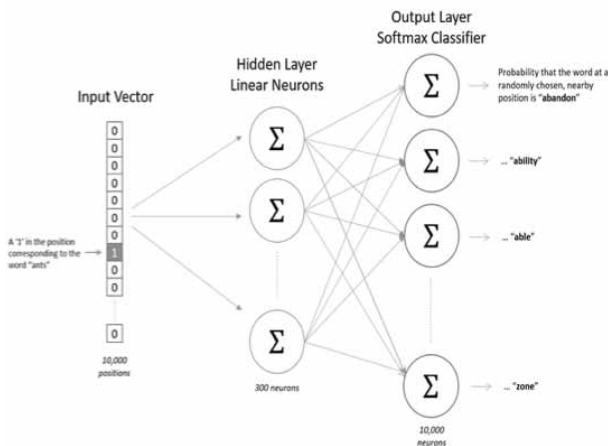


Figure 4: W2V 2-layers flat neural network

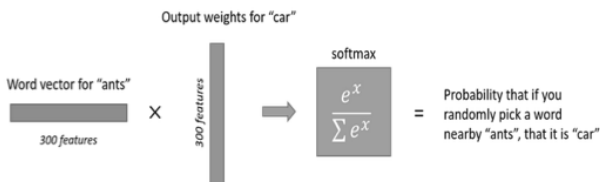


Figure 5: Word vector to word probability

There are several implications of W2V in our project study. The first attempt to use W2V was based on completely unadjusted data. The second attempt at slightly adjusted data (without special characters, HTML, URLs). The third attempt ran on slightly clean data without names.

In addition, the model was operated with several parameters, but the training algorithm was always hierarchical softmax. Context size should depend, among other things, on the length of the sentences to be processed as well as their complexity. Average length of our sentences is 16 words. The author recommends a window size of about 10 for the skip-gram model [22].

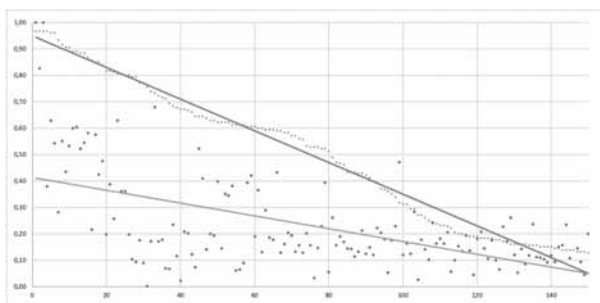


Figure 6: Word vector to word probability

4.5 Evaluation

The main task of the evaluation phase is to evaluate to what extent the results meet the business objectives and the expected or desired results. Furthermore, it must be evaluated whether these results can be successfully deployed. Finally, it should be clarified if there is room for improvement and in what way the project can be optimized [27].

Evaluating clustering results is difficult task: Different methods may yield very different results and there is no right or wrong the clustering result representing some unknown structure discovered [23].

WordSim-353 [24] and SimLex-999 [25] inspired us to evaluate the quality of correlation results of pairs of words by applying human judgment. This procedure was built in the following way: 150 pairs of words from the statistical population of tickets vocabulary were estimated by 18 people for the correlation among themselves on a scale of 1 to 10 (unrelated - 10 related), in addition related pairs of words are also evaluated by algorithm and then these values are compared [26].

5 Results

As observed in Figure 2, the k-means algorithm is successful in creating 30 clusters with small within-cluster heterogeneity. The downside of the k-means algorithm is that it has trouble dealing with the high level of noise in our data. The clusters seem to be reasonable with respect to human judgment, but it is difficult to find analogies between words.

To our surprise we could not observe a positive effect of the vector dimensions on either the word pair relations / Pearson correlation nor the word pair similarity test in the Word2Vec model. Additionally, it was observed that high-dimensional word embeddings perform generally worse on the similarity test opposed to less dimensional word embeddings. This could be due to the nature of our data, which is strongly limited in terms of vocabulary. Our vocabulary consists of about 19.000 distinct words. Therefore, it cannot possibly cover the entirety of the word analogies present in the word similarity test [12, 26].

The efforts of negative sampling to reduce noise in the data set did not produce any significant improvements. On the other hand, the subsampling of frequent words

helped to keep frequent words with low discriminative power out of the clusters. The effect of the context window is noticeable in both the relatedness of word pairs as well as in the analogy task. It was observed that a windows size of 8 words seems to be an optimal value to increase performance in the relatedness whereas window sizes of 5 and 10 performed noticeably better in the word analogy task [28].

Our W2V skip-gram model performed well on the analogy task given the extremely limited vocabulary, the small number of total words in our dataset and the difficulty of the German language in such tasks. Comparing the results of our model to the results of a W2V-model trained on a Wikipedia dataset with a vocabulary size of over 103.000 words and a total word count of over 6 billion, it is impressive that our model performed this well. Looking at the results of [29] which compared different W2V models in different language settings, we can see that the Wikipedia model mentioned before achieved an accuracy in the analogy task of at most 1.17%.

When taking a closer look at the analogy task we can see where our model performed best and worst. Our model scores very poorly in the categories of capitals, nouns and adjectives (0 %) whereas it performs well in the category of verbs (10 %).

To summarize the results of the W2V skip-gram model, neither an increase in dimension nor an increase in training time can produce better results than the default values of the model. Subsampling is necessary to keep unwanted stop words out of the clusters while negative sampling could not reduce noise in our example.

6 Conclusion

To summarize the results of our clustering methods, it is possible to create reasonable clusters according to word relatedness, word analogy and human judgment. After trying many clustering methods Word2Vec turned out to be the most promising method for the correct identification of semantic relatedness and delivered useful clusters from a human point of view. Regardless of our efforts we were not able to obtain a useful base for our prediction model, nevertheless our clusters, even considering the poor data base, delivered appropriate results.

Our approach might be a reasonable solution to clus-

ter issues of semantic incident data. The main reason for the complicated process of clustering is the poor quality and low quantity of data.

In order to make predictions outside of classification tasks it would be necessary to enrich the cluster data with external information. An example would be a regression task in order to inform the user of a potential difficulty and the estimated time to ensure optimal forecasting of human resource deployment.

References

- [1] J. L. Hellerstein, Fan Zhang, and P. Shahabuddin. An approach to predictive detection for service management. In *Integrated Network Management VI. Distributed Management for the Networked Millennium. Proceedings of the Sixth IFIP/IEEE International Symposium on Integrated Network Management. (Cat. No.99EX302)*, pages 309–322, 1999.
- [2] S. M. Weiss, B. F. White, C. V. Apte, and F. J. Damerau. Lightweight document matching for help-desk applications. *IEEE Intelligent Systems and their Applications*, 15(2):57–61, 2000.
- [3] S. Kikuchi. Prediction of workloads in incident management based on incident ticket updating history. In *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, pages 333–340, 2015.
- [4] M. Yun, Y. Lan, and T. Han. Automate incident management by decision-making model. In *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pages 217–222, 2017.
- [5] Colin Shearer. The CRISP-DM Model: The New Blueprint for Data Mining. *Journal of Data Warehousing*, 5(4), 2000.
- [6] Li-Ping Jing, Hou-Kuan Huang, and Hong-Bo Shi. Improved feature selection approach tfidf in text mining. In *Proceedings. International Conference on Machine Learning and Cybernetics*, volume 2, pages 944–946 vol.2, 2002.
- [7] J. De Weerd, S. K. L. M. vanden Broucke, J. Vanthienen, and B. Baesens. Leveraging process discovery with trace clustering and text mining for intelligent analysis of incident management processes. In *2012 IEEE Congress on Evolutionary Computation*, pages 1–8, 2012.

- [8] Gaurav Jain, A. Ginwala, and Y. A. Aslandogan. An approach to text classification using dimensionality reduction and combination of classifiers. In *Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration, 2004. IRI 2004*, pages 564–569, 2004.
- [9] A. El-Hamdouchi and P. Willett. Hierarchic document classification using ward’s clustering method. In *Proceedings of the 9th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’86*, pages 149–156, New York, NY, USA, 1986. ACM.
- [10] B. Borah and D. K. Bhattacharyya. An improved sampling-based dbscan for large spatial databases. In *International Conference on Intelligent Sensing and Information Processing, 2004. Proceedings of*, pages 92–96, 2004.
- [11] Miin-Shen Yang and Kuo-Lung Wu. A similarity-based robust clustering method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(4):434–448, 2004.
- [12] W. Zirkel and G. Wirtz. A process for identifying predictive correlation patterns in service management systems. In *2010 7th International Conference on Service Systems and Service Management*, pages 1–6, 2010.
- [13] Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. Problems with evaluation of word embeddings using word similarity tasks, 2016.
- [14] Francois Chaubard, Rohit Mundra, and Richard Socher. Cs 224d: Deep learning for nlp.
- [15] Yoav Goldberg and Omer Levy. word2vec explained: Deriving mikolov et al.’s negative-sampling word-embedding method. *CoRR*, abs/1402.3722, 2014.
- [16] Sascha Rothe and Hinrich Schütze. Word embedding calculus in meaningful ultradense subspaces, 2016.
- [17] Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. Evaluation methods for unsupervised word embeddings, 2015.
- [18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.
- [19] Deepti Mahajan, Radhika Patil, and Varsha Sankar. Word2vec using character n-grams.
- [20] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation, 2014.
- [21] Chris McCormick. Word2vec tutorial - the skip-gram model, 2016.
- [22] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013.
- [23] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.
- [24] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppín. Placing search in context: The concept revisited. In *Proceedings of the 10th International Conference on World Wide Web, WWW ’01*, pages 406–414, New York, NY, USA, 2001. ACM.
- [25] Felix Hill, Roi Reichart, and Anna Korhonen. Simlex-999: Evaluating semantic models with genuine similarity estimation. *Comput. Linguist.*, 41(4):665–695, 2015.
- [26] Stijn van Dongen and Anton Enright. Metric distances derived from cosine similarity and pearson and spearman correlations. 2012.
- [27] Bhoj Raj Sharma and Aman Paul. Clustering algorithms: Study and performance evaluation using weka tool. *International Journal of Current Engineering and Technology*, (3), 2013.
- [28] Stanislaw Jastrzebski, Damian Lesniak, and Wojciech Marian Czarnecki. How to evaluate word embeddings? on importance of data efficiency and simple supervised tasks. *CoRR*, abs/1702.02170, 2017.
- [29] Stefan Jansen. Word and phrase translation with word2vec. *CoRR*, abs/1705.03127, 2017.

Regelungstechnische Projektierung und Realisierung mechatronischer Systeme am Beispiel von Hochschul-Laboranlagen

Christian Siegle¹, Thomas Pospiech¹, Tobias Loose¹

¹Hochschule Heilbronn, Max-Planck-Straße 39, 74081 Heilbronn

christian_siegle@t-online.de, thomas.pospiech@hs-heilbronn.de, tobias.loose@hs-heilbronn.de

Dieser Beitrag zeigt wie regelungstechnische Fragestellungen praxisnah durch kleine Laborarbeiten als Ergänzung zur Hochschullehre gelöst werden können. Dabei werden aus Sicht von Studierenden Lernerfahrungen aufgezeigt, wie theoretische Erkenntnisse aus der Regelungstechnik in der Praxis anwendbar sind. Ebenso wird der Mehrwert für industriepraktische Tätigkeiten aufgezeigt, d.h. wie Studierende durch diese Laborarbeiten Wissen aufbauen, um erfolgreich in einen Ingenieur-Beruf zu starten.

1 Einleitung

Regelungstechnik gilt als eine der anspruchsvollsten, aber auch reizvollsten Ingenieurdisziplinen, siehe [2, 7, 9, 11]. Durch die Anwendung und Kombination vieler Fachbereiche zu einer *funktionierenden* Gesamtanlage müssen viele Ingenieurfächer hinreichend beherrscht werden, wie z. B. Elektrotechnik, technische Mechanik, aber auch Konstruktionslehre. Das Reizvolle besteht darin, dass das Zusammenspiel der Fachbereiche in einer realen Anlage zur Erfüllung einer bestimmten Zielstellung funktionieren muss. Studierende müssen dabei die teilweise komplexen Zusammenhänge von technischen Systemen verstehen und abstrahieren. Um dieses Verständnis aufzubauen und den Bezug zu realen Anlagen herzustellen, bietet es sich an, die Regelungstechnik mit kleinen Laboranlagen zu praktizieren. Hierzu wurde ein Didaktik-Konzept erarbeitet, bei dem Studierende vorlesungsbegleitend Projekte mit Hilfe von LEGO[®] MINDSTORMS[®] und MATLAB[®] / Simulink[®] erarbeiten, siehe [6, 7, 8]. In diesem Beitrag werden die Lernerfahrungen solcher Projekte und exemplarisch die Realisierung einer Gleichlaufregelung demonstriert.

2 Aufgabenstellung

Vorlesungsbegleitend lassen sich kleine regelungstechnische Projekte mit LEGO[®] MINDSTORMS[®]

durch Studierende selbstständig erarbeiten, weil die Hardware für Studierende selbsterklärend bzw. bereits bestens bekannt ist. Daher lautet die übergeordnete Projektaufgabe, siehe [6, 7]: Es ist eine technische Anlage zur Erfüllung einer bestimmten Aufgabe zu entwickeln. Der Bezug zur Vorlesung und damit die Anwendung ingenieurwissenschaftlicher Methoden ist dabei sicherzustellen. Studierende sollen dabei idealerweise selbstständig bestimmte Anlagen zur Realisierung vorschlagen.

3 Lösungswege und Lernerfahrung

3.1 Realisierung einer Gleichlaufregelung

Beispielhaft wird hier eine Gleichlaufregelung der Positionen x und y einer Hebebühne als Laborarbeit beschrieben, siehe Abbildung 1. Die Hebebühne besteht aus zwei Elektromotoren M_A und M_B , die jeweils über ein Getriebe G die Positionen x und y beeinflussen.

Die Idee derartiger Laborprojekte besteht darin, bestimmte Charakteristiken oder Realisierungen *realer* Anlagen nachzubauen, siehe [6, 7, 8]. Sicherlich lassen sich mit LEGO[®] MINDSTORMS[®] einfachere Konstruktionen für eine Hebebühne bauen, aber das steht hier nicht im Vordergrund, weil hier reale An-

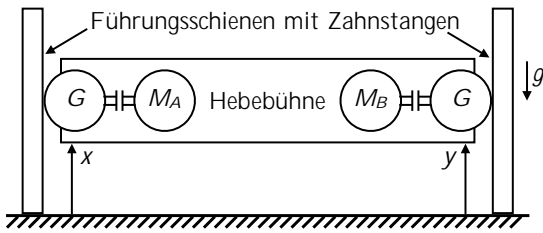


Abbildung 1: Prinzip einer Hebebühne mit zwei Motoren

lagen nachgebildet werden sollen. Bei großen Hebebühnen oder auch Werkzeugmaschinen wie Pressen müssen mitunter mehrere Antriebe vorgesehen werden, die später im Gleichlauf arbeiten. Dieses ist als Laborarbeit nachempfunden und über eine Master-Slave-Regelung realisiert, siehe Abbildung 2.

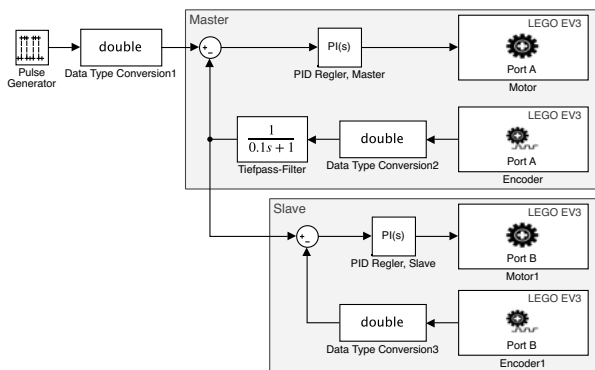


Abbildung 2: Realisierung einer Gleichlaufregelung, die Encoder sind jeweils in den Motoren M_A und M_B integriert

Nachdem die Reglerstruktur mit einem Master-Slave-Regler entworfen ist, müssen regelungstechnische Projektierungsschritte wie

- Signalverarbeitung von Sensordaten mit einem Tiefpass-Filter, um unerwünschte Reglerauswirkung aufgrund digitaler Sensormesswerte auf das Anlagen-Verhalten zu reduzieren (insbesondere mit einem P- und D-Regleranteil, wobei letzterer hier nicht verwendet ist),
- Wahl eines hohen Regler-Verstärkungsfaktors für den Slave-Antrieb, damit der Slave mit dem Master „gut mithalten kann“,
- Wahl geeigneter Reglerparameter, z. B. durch Ziegler-Nichols (die Lego-Anlage kann auch an die Stabilitätsgrenze gefahren werden) oder durch Integralkriterien,
- heuristische und auch „spielerische“ Einstellung der Reglerparameter zur Optimierung oder als

Lernerfahrung auch zur gezielten Verschlechterung des Anlagenverhaltens,

- Berechnung und Bestimmung optimaler Reglerparameter mit vorgelagerter mathematischer Modellbildung

ausgearbeitet werden, siehe [2, 9].

Sehr anschaulich lässt sich auch das Regelstrecken-Modell durch das Sprungantwortverhalten oder durch verschiedene Frequenzen ermitteln, um z. B. das Bode-Diagramm zu erhalten, siehe Abbildungen 3 und 4.

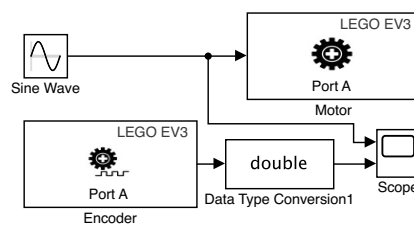


Abbildung 3: Versuchstechnische Ermittlung eines Bode-Diagramms

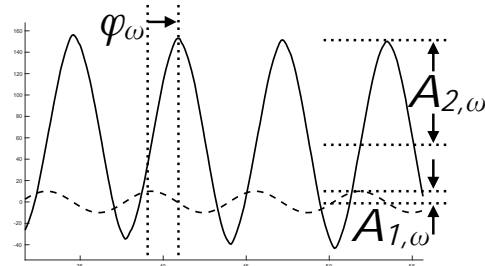


Abbildung 4: Beispiel einer Messung der Encoder-Position (durchgezogen) mit der Phasenverschiebung φ_ω und dem Amplitudenverhältnis $A_{2,\omega}/A_{1,\omega}$ gegenüber der Anregung mit der Kreisfrequenz ω (gestrichelt)

3.2 Lernerfahrung

Nachfolgend werden wertvolle Lernerfahrungen aus Sicht von Studierenden beschrieben, die durch die Ausarbeitung eines realen regelungstechnischen Projektes gesammelt und oftmals als „Aha-Erlebnis“ empfunden werden. Die Liste der Lernerfahrung soll hier nur beispielhaft den Nutzen derartiger Projekte zeigen:

- **Verknüpfung von Theorie und Praxis**
Beim Studium von theoretischen Grundlagen

und mathematischen Beschreibungen von Systemen kann der Bezug zur Realität bei Studierenden mitunter verloren gehen. Bei der Umsetzung in einer realen, physischen Laboranlage werden Studierende allerdings „gezwungen“ die theoretischen Erkenntnisse mit der realen Anlage zu verknüpfen. Das frisch angelesene und gewonnene Wissen aus Vorlesungen kann direkt an der Laboranlage nachvollzogen werden, z. B. das Sprungantwort-Verhalten von Elektromotoren. Beispielsweise wird das Verständnis für eine Übertragungsfunktion maßgeblich verbessert, wenn es eigenständig an realen Anlagen vermessen werden kann, z. B. Beobachtung von Messergebnissen eines Motor-Anlaufverhaltens mit dem Simulink-Block „scope“.

Bei der rein theoretischen Betrachtung einer Gleichlaufregelung ist ein typischer Fehler, dass zwei baugleiche Motoren als identisch angenommen werden. Die Praxiserfahrung und Realisierung einer Gleichlaufregelung zeigt allerdings, dass diese Annahme ein Trugschluss ist, da zwei baugleiche Systeme in Realität niemals identisch sind, z. B. aufgrund Fertigungstoleranzen, unterschiedlicher Reibungseinflüsse. Eine praxistaugliche regelungstechnische Lösung ist hierfür zwingend erforderlich, wie später noch gezeigt wird.

- **Aufbau von Verständnis für die Regelungstechnik**

Im Alltag gibt es viele Beispiele in denen der Mensch Anlagen regelt, z. B. beim Autofahren, beim Greifen von Objekten. Hierbei entwickelt der Mensch ein „Gefühl“ für die angewandte Regelung. Bei der Regelungstechnik übernimmt der Regler diese automatisierungstechnische Aufgabe und muss beim Entwurf entsprechend eingestellt werden. Die Auswirkung und Beeinflussung eines PID-Reglers auf die Regelstrecke ist dabei eine neue Erkenntnis für Studierende. Damit verbunden ist auch die Erfahrung, welche Auswirkung die Änderung von Parametern und Funktionen nach sich zieht, z. B. Vergrößerung von Reglerverstärkungen, falsche Vorzeichen, verschiedene Anregungsfunktionen. Damit kann das Anlagenverhalten stark beeinflusst bis hin zur Instabilität geführt werden. Ebenso ist bei Annäherung an den Grenzbereich auch die Belastungsgrenze der Laboranla-

ge durch z. B. „Anlagen-Ächzen“ frühzeitig und realitätsnah *erlebbar*. Auch, dass nicht beliebige Parameter für den Regler verwendet werden dürfen und es eine Stabilitätsgrenze gibt, ist eine wertvolle Erkenntnis. Das „unscheinbare“ umkehren eines Vorzeichens oder die geringe, „unscheinbare“ Erhöhung eines Parameters kann regelungstechnisch bereits zur Katastrophe führen. Dies sind wertvolle, praxisrelevante Lernerfahrungen, die durch ein reines Literaturstudium oftmals schwer zu vermitteln sind.

- **Anwendung der Mathematik zur Optimierung von Anlagen**

Bis zu einem gewissen Grad lässt sich ein Anlagenverhalten heuristisch verbessern. Eine weitere Optimierung ist dann aber nur möglich, wenn mathematische Modelle zur Beschreibung der Anlagen und des Regelkreises herangezogen werden. Diese Erkenntnis wird für Studierende bei derartigen Projekten deutlich, beispielsweise um das Schwingungsverhalten zu verbessern.

- **Begeisterung für Regelungstechnik**

Bei der ersten Inbetriebnahme von Anlagen werden oftmals suboptimale Ergebnisse erzielt, z. B. Schwingen einer Anlage, bleibende Regeldifferenzen. Studierende erkennen dabei oftmals den eigenen Ehrgeiz und Drang die *eigene* Anlage an ein regelungstechnisches Optimum heranzuführen, wie z. B. die größt mögliche Taktzeit aus der Anlage herauszuholen bei entsprechend hoher Wiederholgenauigkeit und Regelgüte. Dabei wird auch erkannt, dass die Optimierung der Konstruktion einen Einfluss auf das Gesamtergebnis hat, was eine sehr wertvolle Erkenntnis für einen späteren Beruf im Industriealltag ist. Ebenso wird erkannt, dass jedes System seine Grenzen hat, z. B. müssen mitunter Geschwindigkeiten gedrosselt werden, damit die Anlage keinen Schaden nimmt, z. B. das Herausspringen einer Hubbühne aus der Führung.

- **Berufsvorbereitung**

Die Erfahrung, die an einer realen Laboranlage gesammelt werden, können nun mühelos auf andere Anlagen und Systeme übertragen werden. Damit können Studierende bereits industriennahe „Erfahrungswerte“ im Anlagenbau sammeln, um reale Anlagen besser beurteilen und optimieren zu können, z. B. unterschiedliche Methoden zur Erhöhung von Taktzeiten. Auch lassen sich Gefahren viel besser abschätzen, z. B. die Annähe-

rung an einen Grenzbereich.

- **Projektierung und Projektmanagement**

Bei einem Projektvorhaben mit zeitlich beschränktem Ziel während eines Semesters, müssen „nebenbei“ auch Methoden des Projektmanagements angewandt werden, z. B. das Aufstellen eines Projektplans inkl. Zeitplan mit Meilensteinen, die Rücksprache mit Betreuern im Sinne eines Projektsponsors bzw. Lenkungsausschusses, siehe [1]. Damit wird auch deutlich, wie viel Zeit ein bestimmtes, technisches Vorhaben in Anspruch nimmt. Es reift die Erkenntnis, dass gewisse Dinge in der Technik einfach Zeit brauchen, z. B. kann das Verstehen von Anlagen und Zusammenhängen, das Finden von Problemlösungen sehr viel Zeit in Anspruch nehmen. Viele Ingenieure kennen Situationen, bei denen noch bis „Mitten in die Nacht“ an Problemen gearbeitet werden muss, weil die Technik nicht so will, wie der Kunde es bestellt hat. Es wird auch die Erfahrung geschult, um selbst Zeiten abschätzen zu können, wie lange ein technisches Vorhaben in Anspruch nehmen wird. Das ist für eine industriepraktische Anwendung ebenso von großem Nutzen.

Bei einem derartigen Projekt wird auch das lösungsorientierte Denken trainiert, so dass technische Systeme auch nicht zu komplex gestaltet werden. Oftmals gibt es in der Technik einfache, pragmatische und gute Lösungen, die sich am Ende als die Besten herausstellen.

Erfahrenen Ingenieuren sind derartige Kenntnisse und Erfahrungen hinreichend bekannt, siehe [2, 3, 4, 5, 8, 9, 10, 11, 12]. Allerdings sind diese Erfahrungen für Studierende oftmals neu.

4 Zusammenfassung

Dieser Beitrag zeigt Lernerfahrungen von Studierenden, die durch eine vorlesungsbegleitende regelungstechnische Laborarbeiten mit LEGO® MINDSTORMS® gewonnen werden können. Die Idee dabei ist bestimmte Problemstellungen realer Anlagen nachzugestalten, um dann Lösungen zu entwickeln, z. B. der Reglerentwurf einer Gleichlaufregelung, siehe auch [6, 7] für weitere Industrieanlagen.

Literatur

- [1] DIN 69901, Teil 1 bis 5. *Projektmanagement – Projektmanagementsysteme*. Berlin: Beuth. 2009.
- [2] O. Föllinger. *Regelungstechnik*. Berlin, Offenbach: VDE Verlag. 2016.
- [3] R. Hagl. *Elektrische Antriebstechnik*. München: Carl Hanser. 2013.
- [4] H. Jaschek, H. Voos. *Grundkurs der Regelungstechnik*. München: Oldenbourg. 2010.
- [5] T. Loose, H. Mehling, E. Holembowski, S. Arns, K. Kempf, H. Behl. *Im Ziehkissen liegt die Kraft*. In: *BLECH InForm* Nr. 5, S. 104 - 107. 2006.
- [6] T. Loose. *Projektierung mechatronischer Anlagen als Laborarbeit in der Hochschullehre am Beispiel LEGO® MINDSTORMS®*. In: *Proc., 2. IFToMM D-A-CH Konferenz*, S. 1–8 (Tagungsband auf CD-ROM). Universität Innsbruck. 2016.
- [7] T. Loose. *Projektierung mechatronischer Anlagen in der Hochschullehre am Beispiel von Labormodellen*. In: *Proc., Workshop 2017 ASIM/GI-Fachgruppen* (Commerell, W.; Pawletta, T., Hg.), S. 162–167. ARGESIM Verlag Wiens, Hochschule Ulm. 2017.
- [8] T. Loose, Th. Pospiech. *Einbindung von Labormodellen mit industriepraktischen Fragestellungen in der Hochschullehre*. In: *Proc., Workshop 2018 ASIM/GI-Fachgruppen* (Loose, T., Hg.), Hochschule Heilbronn. 2018.
- [9] H. Lutz, W. Wendt. *Taschenbuch der Regelungstechnik*. Haan-Grutten: Europa-Lehrmittel. 2014.
- [10] R. C. Oldenbourg, H. Sartorius. *Dynamik selbsttätiger Regelungen*. München: Oldenbourg. 1951.
- [11] W. Oppelt. *Kleines Handbuch technischer Regelungsvorgänge*. Weinheim/Bergstr.: Verlag Chemie. 1972.
- [12] D. Schröder. *Elektrische Antriebe – Regelung von Antriebssystemen*. Berlin, Heidelberg: Springer Vieweg. 2015.

Standardisierung dynamischer Modelle zur Analyse, Synthese, Simulation und Echtzeitrealisierung

Karlheinz Wolfmüller

Hochschule Heilbronn

karlheinz.wolfmueller@hs-heilbronn.de

Abstract - The nonlinear state space model with the potential- and flow variables $\varepsilon(t)$ and $\phi(t)$ as state space variables is introduced here as a model standard of a dynamic system with symbolic variables. In a general operating point a linearized matrix model is derived depending on the symbolic state space variables, the inputs and the system parameters. With this model all analysis-, synthesis-, simulation- and realtime realizing steps are implemented analytically with symbolic tools. The Software modules of the solution approaches are strictly realized depending on parameters. A standardized simulink model for simulation and for a realtime application is presented. The solution approaches are completely parameter adaptive. The methods presented here are verified for example by a nonlinear unstable process model and by some MIMO-models with nonlinear decoupling control methods.

1 Einleitung

Dynamische Modelle beschreiben das Zeitverhalten realer Vorgänge abstrahiert mit der Sprache der Mathematik und bilden die Basis sämtlicher Aktivitäten, die sich mit der Dynamik eines Systems befassen. In der realen Welt werden dabei immer Zustandsänderungen beschrieben, die durch Materie-, Energie- und Informations-, Transport-, Wandlungs- und Speichervorgängen verursacht werden. Bei der Modellbildung werden dabei verschiedene Formen von Differentialgleichungen mit unterschiedlichen Eigenschaften eingesetzt. Die physikalische Interpretierbarkeit der Modelle hat in der Praxis eine besondere Bedeutung. Durch die Einführung von Potential- und Flußgrößen lassen sich die Energiespeicher in einem System methodisch verallgemeinert darstellen. Mit entsprechenden Analogie Beziehungen läßt sich der Bezug in den Systemen der Elektrotechnik, Mechanik, Thermodynamik, Hydrodynamik und Gasdynamik herstellen. Die nichtlineare Zustandsraum-Modelldarstellung hat in diesem Zusammenhang eine herausragende Bedeutung. Leistungsfähige Modellbildungs-, Analyse-, Synthese-, Simulations- und Echtzeitprogrammierungs-Tools bieten mittlerweile die Voraussetzungen Forschungs-, Entwicklungs- und Entwurfsprojekte auf dem Gebiet der Systemdynamik weitgehend methodisch mit möglichst vielen analytischen und möglichst wenig numerischen Lösungsschritten durchzuführen. Wird diese Zielvorgabe konsequent eingehalten, darf man erwarten, dass am Ende immer eine ganze Problemklasse standardisiert gelöst wird. Die Softwaremodule der Problemlösung sollten sich allgemein ohne programmiertechnische Änderungen einsetzen

lassen. Spezielle problempezifische Lösungen sollten nur noch durch Strukturauswahl, Parametervariation und Anpassung erzeugt werden. Häufig wird bei einem RT-Projekt aus einer grundsätzlichen Aufgabe durch numerische Lösungsschritte ein spezielles Problem mit speziellen Lösungen gemacht. Diese Vorgehensweise ist ineffizient und schränkt die Chancen auf ein Optimum der Lösung stark ein. Es wird hier der Versuch unternommen auf analytischem Wege standardisierte Lösungen bei der Modellbildung, der Systemanalyse und bei der Synthese von Simulations-, Steuerungs-, Regelungs- und Signalaufbereitungs-Algorithmen mit Hilfe von symbolischen Werkzeugen aufzuzeigen. An einem exemplarischen Beispiel wird mit MATLAB/Simulink an einem instabilen nichtlinearen Streckenmodell z.B. ein Parameter adaptiver PI-Zustandsregler mit frei wählbarer Dynamik und einem nichtlinearen Beobachter mit variabler Schätzfehlerdynamik realisiert. Das Konzept kann als struktureller Lösungsstandard für viele Projekte verwendet werden. Alle physikalischen Parameter können im laufenden Simulations- und Echtzeitbetrieb verändert werden. Die Simulink-Lösungen der Simulations- und der Echtzeitprogramme enthalten nur Subsystem-Module, die konsequent auf analytischem Wege mit den Werkzeugen der symbolischen MATLAB-Function Library „RT2lib“ [4] und einem MATLAB Function-Code Generator erzeugt wurden. Diese Programmibliothek benutzt die MATLAB Symbolic Toolbox und wurde im RT2-Labor der Hochschule Heilbronn entwickelt.

2 Dynamische Modelle

Die dynamische Modellbildung erfolgt in der Regel gemäß **Abbildung 1** in mehreren Schritten. Von zentraler Bedeutung dabei ist die Definition des Systems mit der Festlegung von eindeutigen Systemgrenzen. Damit liegt fest, wieviele innere Energiespeicher bzw. dynamische Freiheitsgrade n im System an den Zustandsänderungen beteiligt sind. Im Modell werden diese Energiespeicher durch die inneren Zustandsgrößen x_i , $i = 1, \dots, n$ abgebildet. Die kausalen Ursachen des Systems sind die bekannten p Eingangsgrößen u_1, \dots, u_p und die r unbekanntes Störgrößen z_1, \dots, z_r über die das System aus der Umgebung ausschließlich beeinflusst werden kann. Die Wirkungen sind im Inneren des Systems durch die Zustandsgrößen repräsentiert. In der Umgebung allerdings sind die Wirkungen nur an den q Ausgängen y_1, \dots, y_q sichtbar. Nach dieser Systemdefinition gibt es verschiedene Möglichkeiten die funktionellen Zusammenhänge zwischen Ursachen und Wirkungen darzustellen.

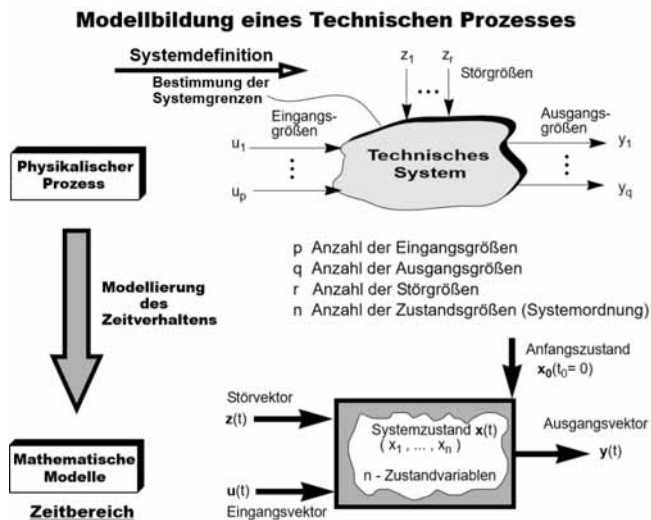


Abbildung 1. Modellbildung eines Prozesses [2],[3]

Mit der Aufstellung der Systemgleichungen läßt sich das Systemverhalten als

- Übertragungsmodell und/oder als
- Zustandsraum-Modell

abbilden. Jedes dieser Modelle verkörpert eine eigene Modellklasse mit unterschiedlichen Eigenschaften.

2.1 Übertragungsmodelle

Zur Darstellung eines MIMO-Systems n -ter Ordnung werden im allgemeinen q nichtlineare Übertragungsdifferentialgleichungen

$$t f_k(y_k^{(n)}, \dots, \dot{y}_k, y_k; \mathbf{u}^{(m)}, \dots, \dot{\mathbf{u}}, \mathbf{u}) = 0$$

$$m \leq n \quad k = 1, \dots, q \quad (1)$$

für jeden Ausgang benötigt, die das kausale Systemverhalten der Wirkungen (Ausgängen \mathbf{y}) in Abhängigkeit der Ursachen (Eingängen \mathbf{u}) und der Systemeigenschaften explizit mit den Größen aus der Umgebung in einem absoluten Koordinatensystem beschreiben. Existieren nur lineare Zusammenhänge bzw. linearisiert man das System Gl. (1) um einen Betriebspunkt $(\mathbf{y}_B, \mathbf{u}_B)$ und entsprechende Ableitungen davon, ergeben sich $(q \times p)$ lineare Übertragungsdifferentialgleichungen

$$y_k^{(n)} + a_{n-1,k,j} y_k^{(n-1)} + \dots + a_{1,k,j} \dot{y}_k + a_{0,k,j} y_k =$$

$$b_{m,k,j} u_j^{(m)} + b_{m-1,k,j} u_j^{(m-1)} + \dots + b_{1,k,j} \dot{u}_j + b_{0,k,j} u_j$$

$$m \leq n, \quad k = 1, \dots, q; \quad j = 1, \dots, p \quad (2)$$

jeweils des j -ten Eingangs zum k -ten Ausgang in einem absoluten oder relativen Koordinatensystem. Im Laplace-Frequenzbereich kann das Modell Gl. (2) auch mit Hilfe einer komplexen $(q \times p)$ Übertragungsmatrix dargestellt werden [3].

2.2 Zustandsraum-Modelle

Zustandsraum-Modelle bilden das Systemverhalten in zwei Schritten ab. Im 1.Schritt werden die inneren dynamischen Freiheitsgrade bzw. die inneren Energiezustände in ihren zeitliche Veränderungen jeweils mit einer Differentialgleichung 1.Ordnung in Abhängigkeit vom inneren Gesamtzustand $\mathbf{x}(t)$ und den Eingängen $\mathbf{u}(t)$ zu einem bestimmten Zeitpunkt t abgebildet. In diesem Modellierungsschritt ergeben sich n Differentialgleichungen 1.Ordnung die zu einem Funktionsvektor Gl. (3)

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \quad (3)$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}, t) \quad (4)$$

zusammengefaßt werden können. Dieser Teil des Modells wird als nichtlineare Zustandsdifferentialgleichung bezeichnet. Die Wirkungen in der äußeren Umgebung werden im 2.Schritt durch einen nichtlinearen, algebraischen Funktionsvektor des Ausgangs $\mathbf{y}(t)$ der Länge q in Abhängigkeit von den inneren Zuständen $\mathbf{x}(t)$ und den Eingängen $\mathbf{u}(t)$ zu einem Zeitpunkt t dargestellt. Gl.(4) wird als nichtlineare Ausgangsgleichung bezeichnet. Die Zustandsdifferentialgleichung Gl.(3) und die Ausgangsgleichung Gl.(4) sind die Zustandsgleichungen des nichtlinearen (NL-)Zustandsraum-Modells.

Die **Abbildung 2** zeigt das Strukturbild des NL-Zustandsraum-Modells.

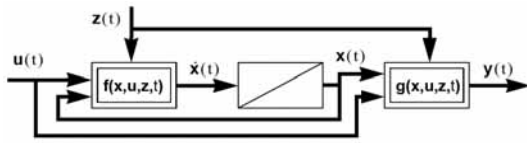


Abbildung 2. Strukturbild NL-Zustandsraum-Modell

Linearisiert man das NL-Modell Gl. (3), Gl. (4) um einen Betriebspunkt $(\mathbf{x}_B, \mathbf{y}_B)$ z.B. der Ruhelage mit

$$0 = \mathbf{f}(\mathbf{x}_B, \mathbf{u}_B), \quad \mathbf{y}_B = \mathbf{g}(\mathbf{x}_B, \mathbf{u}_B) \quad (5)$$

und der Lösung $[\mathbf{x}_B(\mathbf{y}_B); \mathbf{u}_B(\mathbf{y}_B)]$

ergibt sich für die Abweichungen

$$\Delta \dot{\mathbf{x}} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\mathbf{x}_B, \mathbf{u}_B} \cdot \Delta \mathbf{x} + \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{\mathbf{x}_B, \mathbf{u}_B} \cdot \Delta \mathbf{u} \quad (6)$$

$$\Delta \mathbf{y} = \left. \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \right|_{\mathbf{x}_B, \mathbf{u}_B} \cdot \Delta \mathbf{x} + \left. \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \right|_{\mathbf{x}_B, \mathbf{u}_B} \cdot \Delta \mathbf{u} \quad (7)$$

ein Modell in einem relativen Koordinatensystem um den Betriebspunkt. Verändert man nun den Betriebspunkt im physikalischen Wertebereich einer Ausgangstrajektorie $\mathbf{y}(t)$ so entsteht ein Modell für die Abweichungen um einen zeitveränderlichen Betriebspunkt

$$\Delta \dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}(t)) \cdot \Delta \mathbf{x} + \mathbf{B}(\mathbf{u}(t)) \cdot \Delta \mathbf{u} \quad (8)$$

$$\Delta \mathbf{y} = \mathbf{C}(\mathbf{x}(t)) \cdot \Delta \mathbf{x} + \mathbf{D}(\mathbf{u}(t)) \cdot \Delta \mathbf{u} \quad (9)$$

in einer formalen Vektor- und Matrizenschreibweise. Die Modellmatrizen \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{D} sind zeitabhängig von $\mathbf{x}(t)$ und $\mathbf{u}(t)$. Zu erwähnen ist, dass am Betriebspunkt das Änderungsverhalten dieses Modells exakt dem Verhalten des nichtlinearen Originalmodells am Betriebspunkt entspricht. Damit wird das Systemverhalten in einem absoluten bzw. relativen Koordinatensystem im n -dimensionalen Zustandsraum X^n beschrieben.

2.2.1 Physikalische Definition der Zustandsgrößen

In den technischen Disziplinen Elektrotechnik, Mechanik, Thermodynamik, Hydrodynamik und Gasdynamik lassen sich die inneren Zustandsgrößen als abstrakte Potentialgrößen $\varepsilon(t)$ und Flussgrößen $\phi(t)$ von Speicherbauteilen interpretieren [3], wie sie z.B. in Netzwerken benutzt werden. Die Potentialgröße repräsentiert direkt den Energiezustand eines

kapazitiven Speichers. Die Zustandsdifferentialgleichung

$$\dot{\varepsilon}(t) = \frac{1}{C(\varepsilon)} \cdot \phi(t) \quad (10)$$

beschreibt mit der Bauteilkapazität C den Zusammenhang mit der Flussgröße. Die Flussgröße repräsentiert direkt den Energiezustand eines induktiven Speichers. Die Zustandsdifferentialgleichung

$$\dot{\phi}(t) = \frac{1}{L(\phi)} \cdot \varepsilon(t) \quad (11)$$

beschreibt mit der Bauteilinduktivität L den Zusammenhang mit der Potentialgröße.

Es wird nun noch ein Energiewandler als Verbraucherbauteil mit der reellen Bauteilgröße

$$R = f(\varepsilon, \phi) = \varepsilon / \phi \quad (12)$$

eingeführt, in dem mit der Wirkleistung

$$p = \varepsilon \cdot \phi = R \cdot \phi^2 \quad (13)$$

eine gesteuerte thermische Flussquelle thermische Energie erzeugt, die die Temperatur $T(t)$ des Systems als thermische Potentialgröße beeinflusst in Wechselwirkung mit der Temperatur $T_U(t)$ der Umgebung.

Physikal. System	Potentialgröße $\varepsilon(t)$	Flussgröße $\phi(t)$
Elektrotechnik	<u>Spannung</u> u /[V]	<u>Strom</u> i /[A]
Mechanik - Translation - Rotation	<u>Geschwindigkeit</u> v /[m/s] <u>Kreisfrequenz</u> ω /[rad/s]	<u>Kraft</u> F /[N] <u>Moment</u> M /[N/m]
Thermodynamik	<u>Temperatur</u> T /[°K]	<u>Wärmestrom</u> \dot{Q} /[W]
Hydrodynamik	<u>Druck</u> p /[Pa]	<u>Volumenstrom</u> \dot{V} /[m³/s]
Gasdynamik	<u>Druck</u> p /[Pa]	<u>Volumenstrom</u> $\dot{V} = \frac{\dot{m}}{\rho}$ /[m³/s]

Tabelle 1. Analogien der Potential- und Flussgrößen

Da grundsätzlich derartige Bauteile als thermische Energiewandler in jedem System vorhanden sind, werden zur Beschreibung der Temperatur der Bauteile im System immer dynamische thermische Modelle

von thermischen Massen benötigt. Thermische Energiespeicher sind grundsätzlich kapazitive Speicher. In der **Tabelle 1** sind der Potential- und der Flussgröße die Analogiegrößen in den einzelnen technischen Disziplinen zugeordnet.

Zur Ansteuerung der Eingänge **u** und zur Realisierung der Anfangszustände $x_0(t_0)$ der Energiespeicher sind noch eine ideale Potentialquelle und eine ideale Flussquelle mit den entsprechenden Impedanzbedingungen notwendig. In der **Abbildung 3** sind die 5 Netzwerkbauelemente zusammengefasst, die in Parallel- und/oder Reihenschaltung zu komplexen dynamischen Netzwerk-Modellen [3] zusammen geschaltet werden können. Durch Anwendung der Knoten- und Maschensätze bieten diese Modelle die

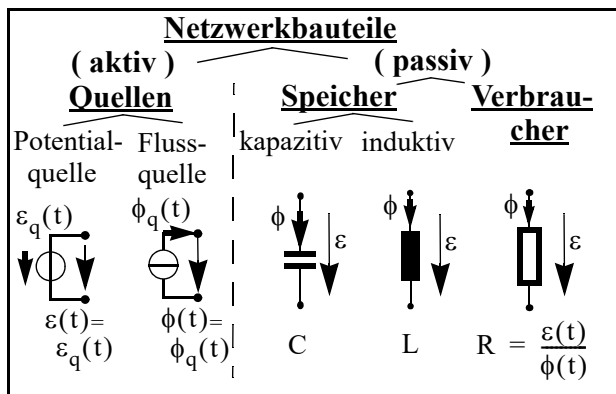


Abbildung 3. Übersicht der Netzwerkbauteile für die dynamische Modellbildung

idealen Voraussetzungen zur systematischen Aufstellung der Modellgleichungen [3].

3 Analyse

Lineare Übertragungsmodelle können direkt in die speziellen Zustandsräume der Regelungsnormform (RNF) und der Beobachtungsnormform (BNF) überführt werden [1]. Dabei sind die Zustandsvariablen direkt mit den Ausgängen und deren Ableitungen verknüpft. Sie sind also nur noch bedingt oder nicht mehr physikalisch interpretierbar.

3.1 Eigenschaften von Übertragungsmodellen

Lineare Übertragungsmodelle werden vornehmlich in der klassischen Regelungstechnik im Laplace-Bildbereich zur Frequenzanalyse und dem Entwurf klassischer PID-Standardregelkreise insbesondere bei SISO-Systemen eingesetzt. Komplexe nichtlineare Mehrgrößensysteme, deren innere physikalische Eigenschaften interessieren, lassen sich mit Übertragungsmodellen nur schwer oder überhaupt nicht physikalisch analysieren.

3.2 Eigenschaften von Zustandsraum-Modellen

Interessiert man sich für die physikalischen inneren Zusammenhänge und Größen eines Systems bieten die nichtlinearen Zustandsraum-Modelle Gl. (3) und Gl. (4) die besten Voraussetzungen. Wählt man zusätzlich die Zustandsvariablen wie es in Kap. 2.2.1 beschrieben wird, ist die bestmögliche physikalische Referenz gewährleistet. Diese Modellform kann als physikalischer Modellstandard zur Beschreibung des Zeitverhaltens eines Systems angesehen werden. Zusätzlich ist die mathematische Form und Struktur **Abbildung 2** der Modellgleichungen relativ einfach und prädestiniert zur Verarbeitung mit einem Digitalrechner. Als exemplarisches Beispiel soll hier das allgemein bekannte akademische Lehrmodell „Magnetfeld Schwebekörper“ dienen. In dem MATLAB-Programm **Code 1** wird konsequent mit symbolischen Variablen das NL-Zustandsraum-Modell gemäß Gl. (3), Gl. (4) im MATLAB-Workspace gebildet.

```

%%
% Parameter Magnetfeldregelung
%-----
syms k kd g m R L Fm f_NL g_NL x u y w
%%
% Nichtlineares Zustandsraum-Modell
%=====
% NL-Zustandsdgl      x'=f_NL(x,u,t)
%-----
f_NL(1,1) = x(2);
f_NL(2,1) = -kd/m*x(2) - k/m*x(3) ...
            ^2/x(1)^2 + g;
f_NL(3,1) = -(R/L)*x(3) + 1/L*u

% NL-Ausgangsglg      y=g_NL(x,u,t)
%-----
g_NL = x(1)

% Matlab-function Generator
create_matfkt('NL_zgl_matfkt', f_NL,
g_NL)
    
```

Code 1. Symbolische NL-Zustandsgleichungen

Die verwendeten MATLAB-functions sind Bestandteile der im RT2-Labor der Hochschule Heilbronn entwickelten MATLAB-function Library „RT2lib“ [4]. Der MATLAB-function Code Generator „create_matfkt(...)“ in der letzten Zeile generiert automatisch den Programm-Code der NL-Zustandsgleichungen **f(x,u)**, **g(x,u)** in einem MATLAB-function m-File mit dem Namen 'NL_zgl_matfkt'. Dieser Code Generator bildet aus einer symbolischen Ausdruck-Liste den MATLAB-Code einer MATLAB-

function in einem m-File mit dem Namen des 1.Übergabe-Strings.

Benutzt man darüber hinaus noch die Betriebspunkt-darstellung Gl. (6), Gl. (7), Gl. (8) und Gl. (9) so können formal die Analysewerkzeuge der linearen Systemtheorie [1] angewendet werden. Es besteht damit die Möglichkeit ein System entlang bestimmter Betriebstrajektorien bezüglich seiner Eigenschaften zu betrachten. Zum Beispiel können Steuerbarkeit, Beobachtbarkeit und Stabilität mit der Eigenwertanalyse mit analytischen Werkzeugen an veränderlichen Betriebspunkten untersucht werden. Der MATLAB-Programm **Code 2** führt diesen Schritt durch.

```
%%
% Stationärer Betriebspunkt
%-----
[xB, uB]=nl_statb(f_NL,g_NL,x,u,y)
%%
% Linearisiertes Modell
%-----
[Asym, Bsym, Csym, Dsym] = ...
    nl2lin(f_NL, g_NL, x, u, y)
%%
% Analyse am Betriebspunkt
%-----
% Symbolische Eigenwerte
Eigsym=eig(Asym)
```

Code 2. Stationärer Betriebspunkt mit linearisiertem Modell und Eigenwertanalyse

Führt man die Analyseschritte konsequent mit symbolischen Mathematik-Tools durch, sind die analytischen Ergebnisse abhängig von den veränderlichen Systemparametern und sind so allgemeingültig interpretierbar. Dies kann als erheblicher Fortschritt gegenüber rein numerischen Lösungen angesehen werden.

4 Synthese

Es gibt kaum allgemein anwendbare, mathematische Werkzeuge zur Synthese von Steuerungs-, Regelungs- und Beobachtungs-Algorithmen an NL-Zustandsraum-Modellen. Alle bekannten linearen Synthesewerkzeuge [1] können natürlich auf die linearisierten Betriebspunkt-Modelle Gl. (8), Gl. (9) angewendet werden. In der „RT2lib“ werden MATLAB-functions bereitgestellt, die die gängigen Syntheseschritte konsequent analytisch an den linearisierten Betriebspunkt-Modellen umsetzen. Dabei wird die MATLAB Symbolic-Toolbox eingesetzt. Nach aktuellem Stand

werden in der „RT2lib“ Werkzeuge für die folgenden analytischen Syntheseschritte zur Verfügung gestellt:

- Zustandsregler (variable Pole und Parameter),
- PI-Zustandsregler (variable Pole und Parameter),
- Adaptiver Zustandsregler (variable Pole und Parameter),
- Adaptiver PI-Zustandsregler (variable Pole und Parameter),
- Linearer Mehrgrößen-Entkopplungsregler (variable Pole und Parameter),
- Nichtlinearer Mehrgrößen-Entkopplungsregler (variable Pole und Parameter),
- Beobachter (variable Dynamik und Parameter),
- NL-Beobachter (variable Dynamik und Parameter).

5 Simulation

In **Abbildung 4** ist ein mit Simulink realisiertes Simulationsmodell dargestellt, das als Regelsystem-Standard betrachtet werden kann.

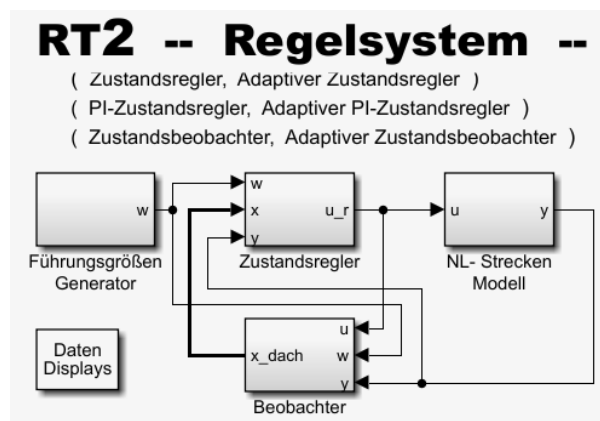


Abbildung 4. Standardisiertes Simulink Regelsystem

Abbildung 5 zeigt die Simulink-Lösung des Subsystems des standardisierten NL-Streckenmodells. Es entspricht strukturell exakt dem mathematischen Modell von **Abbildung 2**. Es können hier z.B. auch die Systemparameter dynamisch während der Simulation bzw. im laufenden Echtzeitbetrieb verändert werden. Sämtliche im Regelsystem beteiligten Komponenten passen sich dynamisch an. Strukturell identisch ist die Simulink Lösung des NL-Beobachters, der als adaptiver Beobachter mit einem von der Messgröße dynamisch gesteuerten Betriebspunkt ent-

worfen wurde. Für die Realisierung der Regelung können sämtliche aufgeführten Zustandsregler-Typen

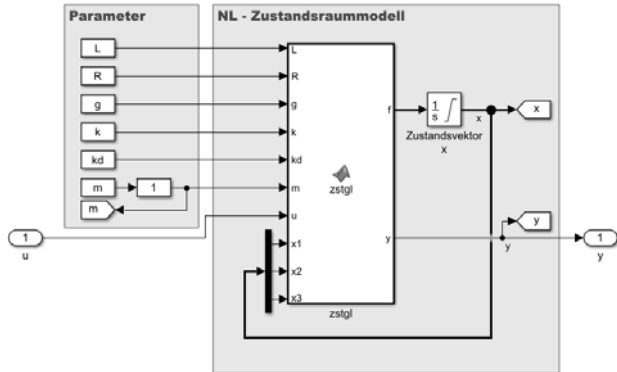


Abbildung 5. NL-Streckenmodell, Simulink-Standard

ausgewählt werden. Auch hier ist die interne Struktur gleich. Beobachter und Regler sind vollständig Systemparameter abhängig und variabel in der Dynamik durch veränderliche Pole.

6 Echtzeitrealisierung

Mit den aktuell verfügbaren Prozessorleistungen ist die Realisierung der Parameter adaptiven Regler- und Beobachter-Algorithmen unter harten Echtzeitbedingungen in der Regel kein grundsätzliches Problem mehr. Der maximale algorithmische Aufwand existiert in der Praxis natürlich nur in der Entwicklungsphase. Nach der Systemoptimierung bzgl. der Zielkriterien der konkreten Anwendung sind viele Parameter im praktischen Betrieb als konstant anzusehen. Der Rechenzeitaufwand kann sich dann stark reduzieren. Das Regelsystem gemäß **Abbildung 4** lässt sich direkt ohne Änderung der Subsysteme mit Real-Time Simulink oder anderen Hardware Support-Packages mit entsprechendem C-Code Generator als Echtzeitprozess bei diskreter Abtastzeit implementieren. Es ist nur die Strecke durch die Sensor/Aktor-Interfaces der Realtime-Hardware an der Stellgröße u und der Messgröße y zu ersetzen. Die Algorithmen der Echtzeit- und der Simulationslösung sind nach der Diskretisierung identisch. Das hier vorgestellte Beispiel wurde mit der Simulink Real-Time Toolbox und PC-Hardware mit PCI-NI6229 I/O-Interfaces unter harten Echtzeitbedingungen (Rt-Online Kernel) realisiert. Selbst mit einem Debian LINUX-Kernel auf einem Raspberry PI-3 konnte eine Abtastzeit von $T_A = 1$ msec für den Regelungsprozess im Rt-Status eingehalten werden.

7 Anwendungsbeispiele

Die hier vorgestellte Vorgehensweise der analytischen Modellierung, Analyse, Synthese, Simulation und Echtzeitrealisierung wurde in mehreren Projekten schon angewendet und verifiziert. Exemplarisch werden hier z.B. genannt:

1. NL-3Tank MIMO-Modell (Entkopplungsregelung mit 2 Eingängen, 2 Ausgängen, 3 Zuständen),
2. NL-gasdynamisches MIMO-Modell (NL-Entkopplungsregelung mit 3 Eingängen, 3 Ausgängen, 4 Zuständen).

In allen Projekten sind die Lösungen der einzelnen Schritten mit den Programm-Modulen strukturell identisch, obwohl völlig unterschiedliche physikalische Problemstellungen behandelt werden.

8 Zusammenfassung

Das NL-Zustandsraum-Modell mit den physikalisch definierten Zustandsgrößen wird hier als Standardmodell eingeführt. Sämtliche Modellierungs-, Analyse-, Synthese-, Simulations- und Echtzeitrealisierungsschritte werden als analytische Lösungen mit symbolischen Ausdrücken umgesetzt. Damit wird eine dynamische Parametrierung der Lösungen in allen Modulen möglich. In den Entwicklungsphasen eines Projekts ist damit die größte Flexibilität mit bestmöglichem Optimum der Lösung gegeben. Es liegt immer eine Lösung der physikalischen Problemklasse vor. Numerische Lösungsschritte, die die Allgemeingültigkeit einschränken, werden konsequent vermieden.

9 Referenzen

- [1] Otto, Föllinger. Regelungstechnik. Heidelberg: Hüthig Verlag 1992, 7. Auflage
- [2] Karlheinz, Wolfmüller. Systemdynamik und Simulation 2. Hochschule Heilbronn, WS2017/18
- [3] Karlheinz Wolfmüller. Modellbildung und Simulation. Hochschule Heilbronn, WS2013/14
- [4] Karlheinz, Wolfmüller. RT2lib, MATLAB-function Library. RT2-Labor, Hochschule Heilbronn, 2018

Generierung von Simulationsmodellen zur Ermittlung des Systemverhaltens von Shuttle-Systemen

Thomas Kriehn¹, Franziska Schloz², Karl-Heinz Wehking², Markus Fittinghoff¹

¹Hochschule Heilbronn, Fakultät für Technische Prozesse

²Universität Stuttgart, Institut für Fördertechnik und Logistik

thomas.kriehn@hs-heilbronn.de

Shuttle-Systeme sind automatische Lagersysteme, die häufig zur Lagerung von Kleinladungsträgern und seltener im Palettenbereich eingesetzt werden. Sie sind für Lager mit hohen Durchsatzanforderungen geeignet. Simulationmodelle zur Ermittlung des Systemverhaltens von Shuttle-Systemen werden häufig eingesetzt. Wenn diese auf Basis von Simulationswerkzeugen entwickelt werden und eine Visualisierung der Prozesse enthalten, können z. T. bestimmte Parameterwerte nur mit höherem Aufwand variiert werden. Dieser Beitrag stellt ein Verfahren zur Generierung von Simulationsmodellen vor. Dadurch ist die Möglichkeit vorhanden, Parameterwerte mit geringem Aufwand zu variieren. Weiterhin sind im Simulationsmodell die Lagerstrategien Chaotische Lagerplatzbelegung und Durchsatzbasierte Lagerplatzbelegung implementiert. Mit den Simulationsmodellen wird das Systemverhalten einfachtiefer gang- und ebenengebundener Shuttle-Systeme ermittelt. Die Parameter zur Beschreibung des Systemverhaltens sind der Durchsatz, die Spielzeit, die Wartezeit und der Nutzungsgrad der Fördermittel. Die Ergebnisse zeigen die Sensitivität des Systemverhaltens des Shuttle-Systems bei Variation der Parameterwerte Geschwindigkeit, Lastübergabezeit, Anzahl Lagerplätze und Anzahl Ebenen. Weiterhin wird das Potential zur Durchsatzsteigerung durch die Durchsatzbasierte Lagerplatzbelegung gezeigt.

1 Einleitung

Der Einsatz von Simulationsmodellen zur Ermittlung des Systemverhaltens von komplexen Materialflusssystemen kommt in vielen Fällen zum Einsatz, wenn eine analytische Berechnung nicht oder nur mit vereinfachenden Annahmen durchgeführt werden kann oder wenn ein analytisches Modell validiert werden soll. Simulationsmodelle kommen häufig in der Planungsphase eines Materialflusssystemes zum Einsatz, zur Detailplanung oder zur Absicherung einer bereits abgeschlossenen Planung. Die Entwicklung von Simulationsmodellen ist meist aufwendig. Simulationsmodelle werden deshalb i. d. R. nur dann eingesetzt, wenn es keine einfachere Methode zur hinreichenden Beantwortung der Fragestellung gibt.

Dieser Beitrag zeigt ein Verfahren zur Minimierung des Aufwandes zur Entwicklung eines ereignisdiskreten Simulationsmodells für Shuttle-Systeme, das die Visualisierung der Prozesse beinhaltet. Dies wird durch die automatische Generierung von Simulationsmodellen erreicht.

Das Simulationsmodell beinhaltet die Möglichkeit bestimmter Parametervariationen, die zu einer Änderung der Visualisierung der Prozesse führen. Bei der Variation der Parameter, welche die Visualisierung beeinflussen, wird jeweils ein neues Simulationsmo-

dell generiert, das die dafür angepassten Prozesse ausführt. Das Simulationsmodell wird zur Ermittlung des Systemverhaltens des Shuttle-Systems mit der vorgegebenen Parameterkombination eingesetzt. Das eingesetzte Simulationswerkzeug ist die Simulationssoftware Automod V 12.6.2.

Die vom Simulationsmodell ermittelten Parameterwerte zur Beschreibung des Systemverhaltens von Shuttle-Systemen sind der Durchsatz, die Spielzeit, die Wartezeit und der Nutzungsgrad der Fördermittel. Shuttle-Systeme sind automatische Lagersysteme. Die vertikale Förderung erfolgt über Lifte, die horizontale über Shuttle-Fahrzeuge. Weit verbreitet sind Shuttle-Systeme mit gang- und ebenengebundenen Shuttle-Fahrzeugen. Dabei kann ein Shuttle-Fahrzeug die Ebene nicht verlassen, dementsprechend befindet sich in jeder Ebene ein Shuttle-Fahrzeug.

Abbildung 1 zeigt ein gang- und ebenengebundenes Shuttle-System mit zwei Behälterliften.



Abbildung 1: Gang- und ebenengebundenes Shuttle-System [1].

In diesem Beitrag werden Abkürzungen verwendet.

a_L	Beschleunigung Lift [m/s^2]
a_{Sh}	Beschleunigung Shuttle-Fahrzeug [m/s^2]
B_{LP}	Breite Lagerplatz [m]
$D_{L,ES}$	Mittlerer Durchsatz Lift ES [LE/h]
DS	Doppelspiel
$D_{Sh,DS}$	Mittlerer Durchsatz Shuttle-Fahrzeug DS [LE/h]
$D_{Sh,ES}$	Mittlerer Durchsatz Shuttle-Fahrzeug ES [LE/h]
E	Anzahl Ebenen
ES	Einzelspiel
H_E	Höhe Ebene [m]
H_Z	Häufigkeit der Zonenansteuerung je Zone
L	Anzahl Lagerplätze je Ebene je Gangseite
LE	Ladeeinheit
LF	Lagerfüllgrad [%]
N_L	Nutzungsgrad Lift [%]
N_{sh}	Nutzungsgrad Shuttle-Fahrzeug [%]
$P_{E/A}$	Position des Ein-/Auslagerungspunktes [m]
PF	Pufferplätze je Ebene
Q_{Min}	Vorgabewert Anzahl Lagerplätze im Lager
$T_{L,Aufab}$	Lastübergabezeit (Aufnahme, Abgabe) Lift [s]
$T_{L,ES}$	Mittlere Spielzeit Lift ES [s]
$T_{L,SP}$	Schalt- und Positionierzeit Lift [s]
$T_{Sh,Aufab}$	Lastübergabezeit Shuttle-Fahrzeug [s]
$T_{Sh,DS}$	Mittlere Spielzeit Shuttle-Fahrzeug DS [s]
$T_{Sh,SP}$	Schalt- und Positionierzeit Shuttle-Fahrzeug [s]
v_L	Geschwindigkeit Lift [m/s]
v_{Sh}	Geschwindigkeit Shuttle-Fahrzeug [m/s]
$w_{Sh,DS}$	Wartezeit Shuttle-Fahrzeug DS [s]
Z	Zonenzuordnung je LP
ZV	Zonenvarianten
Z_A	Anzahl Zonen

Der Beitrag ist wie folgt strukturiert: In Kapitel 1 erfolgt eine Einleitung. Kapitel 2 beinhaltet Beschreibungen zur vorhandenen Literatur (2.1), zum Simulationsmodell (2.2), zur Generierung von Simulationsmodellen (2.3) und zu den implementierten Lagerstrategien (2.4). Das Simulationsmodell wird für eine Analyse unterschiedlicher Shuttle-Systeme angewendet (2.5) und die Ergebnisse der Simulationsstudie gezeigt und interpretiert (2.6). Kapitel 3 enthält die Zusammenfassung des Beitrags sowie einen Ausblick auf mögliche weitere Forschungsarbeiten.

2 Generierung und Anwendung

2.1 Literatur

Zu gang- und ebenengebundenen Shuttle-Systemen existieren mehrere Veröffentlichungen [2-12]. In den meisten der genannten Veröffentlichungen wird ausschließlich die Chaotische Lagerplatzbelegung angewendet. Ausnahmen bilden [2, 3, 4], dabei werden die Durchsatzbasierte Lagerplatzbelegung [2, 3] sowie verschiedene Priorisierungen der Ebenenansteuerung

angewandt [4]. Das Verhältnis der Länge/Höhe des Shuttle-Systems wird häufig optimiert [5, 6, 7, 8, 9]. In [5, 10, 11, 12] dienen die Simulationsmodelle auch der Validierung eines analytischen Modells. Zur Simulation werden entweder Eigenentwicklungen [6, 7, 10, 4] oder Simulationswerkzeuge wie Arena [2, 11, 8], Automod [3], Plant Simulation [9] oder Simio [5] eingesetzt.

In keiner der genannten Veröffentlichungen findet eine Variation der Lastübergabezeiten von Lift und Shuttle-Fahrzeug statt, d. h. die Sensitivität des Systemverhaltens bei Änderung dieses Parameters auf das Systemverhalten ist noch nicht untersucht worden. Auch die Geschwindigkeit wird nur in zwei Veröffentlichungen variiert [6, 10].

2.2 Simulationsmodell

Das Simulationsmodell ermittelt den Grenzdurchsatz, d. h. den maximal möglichen Durchsatz für die gewählte Parameterkombination. In jedem Szenario wird angenommen, dass jederzeit Aufträge zur Bearbeitung zur Verfügung stehen. Abbildung 1 zeigt die verwendete Lagerkonfiguration, ein Shuttle-System mit zwei Behälterliften pro Gang. Die Lagerung erfolgt einfachtief. Die Kapazität zur Ladeeinheitenaufnahme beträgt bei den Liften und den Shuttle-Fahrzeugen 1. Jeder Lagerplatz kann eine Ladeeinheit aufnehmen. Die Kapazität der Pufferplätze je Ebene beträgt 1. Der Lagerfüllgrad beträgt 50 %. Der Ruhepunkt eines Shuttle-Fahrzeugs nach der Bearbeitung eines Auftrags ist an der Position der Pufferplätze.

Folgende Parameter des Simulationsmodells können variiert werden: v_L , a_L , $T_{L,Aufab}$, $T_{L,SP}$, E, H_E , $P_{E/A}$, v_{Sh} , a_{Sh} , B_{LP} , L, $T_{Sh,Aufab}$, $T_{Sh,SP}$, Z, Z_A , H_Z , LF, PF.

2.3 Generierung eines Simulationsmodells

Für die automatische Generierung eines Simulationsmodells wird folgendes Vorgehen gewählt:

1. Entwickeln eines Basis-Modells mit festgelegten Parameterwerten.
2. Eingabe der gewünschten Parameterwerte für das zu generierende Modell in eine Text-Datei.
3. Einlesen der gewünschten Parameterwerte aus der Text-Datei mit der Software Matlab.
4. Die Modell-Dateien des Basis-Modells mit der Software Matlab einlesen. Diese enthalten alle Referenzierungen auf Modellmaße, Anzahl der Komponenten im Modell usw.
5. Variation der Automod-Modell-Dateien durch einen dafür entwickelten Algorithmus.

6. Erzeugen von Ausgabe-Dateien mit der geänderten Codierung.
7. Ersetzen der Dateien des Basis-Modells durch die Ausgabe-Dateien.
8. Speichern des Ausgabe-Modells.

Nach der Generierung kann das Modell mit der Software Automod geöffnet werden und die visuelle Umgebung wird auf Basis der geänderten Dateien erzeugt. Abbildung 2 zeigt zwei generierte Simulationsmodelle ausschnittsweise in der Seitenansicht.

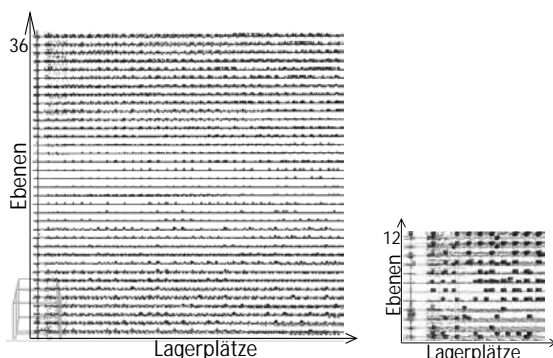


Abbildung 2: Simulationmodelle, mit Automod geöffnet, links hohes Shuttle-System mit 36 Ebenen rechts relativ dazu ein niedriges Shuttle-System mit 12 Ebenen

Der Algorithmus berechnet neue Lagerabmessungen auf Basis der Parameterwerte für die Visualisierung des Simulationsmodells und trägt diese und alle weiteren vorgegebenen Parameterwerte an den richtigen Stellen in der jeweiligen Datei des Simulationsmodells ein. Im Folgenden wird ein Beispiel des Algorithmus zur Generierung eines neuen Modells beschrieben. Dabei werden die einzelnen Ebenen des Lifts in die Lift-Datei eingetragen:

1. Zeilenweises Durchsuchen der Lift-Datei zur ersten Zeile in der die Ebenen eingetragen werden.
2. Solange Ebenenzähler \leq Anzahl der Ebenen gehe zu 3., sonst zu 9.
3. Berechnung der Höhe der jeweiligen Ebene auf Basis der Parameter Anzahl Ebenen und Lagerabmessungen.
4. Durchsuchen der Zeile nach der Position des zu ändernden Parameterwerts.
5. Ändern des Parameterwerts (Angabe der Höhe der jeweiligen Ebene).
6. Ebenenzähler um 1 erhöhen.
7. Gehe in die nächste Zeile.
8. Gehe zu 2.
9. Falls vorhanden, überflüssige Zeilen mit weiteren Ebenen löschen.

2.4 Implementierte Lagerstrategien

Das Simulationsmodell enthält die Lagerstrategien Chaotische und Durchsatzbasierte Lagerplatzbelegung. Bei der Chaotischen Lagerplatzbelegung wird jeder Lagerplatz mit der gleichen Wahrscheinlichkeit für eine Einlagerung ausgewählt. Bei der Durchsatzbasierten Lagerplatzbelegung werden bestimmte Lagerplätze häufiger (A-Zone) und andere seltener (B, C, oder mehr Zonen) zur Einlagerung ausgewählt.

2.5 Anwendung des Simulationsmodells

Zur Ermittlung des Systemverhaltens von Shuttle-Systemen werden für die Lastübergabezeit und die Geschwindigkeit von Lift und Shuttle-Fahrzeug Sensitivitätsanalysen durchgeführt. Weiterhin werden die Anzahl der Ebenen und die Anzahl der Lagerplätze variiert. Die Parameter zur Beschreibung des Systemverhaltens sind der Durchsatz, die Spielzeit, die Wartezeit und der Nutzungsgrad der Fördermittel.

Weiterhin wird das Potential zur Durchsatzsteigerung der Durchsatzbasierten Lagerplatzbelegung gezeigt.

Als konstante Parameterwerte werden in den Analysen folgende verwendet: Breite Lagerplatz $B_{LP}(0.5)$, Höhe Ebene $H_E(0.4)$, Lagerfüllgrad $LF(50)$, Position E/A-Punkt $P_{E/A}(1)$, Schalt- und Positionierzeit Lift $T_{L,SP}(0.5)$ und Shuttle-Fahrzeuge $T_{Sh,SP}(0.5)$.

Für die Sensitivitätsanalyse werden die Parameter Geschwindigkeit Lift $v_L(1,2,3,4,5,6,7,8)$ und Lastübergabezeit Lift $T_{L,Aufab}(1,2,3,4,5,6,7,8)$ variiert. Wenn Parameterwerte variiert werden, dann werden für die jeweils anderen Parameter folgende konstanten Parameterwerte verwendet: Geschwindigkeit Lift $v_L(2)$, Lastübergabezeit Lift $T_{L,Aufab}(2)$, Lastübergabezeit Shuttle-Fahrzeug $T_{Sh,Aufab}(4)$. Dies dient der Vergleichbarkeit, da der Einfluss auf das Systemverhalten durch jeweils einen Parameter ermittelt werden soll. Die Lagerplätze eines Ganges des Shuttle-Systems werden für die Sensitivitätsanalyse auf einen Vorgabewert von $Q_{Min}(4000)$ festgelegt. Es werden fünf Varianten der Höhen- und Längenvariation analysiert, ausgehend von einem sehr niedrigen Shuttle-System mit langen Gängen zu einem sehr hohen Shuttle-System mit kurzen Gängen. Hierbei wird der Vorgabewert $Q_{Min}(4000)$ nahezu konstant gehalten: Variante 1: E(8), L(250), Q(4000), Variante 2: E(16), L(125), Q(4000), Variante 3: E(24), L(83), Q(3984), Variante 4: E(32), L(63), Q(4032), Variante 5: E(40), L(50), Q(4000).

Für die Analyse verschiedener Zonierungsvarianten werden Parametervariationen für die Lastübergabezeit der Lifts $T_{L,Aufab}$ und der Shuttle-Fahrzeuge

$T_{Sh,Aufab}$ gleichzeitig vorgenommen. Dasselbe gilt für Variationen v_L und v_{Sh} . Ziel der Analyse der Zonierungsvarianten ist das Potential zur Durchsatzsteigerung verschiedener Zonierungen für eine Variante eines Shuttle-Systems bei unterschiedlichen Geschwindigkeiten und Lastübergabezeiten zu zeigen. Q_{Min} für die Zonierungsvarianten wird auf 2400 Lagerplätze festgelegt. Es wird ein Shuttle-System mit der Kombination E(12) und L(100) verwendet. Die Variationen beinhalten die Lastübergabezeiten für Lifte $T_{L,Aufab}(1,2,6)$ und Shuttle-Fahrzeuge $T_{Sh,Aufab}(2,4,8)$ sowie die Beschleunigung und Geschwindigkeit für Lifte, $a_L(2,5,6)$ und $v_L(2,7)$, und Shuttle-Fahrzeuge, $a_{Sh}(2,6)$ und $v_{Sh}(2,7)$.

Bei Variation der Parameter werden zur Vergleichbarkeit für die jeweils anderen Parameter folgende konstanten Werte gewählt: $a_L(2,5)$, $a_{Sh}(2)$, $v_L(2)$, $v_{Sh}(2)$, $T_{L,Aufab}(2)$ und $T_{Sh,Aufab}(4)$.

Bei den Zonierungsvarianten ZV0 – ZV4 werden drei Zonen gebildet, A, B, und C. Diese werden mit unterschiedlichen Häufigkeiten angesteuert, A: 60 %, B: 30 % und C: 10 %. Je Zone wird eine bestimmte LP-Anzahl je Gangseite reserviert, A: 240, B: 360 und C: 600. ZV0 steht für die Chaotische Lagerplatzbelegung, d. h. ohne Zonierung. Abbildung 3 zeigt die verwendeten Zonierungsvarianten.

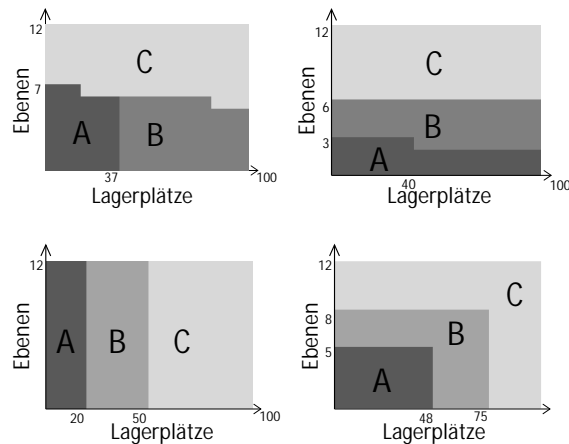


Abbildung 3: Zonierungsvarianten ZV1 - ZV4, ZV1 oben links, ZV2 oben rechts, ZV3 unten links, ZV4 unten rechts

2.6 Ergebnisse

Der Durchsatz des Shuttle-Systems entspricht dem Durchsatz aller eingesetzten Lifte. Das ist dadurch begründet dass die Lifte die Schnittstelle zu angrenzenden Systemen bilden. Über die Lifte werden Ladeeinheiten von angrenzenden Systemen aufgenommen und wieder an diese abgegeben. In den folgenden Analysen werden jeweils ein Lift und ein Shuttle-

Fahrzeug betrachtet. Die Ergebnisse beziehen sich auf einen Gang.

Abbildung 4 zeigt den Einfluss der Anzahl der Ebenen E und der Anzahl der Lagerplätze L auf den Durchsatz eines Lifts, bei Variation der Geschwindigkeit v_L und der Lastübergabezeit $T_{L,Aufab}$ der Lifte.

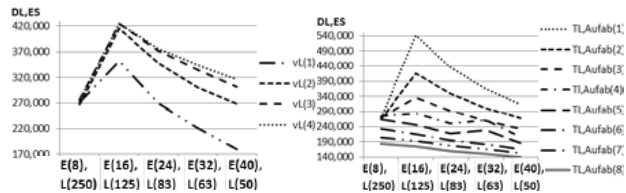


Abbildung 4: Durchsatz eines Lifts, Variation v_L (links) und $T_{L,Aufab}$ (rechts)

Variante 2 mit 16 Ebenen und 125 Lagerplätzen hat bei Geschwindigkeiten des Lifts $v_L(1-4)$ und Lastübergabezeiten $T_{L,Aufab}(1-4)$ die höchsten Durchsätze, d. h. das Verhältnis E(16)/L(125) des Shuttle-Systems ist im Vergleich zu den weiteren Varianten am günstigsten. Bei Lastübergabezeiten des Lifts von $T_{L,Aufab}(4-8)$ stellt der Lift in jeder Variante den Engpass dar. Dementsprechend hat Variante 1 mit nur 8 Ebenen für $T_{L,Aufab}(4-8)$ den höchsten Durchsatz, aufgrund der kürzeren Fahrtwege des Lifts.

Abbildung 5 zeigt den Durchsatz und die Spielzeit eines Lifts für die in Kapitel 2.5 genannten fünf Varianten bei einer Variation der Geschwindigkeit des Lifts.

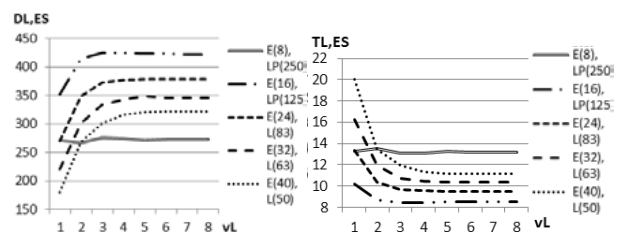


Abbildung 5: Durchsatz (Links) und Spielzeit (rechts) eines Lifts, Variation v_L

Eine Erhöhung der Geschwindigkeit des Lifts bis zu 4 m/s^2 hat eine durchsatzsteigernde (=spielzeitsenkende) Wirkung. Bei höheren Geschwindigkeiten bleibt der Durchsatz weitgehend konstant, da der Einfluss der Geschwindigkeit des Lifts im Vergleich zu dessen Lastübergabezeit immer geringer wird. Dies trifft bei Variante 1 nicht zu, da hier aufgrund der Länge des Shuttle-Systems die Shuttle-Fahrzeuge den Engpass darstellen.

Abbildung 6 zeigt die Wartezeiten und den Nutzungsgrad eines Shuttle-Fahrzeugs. Je schneller der Lift fährt, desto geringere Wartezeiten haben die Shuttle-Fahrzeuge, da die Ebenen häufiger durch den

Lift bedient werden. Der Nutzungsgrad eines Shuttle-Fahrzeugs wird durch die Geschwindigkeitsänderungen des Lifts nur geringfügig beeinflusst.

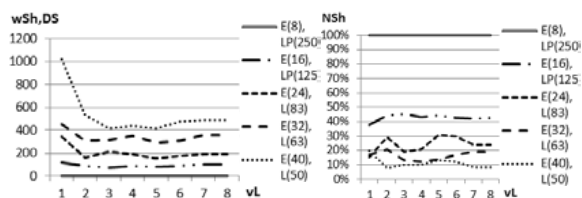


Abbildung 6: Wartezeit (links) und Nutzungsgrad (rechts) eines Shuttle-Fahrzeugs, Variation v_L

Abbildung 7 zeigt den Durchsatz und die Spielzeit eines Lifts bei einer Variation der Lastübergabezeit des Lifts. Eine Senkung der Lastübergabezeit des Lifts hat eine durchsatzsteigernde Wirkung. Der sehr hohe Einfluss der Lastübergabezeit auf Durchsatz und Spielzeit ist erkennbar.

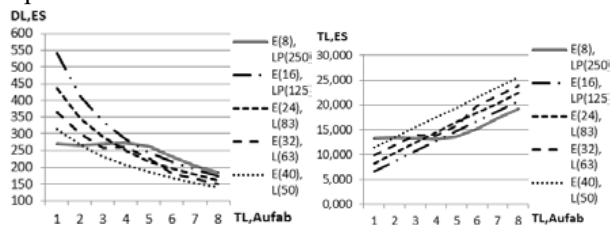


Abbildung 7: Durchsatz (links) und Spielzeit (rechts) eines Lifts, Variation $T_{L,Aufab}$

Abbildung 8 zeigt den Durchsatz eines Shuttle-Fahrzeugs bei einer Variation der Lastübergabezeit des Lifts.

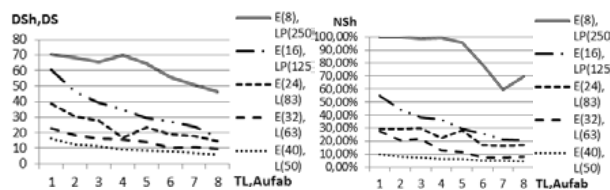


Abbildung 8: Durchsatz (links) und Nutzungsgrad (rechts) eines Shuttle-Fahrzeugs, Variation $T_{L,Aufab}$

Der Durchsatz eines Shuttle-Fahrzeugs ist ebenfalls von den Lastübergabezeiten des Lifts abhängig. Der Durchsatz eines Shuttle-Fahrzeugs ist in Variante 1 am höchsten, da diese Variante die einzige ist, bei welcher der Lift nicht den Engpass darstellt. Je geringer die Lastübergabezeiten des Lifts, desto weniger werden die Shuttle-Fahrzeuge bzgl. ihres Durchsatzes begrenzt. Weiterhin ist die Ebenenanzahl der Varianten zu beachten: Variante 5 hat deshalb den geringsten Durchsatz für ein Shuttle-Fahrzeug, da das Shuttle-System 40 Ebenen hat. Das führt dazu dass der Lift die einzelnen Ebenen seltener bedient als z. B. bei Variante 1 mit 8 Ebenen.

Der Nutzungsgrad eines Shuttle-Fahrzeugs sinkt bei steigender Lastübergabezeit, da die Shuttle-Fahrzeuge dann seltener durch den Lift bedient werden.

Abbildung 9 zeigt den Durchsatz der Zonierungsvarianten bei Variation der Parameterwerte.

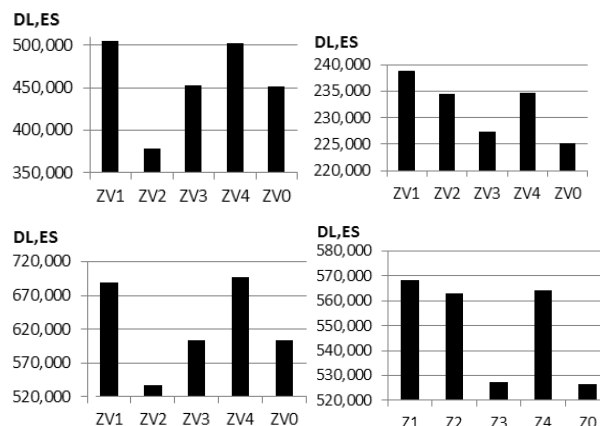


Abbildung 9: Durchsatz des Lifts für unterschiedliche ZV und Parametervariationen, Parameter oben links $T_{L,Aufab}(2)$, $T_{Sh,Aufab}(4)$, oben rechts $T_{L,Aufab}(6)$, $T_{Sh,Aufab}(8)$, unten links $T_{L,Aufab}(1)$, $T_{Sh,Aufab}(2)$, unten rechts $a_L(6)$, $v_L(7)$, $a_{Sh}(6)$, $v_L(7)$.

ZV1 und ZV4 haben die höchsten Durchsätze, da sie die Spielzeiten von Lift und Shuttle-Fahrzeug optimieren. ZV2 ordnet die Zonen reihenweise an, dadurch steuert der Lift nur die unteren Ebenen häufiger an. Der Durchsatz verteilt sich auf wenige Shuttle-Fahrzeuge, diese werden auch nicht bzgl. ihrer Spielzeit optimiert. Daraus resultiert der geringere Durchsatz. ZV2 hat z. T. geringere Durchsätze als ZV0 (keine Zonierung). ZV3 ordnet die Zonen spaltenweise an. Dadurch werden die Shuttle-Spielzeiten optimiert. Der Lift steuert aber weiterhin alle Ebenen mit der gleichen Häufigkeit an. Deutlich zeigt sich bei allen Zonierungsvarianten der Einfluss der Lastübergabezeiten auf den Durchsatz. Bei $T_{L,Aufab}(1)$ und $T_{Sh,Aufab}(2)$ können die höchsten Durchsätze erreicht werden.

Zusammenfassend zeigt die Simulationsstudie dass der Einfluss der Lastübergabezeiten auf den Durchsatz sehr hoch ist und Geschwindigkeitserhöhungen nur bis zu einem bestimmten Grad durchsatzsteigernd sind. Je nach verwendeten Parametern existiert ein optimales Verhältnis zwischen Länge und Höhe des Lagers. Die Durchsatzbasierte Lagerplatzbelegung kann den Durchsatz deutlich steigern, wenn so zониert wird dass die Durchsätze des Lifts und des Shuttle-Fahrzeugs jeweils erhöht werden und in den häufig frequentierten Ebenen in etwa übereinstimmen.

3 Zusammenfassung und Ausblick

In diesem Beitrag wurde eine Möglichkeit zur Generierung von Simulationsmodellen vorgestellt. Die Generierung der Simulationsmodelle bezieht sich auf Shuttle-Systeme. Das Vorgehen zur Generierung könnte aber ebenso auf andere Systeme übertragen werden.

Durch die automatisierte Generierung von Simulationsmodellen ist es möglich ein unternehmensspezifisches Shuttle-System abzubilden.

Das Vorgehen zur Generierung von Simulationsmodellen wurde im Beitrag angewendet und es wurden verschiedene Analysen zur Ermittlung des Systemverhaltens von Shuttle-Systemen durchgeführt. Es zeigte sich ein deutlicher Einfluss der Lastübergabezeiten auf den Durchsatz. Eine Erhöhung der Geschwindigkeit steigert ebenfalls den Durchsatz, allerdings nur bis zu einem Grenzwert, anschließend bleibt der Durchsatz trotz weiterer Erhöhung weitgehend konstant, da die Lastübergabezeiten dann überwiegen. Weiterhin konnte gezeigt werden dass bei Anwendung der Durchsatzbasierten Lagerplatzbelegung Zonierungsvarianten von Vorteil sind, welche die Spielzeiten von Lift und Shuttle-Fahrzeugen optimieren und aufeinander abstimmen.

Ausblick: Das beschriebene Simulationsmodell kann um weitere Lagerkonfigurationen (Shuttle-Systeme mit Fahrzeufliften) und Lagerstrategien erweitert werden. Der Algorithmus zur Anpassung der Automod-Dateien wird entsprechend komplexer, um eine automatische Generierung der Modelle sicherzustellen. Die Variablenzahl wird ebenfalls größer. Weiterhin kann auch die Durchführung von Simulationsläufen automatisiert werden.

4 References

- [1] VDI-2692. *Shuttle-Systeme für Kleinbehälterlagerung*. Berlin: Beuth-Verlag, 2015.
- [2] B. Y. Ekren, Z. Sari und T. Lerher. *Warehouse design unter class-based storage policy of shuttle-based storage and retrieval system*. IFAC-PapersOnLine, Jg. 48, Nr. 3, S. 1152 - 1154, 2015.
- [3] T. Kriehn, F. Schloz, K.-H. Wehking und M. Fittinghoff. *Storage management policies for throughput optimization of shuttle-based storage and retrieval systems*. Proceedings of the XXII International Conference on Material Handling, Constructions and Logistics, Planeta Print, Serbien, Belgrad, S. 177 - 184, 2017.
- [4] S. Zhang, N. Zhao und Y. Zhao. *Universal Simulation Model of Autonomous Vehicle Storage & Retrieval System*. Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014, S. 325 - 330, 2015.
- [5] M. Eder und G. Kartnig. *Durchsatzoptimierung von Shuttle-Systemen mithilfe eines analytischen Berechnungsmodells*. Logistics Journal, Proceedings, 2016.
- [6] T. Lerher. *Design of Experiments for Identifying the Throughput Performance of Shuttle-Based Storage and Retrieval Systems*. Proceedings from the 10th International Scientific Conference Transbaltica 2017, Procedia Engineering 187, S. 324 - 387, 2017.
- [7] T. Lerher, M. Borovinsek, M. Ficko und I. Palcik. *Parametric Study of Throughput Performance in SBS/RS based on Simulation*. International Journal of Simulation Modelling (16), S. 96 - 107, 2017.
- [8] G. Marchet, M. Melacini, S. Perotti und E. Tappia. *Development of a framework for the design of autonomous vehicle storage and retrieval systems*. International Journal of Production and Research, Jg. 51, Nr. 14, S. 4365 - 4387, 2013.
- [9] Z. Ning, L. Lei, Z. Saipeng und G. Lodewijks. *An efficient simulation model for rack design in multi-elevator shuttle-based storage and retrieval system*. Simulation Modelling Practice and Theory, Jg. 67, S. 100 - 116, 2016.
- [10] T. Lerher, B. Ekren, G. Dukic und B. Rosi. *Travel time model for shuttle-based storage and retrieval systems*. International Journal of Advanced Manufacturing Technology, Jg. 78, Nr. 9, S. 1705 - 1725, 2015.
- [11] G. Marchet, M. Melacini, S. Perotti und E. Tappia. *Analytical model to estimate performances of autonomous vehicle storage and retrieval systems for product totes*. International Journal of Production Research, Jg. 50, Nr. 24, S. 7134 - 7148, 2012.
- [12] B. Y. Ekren, A. Akpunar, Z. Sari und T. Lerher. *A Queuing Network Approach for Performance Estimation of Shuttle Based Storage and Retrieval System Design*. Proceedings of the XXII International Conference on Material Handling, Construction and Logistics, Planeta Print, Serbien, Belgrad, S. 185 - 188, 2017.

MbedTarget - A Simulink Target for Cortex-M Microcontrollers

Olaf Hagendorf

HS Wismar, Research Group CEA

olaf.hagendorf@hs-wismar.de

The MATLAB/Simulink add-ons for C and C++ code generation, MATLAB Coder, Simulink Coder and Embedded Coder, are a widely used and well established technologies for rapid prototyping, model based design, real-time simulation and similar technologies. Mathworks provides the prerequisites for code generation for many Simulink blocks and MATLAB functions. Missing is the direct support of peripheral functions like digital and analog input/outputs, communication functionalities and other microcontroller features when the goal is to execute a Simulink model within a small embedded system. To extend the above coders to support these functions, several add-on toolboxes exist, delivered by Mathworks itself or by third-party suppliers. In the following, the programming of microcontrollers is shortly introduced and their resource utilization is compared, starting with assembler up to code generation by a Mathworks coder products. After introducing and comparing different Cortex-M coder toolboxes, the MbedTarget toolbox as an open source alternative is presented. With examples and code snippets, the principle of the code generation process, the work flow of MbedTarget together with the principle to define a Simulink block is shown.

1 Introduction

MATLAB/Simulink is widely used in the engineering education, among others in control theories. Other topics of the student education are microcontroller (MCU) programming and the integration of these into physical processes. Main tools for implementing the algorithms are C and C++ compiler. The interaction with physical systems makes knowledge about control theory often necessary. But the media break between Simulink at one side and C and C++ programming at the other side can make this complicated.

With MATLAB, Simulink and Embedded Coder, toolboxes to create C and C++ code out of Simulink models or MATLAB programs are provided.

To combine both topics, the direct support of peripheral functions like digital and analog input/outputs and other MCU features to interact with a physical system within a Simulink model is necessary. For this, additional packages to extend the coder toolboxes exist, as MATLAB or third-party Add-Ons.

After a short introduction of MCU programming variants in chapter 2 and their resource comparison, the next chapter describes a few coder toolboxes. In chapter 4, the new MbedTarget is introduced as an alternative coder toolbox.

2 Cortex Microcontroller

Cortex MCUs are 32- and 64bit microprocessors, divided into three subfamilies Cortex-M, -R and -A.

They are licensed by ARM Holding. The licensees are producing MCUs with the Cortex microprocessor core but company specific peripheral components.

2.1 Cortex-M family Overview

Members of the Cortex-M family are 32bit MCUs with a broad range of processing power, memory sizes, peripheral components etc. Since the introduction of the first variant, the Cortex-M3 core was released in 2005 and first silicon products were sold in 2006 [1], other variants with more or less powerful cores, e.g. Cortex-M7 and -M0, were introduced. The Cortex-M MCUs have been become a widely used technology, a huge amount of manufacturers are delivering variants. The distributor mouser [2] has ca. 6800 variants of Cortex-M MCUs in its catalog. At all, it lists ca. 40000 MCUs.

2.2 Programming

The following subchapters are shortly introducing and comparing typical embedded programming principles.

2.2.1 Assembler

Since the invention of microprocessors, assembly language has been used [3]. To have the full control over them and their periphery, it is still common to use assembly language. The importance of it decreased over the time in comparison to other languages, but it is still under the 10 most used [4].

2.2.2 C/C++ with HAL library

Many MCU producing companies does not only provide the silicon chip but delivers also software libraries. Experiences show that unlike with previous 8 and 16bit processors, the programming of current 32bit MCUs without these libraries considerably increases the overhead. As an example, STM32Cube by STMicroelectronics (STM) [5] is introduced. The software consists of two parts:

- STM32CubeMX
A tool for C code generation for initialization of hardware functions, adding and configuring of middleware libraries like TCP/IP stack, RTOS, USB, file systems etc. and finally the generation of project files for several integrated development environments (IDEs).
- STM32 MCU packages
These packages implement a hardware abstraction layer (HAL) to provide standardized API calls for all STM MCU families. An additional part of the packages are middleware libraries like TCP/IP stack, RTOS, USB, file systems and graphical libraries.

2.2.3 Mbed library

The Mbed project was started in 2005 by two ARM employers, Simon Ford and Chris Styles, who are helping out in undergraduate and after-school projects. Both were not satisfied with the current situation and developed the idea to ease the MCU development [6]. Among others, main ideas of this are [7]:

- open source MCU debugging and programming hardware
- online toolchain
- object oriented HAL with adaption layer to different company specific HAL libraries

The object oriented HAL is a C/C++ MCU software platform containing object oriented peripheral and library APIs, C adaption layers and startup code for 232 MCUs [7]. Additionally, several Python tools are part of Mbed, providing functions for compiling, testing, library management, exporting project files to several IDEs etc.

Mbed significantly reduces the effort of getting started with ARM based MCUs. A drawback is the amount of additional software layers which leads to an increase of binary sizes. The possibility to compile the same code for different MCUs is advantage and a disadvantage at the same time.

2.2.4 Data flow oriented programming

This programming paradigm, i.e. visual programming, has some advantages in comparison to the above mentioned imperative, text based languages [8, 9]:

- the depiction of a real world object succeed particularly well
- challenging concepts of imperative languages like variables and dynamic data structures are not relevant
- it provides parallel execution without the need of explicit instruction
- the clarity eases prototypical implementation

2.2.5 Conclusion

As a comparison of the different programming principles, the hello world of MCU programming, a LED blinking application, is compared in its resource usage (sizes of read-only and read/write memory) [10]. The effort to program the application, can only be qualitatively estimated. The effort decreases from the usage of assembler language, over a general C and HAL programming and Mbed. The data flow oriented programming will produce the lowest cost in terms of time to create the application. But it also has the highest resource utilization as can be seen in table 1.

programming principle	Flash size (byte)	RAM size (byte)
assembler	88	0
C without library	716	1632
MCU specific HAL	1392 ¹ 2852 ²	1032 ¹ 1032 ²
Mbed HAL	22576 ³ 37716 ⁴	1432 ³ 8484 ⁴
Data flow oriented ⁵	58932	8060

¹ low layer library used [5]

² high layer library used [5]

³ without RTOS

⁴ with RTOS (default config.)

⁵ MbedTarget v1

Table 1. Resource comparison of programming principles using the blinking example [10]

3 MATLAB/Simulink targets for Cortex-M controller

Simulink is a widely used data flow oriented development tool. To program MCUs with it, beside the MbedTarget, introduced in this paper, several other targets exist. They are provided by the Add-Ons manager of MATLAB, or are available as third-party tools. Three of these: Embedded Coder Support Package for STMicroelectronics Discovery Boards [11], Simulink

Coder Support Package for STMicroelectronics Nucleo Boards [12] and STM32-MAT/TARGET [13] are introduced in the following.

3.1 Embedded Coder Support Package for STMicroelectronics Discovery Boards

STM32 Discovery kits are low cost solutions for the evaluation of STM32 Cortex-M MCUs by STM. Beside the MCU and the necessary infrastructure to use it, the kits contains additional peripheral items like displays and sensors. Mouser [2] lists 20 kits in its catalog, the support package can handle three of them: STM32F746G-DISCO, STM32F769I-DISCO and STM32F4-Discovery.

For the STM32F4-Discovery the old, no longer supported standard peripheral library is used, the other two boards are using the Mbed library.

The only supported compiler is GNU GCC.

For the STM32F4-Discovery board, Simulink blocks for analog in, digital in/out, audio in/out and Interrupt handling are available. All MCU pins are usable.

For the other boards, blocks for analog in, digital in/out, serial communication, timer, TCP, UDP and audio in/out are available. The MCU pin usage is restricted to 22 pins.

The supported MCU hardware functionality for all three variants is very basic, the pin usage is partly restricted.

To generate, compile and link the C Code of a Simulink model a single step – Deploy to Hardware – is necessary.

The package requires the Embedded Coder toolbox.

Figure 1 shows an example, an alternating digital pin.

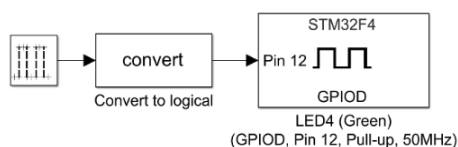


Figure 1. Example with Embedded Coder Support Package for STM Discovery Boards

3.2 Simulink Coder Support Package for STMicroelectronics Nucleo Boards

STM32 Nucleo kits are the lowest cost solutions for the evaluation of STM32 Cortex-M MCUs by STM. Beside the MCU and the necessary infrastructure to use it, the kits does not contain additional peripheral items except a few kits with Ethernet connectivity.

Mouser [2] lists 38 kits in its catalog, the support package supports 9 of them: Nucleo-F401RE, Nucleo-F103RB, Nucleo-F302R8, Nucleo-F031K6, Nucleo-L476RG, Nucleo-L053R8, Nucleo-F746ZG, Nucleo-F411RE and Nucleo-F767ZI.

For all boards the Mbed library is used.

The only supported compiler is GNU GCC.

For all boards, Simulink blocks for analog in, digital in/out, serial communication and timer are available. The MCU pin usage is restricted to 22 pins. The supported MCU hardware functionality is very basic.

To generate, compile and link the C Code of a Simulink model a single step – Deploy to Hardware – is necessary.

The package requires the Simulink Coder toolbox.

Figure 2 shows an example, an alternating digital pin.

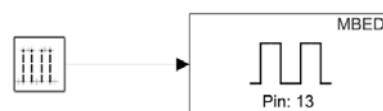


Figure 2. Example with Simulink Coder Support Package for STM Nucleo Boards

3.3 STM32-MAT/TARGET

This packages is provided by STM. It is based on STM32Cube. The Simulink package supports all STM32 MCUs.

All compiler supported by STM32CubeMX can be used: EWARM, Keil MDK V4 and V5, TrueSTUDIO, SW4STM32 and GNU GCC.

Blocks for analog in/out, digital in/out, serial communication, timers, watchdogs are provided. The block set uses the code generation utility STM32CubeMX. Because of this, the supported MCU hardware functionality is considerably large.

To generate, compile and link the C Code of a Simulink model, two steps are necessary: executing the Simulink function: Deploy to Hardware and project building in the chosen IDE.

The package requires the Embedded Coder toolbox.

Figure 3 shows an example, an alternating digital pin.



Figure 3. Example with STM32-MAT/TARGET

3.4 Conclusion

Whereas the provided functionality of the support packages introduced in chap. 3.1 and 3.2 is very basic, the other package offers nearly as much functions as the MCU hardware provides.

All packages are free of charge. But the main disadvantage of all is: they are closed source. Own extensions are difficult to realize or even not possible. The supported MCUs depends on the manufacturer, mainly STM32 products are supported. Mathworks and third parties offers some more targets supporting e.g. NXP KL25Z and K64F, Infineon XMC, Nordic Semiconductor NRF51 and BBC micro:bit. Although the number of Cortex-M MCUs is huge, the selection of Simulink targets and supported MCUs is restricted.

4 MbedTarget

MbedTarget is completely open from both sides: all code and configuration files i.e. MATLAB code, templates, configurations etc. as well as all MCU libraries are available free of charge and as source codes. It is based on MbedOS 5, therefore all MbedOS 5 compatible microcontrollers can be used to run Simulink models.

MbedTarget uses internally GNU GCC to compile the generated source code. Additionally, project files for all Mbed supported IDEs can be generated. With these project files, the generated code can be manually compiled and/or debugged when the Simulink model does not run or own blocks are developed.

The Simulink block library contains two groups of blocks:

- for Mbed functionality
- for sensors, actors, .. based on additional libraries

The first group contains blocks for analog in/out, digital in/out, user LEDs, user buttons, serial communication, timers, Ethernet, RTOS etc.

The second group contains blocks for external analog to digital converter, digital to analog converter, external digital in/out chips, several sensors and actors like temperature, pressure, motion, magnetometer, range, motor control, display, etc. Additionally, MCU specific blocks for encoder, input capture, random number generator and counter are provided.

MbedTarget supports both available coder toolboxes with Simulink model code generation capabilities, Simulink Coder and Embedded Coder.

To generate, compile, link and flash the C Code of a Simulink model a single step – Deploy to Hardware – is necessary.

Figure 4 shows an example, a blinking LED.

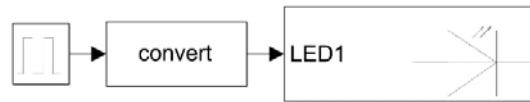


Figure 4. Example with MbedTarget

4.1 Principle of MbedTarget code generation

The principle workflow when processing a Simulink model to an executable binary is shown in figure 5.

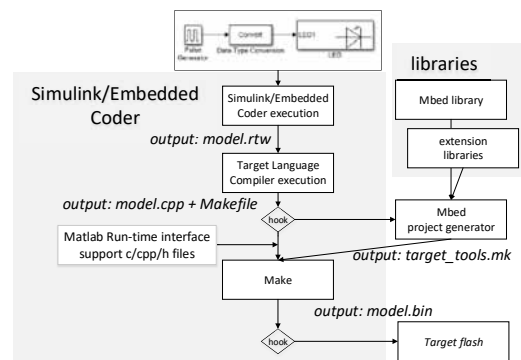


Figure 5. Principle of the MbedTarget code generation

Simulink or Embedded Coder are generating an rtw file from the model. That is a specific textual representation of the model. Together with tlc files (target language compiler files) which are part of MATLAB and of the specific Simulink target, the target language compiler (TLC) generates several c/cpp/h files and a Makefile meanwhile the process can be influenced by hook calls.

MbedTarget uses the code generation principle based on tlc files: `mbed.tlc` for the Embedded Coder or `mbed_grt.tlc` for Simulink Coder as the starting points.

The TLC offers several hooks to customize the code generation process i.e. the transformation process from rtw to c/cpp files. Mainly two hooks are used by MbedTarget: *before_calling_make* for code handling processes and *after_calling_make* for the compiling and flashing process.

The whole procedure can be summarized by the following steps:

1. The TLC uses the current MATLAB working path to store all generated files in a folder with a name constructed using the Simulink model name and

'slprj', e.g. blinky_slprj. During this process Simulink tlc files were used for each standard Simulink block, for each non-Simulink block another tlc file is provided by MbedTarget. Beside the c/cpp/h files generated out of tlc files, also a makefile is created based on template makefiles mbed.tmf or mbed_grt.tmf. This makefile needs an additional make include file, generated within the hook function, described in step 2.

2. After step 1, during the hook call *before_calling_make*, a target specific folder is created. Into this folder, all generated files are copied and a Mbed specific make include file is generated. This include file is very similar to a standard gcc makefile used by Mbed.
3. After executing the hook *before_calling_make* make is called and a bin file is created.
4. During the hook call *after_calling_make*, the bin file is flashed to the target MCU.

4.2 MbedTarget Simulink block usage

This chapter describes shortly the components of an MbedTarget Simulink block. The Digital Output block is chosen because it is one of the simplest blocks. It has only a single input port to write a digital value to MCU pin and is depicted in figure 6.



Figure 6. MbedTarget Digital Output block

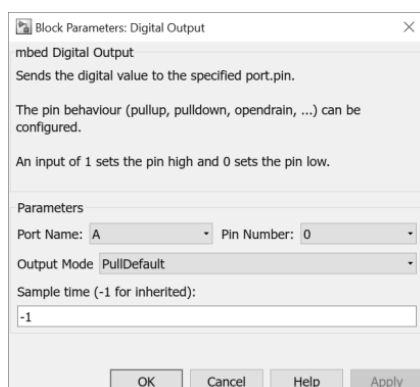


Figure 7. Parameter dialog box

With a double click onto the block, a configuration dialog opens as shown in figure 7. The block has 4 parameters which has to be configured in the dialog:

- Port Name and Pin Number to choose a digital port, e.g. PA0

- Output Mode for pull up, pull down, open drain, ... - options are corresponding to Mbed options of the DigitalInOut C++ class for digital in/out pins.
- Sample time defines the time period, how often the digital value is written to the MCU pin. The value has to be a multiple of the global step size.

4.3 MbedTarget Simulink block creation

To implement custom Simulink blocks, the following elements are necessary:

- a block mask
- a S-function
- a tlc file consisting of a mixture of target language and C/C++ code

The **block** mask defines the outlook of the block as shown in the Simulink model editor, defines the input items available in the block mask dialog. It also defines block title, block help text and the help menu entry.

The block parameters connect the block with a **S-function**, i.e. a binary mexw64 file implemented in C, and a tlc file which has the identical name. In MbedTarget, the S-function mainly describes the behavior of the block: number and type of input and output ports, checks parameters and prepares the transfer of these to the TLC.

A **tlc file** is a mixture of tlc code, a script like language, and C/C++ code in a form of snippets. The script code controls the usage of the C/C++ snippets and how and where they are put into a generated C/C++ file. The tlc file contains mainly three functions: *Setup*, *Start* and *Output*. *Setup* controls the inclusion of additional headers and source files, where *Start* is executed once to generate initialization code and *Output* is called once in every simulation loop. Even if the name is output, the function has to handle also input values, send by Simulink to input ports.

In the following, snippets of the *Start* function are described in detail as an example:

```

1 %assign nPortName=LibBlockParameterValue(PortName,0)
2 %assign nPinNum=LibBlockParameterValue(PinNumber,0)
3 %assign pname="P"+FEVAL("char",nPortName+64)
4 %assign pname=pname+"_"+FEVAL("int2str",nPinNum-1)

```

Lines 1 and 2 fetches the variables entered in the block mask dialog. Lines 3 and 4 creates from these a pin name, e.g. PA_0 as used for STM32 MCUs.

```

5 %assign name = FEVAL(
  "strrep", LibGetFormattedBlockPath(block), "/", "_")
6 %assign name = FEVAL("strrep", name, " ", "_")
7 %assign name = FEVAL("strrep", name, "-", "_")

```

Lines 5 to 8 create a unique name based on the complete block name, e.g. *blinky_Digital_Output* for a block *Digital Output* in a model *blinky*. The Simulink path contains characters, e.g. '/', spaces and '-', which are replaced by '_' to create a valid C identifier.

```

8 %openfile declbuf
9 DigitalInOut %<name>(%<pname>);
10 %closefile declbuf
11 %assign srcFile = LibGetModelDotCFile()
12 %<LibSetSourceFileSection(srcFile, "Declarations",
  declbuf)>

```

Finally, lines 8 to 12 create a single line in the declaration section of the generated source file, shown in line 13:

```
13 DigitalOut blinky_Digital_Output(PA_0);
```

The line 13 is the necessary Mbed code to create a digital output.

A few more lines in the tlc file, using the same principle, are creating the remaining, necessary C code.

4.4 MbedTarget main function

The MbedTarget supports the single task model of Simulink. To create the main function, the target contains template file: `mbed_srmain.tlc` and `mbed_grt_main.cpp`. The creation of multithreaded application by Simulink is not yet supported, but can be done manually with MbedTarget RTOS blocks.

5 Conclusion

MbedTarget was developed to improve and ease the MCU usage in several modules like robotics, sensor/actor systems and embedded control systems of student education. A common characteristic of all these modules is the intensive usage of MATLAB/Simulink in the theoretical part. The practical implementation of the theoretical knowledge is time consuming and error-prone when using typical programming languages like C/C++. By using MbedTarget, the media break between Simulink and C programming could be removed. The disadvantage of an increased resource usage does not play a role in the prototypical implementations in this application field, enough MCU resources are available without problems.

MbedTarget is published at GitHub [14] and will be continuously developed there.

6 References

- [1] J.Yiu. *Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors*. Newnes, UK, 2013.
- [2] <https://www.mouser.de> (accessed 01.2018)
- [3] D.Kusswurm. *Modern X86 Assembly Language Programming*. Springer, USA, 2014.
- [4] <https://www.tiobe.com/tiobe-index/> (accessed 12.2017)
- [5] STM32Cube: <http://www.st.com/en/embedded-software/stm32cube-mcu-packages.html> (accessed 12.2017)
- [6] <https://os.mbed.com/handbook/Founders-interview> (accessed 12.2017)
- [7] <https://www.mbed.com/en/> (accessed 12.2017)
- [8] C.Keltsch. *Ein visuelles Programmiersystem zur Modellierung deskriptiver Untersuchungen in der Datenanalyse*. Diplomarbeit, Diplomica Verlag, 1998.
- [9] W.Maydl. *Komponentenbasierte Softwareentwicklung für datenflußorientierte eingebettete Systeme*. Dissertation, University Passau, Germany, 2005.
- [10] <https://github.com/ATM-HSW/ASIMFachtagung2018> (accessed 01.2018)
- [11] https://de.mathworks.com/hardware-support/st-discovery-board.html?s_tid=AO_HS_info (accessed 01.2018)
- [12] https://de.mathworks.com/hardware-support/st-nucleo.html?s_tid=AO_HS_info (accessed 01.2018)
- [13] http://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-utilities/stm32-mat-target.html (accessed 01.2018)
- [14] https://github.com/ATM-HSW/mbed_target (accessed 01.2018)

Modellbasiert lernende Steuerung für Gelenkarmroboter (Work in Progress)

Georg Kunert, Thorsten Pawletta

Hochschule Wismar - University of Applied Sciences, Technology, Business and Design

Forschungsgruppe Computational Engineering and Automation (CEA)

georg.kunert@cea-wismar.de

Ausgehend vom Entwicklungsstand bei der Steuerungsprogrammierung von Gelenkarmrobotern werden die Zielstellung einer lernenden Steuerung und wesentliche Unterschiede zur konventionellen Herangehensweise aufgezeigt. In diesem Rahmen wird zunächst auf die in der Forschungsgruppe entwickelte Robotic Control and Visualization Toolbox (RCV Tbx) für MATLAB eingegangen. Anhand eines ausgewählten Verfahrens, dem Reinforcement Learning, und einer Beispielanwendung wird gezeigt, wie eine modellbasiert lernende Robotersteuerung für Gelenkarmroboter umgesetzt werden kann.

Es wird explizit der Lernvorgang und die Transformation des erlernten Wissens in eine Steuerung betrachtet. Dabei werden auch die Vor- und Nachteile des untersuchten Verfahrens diskutiert. Abschließend wird kurz die praktische Umsetzung mit MATLAB unter Verwendung der RCV Tbx für einen Kuka Agilus Roboter diskutiert.

Praxisbeispiel: Emulation mit SAP EWM

Alexander Klemm¹, Dominik Wittwer¹

¹io-consultants GmbH & Co. KG

Alexander.Klemm@io-consultants.com

Dominik.Wittwer@io-consultants.com

In diesem Beitrag werden die Möglichkeiten der Simulationssoftware Demo3D an einem Praxisbeispiel einer Emulation mit dem Lagerverwaltungssystem (LVS) „SAP Extended Warehouse Management“ (SAP EWM) vorgestellt.

Im Praxisbeispiel wird ein neues LVS auf Basis von SAP EWM mit der integrierten Komponente „Material Flow System“ (MFS) eingesetzt. Das automatisierte Lager besteht aus einem 8-fach tiefen Kanallager, einem 2-fach tiefen Kommissionier Shuttle, sowie einer Fördertechnik. Die Steuerung wird in MFS und ohne zusätzlichen zwischengeschalteten Materialflussrechner (MFR) umgesetzt. Weiterhin wird ein webbasierter Kommissionierdialog mittels der Web Dynpro Technologie implementiert. Die Verbindung zwischen Simulationssoftware und SAP EWM wird über eine meldepunktbasierte Kommunikation mit Telegrammen realisiert.

1 Projektsituation

Um die Lieferfähigkeit und die flexible und termingerechte Abwicklung von Kundenanforderungen für die nächsten 15 Jahre sicherzustellen, wurde durch den Kunden das Projekt „Lager + Logistik“ ins Leben gerufen. Neben der Schaffung von ausreichender Lagerkapazität und der Beherrschung der zunehmenden Kleinteiligkeit in der Logistik sind die Optimierung und Maximierung von:

- Abläufen
- Verfügbarkeit
- Schnelligkeit
- Zuverlässigkeit
- Kosten

wesentliche Projektziele. Die Schaffung ausreichender Lagerkapazität soll durch den Neubau eines Shuttle-Lagers geschehen, welches in das bestehende Ökosystem integriert wird.

Mit Hilfe einer Emulation wurde dem Kunden eine mögliche Umsetzungsvariante für das entstehende Lager aufgezeigt. Durch die Visualisierung der Kopplung mit dem geforderten LVS SAP EWM wurde ein Ausblick auf den zukünftigen Lagerbetrieb.

2 Verwendete Technologien

2.1 Lagerverwaltungssystem

SAP EWM bildet die Bestandsverwaltung und den Warenfluss im Lager ab. Waren können über das System ein- und ausgelagert, sowie zu verschiedenen Meldepunkten transportiert werden. Die Darstellung von

Bestandstabellen in SAP EWM ist in Abbildung 1 dargestellt.



Typ	Lagerort	Menge BME DA	Bestsch.	Bestandort	Charge	KF	Herku	Elg	Glmer	Verfügbar		
SHL	SHL-01-02-04 02	230000000209	B-70210	Trüffelberrunst	11	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-06 02	730000000302	B-70210	Trüffelberrunst	10	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-02 02	730000000317	B-66610	Saltufo	5	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 01	730000000323	B-66610	Saltufo	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-04 01	730000000331	B-52510	Ornabrücker Friedensschinken	13	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-02-09 02	730000000343	B-52510	Ornabrücker Friedensschinken	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 02	730000000349	B-52510	Ornabrücker Friedensschinken	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-02-01 01	730000000403	B-52510	Ornabrücker Friedensschinken	6	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 01	730000000415	B-70210	Trüffelberrunst	6	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 02	730000000416	B-70210	Trüffelberrunst	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 01	730000000419	B-70210	Trüffelberrunst	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-02 01	730000000423	B-66610	Saltufo	5	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300
SHL	SHL-01-01-09 02	730000000427	B-66610	Saltufo	15	ST	F2	Frei	verwendbar	Lager	LAGER0300	LAGER0300

Abbildung 1. Ansicht Bestand Shuttle-Lager in SAP EWM

2.2 Benutzeroberfläche

Zur Erleichterung der Kommissionier Tätigkeiten an den Arbeitsplätzen wird ein kundenspezifischer Arbeitsplatzdialog mit den einzelnen Arbeitsschritten entwickelt. Dieser Kommissionierdialog wird für einen einfachen Zugriff aus dem Web mittels der SAP Web Dynpro (WD) Technologie umgesetzt



Handlungsmenü: Kommissionierung - Pickplatz

Aktionen: **Salte anlegen (STRO + F7)** | **Artikel scannen (STRO + F6)** | **Salte voll (STRO + F5)** | **Kommissionieren (STRO + F8)** | **Beenden (STRO + F9)**

Status: **Artikel entnehmen und scannen**

Auftragsdaten:

Lieferanten: 31000001114	Lieferdatum: 11.12.2017 20:28:43	Kunde: Kunde 1
Auftr.-Typ: Normallieferungspostion	Ziel-PL: Salte Versand-Salte	Adress: Adress
Lieferanzahl: 1500000253	Ziel-HU: 73000000388	Freigegeben: Hier wird die Freigegeben-Zahl definiert.

Positionen:

Bild	Artikelnr.	Artikel	Quelle-HU	Gefordert	Erliegt
	B-66610	Saltufo	730000000308	2	0
B-70210	Trüffelberrunst		730000000209	1	0

Abbildung 2. WD Kommissionierdialog

Zur Vermeidung einer zeitgesteuerten Aktualisierung des WD Dialogs wird auf die ABAP Push Channel

(APC) Technologie zurückgegriffen. Dies erlaubt eine eventbasierte Aktualisierung des WD Dialogs über das Push Prinzip.

2.3 Simulation

Um das automatisierte Lager abzubilden wird die Simulationssoftware Demo3D [1] verwendet. Diese Software ermöglicht es über ein Baukastenprinzip einzelne Komponenten, wie in diesem Praxisbeispiel ein automatisches Kleinteilelager (AKL), zu visualisieren. Weiterhin kann über JavaScript und C# das Verhalten einzelner Komponenten und die zugrundeliegende Logik des Warenflusses verändert werden.

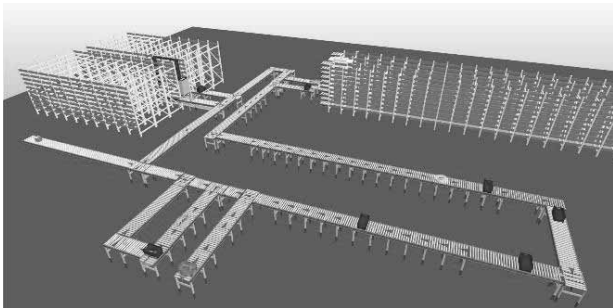


Abbildung 2. Visualisierung mit Demo3D

2.4 Kommunikation zwischen LVS und Simulation

Die Verbindung zwischen SAP EWM und der Simulation erfolgt über Telegramme aufgebaut. In SAP EWM wird die definierte Telegrammstruktur direkt eingepflegt. Zur effizienten Interpretation der Telegrammstruktur in der Simulation wird zusätzlich die Middleware Emulation Controller der Firma Simplan verwendet.

3 Umsetzung

3.1 APC Integration

Der WD Dialog muss bei Ankunft der HU am Arbeitsplatz aktualisiert werden. Mit Hilfe eines Business Add-Ins (BAI) wird nach der Quittierung der Lageraufgabe (LB) mit dem Ziel des Arbeitsplatzes eine Push Meldung angestoßen.

```

1 IF <ls_ltab_vb>-nlpla = 'AKO1-PP01' AND
   <ls_ltab_vb>-vlpla = 'AFT-PPQ01'.
2   TRY.
3     lo_producer_text ?=
       cl_amc_channel_manager=>
         create_message_producer(
           i_application_id = 'ZAMC_MFSPUSH'
           i_channel_id = '/mfs_push').
4     lo_producer_text->send( i_message =
       CONV #( <ls_ltab_vb>-vlenr ) ).
5     CATCH cx_amc_error INTO lx_amc_error.
       MESSAGE lx_amc_error->get_text( )
         TYPE 'E'.
6   ENDMETHOD.
7 ENENDIF

```

Code 1. Push Meldung bei Quittierung einer LB

Die Bearbeitung dieser Push Meldung wird mit Hilfe des WebSocket Protokolls am WD Dialog durchgeführt. Beim Starten des WD Dialogs reicht es aus, die Verbindung zum APC zu öffnen. Jede Meldung nach diesem Zeitpunkt wird ohne weiteres Anfragen am APC empfangen und kann direkt interpretiert werden.

3.2 Telegrammstruktur SAP EWM

Für relevante Lageraufgaben (Warenbewegungen) werden von SAP EWM automatisch Telegramme erzeugt und über die Schnittstelle zur Simulation an die entsprechende speicherprogrammierbare Steuerung (SPS) verteilt. Die verwendeten Telegramme enthalten Quell- und Ziellokation, die Nummer der Handling-Unit (HU) und weitere Informationen für das Routing.

```
MFS SHLSRM1 5737R SLSHLSRMS1 7300000002645HL-01-01-03AFT-DS1 V1 0000 0000 #
```

Abbildung 3. Telegrammstruktur

3.3 Emulation Controller

Da die Software Demo3D im Standard keine Implementierung der TCP/IP-Schnittstelle enthält, kann die Telegrammstruktur nicht direkt interpretiert werden. Der Emulation Controller (EMC) der Firma Simplan bildet die fehlende Funktionalität ab, und erlaubt so die Kommunikation mit SAP EWM über eine TCP/IP Schnittstelle. Er separiert Telegramme nach fixen Zeichenlängen und stellt der Simulation die einzelnen Bestandteile als Objekte zur Verfügung. In der Simulation lassen sich die übergebenen Objekte über die Programmiersprache C# ansprechen und bearbeiten.

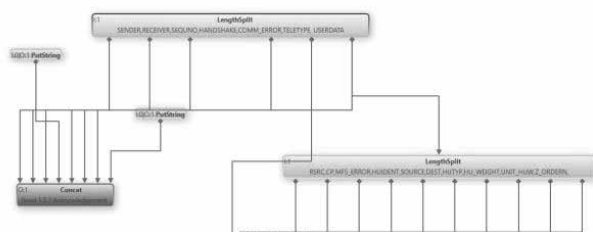


Abbildung 4. Emulation Controller

3.4 Lagerbewegungen in Demo 3D

Das Telegramm in Abbildung 3 stellt eine Kommissionier Aufgabe im Shuttle-Lager dar. Zur Verarbeitung wird es entsprechend Abbildung 4 separiert. Die Emulation setzt dies durch den Quellcodeausschnitt Code 1 um. Die ersten Zeichen der Quell- und der Ziellokation werden geprüft und die entsprechende Warenbewegung ausgeführt. Alle Lagerbewegungen wie Ein-, Aus- und Umlagerung werden analog abgebildet.

```

1 //Auslagerung Rack
2  if (
    inTransportMessage.Data.
    SourceLocation.StartsWith("SHL") &&
    !inTransportMessage.Data.
    DestinationLocation.StartsWith("SHL"))
3  {
4    _document.Run(() =>
    Retrieve(inTransportMessage,
    _comPoint.Outfeeds.Value));
5  }

```

Code 2. Position der Handling-Unit im Lager

Die Quelllokation „SHL-01-01-03“ des Beispieltelegrams aus Abbildung 3 beschreibt den Quell-Lagerplatz und die Ziellokation „DS1“ den „Drop-Stand“ des Shuttle-Lagers. In der Emulation wird der Lagerplatz der Quelllokation in die einzelnen Komponenten Gang, Regal und Kanal aufgeteilt (siehe Code 2). Die zusätzlich übergebene logische Position, gibt die Tiefe an, in der sich die HU auf dem Lagerplatz befindet.

```

1 string[] coords =
  inTransportMessage.Data.SourceLocation.Sp
  lit('-');
2  int rackNrInt =
  Convert.ToInt32(coords[1]);
3  int bay = Convert.ToInt32(coords[2]);
4  int shelf = Convert.ToInt32(coords[3]);
5  int index = Convert.ToInt32(coords[4]);

```

Code 3. Position der HU im Lager

Die eigentliche Warenbewegung wird mit Hilfe der ermittelten Koordinaten aus dem Telegramm und der

Benennung der Meldepunkte in der Simulation gesteuert.

4 Herausforderungen

Die Realisierung der vorgestellten Emulation bringt diverse Herausforderungen mit sich, sowohl innerhalb der einzelnen Systeme als auch in deren Verknüpfung.

4.1 Lagerabbild in der Simulation

In der Simulation wird neben der visuellen Darstellung auch die Logik der einzelnen Komponenten definiert. Das Praxisbeispiel verwendet sowohl ein AKL als auch ein Shuttlelager. Der Baukasten von Emulate3D enthält im Standard lediglich ein 2-fach tiefes AKL. Die Logik des 8-fach tiefen AKL muss vollständig im Quellcode ergänzt werden. Hier muss nicht nur eine visuelle, sondern auch eine grundlegende logische Veränderung vorgenommen werden. Die zusätzliche Anforderung einer visuell fehlerfreien Zulagerung in die Tiefe erhöht die Komplexität deutlich.

Zur Abbildung des Shuttle-Lager wird das AKL des Baukastens als Basis genutzt. Die einzelnen Ebenen des Shuttles werden als horizontale Kräne modelliert, die visuell an die gegebenen Anforderungen angepasst werden.

4.2 WebSocket Integration in WD

Eine direkte Integration des WebSocket Protokolls in eine WD Anwendung ist nicht ohne weiteres möglich. Der clientseitige Aufbau der WebSocket Verbindung lässt sich dennoch durch den Aufruf einer JavaScript (JS) Datei durchführen. Über eine spezielle WD HTML Komponente lässt sich das JS aufrufen und mit einer ABAP Methode verknüpfen.

```

1 try {
2  socket = new WebSocket(webSocketURI);
3  socket.onopen = function() { };
4  var that = this;
5  socket.onmessage = function(wfNotify) {
6    that.callback.fireEvent('ON_MESSAGE',
    wfNotify.data);
7  };
8  socket.onclose = function() {
9  };
10 } catch (exception) {
11 }

```

Code 4. Aufruf einer WebSocket Verbindung in JS

4.3 Synchronisation der Telegramme

In der Schnittstelle zwischen den Systemen müssen neben Warenbewegungs- auch „Life“- und Bestätigungsdiagramme bearbeitet werden. Diese werden von der Simulation bestätigt, dürfen jedoch innerhalb der Simulation nicht weitergegeben werden. Der EMC fängt die entsprechenden Telegramme ab und ändert die Empfangsflags.

Für die Abbildung der Warenbewegungsdiagramme sind im EMC feste Längen definiert, die auch in SAP EWM in einer festen Struktur hinterlegt werden müssen. Auch der Aufbau und die Reihenfolge muss in beiden Systemen aufeinander abgestimmt werden. So ist die Benennung der Meldepunkte für den lagerinternen Warenfluss entscheidend.

5 Fazit

Um die Warenbewegungen in einem neu geplanten Lager sowohl logisch als auch visuell abzubilden wurde das Lagerverwaltungssystem SAP EWM mit der Simulationssoftware Demo3D kombiniert.

Mittels TCP/IP-Verbindung und EMC wurde eine Verbindung zwischen SAP EWM und Demo3D realisiert. Die einzelnen Komponenten innerhalb von Demo3D wurden den abzubildenden Funktion und den Anforderungen des Kunden angepasst. Die Simulation wird über Transporttelegramme gesteuert und der Warenfluss wird vollständig visualisiert.

In Demo3D werden sämtliche Lagerbewegungen aus SAP EWM anschaulich aufgezeigt, wodurch ein umfassendes Verständnis der Lagerprozesse sowie deren Auswirkungen auf die physischen Abläufe im Lager vermittelt wird.

6 References

- [1] Simplan. *Demo3D*. <https://www.simplan.de/software/demo3d/>

Über io-consultants

io-consultants bietet Beratungs- und Planungsleistungen für Produktion, Logistik, Architektur und IT aus einer Hand. Seit der Gründung 1958 begleitet io-consultants ihre Kunden als Generalplaner von der ersten strategischen Überlegung bis hin zur Realisierung komplexer Industrieprojekte in der Supply und Value Chain.

Erfahrene Spezialisten arbeiten in interdisziplinären Teams zusammen. Gemeinsam schaffen sie unternehmensspezifische Strukturen und Prozesse für die Anforderungen ihrer Kunden. Das Ergebnis sind zuverlässige, produktive und effiziente Lösungen, die messbaren Erfolg erbringen.

Mit mehr als 250 Mitarbeitern ist io-consultants weltweit als eines der führenden technischen Beratungs- und Planungsunternehmen an acht Standorten vertreten: Neben dem Hauptsitz in Heidelberg hat das Unternehmen Niederlassungen in München, Leonberg, Berlin, Bethlehem (PA, USA), Singapur, Dubai und Hong Kong.

Maschinelles Lernen für das Energiesystem: Gegenwart und Zukunft

Ralf Mikut,

Karlsruher Institut für Technologie, Institut für Automation und Angewandte Informatik

ralf.mikut@kit.edu

Sowohl das Energiesystem als auch das Gebiet des Maschinellen Lernens durchlaufen zur Zeit einen tiefgreifenden Wandel. Die Herausforderungen beim maschinellen Lernen bestehen in den steigenden Datenmengen (zusammengefasst im Schlagwort "Big Data") sowie im Wunsch, den Entwurf von Lernverfahren immer stärker zu automatisieren. Beim Energiesystem geht es insbesondere um die Beherrschung von Systemen mit vielen lokal verteilten und fluktuierenden Einspeisungen (z.B. Solar- und Windkraftanlagen) im Rahmen der Energiewende durch Smart Grids. Das Ziel dieses Beitrags besteht darin, die Methoden des Maschinellen Lernens näher zu erläutern und typische Anwendungsszenarien für Smart Grids aufzuzeigen.

1 Einleitung

Im Rahmen der Energiewende wird ein Anteil von 80% erneuerbarer Energiequellen zur Stromerzeugung angestrebt [1]. Dieses Ziel ist nur durch dezentrale rückgekoppelte Systeme und die Integration vieler datengenerierender und intelligenter Komponenten erreichbar ("Smart Grid" [2]). Ein solches System muss durch verschiedene Maßnahmen, wie die Integration von Strom-, Gas-, Wärme- und Dampfnetzen inkl. der zugehörigen Datenströme, den Betrieb von verteilten Speichern, die Anpassung des Verbrauches an die jeweilige Erzeugungs- und Netzsituation (Demand Response), ein geeignetes Marktdeign und geeignete regulatorische Vorgaben stabilisiert werden. Jeder dieser Schritte erfordert das Aggregieren von Sensorinformationen, Modellierungen und Visualisierungen zum besseren Problemverständnis sowie das automatische oder zumindest halbautomatische Erstellen von Prognose- und Simulationsmodellen inklusive deren Nutzung in Optimierungsverfahren. Viele dieser Themen setzen Lösungen voraus, die auf Maschinellern Lernen basieren. Hierzu laufen vielfältige Forschungsarbeiten, z.B. in Feldversuchen [3, 4] und im Energy Lab 2.0 des Karlsruher Instituts für Technologie [5].

Das Ziel dieses Beitrags besteht darin, die Methoden des Maschinellen Lernens näher zu erläutern

(Abschnitt 2) und typische Anwendungsszenarien für Smart Grids aufzuzeigen (Abschnitt 3).

2 Maschinelles Lernen

Prinzipiell basieren die meisten Methoden des Maschinellen Lernens auf etablierten Methoden der klassischen Statistik [6, 7], auf Verfahren der Computational Intelligence [8, 9] und vielen Weiterentwicklungen an der Schnittstelle beider Gebiete. Beispiele hierfür sind k -Nearest-Neighbor-Verfahren (Analyse der Ausgangsgröße ähnlicher Beispiele [10]) und Support-Vektor-Maschinen (Suche optimaler Trennflächen in höherdimensionalen Räumen mit integrierten nichtlinearen Transformationen [11]). Zur Computational Intelligence gehören u.a. Künstliche Neuronale Netze [12], die biologisch inspirierte Verschaltungen einfacher Verarbeitungseinheiten zur Nachbildung nichtlinearer Zusammenhänge nutzen, und Cluster-Verfahren zur automatischen Gruppierung ähnlicher Beispiele wie Fuzzy-C-Means [13, 14, 15]. Ein wesentlicher Treiber für die Methoden des Maschinellen Lernens sind massiv ansteigende Datenmengen, die bisherige Methoden an die Grenzen der Skalierbarkeit getrieben haben. Dieser Trend wird unter dem Begriff "Big Data" [16, 17] zusammengefasst und über die folgenden Eigenschaften ("4Vs") definiert [18, 19, 20]:

1. Volume (Größe bzw. Datenmenge, z.B. Zeitreihen mit sehr kleinen Abtastzeiten oder eine Vielzahl von verteilten Sensoren),
2. Velocity (erforderliche Geschwindigkeit beim Generieren, Einlesen, Auslesen oder Übertragen von Daten),
3. Variety (Vielfältigkeit oder Heterogenität, bezogen auf viele verschiedene Datentypen und zu vernetzende Systeme mit unterschiedlichen Eigenschaften, z.B. in Form von Zeitreihen und Logfiles) und
4. Veracity (Vertrauenswürdigkeit, bezogen auf den Anteil fehlerhafter, fehlender oder nur mit Unsicherheiten verfügbaren Daten wie z.B. Intervalldaten, z.B. durch temporäre Sensorausfälle oder Übertragungsprobleme).

Aussichtsreiche Lösungsstrategien umfassen

1. die Nutzung verteilter Umgebungen, inkl. verteilter Filesysteme wie Hadoop, Analysetools wie Apache Spark und Tools zum Cloud Computing, siehe z.B. [21, 22],
2. Ansätze zum automatisiertem Entwurf, wie z.B. Deep Learning [23] und Auto-ML [24],
3. Verfahren zur automatischen Vorverarbeitung sowie zur Beurteilung der Daten- und Modellqualität [25, 26, 9, 27, 28] und
4. rekursive Methoden zur Beherrschung von Datenströmen (also kontinuierlich anfallenden Daten, engl. data streams), die Modelle ständig nachführen und mit neu hinzukommenden Daten verbessern [29].

Eine Auswahl typischer Datenanalysemethoden zeigt Tabelle 1. Für eine erfolgreiche Lösung müssen Experten für das jeweilige Anwendungsgebiet und Experten für Datenanalyse eng zusammenarbeiten, um die jeweilige Fragestellung als ein für Datenanalysemethoden lösbares Problem zu formulieren und um die Ergebnisse zu interpretieren. Hierbei ist entscheidend, den kompletten Auswertebau durch geeignete Softwaretools [30] zu unterstützen und skriptbasiert zu automatisieren. Dazu existieren kommerzielle Produkte wie der IBM SPSS Modeler (<http://www.spss.com/software/modeling/modeler>) oder Open-Source-Lösungen wie KNIME (<http://www.knime.org>, [31]), SciXMiner

(<http://sourceforge.net/projects/scixminer>, [32]), sowie neuere Big-Data-Tools wie Apache Spark (<http://spark.apache.org>, [33]).

3 Maschinelles Lernen im Energiesystem

Mit gegebenen Daten können mit Methoden des Maschinellen Lernens oftmals ganze Modellbibliotheken für verschiedene Einsatzgebiete aufgebaut werden.

Seit Jahren etabliert ist die Prognose der Verbraucherlast [35], oftmals aggregiert auf der Ebene von Stadtwerken oder Verteilnetzen. Hier werden in erster Linie autoregressive Modelle mit externen Eingängen eingesetzt, die sowohl Wettereinflüsse als auch periodische und musterbasierte Komponenten zur Modellierung des Energiebedarfs (Wochen- und Jahresrhythmus, Feiertage, Sondertage mit abweichendem Verhalten usw.) berücksichtigen. Ein Beispiel für eine solche Anwendung ist die Prognose für das Fernwärmenetz der Stadt Karlsruhe. Hier gelang es mit Methoden des Maschinellen Lernens, Modelle für Tages- und Stundenprognosen zu entwickeln und in den praktischen Routinebetrieb zu bringen. Wesentliche Herausforderungen waren dabei der Erhalt der Interpretierbarkeit, der Umgang mit zeitvarianten Effekten durch Zubau im Fernwärmenetz und energetische Sanierungen. In den letzten Jahren spielt die Prognose der Einspeisung von Photovoltaik und Windkraftwerken eine zunehmende Rolle. Für beide Aufgabengebiete existieren etablierte Methoden und Tools (Übersicht siehe [36]).

Repräsentative Last- und Einspeisekurven (optional inkl. der Modellierung von Unsicherheiten) sind u.a. für den Aufbau von Simulations- und Optimierungsstudien nützlich, z.B. für die Optimierung des Speicherbetriebs mit modellprädiktiven Reglern [37] oder als Vorarbeit für den Einsatz von simulationsgestützten Schedulingmodellen [38].

Andere Aufgaben werden in Zukunft an Bedeutung gewinnen und sind derzeit aktuelle Forschungsthemen. Ein Beispiel hierfür ist die Analyse von Smart-Meter-Daten für Demand-Response-Szenarien, also für die Modellierung von Energiepreisen [39, 40] oder der Verbraucherreaktion auf diese variablen Energiepreise [41]. Mit Clustermethoden können zunächst typische Subgruppen des Verbraucherverhaltens ermittelt werden [42, 43] (Abb. 1). Darauf aufbauend kön-

Aufgaben	Ausgewählte geeignete Verfahren
Erkennung atypischer Werte (Ausreißer), Anomaliedetektion	Klassifikation (<i>k</i> -Nearest Neighbor, Support-Vektor-Maschinen), Rauschcluster
Fehlerdetektion	Klassifikation (Bayes-Klassifikatoren, Künstliche Neuronale Netze, Support-Vektor-Maschinen)
Prognose zukünftiger Werte einer Zeitreihe	Regression (Autoregressive Modelle mit oder ohne externe Eingänge, Künstliche Neuronale Netze, <i>k</i> -Nearest Neighbor, Support-Vektor-Regression)
Suche nach Zusammenhängen in Protokolldateien mit Events	Process Mining [34]
Suche nach Subgruppen und typischen Mustern	Cluster-Verfahren (Fuzzy-C-Means, Hierarchisches Clustern), Spezielle Visualisierungen
Verstehen wesentlicher Zusammenhänge	Korrelationsanalysen, Regression (Polynome, Künstliche Neuronale Netze), Spezielle Visualisierungen

Tabelle 1: Ausgewählte Analysemethoden, modifiziert nach [20]

nen das Verbraucherverhalten [44] und der Anteil des Preiseinflusses [45] modelliert werden. Die Zusammenschaltung aller Komponenten in einem Simulationsmodell [46] ermöglicht dann den Vergleich von Szenarien und bis zu einem gewissen Grad auch die gegenseitige Plausibilitätsprüfung der Teilmodelle.

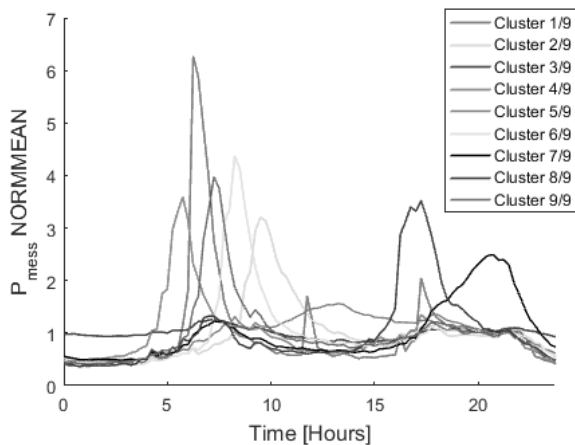


Abbildung 1: Analyse des Energieverbrauchs von Daten aus dem Olympic Peninsula Datensatz [47]: Clusteranalyse mit 9 Clustern der normierten Energieverbrauchszeitreihen für 21894 Tageskurven verschiedener Kunden zum Identifizieren typischer Verbrauchsmuster [48]

Eine andere Herausforderung sind Analysen auf zeitlich hochaufgelösten Daten. Beispielsweise erfordert die Analyse von phasengenauen Strom- und Spannungsmessungen mit einer Abtastfrequenz von 12.8 kHz über längere Zeiträume den Einsatz speziel-

ler Big-Data-Infrastrukturen und -Auswertemethoden [49]. Außerdem besteht die Chance, bisher nicht vollständig gelöste Spezialaufgaben anzugehen, z.B. die Erkennung von Flexibilitätspotenzialen in Batch-Prozessen [50] und Prognosen mit Unsicherheiten [51, 52, 53].

4 Zusammenfassung

Im Energiebereich haben in den letzten Jahren die gesammelten Datenmengen deutlich zugenommen und die Fortsetzung oder gar Verstärkung dieses Trends ist absehbar. Damit besteht die Chance, aus diesen Daten interessante Informationen zu extrahieren, um die zugrundeliegenden Prozesse besser zu verstehen, Simulationsmodelle aufzubauen sowie Planungs-, Optimierungs- und Überwachungsaufgaben im Energiesystem zu verbessern. Prinzipiell stehen dazu etablierte Analysemethoden aus dem Gebiet des Maschinellen Lernens bereit, die für sehr große Datenmengen ("Big Data") gezielt erweitert und angepasst werden müssen.

Acknowledgements

Diese Arbeit wurde von der Helmholtz-Gemeinschaft innerhalb der gemeinsamen Initiative "Energie System 2050 - Ein Beitrag des Forschungsbereichs Energie" unterstützt.

Literatur

- [1] Scholz, R.; Beckmann, M.; Pieper, C.; Muster, M.; Weber, R.: Considerations on Providing the Energy Needs using Exclusively Renewable Sources: Energiewende in Germany. *Renewable and Sustainable Energy Reviews* 35 (2014), S. 109–125.
- [2] Fang, X.; Misra, S.; Xue, G.; Yang, D.: Smart Grid - the New and Improved Power Grid: A Survey. *IEEE Communications Surveys Tutorials* 14(4) (2012), S. 944–980.
- [3] Hofmann, L.; Sonnenschein, M.: Smart Nord - Final Report. Techn. Ber., Lower Saxony Research Network "Smart Nord". 2015.
- [4] Witzsch, M.: Microgrid im Allgäu verändert Zukunft der Stromversorgung. *ew-Magazin für die Energiewirtschaft* (2015), S. 51–53.
- [5] Hagenmeyer, V.; Cakmak, H. K.; Döpmeier, C.; Faulwasser, T.; Isele, J.; Keller, H. B.; Kohlhepp, P.; Kühnapfel, U.; Stucky, U.; Waczowicz, S.; Mikut, R.: Information and Communication Technology in Energy Lab 2.0: Smart Energies System Simulation and Control Center with an Open-Street-Map-based Power Flow Simulation Example. *Energy Technology* 4 (2016), S. 145–162.
- [6] Backhaus, K.; Erichson, B.; Plinke, W.; Weiber, R.: *Multivariate Analysemethoden: Eine anwendungsorientierte Einführung*. Berlin u.a.: Springer. 2000.
- [7] Jain, A. K.; Duin, R. P. W.; Mao, J.: Statistical Pattern Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(1) (2000), S. 4–36.
- [8] Kroll, A.: *Computational Intelligence*. De Gruyter Oldenbourg. 2016.
- [9] Michie, D.; Spiegelhalter, D.; Taylor, C.: *Machine Learning, Neural and Statistical Classification*. Ellis Horwood. 1994.
- [10] Cover, T.; Hart, P.: Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory* 13(1) (1967), S. 21–27.
- [11] Burges, C.: A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining* 2(2) (1998), S. 121–167.
- [12] Haykin, S. S.: *Neural Networks and Learning Machines*. Pearson Education Upper Saddle River. 2009.
- [13] Bezdek, J. C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press. 1981.
- [14] Höppner, F.; Klawonn, F.; Kruse, R.: *Fuzzy Cluster Analysis*. New York: John Wiley. 1999.
- [15] Loose, T.: *Konzept für eine modellgestützte Diagnostik mittels Data Mining am Beispiel der Bewegungsanalyse*. Dissertation, Universität Karlsruhe, Universitätsverlag Karlsruhe. 2004.
- [16] Chen, M.; Mao, S.; Liu, Y.: Big Data: A Survey. *Mobile Networks and Applications* 19 (2014) 2, S. 171–209.
- [17] Wrobel, S.; Voss, H.; Köhler, J.; Beyer, U.; Auer, S.: Big Data, Big Opportunities, Anwendungssituation und Forschungsbedarf des Themas Big Data in Deutschland. *Informatik-Spektrum* 38 (2014) 5, S. 1–9.
- [18] Laney, D.: 3D Data Management: Controlling Data Volume, Velocity and Variety. Techn. Ber., META Group Research Note. 2001.
- [19] Ward, J. S.; Barker, A.: Undefined by Data: A Survey of Big Data Definitions. Techn. Ber., arXiv preprint arXiv:1309.5821. 2013.
- [20] Mikut, R.: Big Data: Perspektiven für Smart Grids und Smart Buildings. In: *Proc., 20. Dresdner Baustatik-Seminar*, S. 107–122. TU Dresden. 2016.
- [21] Döpmeier, C.; Stucky, K.-U.; Mikut, R.; Hagenmeyer, V.: A Concept for the Control, Monitoring and Visualization Center in Energy Lab 2.0. In: *Proc., of the 4th D-A-CH Energy Informatics Conference*, S. 83–94. Cham: Springer International Publishing. 2015.
- [22] Hashem, I. A. T.; Yaqoob, I.; Anuar, N. B.; Mokhtar, S.; Gani, A.; Khan, S. U.: The Rise of "Big Data" on Cloud Computing: Review and Open Research Issues. *Information Systems* 47 (2015), S. 98–115.
- [23] LeCun, Y.; Bengio, Y.; Hinton, G.: Deep Learning. *Nature* 521(7553) (2015), S. 436–444.

- [24] Kotthoff, L.; Thornton, C.; Hoos, H. H.; Hutter, F.; Leyton-Brown, K.: Auto-WEKA 2.0: Automatic Model Selection and Hyperparameter Optimization in WEKA. *Journal of Machine Learning Research* 17 (2016), S. 1–5.
- [25] Batini, C.; Cappiello, C.; Francalanci, C.; Maurino, A.: Methodologies for Data Quality Assessment and Improvement. *ACM Comput. Surv.* 41 (2009) 3, S. 16:1–16:52.
- [26] Madnick, S. E.; Wang, R. Y.; Lee, Y. W.; Zhu, H.: Overview and Framework for Data and Information Quality Research. *Journal of Data and Information Quality (JDIQ)* 1 (2009) 1, S. 2.
- [27] Pimentel, M. A.; Clifton, D. A.; Clifton, L.; Tarrasenko, L.: A Review of Novelty Detection. *Signal Processing* 99 (2014), S. 215–249.
- [28] Doneit, W.; Mikut, R.; Gröll, L.; Pychynski, T.; Reischl, M.: DaMoQ: Eine Open-Source-MATLAB-Toolbox zur Bewertung von Daten- und Modellqualität in Regressionen. *at-Automatisierungstechnik* 65 (2017) 3, S. 207–2018.
- [29] Aggarwal, C.: *Data Streams: Models and Algorithms*. Springer. 2007.
- [30] Mikut, R.; Reischl, M.: Data Mining Tools. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 1 (2011) 5, S. 431–443.
- [31] Berthold, M. R.; Cebron, N.; Dill, F.; Gabriel, T. R.; Kötter, T.; Meinl, T.; Ohl, P.; Sieb, C.; Thiel, K.; Wiswedel, B.: KNIME: The Konstanz Information Miner. In: *Data Analysis, Machine Learning and Applications*, S. 319–326. Springer. 2008.
- [32] Mikut, R.; Bartschat, A.; Doneit, W.; González Ordiano, J. Á.; Schott, B.; Stegmaier, J.; Waczowicz, S.; Reischl, M.: The MATLAB Toolbox SciXMiner: User’s Manual and Programmer’s Guide. Techn. Ber., arXiv:1704.03298. 2017.
- [33] Meng, X.; Bradley, J.; Yuvaz, B.; Sparks, E.; Venkataraman, S.; Liu, D.; Freeman, J.; Tsai, D.; Amde, M.; Owen, S.; et al.: MLLib: Machine Learning in Apache Spark. *Journal of Machine Learning Research* 17 (2016) 34, S. 1–7.
- [34] van der Aalst, W. M.; Reijers, H. A.; Weijters, A. J.; van Dongen, B. F.; De Medeiros, A. A.; Song, M.; Verbeek, H.: Business Process Mining: An Industrial Application. *Information Systems* 32 (2007) 5, S. 713–732.
- [35] Hernandez, L.; Baladron, C.; Aguiar, J. M.; Carro, B.; Sanchez-Esguevillas, A. J.; Lloret, J.; Massana, J.: A Survey on Electric Power Demand Forecasting: Future Trends in Smart grids, Microgrids and Smart Buildings. *IEEE Communications Surveys & Tutorials* 16 (2014) 3, S. 1460–1495.
- [36] González Ordiano, J.; Waczowicz, S.; Hagenmeyer, V.; Mikut, R.: Energy Forecasting Tools and Services. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (2017). In press.
- [37] Appino, R. R.; González Ordiano, J. Á.; Mikut, R.; Faulwasser, T.; Hagenmeyer, V.: On the Use of Probabilistic Forecasts in Scheduling of Renewable Energy Sources Coupled to Storages. *Applied Energy* 210 (2018), S. 1207 – 1218.
- [38] Jakob, W.; González Ordiano, J.; Ludwig, N.; Mikut, R.; Hagenmeyer, V.: Towards Coding Strategies for Forecasting-Based Scheduling in Smart Grids and the Energy Lab 2.0. In: *Proc., Genetic and Evolutionary Computation Conference Companion*, S. 1271–1278. ACM. 2017.
- [39] Keles, D.; Scelle, J.; Paraschiv, F.; Fichtner, W.: Extended Forecast Methods for Day-Ahead Electricity Spot Prices Applying Artificial Neural Networks. *Applied Energy* 162 (2016), S. 218–230.
- [40] Weron, R.: Electricity Price Forecasting: A Review of the State-of-the-Art with a Look into the Future. *International Journal of Forecasting* 30(4) (2014), S. 1030–1081.
- [41] Corradi, O.; Ochsenfeld, H.; Madsen, H.; Pinson, P.: Controlling Electricity Consumption by Forecasting Its Response to Varying Prices. *IEEE Transactions on Power Systems* 28 (2013) 1, S. 421–429.
- [42] Chicco, G.: Overview and Performance Assessment of the Clustering Methods for Electrical Load Pattern Grouping. *Energy* 42 (2012) 1, S. 68–80.

- [43] Waczowicz, S.; Reischl, M.; Hagenmeyer, V.; Mikut, R.; Klaiber, S.; Bretschneider, P.; Konotop, I.; Westermann, D.: Demand Response Clustering - How do Dynamic Prices Affect Household Electricity Consumption? In: *Proc., IEEE PowerTech, Eindhoven*, S. 1–6. IEEE. 2015.
- [44] Waczowicz, S.; Reischl, M.; Klaiber, S.; Bretschneider, P.; Konotop, I.; Westermann, D.; Hagenmeyer, V.; Mikut, R.: Virtual Storages as Theoretically Motivated Demand Response Models for Enhanced Smart Grid Operations. *Energy Technology* 4 (2016), S. 163–176.
- [45] Klaiber, S.; Waczowicz, S.; Konotop, I.; Westermann, D.; Mikut, R.; Bretschneider, P.: Prognose für preisbeeinflusstes Verbrauchsverhalten. *at-Automatisierungstechnik* 65 (2017) 3, S. 179–188.
- [46] Konotop, I.; Waczowicz, S.; Klaiber, S.; Bretschneider, P.; Mikut, R.; Westermann, D.: Investigation and Mathematical Modelling of the Impact of Incentive Signals to Consumers on their Consumption, Load Forecast and Network Operation. *IET Generation, Transmission & Distribution* 10 (2016) 9, S. 2138–2146.
- [47] Hammerstrom, D. J.; Ambrosio, R.; Brous, J.; Carlon, T. A.; Chassin, D. P.; DeSteese, J. G.; Guttromson, R. T.; Horst, G. R.; Järvegren, O. M.; Kajfasz, R.; et al.: Pacific Northwest GridWise Testbed Demonstration Projects, Part I. Olympic Peninsula Project. Techn. Ber., Pacific Northwest National Laboratory (PNNL), Richland, Washington. 2007.
- [48] Waczowicz, S.; Klaiber, S.; Bretschneider, P.; Konotop, I.; Westermann, D.; Reischl, M.; Mikut, R.: Data Mining zur Analyse der Auswirkungen von Preissignalen auf Haushaltsstromkunden. *at-Automatisierungstechnik* 62 (2014) 10, S. 740–752.
- [49] Maaß, H.; Cakmak, H. K.; Bach, F.; Mikut, R.; Harrabi, A.; Süß, W.; Jakob, W.; Stucky, K.-U.; Kühnapfel, U. G.; Hagenmeyer, V.: Data Processing of High Rate Low Voltage Distribution Grid Recordings for Smart Grid Monitoring and Analysis. *EURASIP Journal on Advances in Signal Processing* 1 (2015), S. 1–21.
- [50] Ludwig, N.; Waczowicz, S.; Mikut, R.; Hagenmeyer, V.: Mining Flexibility Patterns in Energy Time Series from Industrial Processes. In: *Proc., 27. Workshop Computational Intelligence, Dortmund*, S. 13–32. 2017.
- [51] Hong, T.; Pinson, P.; Fan, S.; Zareipour, H.; Troccoli, A.; Hyndman, R. J.: Probabilistic Energy Forecasting: Global Energy Forecasting Competition 2014 and Beyond. *International Journal of Forecasting* 32(3) (2016), S. 896 – 913.
- [52] González Ordiano, J. Á.; Doneit, W.; Waczowicz, S.; Gröll, L.; Mikut, R.; Hagenmeyer, V.: Nearest-Neighbor Based Non-Parametric Probabilistic Forecasting with Applications in Photovoltaic Systems. In: *Proc., 26. Workshop Computational Intelligence, Dortmund*, S. 9–30. 2016.
- [53] Join, C.; Fliess, M.; Voyant, C.; Chaxel, F.: Solar Energy Production: Short-Term Forecasting and Risk Management. *IFAC-PapersOnLine* 49 (2016) 12, S. 686–691.

Smooth virtual commissioning mit industrialPhysics

Florian Wegmann¹, Isabelle Kautz²

¹machineering GmbH & Co. KG

Isabelle.kautz@machineering.de

Die 3D-Simulationssoftware industrialPhysics von machineering macht eine schnelle und einfache virtuelle Inbetriebnahme möglich. CAD-Anbindungen und SPS-Schnittstellen ermöglichen ein synchronisiertes mechatronisches Engineering als Basis für Industrie 4.0. Die mit virtueller Inbetriebnahme erreichte Zeitersparnis und Kostenersparungen sind enorm. Freuen Sie sich auf einen interessanten Vortrag!

Die machineering GmbH & Co. KG mit Sitz in München entwickelt die 3D-Simulationssoftware „industrialPhysics“ für die virtuelle Inbetriebnahme mechatronischer Anlagen. Die 3D-Simulation komplexer Materialflüsse und mechatronischer Produktionsanlagen zählt zu den größten Effizienzreserven im Maschinen- und Anlagenbau. So lassen sich über die 3D-Simulation und das virtuelle Prototyping von Anlagen deren Entwicklung und Konstruktion beschleunigen und absichern. In der Projektakquise können geplante Anlagen im Dialog mit dem Kunden vertriebswirksam visualisiert werden. Das größte Potenzial aber liegt in der virtuellen Inbetriebnahme von Anlagen und Produktionsstraßen: Nachweislich lassen sich die Kosten für Installation, Test und Optimierung von Anlagen über die Simulation um bis zu 75% reduzieren.

In dem Vortrag „Smooth virtual commissioning mit industrialPhysics“ werden die bestehenden Schnittstellen zu diversen CAD-Systemen, zu den Steuerungsherstellern und der Robotik beleuchtet. Anhand zahlreicher Praxisbeispiele wird zudem veranschaulicht, welche Herausforderungen mit industrialPhysics gelöst werden und in welchen Branchen die Simulationssoftware zum Einsatz kommen kann.

Um noch tiefer in eine Anlage eintauchen zu können, hat machineering verschiedene VR + AR-Brillen wie die Microsoft HoloLens an die Simulation angebunden. Wenn die geplante, virtuell laufende Anlage als Hologramm in die Fabrikhalle projiziert wird, eröffnen sich für die Kunden neue Dimensionen. Mit dieser Technologie geht machineering einen großen Schritt weiter in Richtung Umsetzung von Industrie 4.0 für Jedermann. Im Rahmen der Veranstaltung können Interessierte diese virtuellen Welten mit industrialPhysics erleben.

1 References

- [1] Forschungsbericht: Methoden für die virtuelle Inbetriebnahme automatisierter Produktionssysteme. Dr. Georg Wunsch, Diss. München, Techn. Univ.; Herbert UTZ Verlag Wissenschaft. 2007

Data Science für Ingenieure

Dr.-Ing. Hans Martin Ritt, Marcus Maurer, Dr. Alexander Diethert

MathWorks GmbH

Martin.Ritt@mathworks.de

MathWorks hat die Vision, dass auch datenorientierte Methoden für Ingenieure leicht verwendbar sein und bleiben müssen. Während dies bei klassischen Methoden wie der Systemidentifikation seit längerem realisiert ist, brauchen die Möglichkeiten von Datenanalytik, Machine Learning und Deep Learning für Anwendungen wie Predictive Maintenance oder Bilderkennung unsere Aufmerksamkeit. Außerdem müssen wir Wege aufzeigen, wie diese Algorithmen weitgehend automatisiert auf verschiedenen Plattformen wie Grafikkarten, Mikroprozessoren oder Big Data Architekturen performant ausgeführt werden können.

Die Fähigkeit, automatisiert auf Basis von gemessenen Daten, neue Zusammenhänge zu erkennen sowie Vorhersagen und Entscheidungen zu treffen, hat unsere Technologie grundlegend verändert.

Es ist also kein Wunder, dass Data Science für die Ingenieurwissenschaften immer wichtiger geworden ist. So entwickeln Ingenieure komplexe Systeme wie smarte Sensoren, vernetzte Maschinen oder autonome Fahrzeuge und prozessieren dabei nicht selten immense Datenmengen, die die Grundlage der Intelligenz dieser Systeme bilden.

den schnellen Zugriff auf dieses immer größer werdende Datenvolumen. Neue Algorithmen kontrollieren das Zusammenspiel der Bauteile in Cloud-basierten Architekturen oder direkt auf den jeweiligen Mikroprozessoren.

Data Scientists helfen bei dem neuen, datenorientierten Teil des Aufgabenspektrums. Jedoch fehlt den Data Scientists häufig eine wichtige Kompetenz: das Know-how des Ingenieurs. Das Potential der vorhandenen Daten wird oft nicht ausgeschöpft, weil eine rein mathematische Betrachtung ohne das entsprechende Fachwissen meist nicht zielführend ist. Bei vielen Entwicklungsteams ist daher die Verteilung der Aufgaben noch nicht optimal. Wie können die Bereiche Entwicklung und Data Science also näher zusammengebracht werden?

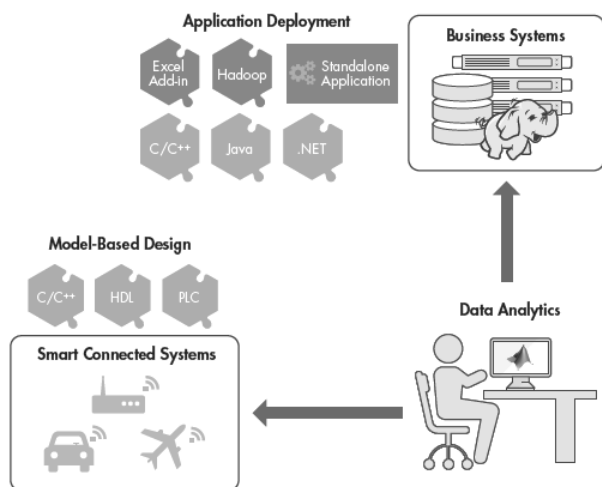


Abbildung 1. Datenanalyse integriert in Business Systemen sowie smarten, vernetzten Systemen

Der Ingenieur wird selbst zum Data Scientist – zum Teil zumindest. Aufgaben wie statistische Datenanalysen, Parameteroptimierung oder Systemidentifikation sind für Ingenieure schnell zu erlernen. Der Weg in Richtung Mustererkennung und die Entwicklung von Vorhersagemodellen ist dann nicht mehr weit. Data Science birgt ein hohes Potential für viele Entwicklungsprojekte. Oft ist ein tieferes Einsteigen in IT Themen gar nicht notwendig.

Verteilte Datenbanksysteme ermöglichen in diesem Zusammenhang die Speicherung und

Baden-Württemberg zu Gast in der Welt – die Welt zu Gast bei uns

Werner Schempp, Ministerialdirigent a.D.

Ehemaliger Abteilungsleiter der Abteilung: „Internationale Angelegenheiten“

im Staatsministerium Baden-Württemberg (bis August 2017)

schemppus@web.de

„Reisen bildet die Jugend und beult Koffer aus!“ - so salopp ist gelegentlich über das Reisen im Volksmund zu hören. Dass Reisen auch im politischen Alltag im Bund und in den Ländern von großer Bedeutung ist, soll an Beispielen von sog. Delegationsreisen baden-württembergischer Ministerpräsidenten mit Blick auf Wirtschaft, Wissenschaft, Hochschulen und auf die regierungsnahen Institutionen (z.B. bw-i Baden-Württemberg international, Stiftung Entwicklungszusammenarbeit SEZ und e-mobil Baden-Württemberg) erläutert werden. Außerdem vermitteln kleine Berichte, amüsante Begebenheiten und Anekdoten am Rande von Delegationsreisen ins Ausland und von Staatsbesuchen in Baden-Württemberg aus eigenem Erleben des Referenten, dass letzten Endes immer und überall Menschen und Menschliches im Zentrum des Geschehens stehen.

1 Kurz zum Inhalt

Sowohl die jeweiligen Bundesregierungen als auch die Landesregierungen der 16 deutschen Länder pflegen in regelmäßigen Abständen Reisen mit internationalen Zielen zu unternehmen. Dahinter stecken zunächst diplomatische Usancen, dann aber im konkreten Fall lange vor Reisebeginn ausgeklügelte Planungen und Zielvorstellungen. Je nach thematischer Ausrichtung, Destination und Reisedauer werden neben einer politischen und administrativen (Arbeits-ebene) Entourage auch jeweils größere und kleiner Fachdelegationen zur Mitreise eingeladen. Einerseits bieten solche Delegationsreisen im Zielland die Möglichkeiten und Leistungen der (baden-württembergischen) Wirtschaft/Wissenschaft/ Hochschulen etc. international vorzustellen. Andererseits kann ein hochrangiger Politiker, ob vom Bund oder aus einem Land, vor allem für Vertreter kleinerer Firmen/ Institutionen als Türöffner für neue Kontakte und Kooperationsmöglichkeiten fungieren.

Viele Informationen zur diplomatischen Reise-tätigkeit und zur protokollarischen Gastgeber-schaft in Bund und in den Ländern

ergänzen den Vortrag. Darüber hinaus lockern eigene Erlebnisse des Referenten aus 32 Dienst-jahren im Staatsministerium Baden-Württemberg und in der Vertretung des Landes Baden-Württemberg beim Bund in Berlin das Thema auf.

Deep Reinforcement Learning zur Bewegungsplanung und -regelung mobiler Roboter

Jan-Niklas Nieland¹, Marco Beesk¹, Frank Tränkle²

¹Neobotix GmbH, Weipertstr. 8-10, D-74076 Heilbronn

²Automotive Systems Engineering, Hochschule Heilbronn, D-74081 Heilbronn
nieland@neobotix.de

Die voranschreitende Digitalisierung der Industrie lässt die Anzahl an mobilen Robotern in diesem Umfeld stark ansteigen. Klassische Algorithmen zur lokalen Bahnplanung und -regelung der mobilen Roboter unter Berücksichtigung von statischen und dynamischen Hindernissen besitzen eine hohe Komplexität und bieten nicht die nötige Agilität. Neue Algorithmen des Deep Reinforcement Learning wie *Deep Deterministic Policy Gradient* (DDPG) [1] bieten das Potential, die lokale Bewegungsplanung und -regelung von mobilen Robotern zu verbessern. Der Einsatz von Deep Reinforcement Learning im Bereich der mobilen Robotik bedarf der Entwicklung eines neuen Konzeptes zur Einbettung in die bestehenden Strukturen. Das im vorliegenden Beitrag vorgestellte Konzept *Instinctive Motion Control* (IMC) vereint die Ebenen der Bewegungsplanung und -regelung aus [2] als Ansatz des *End-To-End Learnings*. Der Beitrag beschreibt die Rahmenbedingungen, Abläufe und Auswahlkriterien, die zur Auswahl von DDPG geführt haben. Für die Verwendung von DDPG wird eine Testumgebung entwickelt und implementiert, in der verschiedene Strategien der künstlichen neuronalen Netze (KNN) getestet und bewertet werden können. Abschließend werden verschiedene Konstellationen von Hyperparametern mit simulierten und realen mobilen Robotern getestet und ausgewertet. Zum Einsatz kommt der mobiler Roboter MP-400 der Firma Neobotix.

1 Einleitung

Die voranschreitende Digitalisierung der Industrie lässt die Anzahl an mobilen Robotern in diesem Umfeld stark ansteigen. Die für einen mobilen Roboter zur Verfügung stehenden Strecken werden von anderen mobilen Robotern und Menschen mit verwendet. Daraus resultieren verschiedene Problemstellungen wie die Planung von Ausweichmanövern. Als Ausweichmanöver wird in dieser Arbeit die Neuplanung von Bahnkurven und die Bewegungsregelung zum Umfahren bei statischen und dynamischen Hindernissen bezeichnet, welche bei der globalen Bahnplanung nicht berücksichtigt wurden. Ein Ausweichmanöver spielt sich somit immer im lokalen Bereich des mobilen Roboters ab, wobei die Größe dieses Bereichs variabel definiert werden kann. Durch eine stetig steigende Anzahl an Verkehrsteilnehmern werden Ausweichmanöver immer wichtiger, um Blockierungen und Verzögerungen zu vermeiden. Gründe für die Probleme bei der Planung von Ausweichmanövern sind vor allem die meist unbekannte Dynamik der mo-

bilien Roboter und das unbekannt Verhalten von dynamischen Hindernissen aller Art.

Eine weitere Problemstellung ist die Ausführung der Ausweichmanöver bzw. die Regelung der mobilen Roboter auf eine zuvor geplante Bahn. Das zuvor geplante Ausweichmanöver bzw. die Bahn soll vom mobilen Roboter befahren werden. Durch Ungenauigkeiten bei der Regelung entstehen Abweichungen von der geplanten Bahn. Vor allem in Engstellen und Kreuzungsbereichen kommt es durch diese Abweichungen zu Blockierungen. Die klassischen Algorithmen zur lokalen Bahnplanung und -regelung der mobilen Roboter unter Berücksichtigung von statischen und dynamischen Hindernissen besitzen eine hohe Komplexität und bieten nicht die nötige Agilität.

Neue Algorithmen des Deep Reinforcement Learning (DRL) wie *Deep Deterministic Policy Gradient* (DDPG) [1] bieten das Potential, die lokale Bewegungsplanung und -regelung von mobilen Robotern zu verbessern. Der Einsatz von Deep Reinforcement Learning im Bereich der mobilen Robotik bedarf der Entwicklung eines neuen Konzeptes zur Einbettung in die

bestehenden Strukturen. Weiterhin wird die Entwicklung und Evaluation dieses Konzepts beschrieben.

2 Instinctive Motion Control

Für das zu entwickelnde Konzept zur Einbettung von DRL in die Strukturen der lokalen Bewegungsplanung und -regelung mobiler Roboter müssen zunächst einige Rahmenbedingungen festgelegt werden, welche die Komplexität der Aufgabenstellung auf ein handhabbares Niveau reduzieren, ohne es dabei auf einzelne Anwendungsszenarien zu beschränken. Eine allgemeine Beschreibung der Strukturen ist in [2] zu finden. Als erste Rahmenbedingung wird eine globale Bahnplanung vorausgesetzt. Diese globale Bahn vom Startpunkt zum Zielpunkt wird als eine Straße mit definierter Breite angenommen. Die Breite der Straße muss so ausgelegt sein, dass ausreichend Platz für den mobilen Roboter bleibt, innerhalb dieser Straße Hindernissen auszuweichen. Ist nicht genug Platz zum Ausweichen, hält der mobile Roboter an und signalisiert eine Blockierung. Zusätzlich wird der Bereich für die lokale Bewegungsplanung und -regelung auf einen von der Größe des mobilen Roboters abhängigen Umkreis beschränkt.

Eine Neuerung, die das Konzept *Instinctive Motion Control* (IMC) bietet, ist die Vereinigung der Ebenen der Bewegungsplanung und -regelung aus [2] als Ansatz des *End-To-End Learnings*. Diese Vereinigung gelingt durch den Einsatz von künstlichen neuronalen Netzen (KNN) und Algorithmen aus dem Bereich des Deep Reinforcement Learnings. Ein KNN lernt dabei den Zusammenhang zwischen einem Zustand und einer Aktion anhand von Belohnungen. Der Zustand stellt den Eingangsvektor des KNN dar und kann in Abhängigkeit der vorhandenen Sensoren und der Darstellung des Zielpunkts definiert werden. Die Aktion ist der Ausgangsvektor des KNN und enthält die Steuersignale für die vorhandenen Motoren. Die Anzahl der Motoren ist nicht begrenzt, was den Einsatz beliebiger Kinematiken ermöglicht. Das Training des KNN wird mit dem Algorithmus *Deep Deterministic Policy Gradient* (DDPG) durchgeführt. DDPG bietet durch seine allgemeinen Auslegung alle wichtigen Aspekte, die für die Umsetzung von IMC wichtig sind:

- Kontinuierliche Werte im Zustandsraum
- Kontinuierliche Werte im Aktionsraum
- Actor-Critic Aufbau

- Off-Policy
- Experience Replay mit ReplayBuffer

Erläuterungen zu diesen und weiteren Aspekten sind in [3] und [1] zu finden.

Innerhalb der in Abschnitt 4 beschriebenen Versuchsserien wird die Zielsetzung der Anfahrt eines Zielpunkts mit Betrachtung der lokalen Umgebung zunächst auf die Zielsetzung der Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung reduziert. Diese Reduzierung soll die Analyse und die Optimierung der noch zu definierten Parameter vereinfachen. Sobald die Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung gelingt, werden Parameter für die eigentliche Zielsetzung unter Berücksichtigung der bis dahin gewonnenen Erkenntnisse definiert.

Ein für die Evaluation von IMC wichtiger Gesichtspunkt ist der Einsatz auf einem realen mobilen Roboter. Dazu wird ein mobiler Roboter der Firma Neobotix GmbH mit dem Namen MP-400 verwendet. Dieser besitzt eine differentielle Kinematik, zwei Laserscanner und Radencoders.

Bei der Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung werden als Eingangsdaten für das Actor-KNN die Ist-Drehzahl des linken und rechten Rades, die X- und Y-Koordinaten des Ziels im Basis-Koordinatensystem des mobilen Roboters sowie die Winkeldifferenz der aktuellen Ausrichtung des mobilen Roboters und des Ziels verwendet. Die Ausgabe des Actor-KNN sind die Soll-Drehzahlen des linken und rechten Rades. Eine schematische Darstellung dieses Aufbaus ist in Abbildung 1 zu finden.

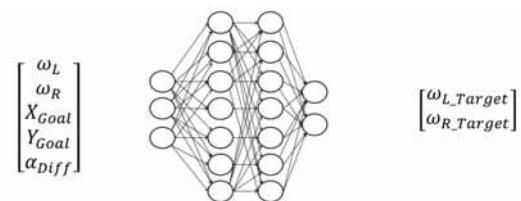


Abbildung 1: Schematische Darstellung des Actor-KNN für die Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung

Bei der Anfahrt eines Zielpunkts mit Betrachtung der lokalen Umgebung werden zusätzlich zu den bereits beschriebenen Daten 48 Abstandswerte der Laserscanner (D1 - D48) verwendet, um die Umgebung wahrzunehmen. Eine schematische Darstellung dieses Aufbaus ist in Abbildung 2 zu finden.

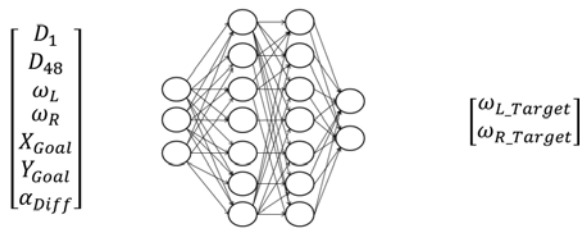


Abbildung 2: Schematische Darstellung des Actor-KNN für die Anfahrt eines Zielpunkts mit Betrachtung der lokalen Umgebung

3 Trainingsumgebung

Für die Optimierung bzw. das Training der in IMC eingesetzten KNNs wird eine eigene Softwarestruktur entwickelt. Die Implementierung wird mit den Software-Bibliotheken TensorFlow, Keras, SQLite und ROS in der Programmiersprache Python durchgeführt. ROS steht für Robot Operating System [4] und ist ein weltweit verwendetes Open Source Framework zur Entwicklung und Implementierung von Steuerungssoftware für Roboter. Diese Softwarestruktur, auch Trainingsumgebung genannt, besteht aus den Komponenten Optimierungsroutine, Episodensteuerungsprogramm und Steuerungssoftware des simulierten und realen mobilen Roboters. Der Aufbau der Trainingsumgebung ist in Abbildung 3 und 4 dargestellt.

Die Optimierungsroutine implementiert den DRL-Algorithmus DDPG und verwaltet den ReplayBuffer. Die Eingliederung des ReplayBuffer in die Optimierungsroutine soll eine effiziente Anbindung der Trainingsdaten an den Optimierungsalgorithmus gewährleisten. Das Laden der Daten aus dem ReplayBuffer ist einer der Flaschenhälse bei der Optimierung. Die Kommunikation mit den anderen Komponenten wird über ROS-Topics und -Services aufgebaut.

Die Hauptfunktion des Episodensteuerungsprogramms ist es, den Ablauf eines Trainingsdurchlaufs zu steuern, welcher die Daten zur Optimierung der KNNs liefert. Dieser Vorgang wird als Episode bezeichnet. Die gewonnenen Daten werden als sogenannte Transitionen im ReplayBuffer gespeichert. Auch das Episodensteuerungsprogramm ist durch die ROS-API mit den anderen Softwarekomponenten verbunden. Ein wichtiger Aspekt für die Nutzung von IMC ist die Möglichkeit der Verwendung von simulierten und realen mobilen Robotern mit derselben Softwarestruktur. Der Einsatz von ROS ermöglicht die Verwendung

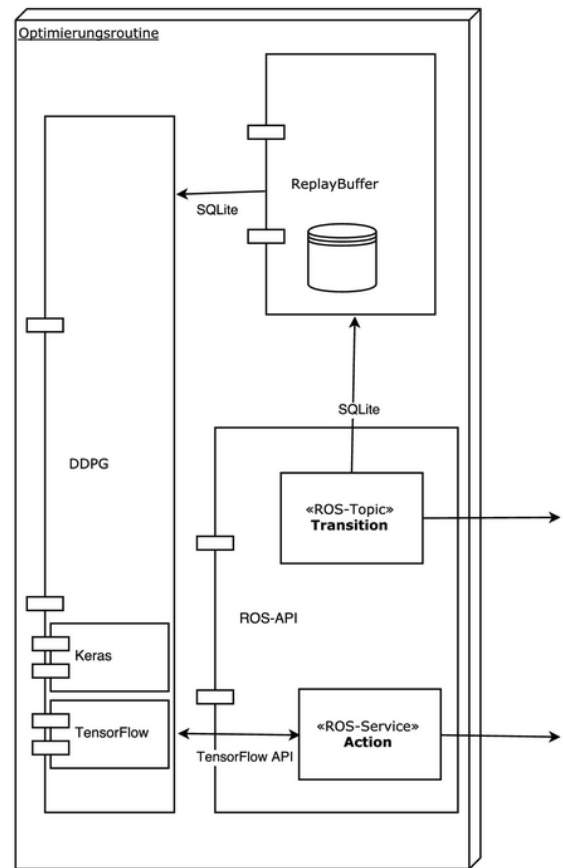


Abbildung 3: Softwarestruktur der Optimierungsroutine

von Robotern und Simulationsumgebungen, die eine ROS-Schnittstelle besitzen.

Für die Simulation der mobilen Roboter wird die Simulationssoftware Gazebo eingesetzt. Gazebo besitzt eine Schnittstelle zu ROS und ist Open Source. Damit ein möglichst direkter Vergleich zwischen Simulation und realem mobilen Roboter durchgeführt werden kann, wird ein MP-400 simuliert.

Die Softwarekomponente mit dem Namen MP-400 stellt die Steuerungseinheit des gleichnamigen mobilen Roboters dar.

4 Versuchsreihen

In einzelnen Versuchen werden verschiedene Konstellationen von Hyperparametern zum Erreichen der Zielsetzung getestet. Hyperparameter sind Parameter, die nicht von DDPG optimiert werden. Als noch zu definierende Hyperparameter stehen u.a. die Topolo-

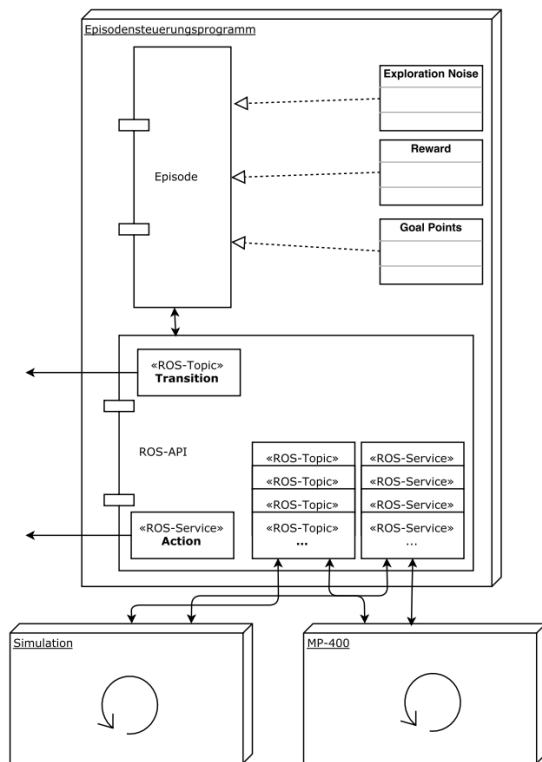


Abbildung 4: Softwarestruktur der Episodensteuerung mit Simulation und Robotersteuerung

gie der KNNs, das Erkundungsrauschen und die Belohnung aus.

Als Topologie des KNN wird die Auslegung der folgenden Parameter bezeichnet:

- Anzahl der Schichten
- Anzahl der Neuronen für jede Schicht
- Anordnung der Schichten
- Aktivierungsfunktion der Neuronen für jede Schicht

Die optimale Topologie der künstlichen neuronalen Netze kann zum aktuellen Zeitpunkt nicht theoretisch bestimmt werden. Aus diesem Grund werden verschiedene Topologien für die zwei verschiedenen Zielsetzungen entwickelt und in einzelnen Versuchen getestet. Beispielhaft für die verschiedenen getesteten Topologien wird in Abbildung 5 der Aufbau eines Actor-KNN für die Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung dargestellt.

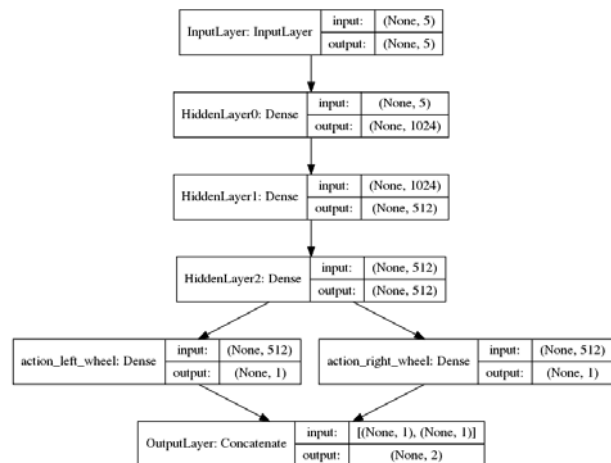


Abbildung 5: Actor-KNN zur Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung

Das Actor-KNN besteht aus der Eingangsschicht, drei verdeckten Schichten (Hidden Layer) und der Ausgangsschicht. Die Neuronen der einzelnen Schichten sind voll verbunden. Die Eingangsschicht und die verdeckten Schichten verwenden eine ReLU-Aktivierungsfunktion. Die Ausgangsschicht verwendet eine tanh-Aktivierungsfunktion.

Das Erkundungsrauschen gibt an, wie vom KNN generierte Aktionen verfälscht werden sollen, um möglichst viele verschiedene Kombinationen von Aktionen zu Erkunden. Für eine Definition des Erkundungsrauschens wird Expertenwissen zu der jeweiligen Zielsetzung benötigt.

Die Belohnung besteht aus einer direkten Belohnung im Wertebereich, welche nach jeder Aktion gewährt wird und einer Abschlussbelohnung, sobald das Ziel erreicht ist. Die direkte Belohnung hängt von der Länge der zurückgelegten Strecke ab und ist positiv, solange sich der mobile Roboter dem Ziel nähert. Entfernt sich der mobile Roboter weiter vom Ziel, wird die direkte Belohnung negativ. Ein wichtiger Aspekt beim Training ist, dass der mobile Roboter schnellstmöglich das Ziel erreicht. Es soll somit mit der geringst möglichen Anzahl an Aktionen erreicht werden. Aus diesem Grund wird für jede Aktion ein fester Wert bei der Belohnung abgezogen. Im folgenden Abschnitt werden die Ergebnisse zu einzelnen Versuchen beschrieben.

5 Ergebnisse

Insgesamt wurden mehr als 30 Versuche durchgeführt und ausgewertet. Die Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung konnte zufriedenstellend absolviert werden. Der Erfolg des Trainings wurde u.a. durch die Verläufe der sogenannten *target_Q_value* und der benötigten Anzahl an Aktionen zum Erreichen des Ziels der Episode überprüft.

Der Verlauf der *target_Q_value* in Abbildung 6 zeigt, nach einer Einschwingphase, einen nahezu stetigen Anstieg über die Iterationen der Optimierung, was dem erwarteten Verlauf eines erfolgreichen Trainings entspricht.

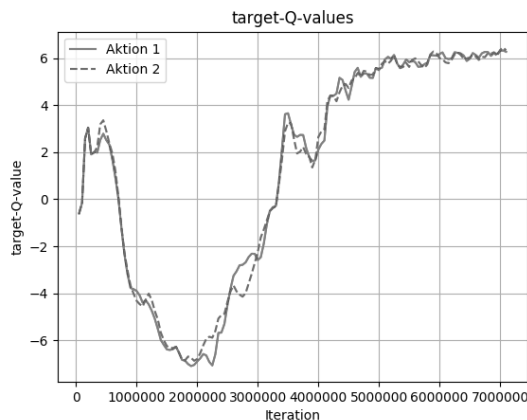


Abbildung 6: Verlauf der *target_Q_value*

Die Anzahl an Aktionen zum Erreichen des Ziels wird in Abbildung 7 dargestellt und zeigt nach einer Anfangsphase, in der die Episoden nach 1000 Aktionen abgebrochen wurden, einen absinkenden Verlauf über die durchgeführten Episoden, bis ein Minimum erreicht wird.

Aus den Versuchen zur Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung sind vor allem Erkenntnisse zur Auslegung des Erkundungsrauschens und der Belohnung gewonnen worden. Es ist ersichtlich, dass die Definition des Erkundungsrauschens eine sehr große Auswirkung auf den Erfolg und die Dauer des Versuchs hat. Bei der Belohnung zeigt sich, dass eine sehr ungenaue Auslegung ausreichend ist, um verschiedene Zielsetzungen zu erreichen. Lediglich gewünschtes und unerwünschtes Verhalten muss differenziert belohnt werden.

Die Anfahrt eines Zielpunkts mit Betrachtung der lokalen Umgebung konnte am Ende der Versuche nur

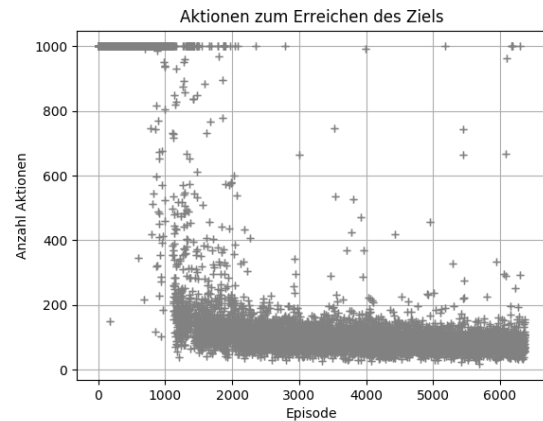


Abbildung 7: Verlauf der benötigten Anzahl an Aktionen zum Erreichen des Zielpunkts

bedingt absolviert werden. Als Grund für das nicht zufriedenstellende Verhalten scheint die Unfähigkeit des KNNs, die Kombination aus der notwendigen Generalisierung bei der Interpretation der Umgebung und der Genauigkeit für das Erreichen eines Zielpunkts zu erlernen.

6 Zusammenfassung

Es konnte durch die Verwendung des DRL-Algorithmus DDPG und der Definition verschiedener Rahmenbedingungen, welche die Komplexität der Aufgabenstellung reduzieren, ein Gesamtkonzept für die Vereinigung von Bewegungsplanung und -regelung entwickelt werden. Durch die Unwissenheit des steuernden KNN über das endgültige Ziel und das somit instinktive Reagieren auf einen bestimmten Zustand wurde der Ansatz *Instinctive Motion Control* (IMC) benannt. Durch die anpassbare interne Struktur des bzw. der KNNs sowie die gewährten Belohnungen zur Bildung des gewünschten Verhaltens, kann das Konzept für beliebige Kinematiken, Dynamiken und Wahrnehmungen verwendet werden. Zur Evaluation von IMC wurden zwei verschiedene Eingangsvektoren und der Ausgangsvektor für einen mobilen Roboter mit differentieller Kinematik definiert. Weiterhin wurde eine komplette Softwarestruktur zum Training entwickelt und implementiert. Bei Versuchen mit verschiedenen Konstellationen von Hyperparametern konnten Erkenntnisse in den Bereichen der Definition des Erkundungsrauschens und der Belohnung gewonnen werden. Die Anfahrt eines Zielpunkts ohne Betrachtung der lokalen Umgebung

konnte vom KNN zufriedenstellend erlernt werden. Die Anfahrt eines Zielpunkts mit Betrachtung der lokalen Umgebung bedarf weiterer Versuche und einem Überdenken der bisher verwendeten Strukturen der KNNs im Bezug auf die Fähigkeit der Generalisierung. Die durchgeführten Versuche haben das zuvor angenommene Potential von DRL-Algorithmen für den Bereich der mobilen Robotik bestätigt.

Literatur

- [1] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.
- [2] Hannah Bast, Daniel Delling, Andrew Goldberg, Matthias Müller-Hannemann, Thomas Pajor, Peter Sanders, Dorothea Wagner, and Renato F Werneck. Route planning in transportation networks. In *Algorithm Engineering*, pages 19–80. Springer, 2016.
- [3] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- [4] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

Konzeption des Entwicklungsprozesses zur Berücksichtigung von Genderaspekten in fahrzeugmechatronischen Systemen

Xiaobo Liu-Henke, Haoqi Tao, Sven Jacobitz

Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau,

Institut für Mechatronik, Salzdhalmmer Str. 46/48, 38302 Wolfenbüttel

x.liu-henke@ostfalia.de

Der vorliegende Beitrag beschreibt einen Entwicklungsprozess zur Berücksichtigung von Genderaspekten für fahrzeugmechatronische Systeme. Es wird der aktuelle Stand der Technik bezüglich des Entwicklungsprozesses für mechatronische Systeme im Kraftfahrzeug dargestellt, um dann hinsichtlich der Genderaspekten zu analysieren. Dabei wird mit Hilfe der Geschlechterforschung die mögliche Lücke des aktuellen Entwicklungsprozesses festgelegt. Daraufhin wird eine Methodik entwickelt, die die neu herausgearbeiteten Anforderungen in den Entwicklungsprozess einbindet und in mechatronischen Fahrzeugsystemen abzubilden bzw. verifizieren.

1 Einleitung und Motivation

Im Zusammenhang mit der Digitalisierung und Vernetzung des Verkehrs und Smart Cities wird autonomes Fahren als eine Schlüsseltechnologie gesehen. Zur Realisierung des autonomen Fahrens ist die Anzahl intelligenter Systeme im Kraftfahrzeug in den vergangenen Jahren stark gestiegen. Nahezu in jedem Teilsystem des Fahrzeugs, in Antrieb und Fahrwerk sowie im Sicherheits- und Komfortbereich, sind mechatronische Systeme im Einsatz, die einen aktiven Eingriff in die Fahrdynamik vornehmen. Eine Vielzahl von diesen Systemen interagiert dabei sehr stark mit den Fahrer_innen. Während der Entwicklung von mechatronischen Fahrzeugsystemen wird derzeit vor den Fahrversuchen im realen Fahrzeug keine subjektive Beurteilung von verschiedenen Fahrer_innen berücksichtigt [1]. Dies führt nicht nur zu einer großen Anzahl an Iterationsschritten und daraus folgenden hohen Entwicklungskosten bzw. -zeiten, sondern kann noch zahlreiche ethische, rechtliche, wirtschaftliche und soziale Probleme hervorbringen. Daher ist es notwendig, dass der Einbezug dieser Aspekte bereits in der frühen Phase der Entwicklung mechatronischer Fahrzeugsysteme erfolgt.

Auf der vierten UN-Weltfrauenkonferenz in Peking 1995 wurde erstmals die begriffliche Kategorie „Gender“ eingeführt. Dabei wird die geschlechtliche Identität als soziale Rolle bzw. als rein kulturelle Konstruktion betont [2]. Nach [3] sind Gender und Technik kooperativ konstruiert und beeinflussen sich wechselseitig, ohne, dass ihr Verhältnis vorher festgelegt ist. Die Beziehung ist variabel und historisch veränderlich ist.

Die häufig als wertneutral und objektive dargestellte ingenieurwissenschaftliche Tätigkeit ist nach [4] stets sozial eingebettet und beeinflusst das Alltagsleben unterschiedlicher Menschen. Geschlechterforschung bietet eine Möglichkeit zur Verbindung zwischen gesellschaftlicher Welt und technischer Entwicklung. Durch Geschlechterforschung lässt sich die menschliche Vielfalt verstehen und in der Technikentwicklung berücksichtigen. Aus diesem Gründen werden Gender- und Diversity-Aspekten zu immer wichtigeren Merkmalen der Entwicklung und der Qualitätsbeurteilung.

Das Ziel dieses Beitrags ist einen Entwicklungsprozess zur Berücksichtigung von Gender-Aspekten für fahrzeugmechatronische Systeme zu konzipieren, um die genderdifferenzierte Betrachtung früherer in der Entwicklungsphase vorzunehmen. Somit können die mechatronischen Systeme im Kraftfahrzeug durch die frühzeitige Einbindung von Nutzer_innen gleichwertiger und marktfähiger entwickelt werden.

2 Stand des Wissens

Wie können Genderaspekte in der Technologieentwicklung erkannt und bewertet werden? Die Fraunhofer Gesellschaft hat im Rahmen ihres Projekts „Discover Gender“ [5] einen praktischen Leitfaden zur Ermittlung der Gender-Aspekte in Forschungsvorhaben zusammengestellt. Dabei werden die Gender-Aspekten hinsichtlich konkreter technischer Fallbeispiele in verschiedenen Forschungsthemen untersucht.

Die genderspezifischen Anforderungen für die Entwicklung einer neuen Lasergravurmaschine unter Berücksichtigung der Mensch-Maschine-Schnittstelle wird in [6] dargestellt. Als Ergebnisse wird der Leitfaden „Design for Gender“ zusammengefasst, welcher Entwicklungsteams unterstützt, genderrelevante Anforderungen zu erfassen und sie im Entwicklungsprozess zu berücksichtigen. Cojocar [6] weist diesbezüglich darauf hin, dass es keine genderspezifischen Anforderungen an Technikentwicklung geben kann, sondern viel mehr eine grundlegende Vielfalt der Nutzer_innen angenommen werden muss, die sich eben nicht auf Geschlechtsunterschiede beschränkt.

In [7] wird ähnliche Ansicht gegeben, dass aus Perspektive der neuen Geschlechterforschung nicht nur auf die Differenzen im Körperbau und weitere körperlichen Unterschiede fokussieren, sondern noch etwa die vielfältigen Bedarfe und Lebenslagen von verschiedenen Menschen in den Blick nehmen müssen. Hierbei dürfen weder ganze Zielgruppen noch bestimmte Nutzer_innen ausgeschlossen werden.

Um den Zusammenhang zwischen Fahrer_innen und fahrzeugmechatronischen Systemen mit den Gender- und Diversity-Aspekten zu untersuchen, ist es unerlässlich das Gesamtsystem *Fahrer-Fahrzeug-Fahrumgebung (3F)* zu untersuchen, da mit der Entwicklung des autonomen Fahrens die Fahrer_innen, das Fahrzeug und die Umgebung des Fahrzeugs in Raum und Zeit enger zusammenwirken. Das dynamische Zusammenspiel dieser drei Faktoren beim Prozess der Fahrzeugführung kann grundsätzlich wie in [1] durch einen geschlossenen Regelkreis dargestellt werden. Die Fahrer_innen nehmen mit ihren Sinnen Informationen über den Fahrzustand wahr. Diese Informationen können sowohl der Straßenverlauf, die Verkehrssituation und darin auftretende Störungen aus der Fahrumgebung, als auch das Fahrverhalten und die Fahrdynamik des Fahrzeugs sein. Nachdem diese aufgenommenen Informationen im Gehirn hinsichtlich ihrer Erkenntnisse und Erfahrungen verarbeitet wurden, wirken die Fahrer_innen über Lenkung, Bremsen, Antrieb und Betätigung der Fahrerassistenzsysteme auf das Fahrzeug.

3 Mechatronischer Entwicklungsprozess

Zur Entwicklung eines Systems mit sehr hoher Komplexität wird die mechatronische Entwicklungsmethodik herangezogen. Zur Gewährleistung eines systema-

tischen Entwurfs ist die Beherrschung der Systemkomplexität elementar. Nach [8] erfolgt die Zerlegung des Systems mittels Modularisierung und Hierarchisierung im Top-Down-Verfahren in Teilsysteme mit mechatronischen Komponenten. Dabei werden die Systemfunktionen mit den definierten Schnittstellen in einer effizienten Art dargestellt. Nach der Strukturierung und Schnittstellendefinition wird das System sukzessiv von unten nach oben im Bottom-up-Verfahren mittels mechatronischer Komposition aufgebaut.

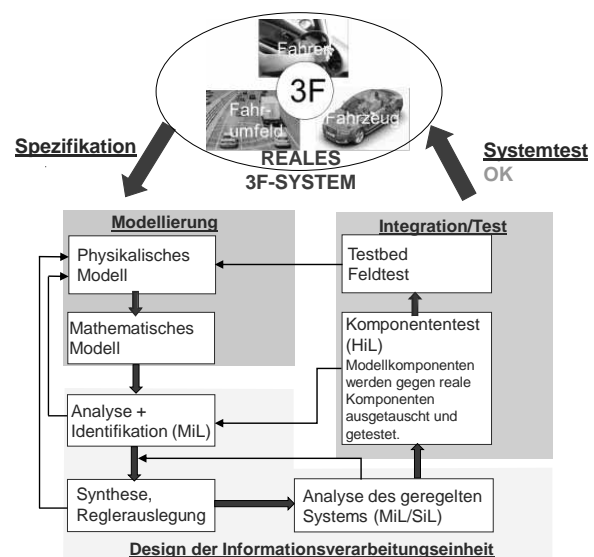


Abbildung 1. Aktueller Entwicklungsprozess für fahrzeugmechatronische Systeme

Zur Auslegung der einzelnen Funktionsmodule wird der in Abbildung 1 dargestellte mechatronische Entwicklungsprozess eingesetzt. Der Entwicklungsprozess kann grob in vier einzelnen Schritte eingeteilt werden: Anforderungsbestimmung, Modellbildung, Auslegung der Informationsverarbeitung und Validierung. Zu Beginn werden die Anforderungen nach den geforderten Systemeigenschaften definiert. Ausgehend davon wird das technische System nach dem Funktionsprinzip in einem physikalischen bzw. mathematischen Ersatzmodell abgebildet. Das Ersatzmodell repräsentiert das Systemverhalten. Darauf basierend wird die Synthese der Softwarefunktionen durchgeführt. Die notwendige Sensorik, Aktorik und Regelalgorithmen werden anhand des Systemverhaltens festgelegt. Anschließend werden die ausgelegten Funktionen in einem durchgängig verifikationsorientierten Prozess mittels Model-in-the-Loop (MiL), Software-in-the-Loop (SiL), und Hardware-in-the-Loop (HiL) validiert und optimiert.

4 Genderspezifische Analyse des vorhandenen Entwicklungsprozesses

In diesem Abschnitt wird der vorhandene Entwicklungsprozess anhand von Gender- und Diversity-Aspekten analysiert. In diesem Zusammenhang werden auch die Lücken im Entwicklungsprozess aufgezeigt. Die Analysen sollen sowohl für einzelne Schritte des Entwicklungsprozesses, als auch für dessen Aufbaustruktur durchgeführt werden.

4.1 Analyse der Anforderungsbestimmung

Die an ein fahrzeugmechatronisches System gestellten Anforderungen sind vielfältig und lassen gerade im Kraftfahrzeug genderspezifische Unterschiede zu, welche jedoch bisher im Entwicklungsprozess nicht hinreichend berücksichtigt werden. Bisher werden solche Anforderungen lediglich ausgehend von einem durchschnittlichen Kunden einer Zielgruppe definiert aufgestellt. Außerdem neigen die Technikentwickler_innen auch dazu, Anforderungen an die Systemen anhand ihrer eigenen Bedürfnisse oder stereotyper Annahmen über Kunden zu stellen, da sie sich selbst als spätere Fahrer_innen sehen [4]. Daher werden die Systeme oft mit diesem, sehr subjektiven, Ziel entwickelt und getestet. Dies führt zu wirtschaftlichen Fehlschlägen, Akzeptanzschwierigkeiten und sogar Gefahren für die Menschen. Ein typisches Vorbild in der Automobilbranche dafür sind normierte Crashtest-Dummys. Die frühen Crashtest-Dummys wurden ausschließlich basierend auf männlichen Normkörpern entwickelt. Später wurden weitere Modellreihen, welche Frauen, Kinder und seit 1996 doch noch Schwangeren simulieren können, nach und nach ergänzt [4].

Will man die tatsächlich vom Benutzer gestellten Anforderungen bei der Entwicklung betrachten, müssen die relevanten Aspekte des gesamten 3F-Systems für die jeweiligen Funktionen berücksichtigt werden [9]. In Abbildung 2 wird dazu das Zusammenspiel von Fahrer_innen, Fahrzeug und Fahrumgebung mit einigen zu berücksichtigenden Aspekten dargestellt.

Im vorhandenen Entwicklungsprozess werden jedoch häufig nur die Reaktionen des Fahrzeugs auf das Lenken, das Beschleunigen und Verzögen über Gas- und Bremspedal, sowie auf die Störungen der Fahrumgebung berücksichtigt. Hierbei werden die Vielfältigkeit der Fahrer_innen und die veränderliche Fahrumgebung zum Großteil vernachlässigt. Auch werden die

Unterschiede zwischen Männern und Frauen bzw. Jungen und Mädchen sowie das Alter für die Interaktionen zwischen Fahrer_innen und Fahrzeug sowie zwischen Fahrer_innen und Fahrumgebung kaum untersucht.

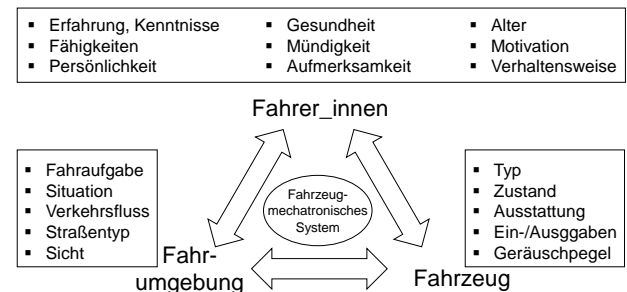


Abbildung 2. Zusammenspiel zwischen Fahrer_innen, Fahrzeug und Fahrumgebung [9]

4.2 Analyse der Modellbildung

In der Modellbildungsphase wird im Allgemeinen ein Fahrzeugmodell und ein Fahrumgebungsmodell aufgebaut. Zur Erprobung wird zusätzlich ein technisches Fahrermodell eingesetzt, dessen Aufgabe es ist, aufgrund der modellierten Fahrumgebung die Interaktion zwischen Fahrer_innen und Fahrzeug vorherzusagen.

In der aktuellen Literatur kann man eine Vielzahl von Fahrermodellen mit verschiedenem Betrachtungsfokus finden. Ein kognitives Modell, welches den Arbeitsprozess der zielgerichteten Tätigkeiten durch einen Menschen allgemein beschreibt, wurde 1983 von Rasmussen veröffentlicht [10]. Auf dieser Grundlage kann danach die Fahraufgabe in drei Ebenen mit unterschiedlicher kognitiver Inanspruchnahme eingeteilt werden. Dieses Drei-Ebenen-Modell ist in Abbildung 3 links dargestellt. Dabei wird zwischen wissensbasiertem, regelbasiertem und fertigkeitbasiertem Verhalten unterschieden [11].

Ein technisches Modell, in Abbildung 3 rechts dargestellt, ist das Modell nach Donges aus dem Jahr 1982 [12]. Dieses wurde aus Ingenieursicht für konkrete Fahraufgabe entwickelt. Auch hier werden drei Ebenen unterschieden: Navigation, Führung und Stabilisierung [11]. Jedoch wird in solchen technischen Fahrermodellen das menschliche Verhalten nur teilweise berücksichtigt. Die Unterschiede zwischen verschiedenen Zielgruppen, welche für eine genderspezifische Auslegung notwendig sind, findet keine Berücksichtigung. Hier besteht daher Handlungsbedarf (s. Pfeile in Abbildung 3), um das menschliche Verhalten und dessen Unterschiede im vorhandenen Fahrermodell zu berücksichtigen und zu integrieren.

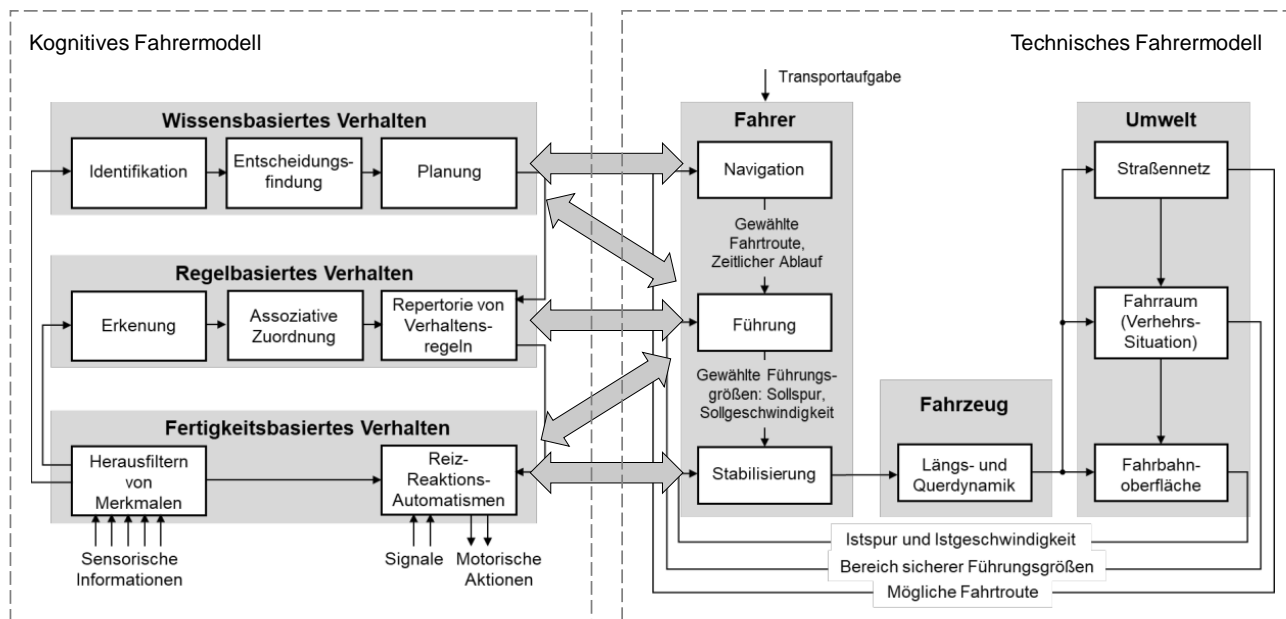


Abbildung 3. Kombinierte Darstellung der technischen und kognitiven Modelle nach Rasmussen und Donges [11]

4.3 Analyse der Auslegung der Informationsverarbeitung und Validierung

Bisher wird die Auslegung der Informationsverarbeitung durch reine Mathematik durchgeführt. Somit lassen sich subjektive Eigenschaften der vielfältigen Fahrer_innen bei Auslegung der Informationsverarbeitung kaum berücksichtigen. Dies ist ein Interdisziplinäres Problem, welches durch Ergänzung von Wahrnehmungs- und Kognitionstheorie sowie Handlungspsychologie gelöst werden muss. Ein ähnliches Problem entsteht ebenfalls während der Validierungsphase, in der die Funktionen mittels qualitativen Methoden oft nur für eine Durchschnittszielgruppe abgesichert werden. Dadurch wird die Zielgruppenvielfalt vernachlässigt.

5 Konzeption des neuen Entwicklungsprozesses

Basierend auf der vorherigen Analyse wird in diesem Abschnitt die prinzipielle Konzeption des Entwicklungsprozesses zur Berücksichtigung von Gender- und Diversity-Aspekten für fahrzeugmechatronische Systeme dargestellt.

Abbildung 4 stellt die prinzipielle Konzeption des genderspezifischen Entwicklungsprozesses dar. Hierbei wird der vorhandene Entwicklungsprozess durch ein Anforderungsmanagement ergänzt und erweitert. Während der Entwicklung wird jeder Schritt des Entwicklungsprozesses anhand des Anforderungsmanage-

ments ausgeführt. Dabei müssen nicht nur der Entwicklungsprozess selbst, sondern gleichzeitig noch die Fahrumgebung ganzheitlich betrachtet werden. Die in Abbildung 4 rot markierten, möglichen, Lücken im gendergerechten Entwicklungsprozess ergeben sich aufgrund der in Abschnitt 4 dargestellten Analyse. Diese sind im weiteren Verlauf der Forschungsarbeit weiter zu untersuchen. Hierbei werden die Lücken genau spezifiziert sowie Lösungsvorschläge zur Schließung erarbeitet und validiert.

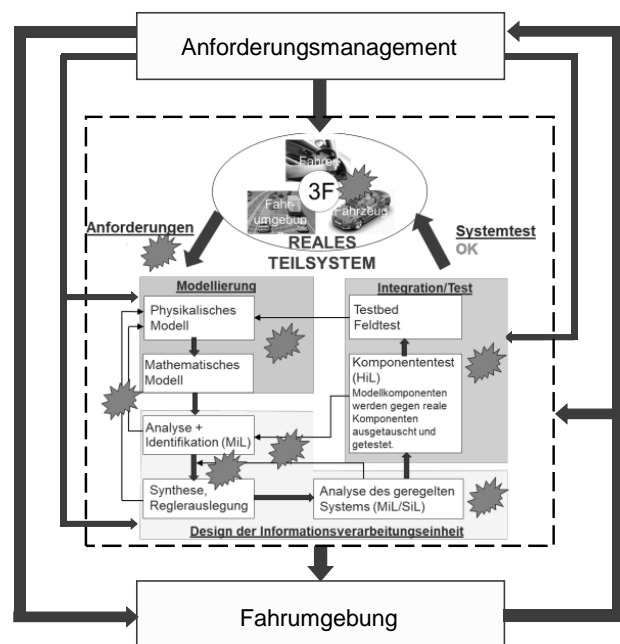


Abbildung 4. Konzept des Entwicklungsprozesses

5.1 Anforderungsmanagement

Um die genauen Anforderungen an das zu entwickelnde System zu stellen, muss ein neuartiges Anforderungsmanagement herangezogen werden. Abbildung 5 stellt die hierarchische Struktur des Anforderungsmanagements dar. Es handelt sich um die technischen Anforderungen und die Anforderungen zur Berücksichtigung von Mensch- und Sozialgerechtigkeit.

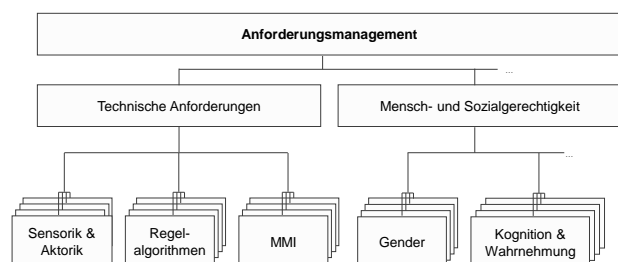


Abbildung 5. Hierarchische Struktur des Anforderungsmanagements

Die Analyse der beiden Anforderungen wird bei der Entwicklung eines mechatronischen Systems parallel durchgeführt. Das Fachwissen und die Methoden zur Ermittlung der technischen Anforderungen stehen für Ingenieure zur Verfügung, z.B. Auswahl der Sensorik und Aktorik, Auslegung von Reglern und MMI. Zur Untersuchung der Anforderungen an die Mensch- und Sozialgerechtigkeit ist es besonders wichtig, das menschliche Verhalten und die Fähigkeiten der Fahrer_innen zu ermitteln. Schwierigkeit ist hierbei, die relevanten Parameter der Fahrer_innen, Fahrzeug und Fahrumgebung hinsichtlich solcher Aspekte für die jeweiligen Funktionen des fahrzeugmechatronischen Systems zu identifizieren, zu beschreiben und zu quantifizieren. Da diese von extrem vielen Faktoren abhängig und sehr komplex sind, muss hierbei die Entwicklung von Systemen neben Ingenieurwissenschaften durch die sozialwissenschaftlichen Erkenntnisse ergänzt werden. Um die gewünschten Daten zu erfassen, können z.B. die Methoden Beobachtung, physiologische Messung und Befragungen während Probandenversuchen verwendet werden.

Im Rahmen der Gender- und Diversity-Aspekte müssen die Fahrer_innen als ein zentrales Element im den

Entwicklungsprozess stehen. Durch Untersuchung der Fahrer_innen können verschiedene Charaktereigenschaften für genderspezifische Aspekte berücksichtigt werden. Hierbei müssen die Interaktionen innerhalb des gesamten 3F-Systems ganzheitlich betrachtet werden. Es muss untersucht werden, wie die Fahrer_innen auf die Fahrumgebung und das Fahrverhalten reagieren und wie das Fahrzeug auf die verschiedenen Fahrer_innen reagieren. Dies wird immer wichtiger für das zukünftige kognitive Kraftfahrzeug [13]. Von diesem kognitiven Kraftfahrzeug wird erwartet, dass es sich selbst und Fahrumgebung sowie Fahrer_innen wahrnehmen kann.

5.2 Modellbildung

Im Rahmen eines gendergerechten Entwicklungsprozesses für fahrzeugmechatronische Systeme, muss im Prozessschritt der Modellbildung neben dem technischen Modell auch ein menschliches Verhaltensmodell, welches den Fahrer_innenzustand repräsentiert, aufgebaut werden. Dieses ergänzt dann das technische Fahrermodell. Das Verhaltensmodell aus Abbildung 3 muss hierzu sowohl Gender- und Diversity-Aspekte also auch um Kognition berücksichtigen und ggf. um diese Punkte erweitert werden. Für diese Erweiterung bestehen zwei Lösungsmöglichkeiten: Veränderung der vorhandenen Modellparameter oder Erweiterung der Modellstruktur. Da eine reine Veränderung der Modellparameter jedoch nur eine Verschiebung der betrachteten Problematik zur Folge hat, wird der Ansatz der Strukturveränderung weiterverfolgt.

Nach [14] wird die prinzipielle Modellstruktur für das menschliche Verhalten durch die Hauptmodule Wahrnehmen, Erkennen, Entscheiden und Handeln abgebildet (siehe Abbildung 6). Jedes Hauptmodul verfügt über eignen Fähigkeitsparameter und wird gleichzeitig durch anderen Eigenschaftsparameter und Fertigungsparameter sowie intrinsische Parameter ergänzt. Eingangsgrößen sind Informationen aus Fahrzeug und Fahrumgebung. Der Ausgang ist durch Veränderungen der Stellgrößen des Fahrzeugs beschrieben.

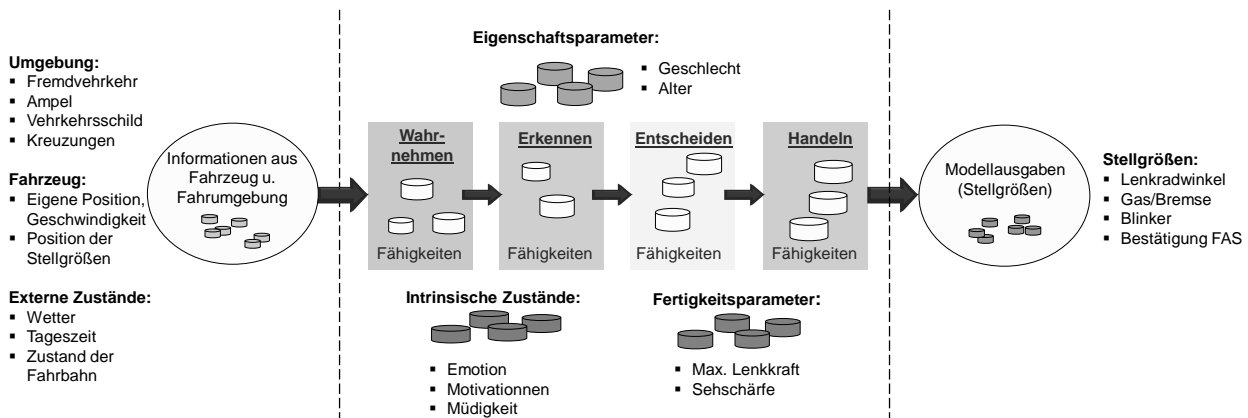


Abbildung 6. Prinzipielle Struktur des menschlichen Verhaltensmodells

Eine besondere Schwierigkeit im Zusammenhang mit psychologischen Fahrermodellen liegt in der Identifizierung der Parameter. Ein Überblick über die Identifizierung vom menschlichen Fahrermodell wird in [15] vorgestellt. Hierbei wird zwischen parametrischen, semi-parametrischen und nicht-parametrischen Verfahren unterschieden.

5.3 Auslegung der Informationsverarbeitung und Validierung

Um ein gendergerechtes System zu entwickeln, müssen die Funktionen bei der Auslegung der Informationsverarbeitung personalisiert entworfen werden. Hierzu müssen sich die Funktionen oder deren Parameter in der Abhängigkeit der verschiedenen Zielgruppen durch Erkennung des Fahrerverhaltens automatisch anpassen können.

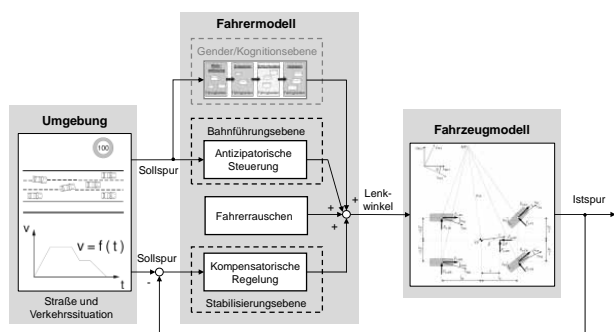


Abbildung 7. Modellvalidierung am Beispiel eines Lenksystems

Nachdem die Modellbildung und Regler Synthese durchgeführt wurde, werden die Softwarefunktionen mittels MiL, SiL und HiL validiert (siehe Abbildung 7). Dabei wird die Integration des menschlichen Verhaltensmodells im vorhandenen technischen Fahrermodell fokussiert. Zur Beurteilung der Ergebnisse können die subjektiven Eigenschaften von Menschen

durch Objektivierung betrachtet werden. Beispielsweise können die Bewertungskriterien Aufgabenerfüllung, körperliche und geistige Arbeitsbelastung während der Querführung beim Fahren durch die folgenden quadratischen Gütemaßfunktionen dargestellt werden [16]:

- Aufgabenerfüllung J_{AE} :

$$J_{AE} = \frac{1}{S} \int_0^t \Delta y^2 d\tau$$
- Körperliche Arbeitsbelastung J_{KAb} :

$$J_{KAb} = \frac{1}{S} \int_0^t \delta^2 d\tau$$
- Geistige Arbeitsbelastung J_{GAb} :

$$J_{GAb} = \frac{1}{S} \int_0^t \dot{\delta}^2 d\tau$$

Dabei ist Δy die Querabweichung, δ der Lenkwinkel und $\dot{\delta}$ die zeitliche Ableitung des Lenkwinkels.

6 Resümee und Ausblick

Der vorliegende Beitrag fokussiert das Konzept eines ganzheitlichen Entwicklungsprozesses zur Berücksichtigung von Genderaspekten in fahrzeugmechatronischen Systemen. Hierbei wurde zunächst der aktuell angewendete Prozess analysiert. Durch Erweiterung des technischen durch ein Psychologisches Fahrermodell wurde ein auf Anforderungsmanagement basierendes Konzept für einen verbesserten Prozess vorgestellt.

Weitere Forschungsarbeit befasst sich mit der Realisierung, Anwendung und Validierung des konzipierten Entwicklungsprozesses an einem konkreten Anwendungsbeispiel sowie Optimierung und Erweiterung des genderspezifischen Entwicklungsprozesses. In der Berücksichtigung von Gender- und Diversity-Aspekten im Entwicklungsprozess wird ein bedeutendes Potenzial gesehen.

Danksagung

Diese Arbeit wurde im Rahmen des Promotionskollegs „Konfiguration von Mensch, Maschine & Geschlecht. Interdisziplinäre Analysen zur Technikentwicklung (KoMMA.G)“ durch das Niedersächsische Ministerium für Wissenschaft und Kultur gefördert. Für die Förderung bedanken sich die Autoren herzlichst.



Niedersächsisches Ministerium
für Wissenschaft und Kultur



KONFIGURATIONEN VON MENSCH,
MASCHINE & GESCHLECHT
Interdisziplinäre Analysen zur Technikentwicklung

7 Literaturverzeichnis

- [1] X. Liu-Henke, M. Göllner, R. Buchta und F. Quantmeyer, „Systemkonzept eines modularen HiL-Systems für modellbasierte Funktionsentwicklung fahrzeugmechatronischer Systeme,“ in *ASIM/GI STS/GMMS Workshop 2017*, Ulm, 2017.
- [2] B. J. Krings, „Genderforschung und Technikentwicklung,“ pp. 5-8, 07 2002.
- [3] C. Bath, „Gender und Technik,“ Ostfalia HAW, Wolfenbüttel, 2015.
- [4] C. Bath, „Gender, Technik und Mobilität - Innovative, soziotechnische Lösungen für gesellschaftliche, wissenschaftliche und wirtschaftliche Herausforderungen,“ Braunschweig, 2015.
- [5] S. Bühner und M. Schraudner, „Gender-Aspekte in der Forschung,“ Fraunhofer, Institut System- und Innovationsforschung, Karlsruhe, 2006.
- [6] E. Cojocar, „Design for Gender: Bedienungsgerechte Maschinenentwicklung durch Expertise von MaschinenbedienerInnen,“ in *Agenda Gute Arbeit: geschlechtergerecht!*, Wien, VSA: Verlag, Hamburg, 2014, pp. 158-168.
- [7] C. Bath, „„Discover Gender“ in Forschung und Technologieentwicklung? Re-discover Gender Studies, Wissenschafts- und Gesellschaftskritik,“ *Sozial Technik, Zeitschrift für sozial- und umweltverträgliche Technikgestaltung*, pp. 3-5, 2007.
- [8] X. Liu-Henke, „Mechatronische Entwicklung der aktiven Feder-/Neigetchnik für das Schienenfahrzeug RailCab,“ Dissertation, Universität Paderborn, 2004.
- [9] W. König, „Nutzergerechte Entwicklung der Mensch-Maschine-Interaktion von Fahrerassistenzsystemen,“ in *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, Springer Vieweg, 2015, pp. 33-42.
- [10] J. Rasmussen, „Skills, Rules and Knowledge; Signals, Signs and Symbols and other Distinctions in Human Performance Models,“ *IEEE Trans. on Systems, Man and Cybernetics*, 1983.
- [11] E. Donges, „Fahrerverhaltensmodelle,“ in *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*, Springer Vieweg, 2015, pp. 17-54.
- [12] E. Donges, „Aspekte der aktiven Sicherheit bei der Führung von Personkraftwagen,“ *Automobil-Industrie* 27, 1982.
- [13] B. Deml, H. Neumann, A. Müller und H. J. Wünsche, „Fahrermodellierung im Kontext kognitiver Automobile,“ *Automatisierungstechnik*, pp. 601-608, 11 2008.
- [14] S. Kupschick und T. Jürgensohn, „Entwicklung einer modularen Plattform für menschliche Fahrermodelle,“ in *Fahrermodellierung in Wissenschaft und Wirtschaft*, Berlin, VDI, 2015, pp. 83-93.
- [15] W. Wang, J. Xi und H. Chen, „Modeling and Recognizing Driver Behavior Based on Driving Data: A Survey,“ in *Mathematical Problems in Engineering*, Peking, 2014.
- [16] W. Wang, J. Xi und J. Wang, „Human-Centered Feed-Forward Control of a Vehicle Steering System Based on a Driver’s Steering Model,“ in *American Control Conference*, Chicago, 2015.
- [17] G. Hiesgen, Effiziente Entwicklung eines menschenzentrierten Querführungsassistenzsystems mit einem Fahrsimulator, Oberhausen: Universität Duisburg-Essen, 2011.

Systemkonzept einer durchgängigen Low-Cost RCP-Plattform

Xiaobo Liu-Henke, Sven Jacobitz, Marian Göllner, Sören Scherler
Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau,
Institut für Mechatronik, Salzdahlumer Str. 46/48, 38302 Wolfenbüttel
x.liu-henke@ostfalia.de

Mechatronische Systeme finden sich heutzutage in allen Lebensbereichen wieder. Bei immer kürzeren Produktentwicklungszeiten ist ein strukturiertes Vorgehen bei der Entwicklung neuer Funktionen unabdinglich. Daher wird immer häufiger der durchgängig modellbasierte Ansatz des Rapid Control Prototyping (RCP) für das Systemdesign verwendet. Dieser zeichnet sich durch Nutzung einer durchgängigen Werkzeugkette, bestehend aus Hard- und Software, mit hohem Automatisierungsgrad aus. Insbesondere für kleine und mittelständische Unternehmen (KMU) sind durchgängige RCP-Werkzeugketten mit einem hohen Anschaffungspreis verbunden, wodurch die Wettbewerbsfähigkeit bei der mechatronischen Systementwicklung gehemmt wird. Der vorliegende Beitrag stellt daher das Systemkonzept einer Low-Cost RCP-Plattform, bestehend aus Offline Simulationsumgebung, Echtzeithardware und Mensch Maschine Schnittstelle für Onlineexperimente, vor. Hierzu erfolgt zunächst eine allgemeine Analyse von RCP-Systemen, eine Übersicht aktuell auf dem Markt verfügbarer Systeme sowie eine abschließende Konzeption des Systems.

1 Einleitung

Zur Verbesserung wichtiger Systemeigenschaften wird immer mehr intelligente Hard- und Software in technische Systeme eingebettet. Kern dieser intelligenten Systeme sind Steuergeräte und die auf dem Steuergerät implementierten, hochwertigen Reglerfunktionen einschließlich Signalverarbeitung, in Gesamtheit ein sogenanntes Embedded Control System. Zusammen mit der Sensorik und Aktorik bildet dieses ein eingebettetes, mechatronisches System mit hohem Vernetzungsgrad mit hochgradig komplexer Software für Steuerungs- und Regelsysteme. Das Batteriemanagementsystem für Elektrofahrzeuge, Herzschrittmacher in der Medizintechnik, Steuerungs- und Regelsysteme für intelligente Gebäude und für die digitale Produktionstechnik sind nur einige Anwendungsbeispiele aus verschiedenen Gebieten unserer Gesellschaft.

Zur Funktionsauslegung eingebetteter mechatronischer Systeme mit hochgradig komplexer Software ist eine ganzheitliche, modellbasierte Entwicklungsmethodik erforderlich, um die steigende Komplexität beherrschen zu können. Kennzeichnend für diese Methodik sind die hohe Durchgängigkeit und der Automatisierungsgrad, von der Modellbildung, der modellbasierten Funktionsauslegung über die automatische Codegenerierung bis hin zur Echtzeit-Realisierung [1]. Diese Entwicklungsmethodik, begleitet durch eine durchgängige, computergestützte, vollautomatisierte CAE-Plattform, wird als Rapid Control Prototyping (RCP) bezeichnet.

Derzeit wird diese modellbasierte Funktionsentwicklung ausschließlich durch kostspielige CAE-Werkzeugketten (z.B. mittels MATLAB/Simulink, kombiniert mit einem RCP-System der Firma dSPACE) als „High-Cost-RCP CAE-Werkzeugkette“ unterstützt [2]. Diese sind jedoch mit erheblichen Anschaffungskosten verbunden, weshalb sie primär nur von größeren Firmen eingesetzt werden können.

KMU verfügen selten über die Finanzmittel zur Nutzung solcher RCP-Entwicklungsplattformen. Sie haben praktisch keine Chance im Gebiet der eingebetteten mechatronischen Systeme eine Wettbewerbskompetenz aufzubauen. Weiterhin liegt den aktuellen Entwicklungen der KMU häufig Einzelstück-Charakter zugrunde, welcher mit einem sehr hohen personellen, zeitlichen und finanziellen Aufwand aufgrund manueller Programmierung für Steuerungs- und Regelsysteme verbunden ist. Zudem lassen sich Anforderungen des Qualitätsmanagements, aufgrund der hauptsächlich zufälligen Fehler, nur mit großem Kontrollaufwand erfüllen [3]. Durch Nutzung einer durchgängigen RCP-Entwicklungsplattform können sowohl der zeitliche, als auch der finanzielle Aufwand reduziert und das Qualitätsmanagement nachhaltig integriert werden [4].

Aus diesem Grund wird an der Ostfalia Hochschule im Rahmen eines EFRE geförderten Projektes eine Low-Cost RCP-Plattform entwickelt. Der vorliegende Beitrag fokussiert das Systemkonzept der zu entwickelnden Plattform. Es werden RCP-Systeme im Allgemeinen untersucht, ein Überblick über den aktuellen Stand

der Technik gegeben und das Konzept einer Low-Cost Entwicklungsplattform dargestellt.

2 RCP Entwicklungsmethodik

Der durchgängige RCP-Prozess besteht aus einer modellbasierten, verifikationsorientierten Entwicklungsmethodik mit den drei Entwicklungsstufen Model-in-the-Loop (MiL), Software-in-the-Loop (SiL) und Hardware-in-the-Loop (HiL) [5]. Entscheidende Vorteile dieser Methodik sind verkürzte Entwicklungszeiten durch frühzeitige Iterationen und ein geringer Fehleranteil im Quellcode, da dieser automatisiert erzeugt wird (siehe Abbildung 1). Weiterhin erfolgt mithilfe dieser Methodik eine durchgängige Funktionsabsicherung [6].

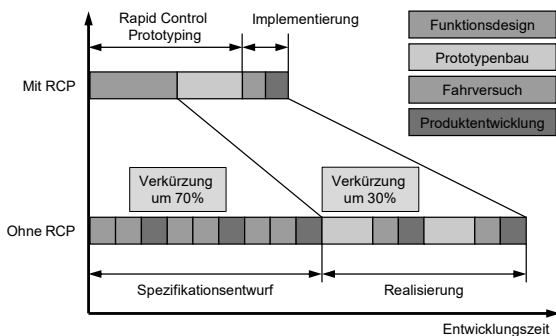


Abbildung 1: Verkürzung der Entwicklungszeit durch den Einsatz von RCP [7]

Um die Systemkomplexität zu reduzieren, erfolgt mittels Modularisierung zunächst die Zergliederung des Gesamtsystems in eine hierarchisch angeordnete Struktur (mechatronische Strukturierung). Nach Definition der Schnittstellen zwischen den Modulen erfolgt, angefangen bei der untersten Ebene, die Auslegung der Teilfunktionen sowie dessen Integration zum Gesamtsystem (mechatronische Komposition) [5]. Der Prozess der Komposition, inklusive Validierung und Optimierung mittels MiL-, SiL- und HiL-Simulationen, wird durch Abbildung 2 illustriert.

Wird der Prozess aus Abbildung 2 genauer analysiert, ergeben sich, angefangen bei Bibliotheken zur Modellierung bis hin zur Simulation auf der Zielhardware, vier Prozessschritte, welche durchlaufen werden (vgl. Abbildung 4, links): Modellierung und Analyse, Erzeugung von Programmcode, Erzeugung von ausführbarem Code und Codeimplementierung.

Während der Modellierung und Analyse werden sowohl die Regelstrecke als auch die Reglerfunktionen als Modell nachgebildet und analysiert. Aus anhand von Offlinesimulationen optimierten Algorithmen und

Parametern erfolgt die automatisierte Transformation in Programmcode sowie die Erweiterung und Optimierung für die Echtzeitverarbeitung. Hieraus entsteht ein ausführbarer Code, welcher automatisiert kompiliert und auf der Zielhardware implementiert wird.

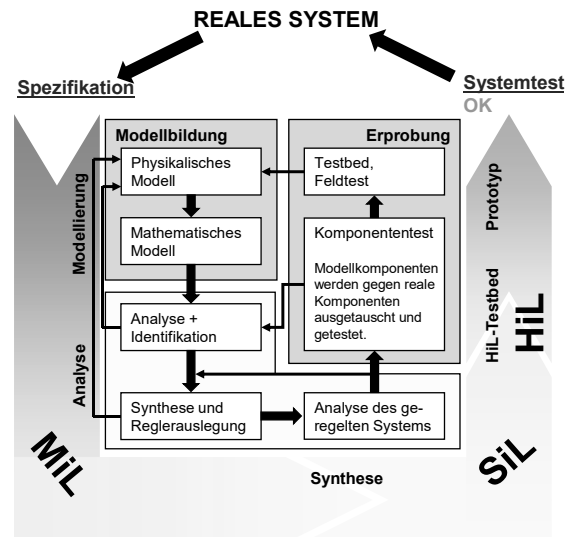


Abbildung 2. Modellbasierter, verifikationsorientierter Entwicklungsprozess nach [5]

3 Aufbau von RCP-Systemen

Bei der systematischen Analyse von RCP-Systemen lassen sich die verwendeten Werkzeuge grundlegend in drei Kategorien einteilen: Offline-Simulationsumgebung, Echtzeithardware zur Onlinesimulation und Werkzeug mit Mensch-Maschine-Interface (MMI) zur Durchführung von Onlinesimulationen (sogenannte Experimentiersoftware).

3.1 Offline Simulationsumgebung

In der Simulationsumgebung werden sowohl die Softwarefunktionen (z.B. hochwertige Regelalgorithmen) als auch die Steuergeräteumgebung (z.B. die Regelstrecke) mittels blockschaltbildorientierter Beschreibungssprache abgebildet, wofür entsprechende Modellbibliotheken benötigt werden. Auch eine Dokumentation des Modells sollte in dieser Softwareumgebung möglich sein. Eine gut dokumentierte Simulation stellt gleichzeitig eine ausführbare Spezifikation dar, was Verständlichkeit und Transparenz innerhalb des Entwicklungsprozesses steigert [3].

Neben der Simulation werden auch Werkzeuge zur Modellidentifikation und Analyse benötigt. Durch eine Simulationsumgebung, welche all diese Funktionen integriert werden Fehler, z.B. durch manuelle Modell- oder Parameterübertragung, vermieden und Arbeitsabläufe erleichtert.

Ist eine ausreichende Güte der zu entwickelnden Funktionen erreicht, wird aus dem Blockschaltbild automatisiert Programmcode generiert. Dieser kann dann getestet und optimiert sowie anschließend in von der Zielhardware ausführbaren Code transformiert werden. Durch den automatisierten Ablauf dieser Funktion und Aufruf der nachfolgenden Softwaretools (Compiler, etc.) werden wiederum Fehler vermieden und Arbeitsabläufe optimiert [1].

3.2 Echtzeithardware zur Onlinesimulation

Mit Fortschreiten der Entwicklung werden zur weiteren Optimierung und Absicherung Onlinesimulationen unter Verwendung realer Komponenten durchgeführt. Hierzu müssen die entworfenen Algorithmen auf einer entsprechend leistungsfähigen Echtzeithardware implementiert werden. Nach Möglichkeit erfolgt dies wie in Abschnitt 2 beschrieben ohne manuelle Programmierung, vollständig automatisiert, aus der Simulationsumgebung heraus. Die Echtzeithardware muss über ausreichende Rechenleistung, Speicher und Schnittstellen verfügen [8].

Die zur Onlinesimulation notwendige Echtzeithardware lässt sich hauptsächlich in zwei Kategorien einteilen (siehe Abbildung 3): Ausführung eines Target-Programms auf einem Echtzeitrechner und Überwachung / Kontrolle durch ein Hostprogramm auf einem nicht echtzeitfähigen PC (Abbildung 3a) oder Ausführung der Softwarefunktionen auf einem speziell konfigurierten Host-PC mit Nutzung eines Serverprogramms auf einer Echtzeithardware als Schnittstellenkarte (Abbildung 3b)[9]. Letztere Konfiguration ist in der Regel nur für langsame Regelungssysteme geeignet und erfüllt keine harten Echtzeitanforderungen, also keine garantierte Einhaltung der vorgegebenen Zykluszeit.

3.3 Experimentiersoftware

Während der Onlinesimulation muss die Möglichkeit bestehen, Signale zu beobachten und Parameter zu verstellen (sogenannte Mess- und Verstellaufgaben) [6]. Wird eine Topologie nach Abbildung 3a verwendet, ist hierzu eine entsprechende Kopplung zwischen der Echtzeithardware und dem Host-PC notwendig.

In der Experimentiersoftware muss die Möglichkeit bestehen, intuitiv und schnell übersichtliche Oberflächen für verschiedene Onlinesimulationen zu erstellen. Hierzu müssen verschiedene Anzeige- und Eingabelemente platziert und mit Signalen verknüpft werden. Neben den Mess- und Verstellaufgaben können noch

diverse weitere Aufgaben wie Daten- und Versionsmanagement durch die Experimentiersoftware erfüllt werden. Hierdurch wird eine hohe Durchgängigkeit des RCP-Prozesses erzielt.

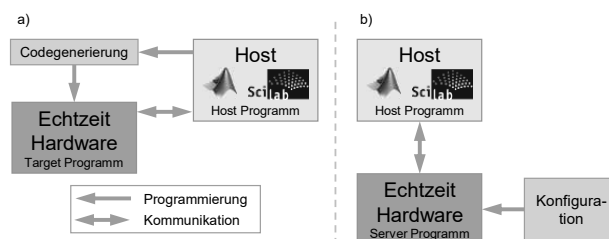


Abbildung 3. Mögliche RCP Hardware Architekturen

4 Stand der Technik

Für die in Kapitel 3 dargestellten Werkzeuge gibt es diverse Realisierungsansätze. Wird die Analyse zunächst auf die Simulationsumgebung (SW) und die Echtzeithardware (HW) beschränkt, ergeben sich grundsätzlich vier Möglichkeiten, SW und HW zu kombinieren:

1. High-Cost SW + High-Cost HW: Im Folgenden als High-Cost System bezeichnet.
2. High-Cost SW + Low-Cost HW: Im Folgenden als teilweise Substitution bezeichnet.
3. Low-Cost SW + High-Cost HW: Wird derzeit nicht unterstützt.
4. Low-Cost SW+Low-Cost HW: Im Folgenden als vollständige Substitution bezeichnet.

4.1 High-Cost Systeme

Wie im Kapitel 1 dargestellt, hat sich zur Unterstützung der Entwicklungsmethodik die Verwendung einer kommerziellen „High-Cost“ CAE-Werkzeugkette, bestehend aus MATLAB/Simulink und Echtzeit-Hardware und Software der Firma dSPACE als Quasistandard in der Automobilindustrie etabliert [10].

Von dieser High-Cost-Plattform stehen Bibliotheken bereit, welche neben Funktionsblöcken auch Schnittstellen zwischen Hard- und Software beinhalten. Zur automatisierten C-Code-Erzeugung steht der Simulink Coder sowie für Echtzeitcode der Embedded Coder zur Verfügung. Mithilfe des Real-Time Interfaces (RTI) von dSPACE wird das zuvor generierte ausführbare Echtzeitprogramm automatisiert auf die dSPACE-Hardware übertragen. Ein Mensch-Maschine-Interface (MMI) zur interaktiven Kommunikation ist durch das Programm ControlDesk gegeben [6]. Der Prozessablauf sowie die genutzten Werkzeuge sind in der Spalte High-Cost in Abbildung 4 illustriert.

Insgesamt handelt es sich bei dieser High-Cost RCP-Entwicklungsplattform um einen in der Industrie etablierten, durchgängigen Prozess, welcher die Modellbildung, Funktionsauslegung und Echtzeiteralisierung der Funktionen unterstützt [11].

4.2 Teilweise Substitution

Mit der Simulationsumgebung MATLAB/Simulink ist es seit geraumer Zeit möglich für verschiedene, auch kostengünstige, Hardwaresysteme Code zu generieren und automatisch zu implementieren [11]. Viele Veröffentlichungen, welche sich mit dem Thema Low-Cost RCP befassen, nutzen diese Funktion und substituieren die High-Cost Hardware meist durch für die vorgesehene Anwendung ausreichend dimensionierte und vorkonfigurierte Mikrocontroller.

Weit verbreitet in diesem Bereich ist das *Waijung Blockset* der Firma Aimagin. Dies ermöglicht die Programmierung von ARM-Cortex-M basierten Systemen. Insbesondere Controller der Serie STM32 sind durch vorhandene Konfigurationen schnell und einfach in Betrieb zu nehmen [12]. Durch Erweiterungen kann eine stabile Kommunikation mit MATLAB/Simulink aufgebaut werden, welches hierdurch für Mess- und Verstellaufgaben eingesetzt werden kann. Andere Simulink Erweiterungen fokussieren die Nutzung Arduino-basierter Systeme oder gar für einen speziellen Anwendungsfall konfigurierte Mikrocontroller (vgl. [13]).

Für Anwendungen mit der Topologie nach Abbildung 3b stellt Mathworks die Erweiterung *Real-Time Windows Target* zur Verfügung. Diese verwendet spezielle, hoch priorisierte, Windowstasks für die Berechnung. Nach [14] sind damit Zykluszeiten von bis zu 2 ms möglich. Durch das Funktionsprinzip können Signale direkt aus Simulink heraus manipuliert und beobachtet werden. Ein detailliertes Beispiel zur Anwendung der Funktionsweise ist [15] zu entnehmen.

4.3 Low-Cost Systeme

Die Funktionalitäten der High-Cost CAE-Werkzeugkette können auch durch Low-Cost Werkzeuge erreicht werden. Hierzu wird MATLAB/Simulink durch z.B. Scicoslab oder Scilab/Xcos als Open Source Simulationssoftware und die teure Echtzeithardware durch einen Mikrocontroller inkl. Betriebssystem als Hardware ersetzt. Dies ermöglicht eine Kostenersparnis bis um den Faktor 400 verglichen zur High-Cost RCP-Entwicklungsplattform [11].

Der am meisten verbreitete Ansatz basiert auf der Architektur nach Abbildung 3b und basiert auf einem modifiziertem Linux-Betriebssystem (RTAI) sowie verschiedenen Schnittstellenkarten als Echtzeithardware [16]. Dieses System nutzt Scilab/Scicos als Simulationsumgebung und weist softwaretechnische Schnittstellen für Mess- und Verstellaufgaben auf, welche durch verschiedene Linux-Anwendungen genutzt werden können (z.B. RTAILab [17]).

Ansätze zur Topologie nach Abbildung 3a unter Nutzung Scilab/Xcos werden z.B. durch die Kombination E4Coder/MicroDAQ, X2C mit einem STM32 Mikrocontroller oder der Xcos STM32 Toolbox verfolgt. Diese Werkzeuge sind einfach aufgebaut und konzentrieren sich auf die Programmierung der Zielhardware aus Scilab/Xcos und sind daher nicht ohne Weiteres für einen durchgängigen Prozess nutzbar.

Im Bereich kostengünstiger Hardware gibt es, wie bereits in Abschnitt 4.2 deutlich wird, seit geraumer Zeit diverse Alternativen. Diese reichen von einfachen Mikrocontrollern, wie dem Arduino UNO, über leistungsfähige Systeme mit vielen integrierten Schnittstellen, wie dem STM32F4, bis hin zu PowerPC-Strukturen wie dem Raspberry Pi.

Um mögliche Leistungseinschränkungen gegenüber herkömmlicher RCP-Hardware zu kompensieren, werden auch Ansätze zur auf mehreren Rechnerkernen verteilten Onlinesimulation verfolgt. Zu nennen sind hier z.B. das Projekt AMALTHEA, welches aus der Automobilindustrie stammt oder das aktuell laufende Projekt ARGO des DLR (siehe [18]).

5 Konzept der Low-Cost Plattform

Wie aus Kapitel 4 hervorgeht, existieren bereits einige Lösungsansätze zu Low-Cost RCP-Werkzeugen. Viele dieser Ansätze basieren jedoch auf der inzwischen veralteten Softwareumgebung Scicos und sind mit dem Nachfolger Xcos nicht kompatibel. Da keine aktive Weiterentwicklung bzw. Support erfolgt sind diese Lösungen nicht für den Einsatz in KMU geeignet.

Die für die aktuelle Low-Cost Simulationsumgebung Scilab/Xcos verfügbaren RCP-Werkzeuge konzentrieren sich auf die automatische Codegenerierung, weshalb solche Low-Cost-Kombinationen mit diversen Lücken behaftet sind (siehe Abbildung 4, Spalte „Low-Cost mit Lücken“). Aufgrund der vorhandenen Lücken sind zahlreiche manuelle Arbeiten, u. a. Handcodierung, notwendig. Dies führt zu hohem Zeitaufwand

und Programmieren als Fehlerquellen, was die Reproduzierbarkeit möglicher Fehler nicht gewährleistet und folglich deren Beseitigung erschwert. Das bedeutet einen langwierigen Iterationszyklus, da die Fehlersuche

einen hohen Zeitanteil bei der Entwicklung von Software einnimmt [19]. Daher können solche lückenhaften Low-Cost-Kombinationen nicht im Entwicklungsprozess eingesetzt werden.

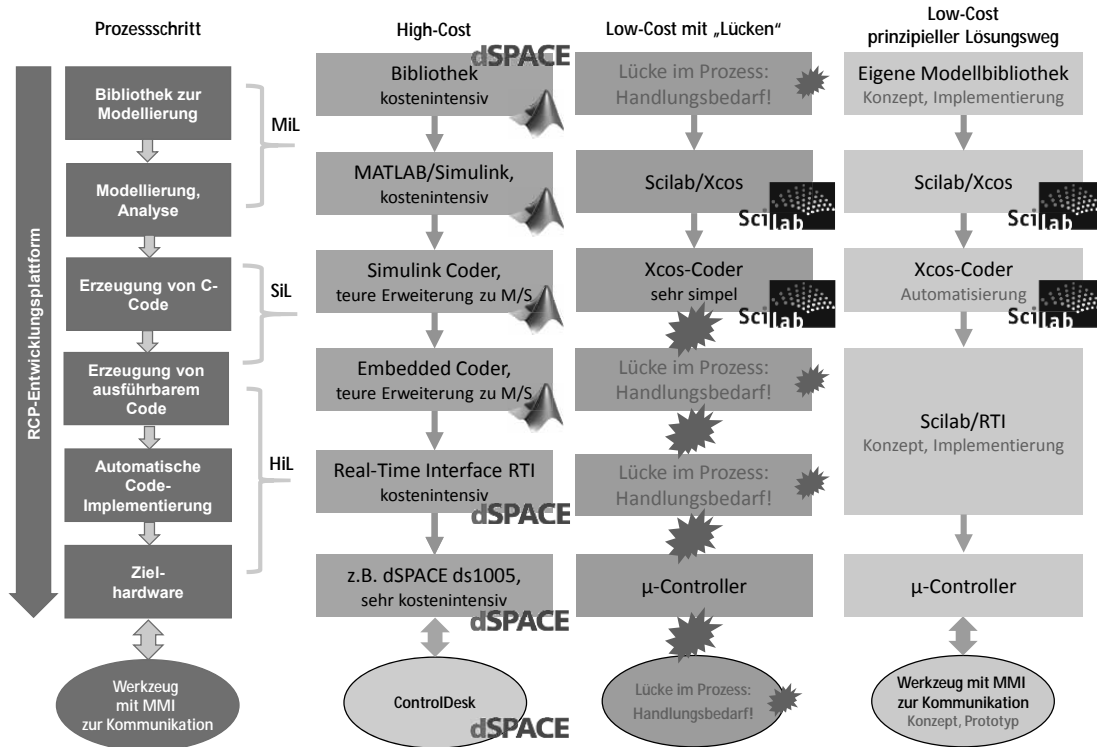


Abbildung 4. Prozess des RCP, Lücken in der Low-Cost Umgebung und prinzipiell verfolgter Lösungsweg

Der prinzipiell verfolgte Lösungsweg zur Realisierung eines durchgängigen Low-Cost RCP-Systems mit Open Source Plattform auf Basis von Scilab/Xcos ist in Abbildung 4 rechts illustriert. Aufgrund der Struktur von Scilab können neue Funktionen und Erweiterungen gut entwickelt und integriert werden, was eine Grundvoraussetzung für die Nutzung als Softwareplattform für das Low-Cost RCP-System darstellt.

Zunächst sind entsprechende Modellbibliotheken zu schaffen, die es ermöglichen Softwarefunktionen durchgängig in Scilab/Xcos zu entwickeln. Der integrierte Xcos-Coder ist sehr einfach aufgebaut und nicht dafür geeignet echtzeitfähigen Code zu generieren. Dieser muss verbessert oder durch einen Codegenerator mit entsprechenden Eigenschaften ersetzt werden. Eine Schnittstelle zur Echtzeithardware muss mit automatisiertem Kompilierungs- und Implementierungsprozess geschaffen werden. Letztendlich ist ein entsprechendes Hardwarekonzept zu wählen und ein Werkzeug mit MMI zur Durchführung von Onlinesimulationen zu schaffen.

6 Zusammenfassung und Ausblick

Der vorliegende Beitrag fokussiert das Systemkonzept eines Low-Cost RCP-Systems mit Open Source Plattform. Nach der Vorstellung des zugrundeliegenden Entwicklungsprozesses erfolgte eine allgemeine Analyse von RCP-Systemen. Hierzu wurden diese in die Bereiche Simulationssoftware, Echtzeithardware und Experimentiersoftware eingeteilt. In der anschließenden Darstellung zum Stand der Technik wurden vorhandene High-Cost-, Low-Cost- und Mischsysteme vorgestellt und analysiert. Abschließend erfolgte eine Analyse des aktuell realisierbaren Low-Cost RCP-Prozesses sowie die Vorstellung des durch die Ostfalia verfolgten Konzepts zur Entwicklung eines durchgängigen RCP-Systems.

Folgende Arbeiten konzentrieren sich auf die Schließung der Lücken im Low-Cost RCP-Prozess. Ein besonderes Augenmerk liegt hierbei auf dem Interface zwischen Onlinesimulation und Host-PC. Die Ergebnisse werden anhand des durchgängigen Anwendungsbeispiels eines Batteriemanagementsystems validiert.

7 Danksagung

Der vorliegende Beitrag wurde im Rahmen des Forschungsprojektes *LoCoRCP* von dem EFRE Fonds der Europäischen Union unter dem Kennzeichen ZW 6-85003460 gefördert. Die Verantwortung für den Inhalt liegt bei den Autoren. Für die Förderung bedanken sich diese herzlichst.



8 Referenzen

- [1] Abel, D. und Bollig, A.: *Rapid Control Prototyping. Methoden und Anwendungen*. Springer-Verlag, Berlin Heidelberg, 2006
- [2] Liu-Henke, X., Buchta, R. und Quantmeyer, F.: *Simulation eines mechatronischen Lenkungsmoduls für ein Elektrofahrzeug mit dezentralen Direktantrieben*. ASIM STS/GMMS Workshop 2011 - 'Simulation Technischer Systeme', Krefeld, 24.-25.02, 2011
- [3] Otterbach, R. und Schütte, F.: *Effiziente Funktions- und Software-Entwicklung für mechatronische Systeme im Automobil*. 2. Paderborner Workshop Intelligente Mechatronische Systeme, Paderborn, 25. und 26. März 2004
- [4] Broy, M. et al.: *Einfluss des Software-Designs auf die Wirtschaftlichkeit von Software-Entwicklungen*. ATZ Elektronik Nr. 2 2011, S. 34–37
- [5] Liu-Henke, X.: *Mechatronische Entwicklung der aktiven Feder-Neigetchnik für das Schienenfahrzeug RailCab*. Dissertation, Universität Paderborn 2005
- [6] Hanselmann, H.: *Beschleunigte Mechatronik-Entwicklung durch Rapid Control Prototyping und Hardware-in-the-Loop-Simulation*. at - Automatisierungstechnik Nr. 3 1998, S. 113–119
- [7] Hanselmann, H.: *Vom Modell zum Serienelement*. Elektronik automotive Nr. 3 2003
- [8] Lee, W., Shin, M. und Sunwoo, M.: *Target-identical rapid control prototyping platform for model-based engine control*. *Proceedings of the Institution of Mechanical Engineers, Journal of Automobile Engineering* Nr. 7 2004, S. 755–765
- [9] Liu-Henke, X et al.: *Investigation of low-cost open-source platforms for developing of mechatronic functions with rapid control prototyping*. 10th International Conference on MSM. Opole, Polen 2014
- [10] Köster, L., Thomsen, T. und Stracke, R.: *Von Simulink nach OSEK. Automatische Codegenerierung für Echtzeitbetriebssysteme mit Target-Link*. Embedded Intelligence. Nuremberg 2001
- [11] Liu-Henke, X., Feind, R. u. Roch, M.: *Machbarkeitsstudie zu einem Low-Cost RCP-System mit Open Source Plattform für die Entwicklung von Batterieregelsystemen*, Projektabschlussbericht., Wolfenbüttel, 2014
- [12] Tränkle, F.: *Eine Anleitung zur Entwicklung von Simulink-Targets für die Lehre*. Workshop Modellbasierte Entwicklung eingebetteter Systeme X., Dahstuhl, 2014
- [13] Zhong, Q.-C. et al.: *Low-cost Rapid Control Prototyping Paradigm*. UKACC International Conference on CONTROL. 2010, S. 1269–1273
- [14] The MathWorks: *Real-Time Windows Target. Run Simulink models in real time on Microsoft Windows PCs*, 2012
- [15] Xu, Z. und Tang, S.: *Design of Rapid Control Prototyping Platform Based on RTX*. ICCE. Melbourne, Australien, 2012, S. 427–434
- [16] Bucher, R. und Balemi, S.: *Scilab/Scicos and Linux RTAI - a unified approach*. IEEE Conference on Control Applications, Toronto, Kanada, 2005, S. 1121–1126
- [17] Serna, J. M. et al.: *Real-time control platform based on free software and USB communication*. CACSD, Buenos Aires, Argentinien, 2016, S. 845–849
- [18] Reder, S.: *The ARGO Approach: Parallelization Toolchain for Model-based Real Time Applications*. 12th HiPEAC conference, Stockholm, Schweden. 2017
- [19] Liu-Henke, X. u. Duym, S.: *Modellgestützte Funktionsabsicherung des vernetzten mechatronischen Kraftfahrzeugs*. VDI-Berichte 1892 "Mechatronik 2005 - Innovative Produktentwicklung", VDI-Verlag, Düsseldorf, 2005, S.1073-1090

Simulationsgestützte Konzeption der Antriebstopologie eines fahrerlosen Transportfahrzeugs

Xiaobo Liu-Henke¹, Sören Scherler, Marian Göllner, Johannes Maisik, Matthias Fritsch

Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau,

Institut für Mechatronik, Salzdhahmer Str. 46/48, 38302 Wolfenbüttel

x.liu-henke@ostfalia.de

Im vorliegenden Beitrag wird als Grundlage der weiteren Entwicklung eines fahrerlosen Transportsystems (FTF, engl. AGV) die Herleitung der kinematischen Beziehungen der Antriebe vorgestellt. Diese werden allgemeingültig hergeleitet, sodass eine Vielzahl verschiedener Antriebstopologien durch entsprechende Parametrierung erzeugt und untersucht werden können.

1 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 KMU bewertet werden sollen, um KMU bei der 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 Integration fahrerloser Transportfahrzeuge (FTF) in intelligente, cyber-physische Produktionsanlagen. Aufgabe eines FTF ist der autonome Gütertransport innerhalb des Produktionsumfeldes.

Im vorliegenden Beitrag wird die Modellbildung der Antriebstopologien für eine variierende Anzahl von Mecanum-Rädern in unterschiedlichen geometrischen Anordnungen dargestellt. Ziel ist die Herleitung symbolischer Modellgleichungen, mit welchen durch entsprechende Parametrierung eine Vielzahl von Varianten abgebildet werden können. Aufbauend darauf werden diese Antriebstopologien mithilfe verschiedener Simulationen bewertet und miteinander verglichen, um die für das beschriebene Projektvorhaben ideale Antriebstopologie auswählen zu können.

2 Stand des Wissens

Bei der Kraftübertragung des Antriebsmoments auf die Fahrbahn werden bei dem Großteil von Fahrerlosen Transportfahrzeugen herkömmliche Räder verwendet. Dabei werden hauptsächlich Vollgummireifen oder Kunststoffreifen und nur selten Luftreifen verwendet. Bei dieser Art von Reifen werden nicht nur Kräfte senkrecht zur Hauptachse des Rades, sondern auch Querkräfte, beispielweise bei Kurvenfahrten, übertragen. Dies hat zur Folge, dass diese Fahrzeuge nicht frei im Raum rangieren können und dass bei Kurvenfahrten je nach Größe des Rads und Bodenkontaktfläche immer ein entsprechender Kurvenradius zu berücksichtigen ist.

Je nach Anwendung lassen sich verschiedene Antriebstopologien realisieren. Die DIN 2510 zeigt beispielsweise Topologien mit drei Rädern, bei dem das angetriebene Rad drehbar ist, und mit vier Rädern, von denen zwei Räder sowohl für den Antrieb als auch mittels Differentiallenkung für die Lenkung sorgen und zwei eine stützende Funktion erfüllen.

Um omnidirektionale Fahrmanöver zu realisieren und aus beliebiger Orientierung in jede Raumrichtung zu manövrieren, wurden u.a. das sogenannte Allseitenrad und das Mecanum Rad (**Abbildung 1**) entwickelt. Letzteres wurde 1972 von seinem Erfinder Bengt Ilon als Patent angemeldet [1]. Bei diesen Rädern bestehen die Laufflächen aus Rollen, deren Drehachse bei einem Allseitenrad im rechten Winkel und bei einem Mecanum-Rad meistens in einem 45° Winkel zur Drehachse des Hauptrades liegt. Das gesamte Rad wird lediglich über einen Motor angetrieben, nicht die einzelnen Rollen. Der Kontakt zum Boden erfolgt jedoch ausschließlich über die Rollen.



Abbildung 1. Mecanum-Rad [2].

Mecanum-Räder können aufgrund der Anordnung der Rollen höhere Lasten als Allseitenräder tragen. Durch den Winkelversatz der Achsen zwischen der Hauptachse des Rades und der Drehachsen der einzelnen Rolle, der bei dem Mecanum-Rad i.d.R 45° beträgt, wird zusätzlich das Abrollen des Rades von einer Rolle zur nächsten fließender umgesetzt, was im Gegensatz zum Allseitenrad eine höhere Laufruhe zur Folge hat.

Fahrerlose Transportfahrzeuge mit Mecanum-Rädern werden mit einer unterschiedlichen Anzahl an Rädern realisiert. Das FTF der Fa. Hangfa verfügt über acht Mecanum-Räder [4], der youbot der Fa. Kuka hingegen über insgesamt vier Mecanum-Räder [5].

3 Methodik

Die durchgängig modellbasierte und verifikationsorientierte Funktionsauslegung und -absicherung (vgl. **Abbildung 2**) hat sich für die Entwicklung mechatronischer Systeme als zeit- und kosteneffizient erwiesen.

Ausgehend von Anforderungen und Lastenheft erfolgt die Model-in-the-Loop-Simulation (MiL), in welcher Regelalgorithmen ohne Anspruch auf Echtzeitfähigkeit entwickelt und an einem Fahrdynamikmodell erprobt werden. Aus den simulativ erprobten Regelalgorithmen wird im Rahmen der Software-in-the-Loop-Simulation (SiL) mittels automatischer Code-Generierung ein ausführbarer Target-Code erzeugt, welcher wiederum mit einem Fahrdynamikmodell auf Funktion und Fehler getestet wird. Daraufhin erfolgt die Hardware-in-the-Loop-Simulation (HiL), bei welcher ein um physische Teilkomponenten ergänztes, echtzeitfähiges Dynamikmodell zur Funktionsabsicherung und -optimierung verwendet wird.

Mithilfe dieser Methodik wird eine durchgängige Entwicklung basierend auf virtuellen Prototypen bis zum fertigen Produkt sichergestellt.

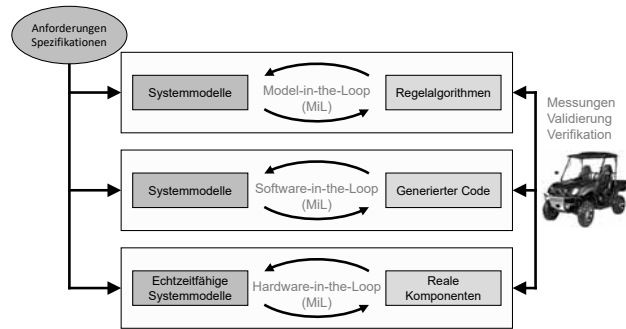


Abbildung 2. Durchgängig modellbasierte, verifikationsorientierte Entwicklungsmethodik nach [6].

Im Rahmen dieser Arbeit wird die Modellbildung der Antriebstopologie eines FTF als Grundlage dieses Entwicklungsprozesses durchgeführt. Der Modellbildungsprozess (vgl. **Abbildung 3**) 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.

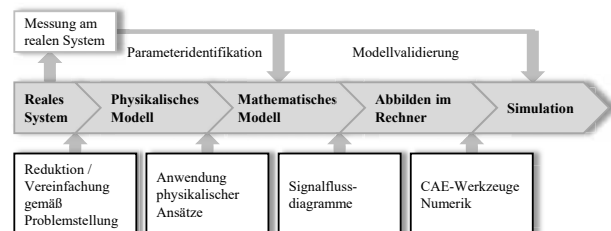


Abbildung 3. Modellbildungsprozess.

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.

4 Modellbildung

Ziel der Modellbildung ist zunächst die Herleitung eines allgemeingültigen Ansatzes zur Beschreibung der Kinematik eines FTF mit mehreren Mecanum-Rädern.

4.1 Mecanum-Rad

Abbildung 4 zeigt die kinematischen Beziehungen eines Mecanum-Rades angelehnt an [7]. Die Umfangsgeschwindigkeit

$$v = \omega \cdot r \tag{1}$$

ergibt sich aus der Winkelgeschwindigkeit ω und dem Radradius r .

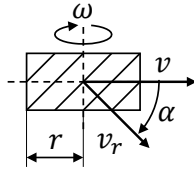


Abbildung 4. Kinematik eines Mecanum-Rades.

Zusätzlich ergibt sich aufgrund der im Winkel α zur Radachse versetzt stehenden Rollen eine zu den Rollen senkrechte Geschwindigkeitskomponente v_r .

$$v_r = \omega \cdot r \cdot \cos \alpha \quad (2)$$

Die orthogonale Querbewegung des Mecanum Rades ist durch dessen Rollen frei. Das heißt eine Drehung des Rades wird über die orthogonale mit einer Drehung der Rollen abgeleitet:

$$\omega \cdot r \cdot \sin \alpha - r_{SW} \cdot \dot{\varphi}_{SW} = 0 \quad (3)$$

Unter der Berücksichtigung dieser Eigenschaften kann die Lage und Orientierung des Rades mit den bekannten Axiomen der Kinematik beschrieben werden.

4.2 Kinematik eines FTF mit einem Mecanum-Rad

Um die kinematischen Beziehungen für eine beliebige Anzahl an Rädern in beliebiger Lage (nur beschränkt durch den Momentanpol) herleiten zu können, wird zunächst ein FTF mit einem Mecanum-Rad betrachtet (vgl. **Abbildung 5**).

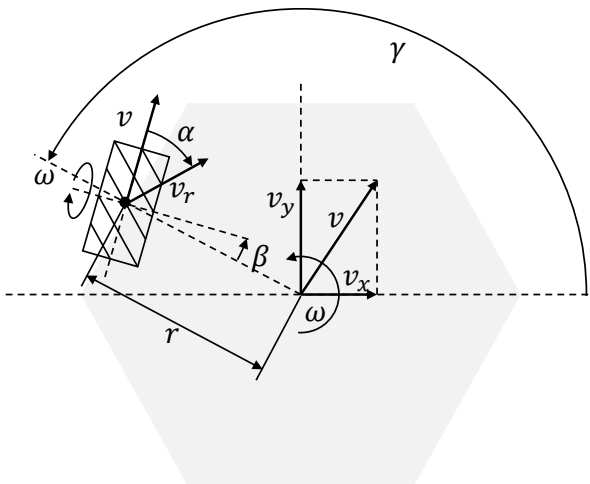


Abbildung 5. Kinematik eines FTF mit einem Mecanum-Rad.

Die Lage und Orientierung dieses FTF kann über zwei translatorische Freiheitsgrade in x- und y-Richtung und einen rotatorischen Freiheitsgrad um seine Hochachse definiert werden. Lage und Orientierung des Mecanum-Rades lassen sich nun durch die Winkel β

und γ sowie den Abstand r zum Schwerpunkt des FTF eindeutig bezogen auf das Koordinatensystem des FTF beschreiben.

Durch komponentenweise Betrachtung der Geschwindigkeiten in x- und y-Richtung kann eine mathematische Beschreibung der Kopplung zwischen Mecanum-Rad und FTF hergeleitet werden. In x-Richtung gilt

$$v_x - \sin(\gamma) \omega r = -\sin(\alpha - \beta - \gamma) v_r + \sin(\beta + \gamma) v \quad (4)$$

und in y-Richtung

$$v_y + \cos(\gamma) \omega r = -\cos(\alpha - \beta - \gamma) v_r - \cos(\beta + \gamma) v. \quad (5)$$

Durch Umstellen von Gleichung „2“ ergibt sich mit

$$v_r = \frac{-v_y - \cos(\gamma) \omega r - \cos(\beta + \gamma) v}{\cos(\alpha - \beta - \gamma)} \quad (6)$$

eine Gleichung zur Kopplung von „1“ und „2“. Durch Einsetzen von „3“ in „1“ und Äquivalenzumformungen ergibt sich die Umfangsgeschwindigkeit des Mecanum-Rades

$$v = \frac{\cos(\alpha - \beta - \gamma) v_x - \sin(\alpha - \beta - \gamma) v_y - \sin(\alpha - \beta) \omega r}{\sin(\alpha)} \quad (7)$$

in Abhängigkeit der Bewegung des FTF. In Matrixschreibweise ergibt sich

$$[v] = \begin{bmatrix} \frac{\cos(\alpha - \beta - \gamma)}{\sin(\alpha)} & \frac{-\sin(\alpha - \beta - \gamma)}{\sin(\alpha)} & \frac{-\sin(\alpha - \beta) r}{\sin(\alpha)} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix}. \quad (8)$$

4.3 Allgemeine Beschreibung eines AGV

Da die Lage der Winkel $\alpha_i, \beta_i, \gamma_i$ jedes Rades von $i = 1$ bis zum n -ten Rad durch den konstruktiven Aufbau bestimmt und fortan ortsfest ist, sollen diese durch die konstruktiven Ersatzwinkel

$$\delta_i = \alpha_i - \beta_i - \gamma_i \quad (9)$$

und

$$\varepsilon_i = \alpha_i - \beta_i \quad (10)$$

substituiert werden.

Für ein AGV mit mehreren Rädern kann basierend auf Gleichung 8 eine allgemeine Darstellungsform

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} \frac{\cos(\delta_1)}{\sin(\alpha_1)} & \frac{-\sin(\delta_1)}{\sin(\alpha_1)} & \frac{-\sin(\varepsilon_1) r_1}{\sin(\alpha_1)} \\ \frac{\cos(\delta_2)}{\sin(\alpha_2)} & \frac{-\sin(\delta_2)}{\sin(\alpha_2)} & \frac{-\sin(\varepsilon_2) r_2}{\sin(\alpha_2)} \\ \vdots & \vdots & \vdots \\ \frac{\cos(\delta_n)}{\sin(\alpha_n)} & \frac{-\sin(\delta_n)}{\sin(\alpha_n)} & \frac{-\sin(\varepsilon_n) r_n}{\sin(\alpha_n)} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (11)$$

gefunden werden. Durch Einsetzen der konstruktiven Winkel repräsentiert diese Matrix die Rückwärtskinematik des FTF. Durch Vorgabe der Geschwindigkeiten v_x, v_y in x- und y- Richtung des Ursprungskoordinatensystems und der Drehwinkelgeschwindigkeit ω können die notwendigen Radgeschwindigkeiten v_i berechnet werden.

4.4 Anordnung mit drei Mecanum-Rädern

Die in **Abbildung 6** gezeigte Antriebstopologie besteht aus drei Rädern die jeweils auf einem Radius in einem Winkel von 120° zueinander gleichmäßig verteilt sind. Dabei ist die Drehachse der einzelnen Räder jeweils zum Koordinatenmittelpunkt des FTF ausgerichtet.

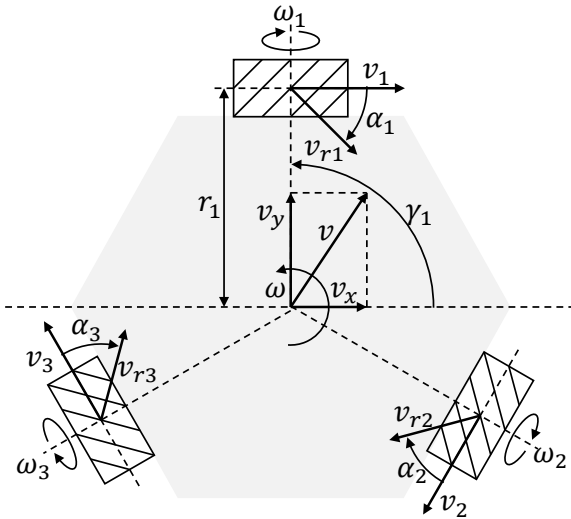


Abbildung 6. Exemplarische Kinematik mit drei Mecanum-Rädern.

Es ergeben sich die folgenden Winkel für die Lage und Orientierung der einzelnen Räder im Raum:

$$\alpha_i = 45^\circ \quad (12)$$

$$\delta_i = \alpha_i - (90^\circ + i \cdot 120^\circ) \quad (13)$$

$$\varepsilon_i = \alpha_i \quad (14)$$

Setzt man diese Winkel in die Gleichung 11 ein, so entsteht folgende Matrix für die Rückwärtskinematik dieses Systems:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \cos(\delta_1) & -\sin(\delta_1) & -\sin(\varepsilon_1)r_1 \\ \sin(\alpha_1) & \sin(\alpha_1) & \sin(\alpha_1) \\ \cos(\delta_2) & -\sin(\delta_2) & -\sin(\varepsilon_2)r_2 \\ \sin(\alpha_2) & \sin(\alpha_2) & \sin(\alpha_2) \\ \cos(\delta_3) & -\sin(\delta_3) & -\sin(\varepsilon_3)r_3 \\ \sin(\alpha_3) & \sin(\alpha_3) & \sin(\alpha_3) \end{bmatrix} R_z \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (15)$$

Nach Berechnung der Winkelfunktionen ergibt sich:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -r \\ -\frac{\sqrt{6}+\sqrt{2}}{\sqrt{2}} & -\frac{\sqrt{6}-\sqrt{2}}{\sqrt{2}} & -r \\ -\frac{\sqrt{6}-\sqrt{2}}{\sqrt{2}} & -\frac{\sqrt{6}+\sqrt{2}}{\sqrt{2}} & -r \end{bmatrix} R_z \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (16)$$

Diese Matrix kann als Konstant angesehen und fortan zur direkten Berechnung der Radgeschwindigkeiten verwendet werden.

4.5 Anordnung mit vier Mecanum-Rädern

Die kinematischen Beziehungen eines AGV mit vier symmetrisch angeordneten Mecanum-Rädern sind in **Abbildung 7** dargestellt.

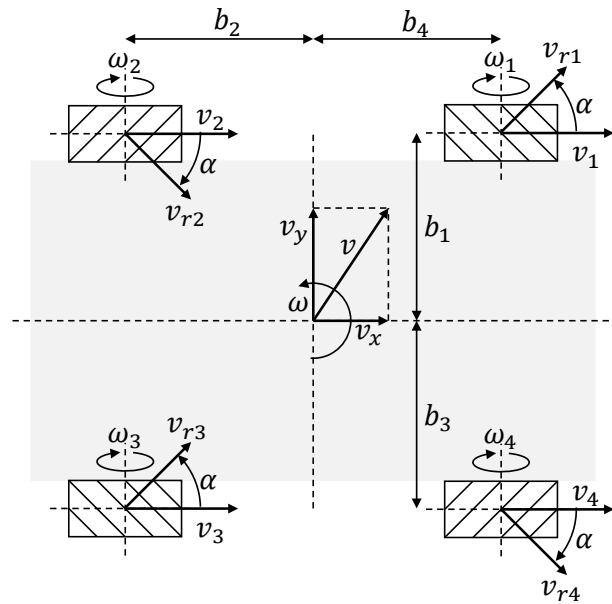


Abbildung 7. Kinematik mit vier Mecanum-Rädern.

Folgende Winkel und Radien beschreiben die Lage und Orientierung der einzelnen Räder in Polarkoordinaten ausgehend vom Koordinatenursprung:

$$\alpha_i = (-1)^i \cdot 45^\circ \quad (17)$$

$$\beta_i = \arcsin\left(\frac{b_i}{r_i}\right) - (i-1) \cdot 90^\circ \quad (18)$$

$$\gamma_i = \arcsin\left(\frac{b_i}{r_i}\right) + (i-1) \cdot 90^\circ \quad (19)$$

$$r_1 = \sqrt{b_i^2 + b_n^2}, r_{i>1} = \sqrt{b_{i-1}^2 + b_i^2} \quad (20)$$

Diese werden in die Gleichung 11 eingesetzt:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} \cos(\delta_1) & -\sin(\delta_1) & -\sin(\varepsilon_1)r_1 \\ \sin(\alpha_1) & \sin(\alpha_1) & \sin(\alpha_1) \\ \cos(\delta_2) & -\sin(\delta_2) & -\sin(\varepsilon_2)r_2 \\ \sin(\alpha_2) & \sin(\alpha_2) & \sin(\alpha_2) \\ \cos(\delta_3) & -\sin(\delta_3) & -\sin(\varepsilon_3)r_3 \\ \sin(\alpha_3) & \sin(\alpha_3) & \sin(\alpha_3) \\ \cos(\delta_4) & -\sin(\delta_4) & -\sin(\varepsilon_4)r_4 \\ \sin(\alpha_4) & \sin(\alpha_4) & \sin(\alpha_4) \end{bmatrix} R_z \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (21)$$

Ein Sonderfall dieser Topologie ist die Anordnung aller Räder auf einem gemeinsamen Radius und in gleichen Abständen, so dass:

$$b_1 = b_2 = b_3 = b_4 = b \quad (22)$$

Dadurch entstehen die folgenden expliziten Winkel und Radien:

$$\alpha_i = (-1)^i \cdot -45^\circ \quad (23)$$

$$\beta_i = -45^\circ \quad (24)$$

$$\gamma_i = 45^\circ + (i - 1) \cdot 90^\circ \quad (25)$$

$$r_i = b \cdot \sqrt{2} \quad (26)$$

Aus diesen wird die konstante Matrix zur Berechnung der Rückwärtskinematik gebildet:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} 1 & -1 & -b \\ 1 & 1 & -b \\ 1 & -1 & b \\ 1 & 1 & b \end{bmatrix} R_z \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (27)$$

Die gefundene Matrix kann nur zur kinematischen Analyse herangezogen werden.

5 Modellparametrierung

Zur Parametrierung der in Polarkoordinaten vorliegenden Kinematik Matrize wird nachfolgend ein Algorithmus beschrieben, mit dem die Lage des FTF mithilfe von kartesischen Koordinaten und der gewünschten Winkelorientierung festgelegt werden kann. Gesucht werden die Winkel β und γ sowie der Abstand r .

Zunächst wird der Radius r mithilfe der Abstände a und b zwischen Schwerpunkt des FTF und Schwerpunkt des Mecanum-Rades definiert (vgl. **Abbildung 8**). Mithilfe des Satzes des Pythagoras ergibt sich der Radius

$$r = \sqrt{a^2 + b^2}. \quad (28)$$

Ebenfalls ergibt sich aus den beiden Abständen der Winkel

$$\gamma = \arctan\left(\frac{b}{a}\right). \quad (29)$$

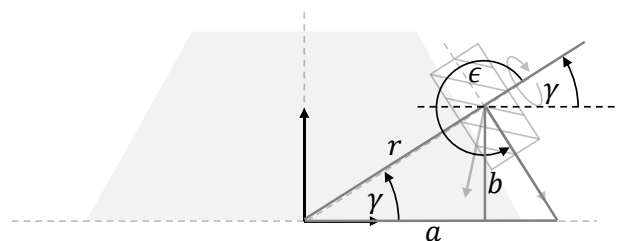


Abbildung 8. Schaubild zur Berechnung der Radposition und Orientierung.

Unter der Annahme, dass der Winkel β null beträgt, ergibt sich die Ausrichtung des rechtwinklig zu r stehenden Mecanum-Rades zu

$$\epsilon = \gamma - 90^\circ, \quad (30)$$

woraus der Winkel

$$\beta = \epsilon_{des} - \epsilon = \epsilon_{des} - \gamma + 90^\circ \quad (31)$$

mithilfe der Wunschausrichtung ϵ_{des} berechnet werden kann. Der Winkel α kann nachträglich definiert werden und beschreibt lediglich die Richtung der Querkraft v_r .

6 Verifikation/Untersuchung

Es wird eine Topologie mit vier Mecanum-Rädern wie in **Abbildung 7** betrachtet, um zum einen den Algorithmus zur Parametrierung zu demonstrieren und zum anderen eine Verifikation durchzuführen. Die Abstände b_1 und b_3 betragen 1 m , die Abstände b_2 und b_4 betragen 2 m . Die weiteren Parameter werden nach Kap. 5 berechnet und in **Tabelle 1** zusammengefasst.

	Rad 1	Rad 2	Rad 3	Rad 4
α	-45°	45°	-45°	45°
r	$2,24\text{ m}$	$2,24\text{ m}$	$2,24\text{ m}$	$2,24\text{ m}$
γ	$26,6^\circ$	$153,4^\circ$	$206,6^\circ$	$333,4^\circ$
ϵ_{des}	0°	0°	0°	0°
β	$63,4^\circ$	$296,6^\circ$	$243,4^\circ$	$116,6^\circ$

Tabelle 1. Parametrierungsbeispiel für ein AGV mit vier Mecanum-Rädern.

Die Ergebnisse der Verifikation werden in **Tabelle 2** dargestellt. In der ersten Spalte ist jeweils der vorgegebene Vektor der AGV-Bewegung

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \quad (32)$$

und in den Rad-Spalten die entsprechende Radgeschwindigkeit v eingetragen.

	Rad 1	Rad 2	Rad 3	Rad 4
$\begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix}$	$5 \frac{m}{s}$	$5 \frac{m}{s}$	$5 \frac{m}{s}$	$5 \frac{m}{s}$
$\begin{bmatrix} 0 \\ 5 \\ 0 \end{bmatrix}$	$-5 \frac{m}{s}$	$5 \frac{m}{s}$	$-5 \frac{m}{s}$	$5 \frac{m}{s}$
$\begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}$	$-15 \frac{m}{s}$	$-15 \frac{m}{s}$	$15 \frac{m}{s}$	$15 \frac{m}{s}$

Tabelle 2. Verifikationsergebnisse.

Erkennbar sind kinematisch plausible Ergebnisse: Im ersten Szenario, einer Fahrt des FTF in x-Richtung, zeigen alle Radumfangsgeschwindigkeit definitionsgemäß in die gleiche Richtung und sind positiv. Im zweiten Szenario, einer Fahrt in y-Richtung, ist erkennbar, dass die Räder alternierende Vorzeichen aufweisen. Dies ist auf die unterschiedlichen Winkel α zurückzuführen, aus welchen unterschiedliche Querkraftübertragungsrichtungen resultieren. Das dritte Szenario zeigt eine Drehung des FTF um seine Hochachse. Die Räder „oberhalb“ des Schwerpunkts drehen entgegen ihrer definierten Richtung, die Räder „unterhalb“ des Schwerpunkts in ihre definierte Richtung. Diese einfachen Szenarien verifizieren die hergeleitete Kinematik für das FTF mit vier Mecanum-Rädern

7 Zusammenfassung und Ausblick

Im vorliegenden Beitrag wurde in allgemeiner Form die Kinematikberechnung für FTF in beliebiger Konfiguration bzgl. Anzahl und Anordnung der antreibenden Mecanum-Räder hergeleitet. Diese Berechnungsmethodik soll zunächst im Rahmen der modellbasierten Reglerauslegung als Grundlage für die Berechnung der Dynamik des Gesamtfahrzeugs FTF genutzt werden. Weiterhin können die kinematischen Zusammenhänge als Rückwärtskinematik für die Verkopplung der Stellgrößen des zu entwickelnden Regelalgorithmus herangezogen werden.

8 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



9 Literatur

- [1] B. E. Ilon. *Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base*. US Patents and Trademarks office, Patent 3,876,255, 1975
- [2] Imetron. *Bewegung wie auf Luftkissen bei voller Bodenhaftung*. <https://www.donkey-motion.de/donkeymotion/mecanum>, Zugriff am 11.01.2018
- [3] T. Rennekamp. *Analyse und Kompensation von Fahrfehlern bei Mecanum-Wheel Fahrzeugen*. Autonome Mobile Systeme 2003, Karlsruhe, 4. - 5. Dezember, 2003
- [4] Chengdu Hangfa Hydraulic Engineering Co. LTD. Navigator, Compass and Discovery series Robots, <http://www.hangfa.com/EN/robot.html>, Zugriff am 11.01.2018
- [5] Kuka Roboter GmbH. *KUKA omniMove Antriebstechnologie*. <https://www.kuka.com/de/technologien/kuka-omnimove-antriebstechnologie>, Zugriff am 11.01.2018
- [6] X. Liu-Henke, S. Duym. *Modellgestützte Funktionsabsicherung des vernetzten mechatronischen Kraftfahrzeugs*. VDI-Tagung Mechatronik 2005, Wiesloch, 01. - 02. Juli, 2005.
- [7] E. Maulana, M. A. Muslim, V. Hendrayawan. *Inverse Kinematic Implementation of Four-Wheels Mecanum Drive Mobile Robot Using Stepper Motors*. International Seminar on Intelligent Technology and Its Applications (ISITIA 2015), Surabaya, Indonesia, May 20-21, 2015.

Gesamtkonzept des Forschungsprojekts iREX 4.0

Sören Scherler¹, Xiaobo Liu-Henke²

Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau,

Institut für Mechatronik, Salzdhahmer Str. 46/48, 38302 Wolfenbüttel

¹so.scherler@ostfalia.de, ²x.liu-henke@ostfalia.de

Ziel des Projekts *Intelligente Elektrofahrzeuge mit Range-Extender in Verkehrssystemen mit Fahrzeug 4.0 (iREX 4.0)* ist die Entwicklung eines prädiktiven, elektronischen Fahrzeugmanagements, welches den autonomen und energieoptimierten Fahrbetrieb eines Elektrofahrzeugs mit zwei Energiequellen (Lithium-Ionen-Batterie und Brennstoffzelle) ermöglicht. Grundlage des Entwicklungsprozesses ist die Modularisierung und Hierarchisierung des Gesamtsystems zur Beherrschung der Systemkomplexität und eine durchgängig modellbasierte, mechatronische Entwicklungsmethodik ergänzt um eine flexibel konfigurierbare Prüfinfrastruktur.

1 Einleitung

An der Ostfalia Hochschule wird vom 01.10.2016 bis zum 30.09.2020 das vom MWK Niedersachsen und der Volkswagenstiftung geförderte transdisziplinäre Verbundprojekt *Zukünftige Fahrzeugtechnologien im Open Region Lab (ZuFOR)* durchgeführt.

Das Verbundprojekt umfasst ein integratives Querschnittsprojekt, welches den Wissensaustausch zwischen dem Verbund und Unternehmen, Forschungseinrichtungen sowie Netzwerken im Rahmen eines Open Region Labs fördern soll. Weiterhin umfasst es drei technische Teilprojekte mit Schwerpunkten auf innovativen Fahrzeugtechnologien der aktiven Sicherheitssysteme für PKW, der leichten Kunststoffstrukturen im Automobilbau sowie der intelligenten und vernetzten Elektrofahrzeuge mit Range Extender.

Im Rahmen dieses Beitrags soll das Gesamtkonzept des Teilprojekts *Intelligente Elektrofahrzeuge mit Range-Extender in Verkehrssystemen mit Fahrzeug 4.0 (iREX 4.0)* dargestellt werden. Ziel ist die Entwicklung eines prädiktiven, elektronischen Fahrzeugmanagements, welches den autonomen, energieoptimierten Fahrbetrieb ermöglicht.

2 Methodik

Aufgrund der steigenden Anforderungen und Komplexität fahrzeugtechnischer Systeme ist die Verwendung einer klar strukturierten Entwicklungsmethodik unerlässlich.

Die verwendete Methodik basiert auf der mechatronischen Strukturierung mittels Modularisierung und

Hierarchisierung im Top-Down-Verfahren. Das komplexe Gesamtsystem wird in intelligente, gekapselte Teilsysteme bestehend aus mechatronischen Komponenten mit definierten Schnittstellen geteilt und hierarchisch strukturiert. **Abbildung 1** zeigt exemplarisch die hieraus resultierende mechatronische Struktur des Forschungselektrofahrzeugs Fredy (**F**unktions-träger für **r**enerative **E**lektromobilität und **F**ahrd**y**namikregelsysteme, vgl. Kap. 4.2) mit den vier Hierarchieebenen der Mechatronischen Funktionsmodule (MFM), der Mechatronischen Funktionsgruppen (MFG), der Autonomen Mechatronischen Systeme (AMS) und der Vernetzten Mechatronischen Systeme (VMS) nach [1].

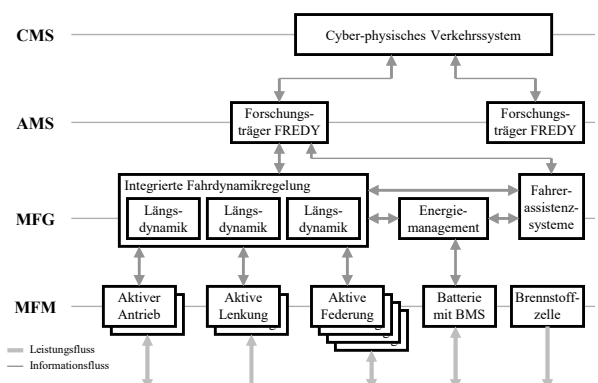


Abbildung 1. Mechatronische Strukturierung des FREDY.

Das MFM stellt die niedrigste Hierarchieebene dar. Es handelt sich um ein nicht weiter teilbares mechatronisches System bestehend aus einer mechanischen Tragstruktur, Sensorik, Aktorik und Informationsverarbeitung. Dieses gekapselte Modul erfüllt eine definierte Funktionalität, bildet das dynamische System-

verhalten ab und repräsentiert Kinematik, Dynamik und mechatronische Funktionen. Durch die mechanische oder informationstechnische Kopplung mehrerer MFM ergeben sich MFG mit eigener Informationsverarbeitung, welche wiederum zu AMS kombiniert werden können. Die Vernetzung mehrere AMS zu VMS stellt die höchste Hierarchieebene dar. Bei einem (autonomen) Fahrzeug handelt es sich um ein AMS, welches durch Digitalisierung und das Internet der Dinge mit anderen Fahrzeugen zu einem VMS vernetzt wird. Die VMS stellen die Grundlage des autonomen Fahrens dar.

Nach der mechatronischen Strukturierung des Gesamtsystems erfolgt die mechatronische Komposition. Die Informationsverarbeitung wird im Bottom-Up-Verfahren ausgelegt. Im folgenden Abschnitt wird die Konzeption dieser hierarchischen Informationsverarbeitung beschrieben.

3 Konzeption

3.1 Problemstellung

Die geringe Reichweite ist neben dem hohen Anschaffungspreis und der nicht flächendeckenden Ladeinfrastruktur ein wesentlicher Grund für die geringe Kundenakzeptanz von Elektrofahrzeugen [2]. Das Reichweitenproblem kann durch Verbrennungsmotoren mit einem Generator oder Brennstoffzellen gelöst werden, welche zugeführten Kraftstoff in elektrische Energie wandeln [3]. Aus ökologischer Sicht ist hierbei die Brennstoffzelle dem Verbrennungsmotor vorzuziehen, da diese, je nach Art der Stromerzeugung, zumindest lokal emissionsfrei betrieben werden kann.

Aus den beiden Energiequellen Batterie und Brennstoffzelle mit ihren unterschiedlichen Wirkungsgradcharakteristika ergibt sich ein Optimierungsproblem, da der Gesamtwirkungsgrad der Energiebereitstellung und somit auch die resultierende Reichweite von der Leistungsaufteilung zwischen den beiden Quellen abhängig sind. Mithilfe prädizierter Daten und Informationen aus Car-2-X-Kommunikation kann die Leistungsverteilung vorausschauend erfolgen und Restriktionen wie verfügbare Kraftstoffmenge oder Lade- und Tankmöglichkeiten berücksichtigen.

Ein weiteres Optimierungspotenzial ergibt sich in der Fahrzeugführung: Mithilfe der prädizierten Daten kann eine energie-, infrastruktur- und verkehrsoptimierte Route gewählt werden, aus welcher ein energieoptimales Geschwindigkeitsprofil gebildet wird. Zur Realisierung dieses Geschwindigkeitsprofils wird

ein autonomer Fahrbetrieb benötigt, da ein menschlicher Fahrer nicht in der Lage ist, ein Fahrzeug exakt nach einer Vorgabe im Wirkungsoptimum zu führen.

3.2 Fahrzeugkonfiguration

Das in diesem Projekt zur exemplarischen Lösung der Problemstellung fokussierte Fahrzeugkonzept (vgl. **Abbildung 2**) sieht ein Elektrofahrzeug mit zwei radnahen, elektrischen Antrieben an der Hinterachse vor, welche mechanisch durch je ein Planetengetriebe mit den Rädern verbunden sind. Jeder Antrieb verfügt über eine bidirektionale Leistungselektronik, die je nach Lastfall als Gleich- oder Wechselrichter fungiert und so neben der Versorgung des Antriebs auch die Stromrückspeisung in den Zwischenkreis ermöglicht.

Der Zwischenkreis kann entweder durch einen unidirektionalen Gleichspannungswandler von der Brennstoffzelle oder durch einen bidirektionalen Gleichspannungswandler von der Hochvolt-Batterie gespeist werden. Mithilfe des bidirektionalen Gleichspannungswandler kann in den Zwischenkreis gespeiste Energie bei Bedarf in das HV-Bordnetz transportiert werden. Das Niedervolt-Bordnetz zur Versorgung leistungsschwacher Komponenten mit eigener Batterie wird aus dem HV-Bordnetz mit Energie gestützt.

Eine konventionelle hydraulische Bremsanlage gewährleistet ein sicheres Bremsen, wenn das rekuperative Bremsen mit den elektrischen Antrieben aufgrund zu kleiner Geschwindigkeiten oder der Maschinengrenzen nicht ausreicht. Mithilfe radindividueller Steer-by-Wire-Lenkmodulen an der Vorderachse können neben herkömmlichen Lenkmanövern bspw. aktive Spurwinkelverstellungen vorgenommen und somit das Fahrverhalten sowie Energieverbrauch situationsabhängig verbessert werden.

Zur Realisierung wird eine zentrale Informationsverarbeitung benötigt, welche Sensor- und Car-2-X-Daten verarbeitet und Sollwerte für die Aktorik des Fahrzeugs generiert. Bei der autonomen Fahrt übernimmt die sie die Aufgaben des Fahrers und gewährleistet die sichere Quer- und Längsführung.

3.3 Anforderungen

Aus dem Fahrzeugkonzept und der geschilderten Problemstellung ergeben sich folgende Anforderungen an die zu entwickelnde Informationsverarbeitung:

- Für einen energieoptimalen Fahrbetrieb müssen die Zielführung optimiert und ein ideales Geschwindigkeitsprofil erzeugt werden.

- Zur Einregelung dieses Geschwindigkeitsprofils muss das Fahrzeug autonom agieren können, da ein menschlicher Fahrer einer Vorgabe nicht exakt folgen kann.
- Für den autonomen Fahrbetrieb müssen Informationen über den aktuellen Fahrzustand, die vorausliegende Strecke, die Fahrumgebung und andere Verkehrsteilnehmer verfügbar sein.
- Basierend auf der Zielführung müssen Sollwerte für die Längs- und Querverführung generiert werden, um das Fahrzeug sicher zu führen.
- Eine Fahrdynamikregelung muss die Längs-, Quer- und Vertikaldynamik regeln.
- Ein Algorithmus zur Leistungsaufteilung zwischen Batterie und Brennstoffzelle muss ein Wirkungsgradoptimum erzielen, um den energieoptimalen Fahrbetrieb zu ermöglichen.
- Der Leistungsfluss im Antriebsstrang und Bordnetz muss gesteuert werden können, um eine stabile Bordnetzspannung zu gewährleisten und Komponenten wie die Batterie vor unzulässigen Zuständen zu schützen.

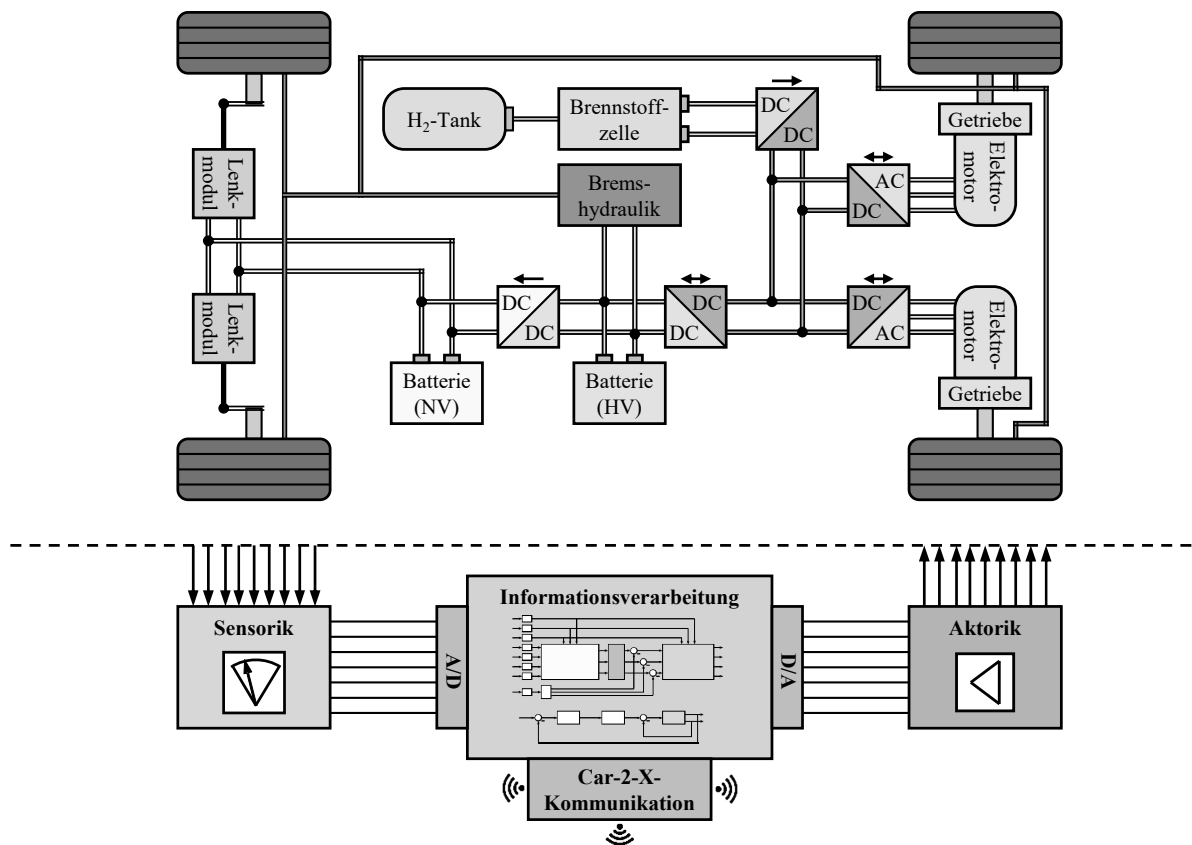


Abbildung 2. Fahrzeugkonfiguration.

3.4 Struktur der Informationsverarbeitung

Gemäß mechatronischer Entwicklungsmethodik (vgl. Kap. 2) wird mithilfe der Anforderungen (Kap. 3.3) die Struktur der hierarchischen Informationsverarbeitung hergeleitet (vgl. **Abbildung 3**).

Kern ist das prädiktive, elektronische Fahrzeugmanagement (peFZM), welches die höchste Instanz im Fahrzeug darstellt. Es ist Teil eines cyber-physischen Verkehrssystems, in welchem das Fahrzeug mit anderen Verkehrsteilnehmern und der Infrastruktur mittels drahtloser Kommunikation (z.B. in ad-hoc-

Netzwerken nach IEEE 801.11p oder Mobilfunk) vernetzt ist und Informationen austauscht. Das peFZM übernimmt die Aufgabe des menschlichen Fahrers und plant basierend auf den verfügbaren Informationen über Fahrzustand und Umwelt die Zielführung. Die Zielführung basierend auf erweiterten Kartendaten soll den Energieverbrauch, die verfügbare Ladeinfrastruktur sowie das Verkehrsgeschehen berücksichtigen, um eine optimale Route auszuwählen und ein Geschwindigkeitsprofil zu planen. Aus der Zielführung werden Sollwerte für Längs- und Querverführung abgeleitet.

Diese Sollwerte werden zum einen dem unterlagerten Fahrwerksmanagement, welches für die Fahrdynamikregelung und Stellgrößenverteilung sorgt sowie weitere Fahrwerksassistentenfunktionen umfasst, und zum anderen dem prädiktiven Energiemanagement übergeben. Das prädiktive Energiemanagement wiederum besteht aus Energieassistentenfunktionen zur Prädiktion zukünftiger Bordnetzzustände und Leistungsbedarfe, dem thermischen Energiemanagement zur Steuerung und Optimierung von Wärmeströmen

und dem elektrischen Energiemanagement mit unterlagerten Funktionen wie dem Batterie-, Brennstoffzellen-, Antriebs- oder Bordnetzmanagement. Diese unterlagerten Funktionen dienen dem Monitoring und der Sicherheitsüberwachung sowie der Betriebsführung der jeweiligen Aggregate auf lokaler Ebene. Das elektrische Energiemanagement umfasst zudem die Leistungsaufteilung zwischen Batterie und Brennstoffzelle zur Erhöhung des Wirkungsgrads im Antriebsstrang.

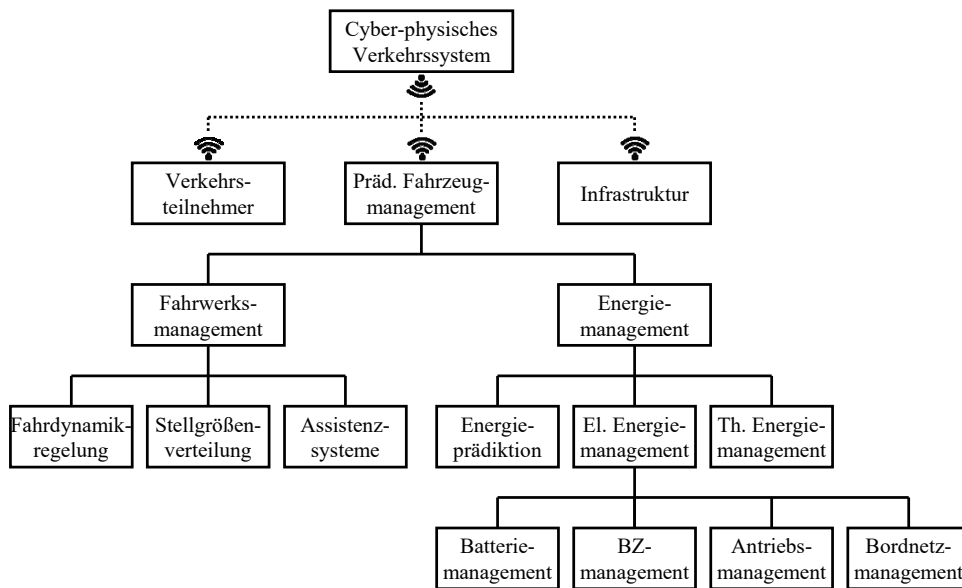


Abbildung 3. Hierarchische Struktur des prädiktiven Fahrzeugmanagements.

4 Realisierung

4.1 Flexibel konfigurierbarer HiL-Prüfstand

Zur modellbasierten, durchgängigen Entwicklung und Absicherung wird ein flexibel konfigurierbarer HiL-Prüfstand entworfen. Durch den modularen Aufbau können verschiedene Anwendungsfälle von der Parameteridentifikation und Validierung von Simulationsmodellen bis zur Erprobung, Verifikation und Optimierung entwickelter Funktionen abgedeckt werden. Grundlage des Prüfstands ist die in Kap. 3.2 beschriebene Fahrzeugkonfiguration.

Der Prüfstand (Abbildung 4) besteht aus einem Echtzeitmodul (EZM), einem Softwaremodul (SWM) und mehreren modularen Prüfstandsmodulen (PSM). Das Echtzeitmodul beinhaltet die Aufbereitung und Digitalisierung der Messsignale der PSM, die Berechnung der Regelalgorithmen auf einem digitalen Signalprozessor (DSP), die Vorgabe von Sollwerten für die PSM-Aktorik sowie die Kommunikation mit

den PSM. Das Softwaremodul umfasst Modelle nicht real vorhandener Systembestandteile, Regelalgorithmen sowie die Schnittstelle zwischen Nutzer und Prüfstand zur Zustandsvisualisierung und Prüfstandsteuerung. Die Modelle und Regler werden mittels automatisierter Codegenerierung auf den DSP geladen. Bei den PSM handelt es sich um ein Brennstoffzellenmodul, ein Batteriemodul, ein Antriebsmodul sowie ein Last- und Quellenmodul, welche gekapselt betrieben die folgenden Anwendungsszenarien ermöglichen:

- **Brennstoffzellenmodul:** Untersuchung von Brennstoffzellen zur Parameteridentifikation.
- **Batteriemodul:** Untersuchung von Batteriezellen und -systemen sowie Erprobung und Optimierung von Batteriemagementsystemen unter definierten klimatischen Bedingungen sowohl durch Gleichstrommessungen als auch Elektrochemische Impedanzspektroskopie.

- **Antriebsmodul:** Untersuchung von elektrischen Antrieben und Leistungselektronik sowie Erprobung und Optimierung von Antriebsregelungen.
- **Last- und Quellenmodul:** Dieses Hilfsmodul liefert regelbare Versorgungsspannungen und -ströme sowie regelbare Lastströme, um Komponenten des Bordnetzes zu simulieren.
- **Batterie-, Brennstoffzellen- und Last-/Quellenmodul:** Untersuchung von Regelkonzepten zur Leistungsaufteilung zwischen Batterie und Brennstoffzelle.
- **Batterie-, Antriebs- und Last/Quellenmodul:** Erprobung von Algorithmen zur Rekuperation.
- **Brennstoffzellen-, Batterie-, Antriebs- und Last-/Quellenmodul:** Untersuchung des Antriebsstrangs bestehend aus Brennstoffzelle, Batterie, Antrieb und Quelle sowie Last, die weitere Bordnetzkomponenten repräsentieren.

Durch entsprechende Schnittstellen kann im gekoppelten Betrieb mehrerer Prüfstandsmodule eine Vielzahl weiterer Anwendungsszenarien abgedeckt werden:

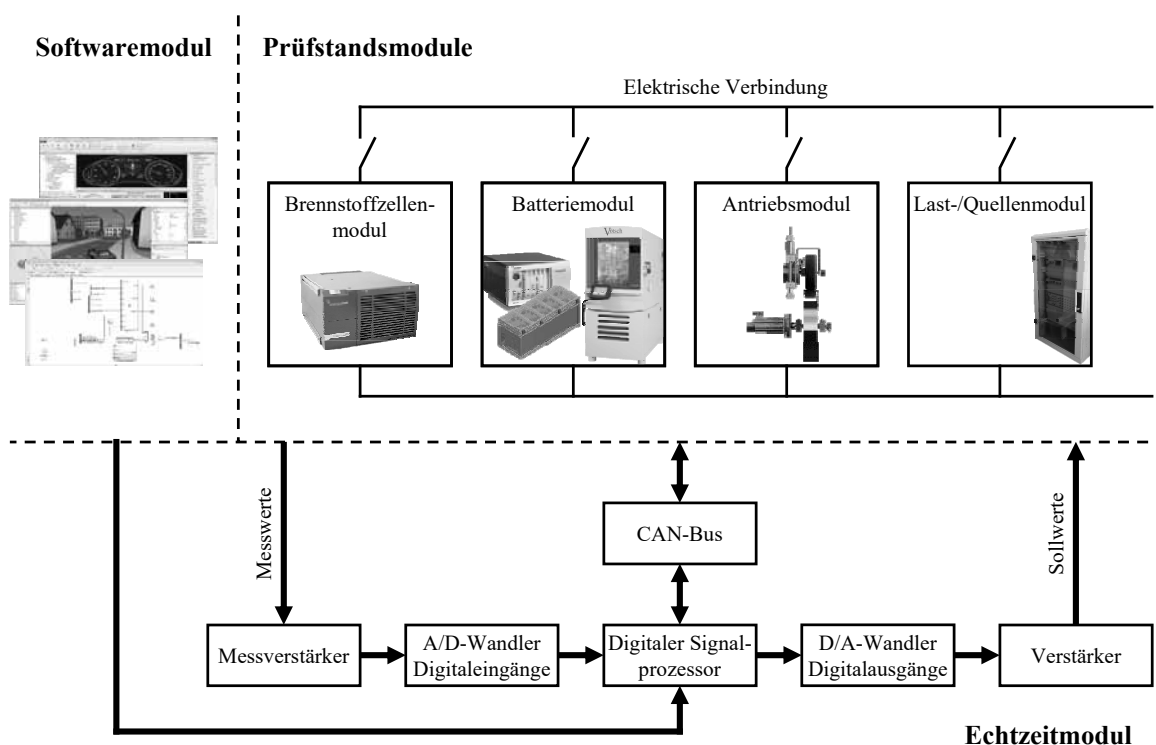


Abbildung 4. Konzept des flexiblen HiL-Prüfstands.

4.2 Funktionsträger FREDY

Der Funktionsträger für regenerative Elektromobilität und Fahrdynamikregelsysteme (FREDY) dient der Validierung und Optimierung von Fahrzeugfunktionen in der Realität, nachdem sie in Simulationen und Prüfstandsversuchen hinreichend untersucht und optimiert wurden.

Es handelt sich um ein Elektrofahrzeug, welches dem Fahrzeugkonzept aus **Abbildung 2** bis auf die Brennstoffzelle, welche noch nachgerüstet wird, entspricht. Der FREDY (**Abbildung 5**) verfügt über zwei radnahe, elektrische Antriebsmaschinen, welche mithilfe je eines Planetengetriebes mechanisch mit den Rädern verbunden sind. Die Antriebsmaschinen werden mithilfe bidirektionaler Leistungselektroniken aus zwei

Lithium-Ionen-Batteriepaketen mit Leistung versorgt. Eine MicroAutoBox der Fa. dSPACE stellt die zentrale Informationsverarbeitung dar und erfüllt alle Mess- und Regelaufgaben.

In der zukünftigen Arbeit wird der FREDY um intelligente Hard- und Software zur Digitalisierung und Vernetzung erweitert, sodass der FREDY mit anderen Verkehrsteilnehmern und der Infrastruktur ein cyberphysisches Verkehrssystem bildet. Somit wird die Realisierung von Car-2-Car-, Car-2-Infrastructure- und anderen Car-2-X-Anwendung ermöglicht.

Weitere Informationen zum FREDY können [4], [5] oder [6] entnommen werden.

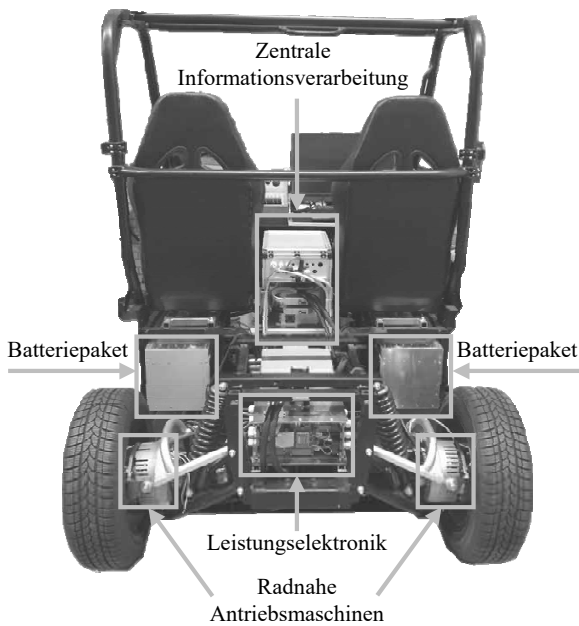


Abbildung 5. Aktueller Ausbaustand des FREDY.

5 Resümee und Ausblick

Der vorliegende Beitrag zeigt das Gesamtkonzept des Forschungsprojekts iREX 4.0. Nach einer Darstellung der verwendeten, mechatronischen Entwicklungsmethodik erfolgte eine Darstellung der Problemstellung, aus der sich unter Berücksichtigung der Fahrzeugkonfiguration Anforderungen an die hierarchische Informationsverarbeitung ergeben, deren Struktur und Funktionsweise dargestellt wurde. Daraufhin wurden ein flexibel konfigurierbares HiL-Prüfsystem zur Systemidentifikation und Funktionsverifikation sowie das Forschungselektrofahrzeug FREDY vorgestellt.

Die nächsten Arbeitsschritte umfassen die sukzessive Entwicklung der Teilfunktionen der hierarchischen Informationsverarbeitung sowie den parallelen Aufbau des flexibel konfigurierbaren HiL-Prüfstands zur Systemidentifikation und Funktionsverifikation und -validierung. Weiterhin fließen die Projektergebnisse direkt in die Weiterentwicklung des Funktionsträgers FREDY ein.

6 Danksagung

Dieser Beitrag wurde im Rahmen des Projekts *Zukünftige Fahrzeugtechnologien im Open Region Lab* durch das MWK Niedersachsen und die Volkswagenstiftung unter dem Förderkennzeichen VWZN3236 gefördert. Die Verantwortung für den Inhalt liegt bei den Autoren.



Niedersächsisches Ministerium
für Wissenschaft und Kultur



VolkswagenStiftung

7 Referenzen

- [1] X. Liu-Henke. *Mechatronische Entwicklung der aktiven Feder-/Neigetechnik für das Schienenfahrzeug RailCab*, VDI-Fortschritt-Berichte, Reihe 12, Nr. 589, VDI-Verlage, Düsseldorf, 2004.
- [2] S. Hüpkes, G. Hillebrand, W. Brings. *Flexibler Antriebsstrang für Elektrofahrzeuge*. ATZ, Band 113, Heft 12, S. 926-931, 2011.
- [3] J. Gorocny, W. Pester. „Range Extender“ gibt Elektroautos nötigen Charme. VDI-Nachrichten, Band 65, Nr. 37, S. 11, 2011.
- [4] X. Liu-Henke, M. Göllner, M. Fritsch, R. Feind, F. Quantmeyer. *FreDy – an electric vehicle with intelligent chassis control systems*, 10th IEEE International Conference on Ecological Vehicles and Renewable Energies (EVER), Monaco, March 31 - April 2, 2015.
- [5] X. Liu-Henke, S. Scherler, M. Fritsch, F. Quantmeyer. *Holistic development of a full-active electric vehicle by means of a model-based systems engineering*. 2nd IEEE International Symposium on Systems Engineering (ISSE), Edinburgh, Scotland, October 4 -5, S. 151 - 157, 2016.
- [6] X. Liu-Henke, S. Scherler, M. Göllner. *System Architecture of a full active Autonomous Electric Vehicle*. IEEE International Transportation Electrification Conference and Expo (ITEC), Chicago, Illinois, USA, June 26 - 28, 2017.

Modellbasierte Entwicklung einer Spurfolgeregelung mittels zeitdiskreten modellprädiktiven Regelalgorithmen (DMPC)

Jie Zhang¹, Marian Göllner¹, Xiaobo Liu-Henke¹

¹ Ostfalia Hochschule für angewandte Wissenschaften, Fakultät Maschinenbau,
Institut für Mechatronik, Salzdhahmer Str. 46/48, 38302 Wolfenbüttel

jie.zhang@ostfalia.de

Im vorliegenden Beitrag wird die modellbasierte Entwicklung einer Spurfolgeregelung mittels zeitdiskreten modellprädiktiven Regelalgorithmen vorgestellt. Dazu wird zunächst ein vereinfachtes lineares Zustandsraummodell des zu regelnden Fahrzeugs hergeleitet und erweitert. Anschließend werden Regelalgorithmen ausgelegt welche mittels diesem in einem Prädiktionshorizont das Spurfolgen optimieren

1 Einleitung

Zentraler Bestandteil der autonomen Fahrt ist neben der effizienten Planung, das Folgen einer Trajektorie. Diese wird auf globaler und lokaler Ebene durch Navigation und Umfelderkennung generiert [1] und stellt im eigentlichen Sinne eine Folge von abzufahrenden Wegpunkten dar, welche einer unterlagerten Regelung als Sollwerte übergeben werden. Werden nun einfache lineare Regler zum folgen dieser Punkte genutzt, wird zwar bei entsprechender Regelgüte die Trajektorie abgefahren, jedoch auf die Änderungen der Zielkoordinaten wie auf einen Sollwertsprung reagiert. Bessere Ergebnisse liefern also Regelstrategien welche den nächsten zu erwartenden Sollwertpunkt mitberücksichtigen [2].

Aus diesem Grunde sollen im Rahmen eines Forschungsvorhabens der Ostfalia Hochschule für angewandte Wissenschaften zur Spurfolgeregelung des Forschungsträgers M-Mobile ein modellprädiktiver Regelalgorithmus (DMPC) entwickelt und validiert werden. Das M-Mobile ist ein autonomes, vollaktives Elektrofahrzeug im Maßstab 1:3 mit dezentralen Direktantrieben und fungiert als Funktionsträger zur Integration von Elektromobilität und intelligenten Fahrwerksystemen [3].

Zunächst soll ein lineares Fahrdynamikmodell entwickelt werden welches die Prädiktion zukünftiger Fahrzustände erlaubt und so die direkte Integration von folgenden Wegpunkten ermöglicht. Dieses lineare Modell basiert auf einem Einspurmodell und berücksichtigt bereits die unterlagerten Antriebs- und Brems-, Federungs- und Lenkmodule. Um diese integrale Funktionalität im Modell einzubetten, wird es in Differenz-

schreibweise formuliert und durch die Ausgangsgrößen erweitert. Auf Basis des erweiterten Zustandsraummodells kann der zukünftige Verlauf der Zustandsgrößen des Systems, die auf den aktuellen Zustandsgrößen und den zukünftigen Änderungen der Stellgröße basieren, prädiziert werden [4]. Da die zugrundeliegende Modellbeschreibung nicht vollständig mit dem realen System übereinstimmt, wird das Prinzip der „Receding horizon control“ angewandt [5].

Im Rahmen des Vorliegenden Beitrags wird der modellbasierte Entwicklungsprozess dieses zeitdiskreten modellprädiktiven Regelalgorithmus (DMPC) beschrieben. Zudem wird die Funktionalität durch Verifikation mittels MiL- SiL und HiL getestet.

2 Methode

Zur Entwicklung dieses komplexen Regelsystems wird der in **Abbildung 1** dargestellte mechatronische Entwicklungskreislauf herangezogen. Am Beginn der theoretischen Untersuchung steht die Modellbildung. Das technische System wird nach Funktionsprinzipien im Hinblick auf die Anforderung in ein physikalisches bzw. mathematisches Ersatzmodell aus unterschiedlichen Fachdisziplinen wie Mechanik, Hydraulik und Elektronik abgebildet. Es repräsentiert das Systemverhalten. Die Regelstrategie, die vom einfachen Regler bis hin zu hierarchisch angeordneten Mehrgrößenregelstrukturen aufgebaut sein kann, wird anhand des Systemverhaltens festgelegt.

Die anschließende modellbasierte Komposition erfolgt gemäß des Rapid Control Prototypings (RCP) in einem durchgängig verifikationsorientierten Prozess aus Model-in-the-Loop (MiL), Software-in-the-Loop (SiL) und Hardware-in-the-Loop (HiL) [3].

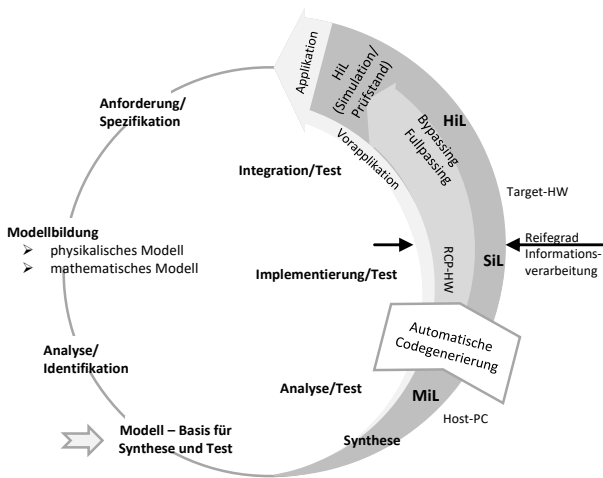


Abbildung 1. Mechatronischer Entwicklungskreislauf

3 Modellbildung

Eine wesentliche Grundlage für die Auslegung und Entwicklung eines leistungsfähigen prädizierenden Regelungssystems ist ein mathematisches Modell der Regelstrecke. Das Streckenmodell des Fahrzeugprototyps M-Mobile wird hierfür, wie in der folgenden **Abbildung 2** dargestellt, durch ein vereinfachtes lineares Einspurmodell angenähert, welches die Kinematik und Dynamik des Systems repräsentiert.

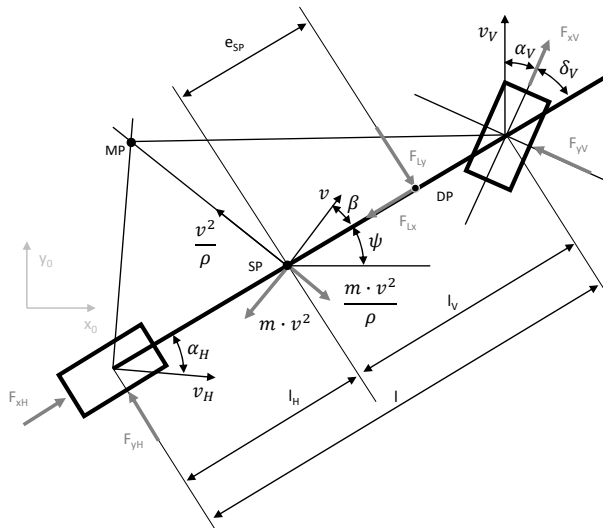


Abbildung 2. Physikalisches Fahrzeugmodell [6]

3.1 Einspurmodell

Anhand der Zusammenfassung und auch Vereinfachung der Kinematik und Dynamik des Fahrzeugs werden die Bewegungsgleichungen durch Impulsbilanz in y-Richtung sowie die Drehimpulsbilanz um die z-Achse aufgestellt:

$$J_z \ddot{\psi} = F_{yV} l_V - F_{yH} l_H \quad (1)$$

$$m v_x (\dot{\beta} + \dot{\psi}) = F_{yV} + F_{yH} \quad (2)$$

Unter Annahme kleiner Seitenkräfte und unter Vernachlässigung des transienten Reifenverhaltens bei trockener Fahrbahn ergibt sich eine lineare Abhängigkeit zwischen Schräglaufwinkel vorn α_V bzw. hinten α_H und der entsprechenden Reifenseitenkraft:

$$F_{yV} = C_{\alpha V} \alpha_V \quad (3)$$

$$F_{yH} = C_{\alpha H} \alpha_H \quad (4)$$

Aus dem Zusammenhang zwischen den Schräglaufwinkeln, dem Lenkwinkel an der Vorderachse δ_V und dem Schwimmwinkel β vereinfacht sich die Gleichung für den Schräglaufwinkel von Vorder- und Hinterachse:

$$\alpha_V \approx \delta_V - \beta - \frac{l_V}{v_x} \dot{\psi} \quad (5)$$

$$\alpha_H \approx -\beta + \frac{l_H}{v_x} \dot{\psi} \quad (6)$$

Anhand des physikalischen Modells und des dynamischen Verhaltens erhält man, wie bereits erwähnt, das vereinfachte mathematische Modell des M-Mobile bei einer bestimmten konstanten Geschwindigkeit v_x , welche im Zustandsraum dargestellt wird:

$$\dot{x} = A x + B \delta_V \quad (7)$$

Die Größen x , A_c und B_c werden definiert zu:

$$x = [\beta \ \dot{\psi}]^T, \quad (8)$$

$$A = \begin{bmatrix} -\frac{C_{\alpha V} + C_{\alpha H}}{m v_x} & \frac{C_{\alpha H} l_H - C_{\alpha V} l_V}{m v_x^2} - 1 \\ \frac{C_{\alpha H} l_H - C_{\alpha V} l_V}{J_z} & -\frac{C_{\alpha H} l_H^2 + C_{\alpha V} l_V^2}{J_z v_x} \end{bmatrix},$$

$$B = \begin{bmatrix} \frac{C_{\alpha V}}{m v_x} & \frac{C_{\alpha V} l_V}{J_z} \end{bmatrix}^T.$$

3.2 Erweitern des Zustandsraums

Folgend wird der Zustandsraum so angepasst, dass er sich für den Einsatz in einer modellprädiktiven Regelung der lateralen Fahrzeugposition eignet.

Durch Einsetzen der zusätzlichen Zustandsgrößen bzw. der lateralen Fahrzeuggeschwindigkeit v_y , lateralen Fahrzeugposition x_y und Gierrate ψ kann der Zustandsraum wie folgt umgeformt werden:

$$\dot{x} = A_c x + B_c \delta_V$$

$$y = C_c x \quad (9)$$

$$x = [\beta \ \dot{\psi} \ v_y \ x_y \ \psi]^T,$$

$$y = [\dot{\psi} \ x_y]^T$$

Mit der Dynamikmatrix A_c , der Eingangsmatrix B_c und der Ausgangsmatrix C_c :

$$A_c = \begin{bmatrix} \frac{-C_{aV}-C_{aH}}{mv_x} & \frac{C_{aH}l_H-C_{aV}l_V}{mv_x^2} - 1 & 0 & 0 & 0 \\ \frac{C_{aH}l_H-C_{aV}l_V}{J_Z} & \frac{-C_{aH}l_H^2-C_{aV}l_V^2}{J_Z v_x} & 0 & 0 & 0 \\ \frac{-C_{aV}-C_{aH}}{m} & \frac{C_{aH}l_H+C_{aV}l_V}{mv_x} & 0 & 0 & 0 \\ 0 & v_x & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (10)$$

$$B_c = \left[\frac{C_{aV}}{mv_x} \quad \frac{C_{aV}l_V}{J_Z} \quad \frac{C_{aV}}{m} \quad 0 \quad 0 \right]^T,$$

$$C_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Somit sind neben den herkömmlichen Zustandsgrößen β und $\dot{\psi}$ weitere wichtige Größen zum Beschreiben der Fahrzeugbewegung verfügbar. Als Ausgangsgrößen des Systems werden die Gierrate und die laterale Position des Fahrzeugs gewählt.

4 Modellbasierte Reglerauslegung

Zuerst muss eine geeignete mathematische Beschreibungsform des dynamischen Systems gefunden werden, da die gewünschte Realisierung der modellprädiktiven Regelung soll auf einer Echtzeithardware mit fester Schrittweite erfolgen soll. Um die Integration zu erleichtern, ist eine diskrete Beschreibungsform des Systems erstrebenswert. Die kontinuierliche Zustandsraumdarstellung in Gl. (9) wird dazu in eine diskrete Darstellung überführt:

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d \delta_v(k) \\ y(k) &= C_d x(k) \end{aligned} \quad (11)$$

Die Berechnung der Zustandsgröße kann ebenfalls wie folgt beschrieben werden:

$$x(k) = A_d x(k-1) + B_d \delta_v(k-1) \quad (12)$$

Wird nun die Änderung der Zustandsgröße innerhalb eines Zeitschrittes zu

$$\begin{aligned} \Delta x(k) &= x(k) - x(k-1) \\ \Delta \delta_v(k) &= \delta_v(k) - \delta_v(k-1) \end{aligned} \quad (13)$$

Definiert, kann die Gl. (10) in folgender Differenzschreibweise dargestellt werden:

$$\begin{aligned} \Delta x(k+1) &= A_d \Delta x(k) + B_d \Delta \delta_v(k) \\ \Delta y(k+1) &= C_d \Delta x(k+1) \\ &= C_d A_d \Delta x(k) + C_d B_d \Delta \delta_v(k) \end{aligned} \quad (14)$$

Dargestellt in einem erweiterten Zustandsraummodell kann das dynamische System wie folgt beschrieben werden:

$$\begin{aligned} x(k+1) &= A x(k) \\ \begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix} &= \begin{bmatrix} A_d & 0_d^T \\ C_d A_d & I_{q \times q} \end{bmatrix} \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B_d \\ C_d B_d \end{bmatrix} \cdot \Delta \delta_v(k) \\ y(k) &= [A_d \quad I_{q \times q}]^C \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} \end{aligned} \quad (15)$$

Ein Vorteil dieses erweiterten Zustandsraummodells ist, dass die Ausgangsgrößen des Systems direkt in einer Gleichung berechnet werden können. Somit können die Abweichungen von den gewünschten Werten direkt ebenfalls in dieser Gleichung berücksichtigt werden.

Auf Basis des erweiterten Zustandsraummodells können nun zukünftige Zustandsgrößen des Systems prädiziert werden. Die hier definierte Zustandsgröße ändert sich wie folgt:

$$\begin{aligned} x(k_i+1|k_i) &= A x(k_i) + B \Delta \delta_v(k_i) \\ x(k_i+2|k_i) &= A x(k_i+1) + B \Delta \delta_v(k_i+1) \\ &= A^2 x(k_i) + A B \Delta \delta_v(k_i) + B \Delta \delta_v(k_i+1) \\ &\vdots \\ x(k_i+N_p|k_i) &= A^{N_p} x(k_i) + A^{N_p-1} B \Delta \delta_v(k_i) + \dots \\ &\quad + A^{N_p-N_c} B \Delta \delta_v(k_i+N_c-1) \end{aligned}$$

Die prädizierten Ausgangsgrößen dementsprechend:

$$\begin{aligned} y(k_i+1|k_i) &= C A x(k_i) + C B \Delta \delta_v(k_i) \\ y(k_i+2|k_i) &= C A x(k_i+1) + C B \Delta \delta_v(k_i+1) \\ &= C A^2 x(k_i) + C A B \Delta \delta_v(k_i) + C B \Delta \delta_v(k_i+1) \\ &\vdots \\ y(k_i+N_p|k_i) &= C A^{N_p} x(k_i) + C A^{N_p-1} B \Delta \delta_v(k_i) + \dots \\ &\quad + C A^{N_p-N_c} B \Delta \delta_v(k_i+N_c-1) \end{aligned}$$

Die prädizierten Größen basieren auf den aktuellen Zustandsgrößen bei $t = k_i$ und den zukünftigen Änderungen der Stellgröße (bis zu N_c). Allgemein muss der Prädiktionshorizont N_p größer gleich dem Kontrollhorizont N_c sein.

Dargestellt in einer kompakten Matrixform ergibt sich:

$$Y = F x(k_i) + \Phi \Delta \delta_v \quad (16)$$

Mit

$$F = \begin{bmatrix} C A \\ C A^2 \\ C A^3 \\ \vdots \\ C A^{N_p} \end{bmatrix}; \Phi = \begin{bmatrix} C B & 0 & \dots & 0 \\ C A B & C B & \dots & 0 \\ C A^2 B & C A B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C A^{N_p-1} B & C A^{N_p-2} B & \dots & C A^{N_p-N_c} B \end{bmatrix} \quad (17)$$

Die Kostenfunktion J wird auf Basis der vorgestellten Beschreibungsform des dynamischen Systems formuliert. Bei der Wahl eines quadratischen Ansatzes der Kostenfunktion kann sichergestellt werden, dass das globale Minimum gefunden wird.

$$J = (R_S - Y)^T (R_S - Y) + \Delta\delta_v^T R \Delta\delta_v \quad (16)$$

Mit R_S als Sollwertvektor und R zur Beeinflussung der Schnelligkeit des geschlossenen Regelkreises. Im ersten Teil der Kostenfunktion wird die Abweichung von den Sollwerten, im zweiten Teil die Stellgrößenänderung gewichtet. Das Minimum durch partielles Ableiten nach der Stellgrößenänderung bestimmt:

$$\Delta\delta_v = (\Phi^T \Phi + R)^{-1} \Phi^T (R_S - Fx(k_i)) \quad (17)$$

Dieser Vorschrift entsprechend werden die Stellgrößenänderungen über den gesamten Kontrollhorizont N_C berechnet. Da die zugrundeliegende Modellbeschreibung nicht vollständig mit den realen System übereinstimmt, wird in jedem Berechnungsschritt lediglich das erste Element des Vektors $\Delta\delta_v$ als Stellgröße ausgegeben und die restlichen Elemente verworfen [5]:

$$\begin{aligned} \Delta\delta_v &= [1 \ 0 \ \dots \ 0]_{1 \times N_C} (\Phi^T \Phi + R)^{-1} \Phi^T (R_S - Fx(k_i)) \\ &= K_y r(k_i) - K_{MPC} x(k_i) \end{aligned} \quad (18)$$

wobei K_y das erste Element von $(\Phi^T \Phi + R)^{-1} \Phi^T R_S$ und K_{MPC} die erste Reihe von $(\Phi^T \Phi + R)^{-1} \Phi^T F$ ist. Dieses Vorgehen wird im nächsten Berechnungsschritt wiederholt und als gleitender Horizont beschrieben. D.h. das Optimierungsproblem ist in einem beweglichen Zeithorizontfenster zu lösen.

Mithilfe des Zustandsreglers K_{MPC} und des Vorfilters K_y wird der geschlossene Regelkreis bestimmt:

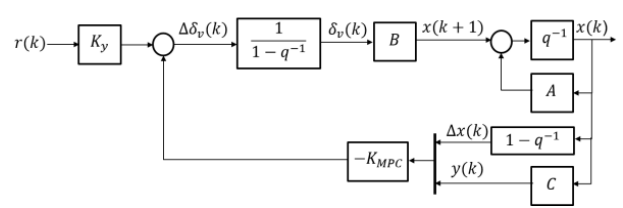


Abbildung 3. Geschlossener Regelkreis

5 Funktionsabsicherung mittels MiL

Zur Funktionsabsicherung wird zunächst die Modell-in-the-Loop-Simulation herangezogen.

Ziel der Spurfolgeregelung ist primär das das M-Mobile einer Solltrajektorie so schnell und exakt wie möglich folgt. Um dies zu Testen wurde eine polynomische

Gleichung als Solltrajektorie für einen Doppelspurwechsel auf den Regler gegeben. **Abbildung 4** stellt die Querverschiebung Y des Fahrzeugschwerpunktes über die longitudinale Bewegung X bei $v_x = 1m/s$ dar.

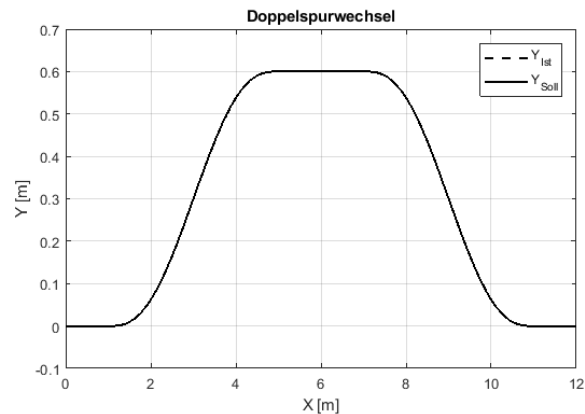


Abbildung 4. Doppelspurwechsel

Aus dem Simulationsergebnis ist ersichtlich das der Regelkreis fast genauso schnell agiert wie es die Solltrajektorie vorgibt. Die **Abbildung 4** zeigt entsprechend das die Soll- und Istquerverschiebung sich ohne Abweichung überlagern.

Der Verlauf des Lenkwinkels an der Vorderachse wird in **Abbildung 5** dargestellt. Bei $t = 1s$ verändert sich der Lenkwinkel um der Solltrajektorie zu Folgen. Bei $t = 5s$ erreicht das Fahrzeug die geplante Sollposition. Anschließend folgt das Fahrzeug der neuen Fahrspur in Vorwärtsrichtung und der Lenkwinkel ändert sich nicht. Bei $t = 7s$ befährt das Fahrzeug wieder die ursprüngliche Fahrspur und erreicht die geplante Endposition.

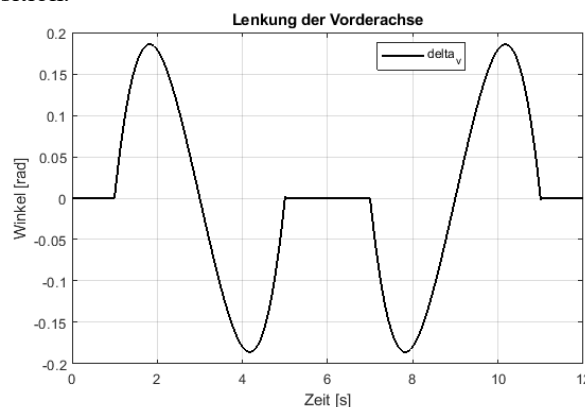


Abbildung 5. Lenkwinkel der Vorderachse

6 Resümee

In dieser Arbeit wurde ein modellprädiktiver Regler anhand eines vereinfachten linearere Zustandsraummodells einer Einspurfahrzeugs als Näherung des M-

Mobile entwickelt. Dazu wurde dieses Modell um für die Regelung notwendige Zustandsgrößen erweitert und für die spätere Ausführung auf Target Hardware in fester Schrittweite diskretisiert. Der diskrete Modellprädiktive Regelalgorithmus kann mit dessen Hilfe in einem beweglichen Ereignishorizont zukünftige Stellgrößen vorhersagen. Dies wurde mittels einer Modell-in-the-Loop Simulation anhand eines Referenzmodells des M-Mobile nach [3] validiert.

Im weiteren Verlauf dieser Arbeit wird der gefundene Algorithmus in das bestehende Fahrzeugmanagementsystem des M-Mobile eingebettet und in Hardware-in-the-Loop Test weiter verifiziert werden.

7 References

- [1] Stenzel, R.: *Steuerungsarchitektur für autonome mobile Roboter*. Dissertation, RWTH Aachen, 2002
- [2] Jezierski, A.; Mozaryn, J.; Suski D.: *A Comparison of LQR and MPC Control Algorithms of an Inverted Pendulum*. In: Mitkowski, W. et al: Trends in Advanced Intelligent Control, Optimization and Automation. Advances in Intelligent Systems and Computing, vol 577. Springer, Cham, 2017
- [3] Buchta, R.: *Mechatronische Entwicklung eines Forschungselektrofahrzeugs zur Erprobung von Fahrdynamikregelungen und Fahrerassistenzsystemen*. Dissertation, Universität Magdeburg, 2016
- [4] Wang, L.: *Model Predictive Control System Design and Implementation Using MATLAB*. Springer, 2009
- [5] Kwon, W.H. and Han, S.: *Receding Horizon Control- model predictive control for state models*. Springer, 2005.
- [6] Liu-Henke, X.: *Fahrzeugdynamikregelung*, Vorlesungsskript zur Vorlesung Fahrdynamikregelung, Hochschule Ostfalia, Wolfenbüttel, 2016

IoT gestütztes Maschinen-Monitoring – „Wearables“ für Maschinen

Prof. Dr.-Ing. Martin Strube

Ostfalia Hochschule für angewandte Wissenschaften – Hochschule Braunschweig Wolfenbüttel

m.strube@ostfalia.de

Die zunehmende Digitalisierung im Maschinen- und Anlagenbau führt zu einer stetig voranschreitenden horizontalen und vertikalen Vernetzung, von der Feld- bis hin zur ERP-Ebene. Damit einhergehend nimmt der Stellenwert von Methoden aus den Bereichen des Knowledge Discovery in Database und der Künstlichen Intelligenz für die Prozess- und Produktionsoptimierung sowie für Ansätze des Predictive Maintenance zu. Dem gegenüber stehen heute oftmals erhebliche Investitionskosten in „Brownfield“ geprägten Hochlohnländern, um heterogene Maschinen- und Anlagenparks mit den notwendigen Messdatenerfassungs- und Übertragungslösungen auszustatten. Um dem entgegenzuwirken, wird in diesem Beitrag ein Ansatz vorgestellt, der sich an bereits etablierten Soft- und Hardwarelösungen für Human Activity Recognition orientiert und diese mit Datenübertragungslösungen des Industrial Internet of Things kombiniert.

1 Einleitung

In vielen Produktionsbereichen finden sich heute Maschinen aus unterschiedlichen Dekaden wieder. Im Zuge der Digitalisierung, insbesondere älterer Maschinen, entstehen häufig erhebliche Kosten durch den notwendigen Ausbau der Netzwerkinfrastruktur, zusätzliche Messtechnik, Schnittstellenanpassungen, Anpassungen in der Automatisierungssoftware zur Bereitstellung der erfassten Daten und nicht zuletzt durch die Einhaltung von Auflagen aus dem Bereich der IT-Sicherheit [1]. Da sich ein Return On Investment dieser Maßnahmen meist nur sehr schwer ermitteln lässt, stellen die damit verbundenen Investitionen oft ein Hemmnis dar.

Im Kontrast dazu finden sich am Markt immer günstigere Soft- und Hardwarelösungen für das Erfassen menschlicher Aktivitäten und für die Machine-to-Machine-Kommunikation (M2M) mit Hilfe von IoT-Funktechnologien wieder. Vor diesem Hintergrund beschreibt dieser Beitrag einen Ansatz, um die zuvor genannten Entwicklungen aus dem Consumer-Bereich auf industrielle Anwendungen zu übertragen.

2 Human Activity Recognition

Der Markt für Wearables ist im Jahr 2017 um ca. 17 % gewachsen, so dass in diesem Jahr weltweit mehr als 300 Mio. dieser Geräte verkauft wurden [2]. Verbunden mit dem enormen Wachstum dieses Marktes hat sich innerhalb der letzten Jahre ein starkes Forschungsfeld im Bereich Human Activity Recognition (HAR) etabliert.

Unter HAR lassen sich verschiedene Technologien zusammenfassen, die das Ziel verfolgen, menschliche Aktivitäten mit Hilfe von Sensordaten in Real-Life Szenarien zu identifizieren. In Kombination mit Wearables, wie z. B. Fitness-Armbändern, findet HAR heute bereits eine breite Anwendung in medizinischen Bereichen sowie im Freizeit- und Profisport (vergleiche hierzu [3], [4], [5]).

Bei der Erkennung von menschliche Aktivitäten mit Hilfe von Sensordaten finden zwei unterschiedliche Herangehensweisen Anwendung. Diese werden fachsprachlich durch die Begriffe Activity Recognition und Activity Pattern Discovery beschrieben [6]. Beim Activity Recognition werden zunächst Aktivitäts-Modelle für die zu identifizierenden Bewegungsabläufe definiert und darauf basierend geeignete Sensorsysteme zur Detektion dieser Bewegungsabläufe entwickelt. Die Modellbildung kann dabei z. B. mit Hilfe des Hidden Markov Models erfolgen (vergleiche hierzu [6], [7], [8]).

Im Gegensatz dazu werden beim Activity Pattern Recognition zunächst Sensordaten bestimmter Bewegungsabläufe aufgezeichnet, um diese dann auf Muster hin zu untersuchen, die eine eindeutige Zuordnung zu bestimmten Aktivitäten ermöglichen. Bei der Mustererkennung finden Methoden wie Adaptive Neuro-Fuzzy Inference Systems, Classification and Regression Tree und Iterative Dichotomiser 3 Anwendung [9].

Infolge der zahlreichen Forschungsbemühungen auf dem Gebiet HAR stehen heute eine Vielzahl von

modellbasierten Ansätzen für die Aktivitätserkennung von menschlichen Individuen zur Verfügung.

Gemessen an der Anzahl der genutzten Sensoren und der Funktionalität, den Großteil der Messdaten direkt auf dem Wearable ohne zusätzlichen Konfigurationsaufwand zu verarbeiten, stellen diese Systeme im Vergleich zu industriellen Lösungen für die Maschinendatenerfassung und -auswertung ein hervorragendes Preis-Leistungs-Verhältnis dar.

3 Industrial Internet of Things

„Der Wandel der Automatisierungsindustrie [...] ist ohne den kombinierten Einsatz moderner Technologien und deren Vernetzung nicht umsetzbar, denn die Verknüpfung aller Komponenten ist Voraussetzung für eine hochdynamische und flexible Industrie 4.0-Produktion.“ [10]

Mit dem Ausbau bereits verfügbarer Low Power Wide Area Network (LPWAN) Kommunikationslösungen des Internet of Things, wie z.B. SigFox [11], LoRa [12] und NB-IoT [13], und der Weiterentwicklung dieser Funktechnologien im Rahmen von Programmen wie Horizon 2020 [14], entsteht auch für industrielle Anwendungen ein neues Portfolio an Datenübertragungslösungen.

Diese zeichnen sich nicht nur durch sehr kostengünstige und einfach inbetriebzunehmende Hardwarelösungen, sondern auch durch vergleichsweise sehr niedrige Datenübertragungskosten aus.

„Mit der Entwicklung neuer Funktechnologien werden digitale drahtlose Kommunikationsnetze zukünftig eine Vielzahl neuer industrieller Anwendungen erschließen und technische Anforderungen erfüllen, die bisher nur durch kabelgebundene Netze erfüllt werden konnten.“ [10]

Ein prädestinierter Anwendungsbereich für Industrial Internet of Things-Funktechnologien (IIoT) ist z. B. die Erweiterung von Maschinen und Anlagen mit drahtlosen Sensorknoten, um zusätzliche Informationen für Wartung, Diagnose und Zustandsüberwachung zu gewinnen. Die Energieeffizienz der LPWAN Technologien ermöglicht dabei zum Teil Realisierungen mit batteriebetriebenen Sensorknoten, wodurch Messungen auch an Stellen ohne das Herstellen einer externen Energieversorgung möglich werden. Des Weiteren ist auch die Ergänzung von Anlagen- und Maschinensteuerungen und Prozessleit-

systemen durch drahtlose Sensorknoten ein naheliegender Anwendungsbereich.

Die Herausforderung bei der Auswahl geeigneter IIoT-Funktechnologien liegt in den anwendungsbezogen stark variierenden Anforderungen, wie z.B. Reichweite, Latenz, Robustheit und Quality of Service, und den am Einsatzort befindlichen Umgebungsbedingungen.

Eine weitere Einschränkung der Anwendungsszenarien ergibt sich durch die im Vergleich zu kabelgebundenen oder etablierten Funktechnologien, wie WLAN oder LTE, verminderten Datenübertragungsraten, wodurch hochfrequent erfasste Sensordaten nicht ohne Weiteres übertragen werden können.

4 „Wearables“ für Maschinen

Betrachtet man die aktuellen Entwicklungen aus dem Bereich der Kommunikationslösungen des Industrial Internet of Things gemeinsam mit den Fortschritten der Forschung zum Thema HAR, so ergeben sich in Bezug auf die in Abschnitt 3 aufgeführten Anwendungsbereiche zwei Fragestellungen:

1. Lassen sich für Wartung, Diagnose und Zustandsüberwachung mit nachträglich implementierten kostengünstigen Sensorknoten, ähnlich wie bei einem Fitness-Armband, relevante Informationen mit Hilfe von Methoden des HAR generieren? Ein Beispiel hierfür wären Rückschlüsse auf den Werkzeugverschleiß auf Basis der Schwingungsmessung an der Außenhaut einer Werkzeugmaschine.
2. Können Ansätze des HAR auf maschinenbauliche Fragestellungen übertragen werden, um hochfrequent erfasste Sensordaten im Sinne des Edge-Computing direkt am Sensorknoten soweit vorzuverarbeiten, dass Funktechnologien mit geringeren Datenübertragungsraten für die Anbindung an bestehende Monitoring- und Automatisierungslösungen Anwendung finden können?

In der aktuellen Forschungsarbeit wird untersucht, wie sich Ansätze aus dem Bereich Human Activity Recognition mit Hilfe von industrietauglichen Sensorplatinen mit IIoT-Funkmodulen auf für den Maschinen- und Anlagenbau relevante Fragestellungen übertragen lassen. Dabei werden diese Sensorplatinen ähnlich wie „Wearables“ nachträglich an leicht zugänglichen Stellen der Maschinen angebracht. Des

Weiteren sind die Platinen mit kostengünstigen Sensoren, z. B. Beschleunigungssensoren wie sie in Smartphones oder Wearables zum Einsatz kommen, bestückt.

In Anlehnung an das in Abschnitt 2 vorgestellten Activity Pattern Discovery werden in einem ersten Schritt Versuche mit Werkzeugmaschinen, z.B. Fräs- und Drehmaschinen, verschiedener Generationen durchgeführt. Um Messdaten zu gewinnen, werden die Werkzeugmaschinen dabei mit unterschiedlichen Sensorplatinen an verschiedenen Stellen ausgerüstet. Abbildung 1 zeigt exemplarisch eine der Maschinen, an denen die Versuche durchgeführt werden.



Abbildung 1. Fräsmaschine von DMG Mori [15]

Abbildung 2 zeigt ein Beispiel der verwendeten Sensorplatinen mit den Abmessungen 50 mm x 50 mm x 15 mm. Auf der hier dargestellten Oberseite befindet sich ein Beschleunigungssensor, ein GPS-Modul und ein Mikrocontroller. Das IoT-Funkmodul befindet sich auf der Unterseite dieser Platine.

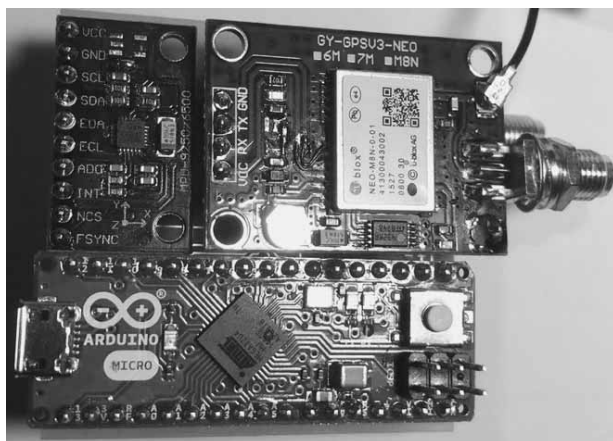


Abbildung 2. Sensorplatine

Im Anschluss werden Zusammenhänge zwischen den von der Maschinensteuerung aufgezeichneten Daten und den Sensordaten der Sensorplatinen untersucht.

Des Weiteren finden Methoden der Mustererkennung des Activity Pattern Discovery Anwendung, um nach Mustern für verschiedene maschinenbauliche Fragestellungen zu suchen. Diese reichen von der Ermittlung OEE relevanter Kenngrößen, über die Erkennung von Werkzeugverschleiß und qualitätsrelevanten Parametern bis hin zu Predictive Maintenance Anwendungen.

In einem zweiten Schritt sollen wie beim Activity Recognition Modelle für die zu identifizierenden Abläufe ausgewählter Werkzeugmaschinen definiert und darauf basierend geeignete Sensorplatinen zur Detektion dieser Abläufe entwickelt werden. Darauf aufbauend muss untersucht werden, wie generalisierungsfähig die Modelle im Hinblick auf das Übertragen der Ergebnisse auf verschiedene Maschinen eines Typs sind.

Aufbauend auf den Ergebnissen der zuvor beschriebenen Arbeiten, gilt es zu untersuchen, ob sich die entwickelten Modelle und Algorithmen im Sinne des Edge-Computings eignen, um hochfrequent erfasste Sensordaten informationsverlustfrei mit Hilfe eines Mikroprozessors auf der Sensorplatine soweit vorzuverarbeiten, dass die daraus resultierenden Informationen mittels IIoT-Funkmodulen an bestehende Monitoring- und Automatisierungslösungen übertragen werden können. Auch hier lassen sich Parallelen zu den technischen Realisierungen von Wearables aus dem Consumer Bereich ziehen, denn auch diese verarbeiten und visualisieren einen Teil der Messdaten direkt auf dem Wearable und übertragen die Ergebnisse für komplexere Auswertungen an Apps oder Cloud-basierte Dienste.

5 References

- [1] Bundesministerium für Wirtschaft und Energie. IT-Sicherheit für die Industrie 4.0 – Abschlussbericht. 2016.
- [2] Gartner. Forecast: Wearable Electronic Devices Worldwide, 2017.
- [3] A. Avci et. al. Activity Recognition Using Inertial Sensing for Healthcare, Wellbeing and Sports Applications: A Survey. 23th International Conference on Architecture of Compu-

- ting Systems, Hannover, Germany, S. 1-10, 2010.
- [4] U. Jensen et al. A wearable real-time activity tracker. *Biomedical Engineering Letters*, Volume 5, S. 147-157, 2015.
- [5] H.-J. Kim und Y. S. Choi. Eating Activity Recognition for Health and Wellness: A Case Study on Asian Eating Style. *IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, USA, S. 446-447, 2013.
- [6] E. Kim et al. Human Activity Recognition and Pattern Discovery. *IEEE Pervasive Computing*, Volume 9, S. 48-53, 2010.
- [7] S. Shaily, V. Mangat. The Hidden Markov Model and its Application to Human Activity Recognition. *2nd International Conference on Recent Advances in Engineering & Computational Sciences (RAECS)*, Chandigarh, India, 2015.
- [8] Y.-J. Kim et al. Hidden Markov Model Ensemble for Activity Recognition using Tri-axis Accelerometer. *IEEE International Conference on Systems, Man, and Cybernetics*, Kowloon, China, S. 3036-3041, 2015.
- [9] L. C. Jatobá et al. Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity. *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vancouver, Canada, S. 5250-5253, 2008.
- [10] VDE-Positionspapier Funktechnologien für Industrie 4.0. *Verband der Elektrotechnik Elektronik Informationsrechnik e.V.* 2017.
- [11] <https://www.sigfox.com/>
- [12] <https://www.lora-alliance.org>
- [13] <https://www.telekom.com/de/medien/medieninformationen/detail/erste-narrowband-iot-servicepakete-in-deutschland-auf-dem-markt-497480>
- [14] <http://www.horizont2020.de>
- [15] <https://de.dmgmori.com/produkte/maschinen/fraesen/vertikal-fraesen/cmx-v>

Approximation von Materialflussmodellen durch neuronale Netze und Gradient-Boost Verfahren

Bernd Allmendinger¹

Stefan Pfaff¹

¹PPI-Informatik

Bernd.Allmendinger@PPI-Informatik.de, Stefan.Pfaff@PPI-Informatik.de

Das Bilden von Ersatzmodellen in der Simulation hat sich in der Praxis durchgesetzt. Durch das Ersatzmodell können sehr schnell und effizient Berechnungen für neue Parameter Konstellationen durchgeführt werden. Bisher werden diese Modelle meist durch lineare Regressionsverfahren ermittelt. Wir zeigen hier an einem Beispiel einer realen Materialflussplanung und deren Simulation, dass sich Gradient-Boost Verfahren wie z.B. mit XGBoost und neuronale Netze (z.B. mit Keras) viel besser als Ersatzmodelle eignen, da sie die nichtlinearen funktionale Zusammenhänge in Simulationsmodellen meist viel besser approximieren können.

1 Einleitung

Simulieren¹ heißt probieren. Bei real existierenden Systemen wie z.B. Fertigungslinien kann man jedoch aus naheliegenden Gründen nur sehr eingeschränkt "probieren". Und bei Systemen im Planungsstadium geht das erst recht nicht. Hier bietet die Simulation den Ausweg: Man erstellt zunächst ein realistisches Modell des zu untersuchenden Systems mit dem man dann systematisch experimentieren kann.

Simulationsmodelle werden eingesetzt, um z.B. in komplexen Systemen Engpässe zu erkennen, Grenzkapazitäten zu ermitteln oder um Zielgrößen wie den Durchsatz zu optimieren. Dabei werden die Prozessparameter vorgegeben, die im Modell systematisch variiert und die Auswirkungen dieser Variationen auf das Verhalten des Simulationsmodells mit Hilfe geeigneter Kenngrößen untersucht. Ziel ist es, aus den einzelnen Experimenten den Zusammenhang zwischen den vorgegebenen Prozessparametern und den daraus resultierenden Zielgrößen des simulierten Modellprozesses abzuleiten.

In der Praxis führt das jedoch aufgrund der meist sehr großen Anzahl sinnvoller Variationen der Prozessparameter auf eine entsprechend große Anzahl von Modellexperimenten, wenn man die wesentlichen Kombinationen untersuchen will. Außerdem führen die Experimente u.U. zu extrem große Datenmengen, die sich einer Auswertung mit „klassischen“ Methoden entziehen.

Hier setzt die Einbeziehung von Verfahren des maschinellen Lernens ein, denn die Untersuchung großer

Datenmengen hinsichtlich der darin enthaltenen Regeln und Muster ist eine klassische Anwendung des maschinellen Lernens. Das maschinelle lernen umfasst sehr viele unterschiedliche Methoden/Verfahren. Die große Herausforderung ist, für ein gegebenes Problem das richtige Verfahren auszuwählen oder mehrere unterschiedliche Verfahren mit einander zu kombinieren (Bildung von Ensembles).

In der Praxis, sowie bei Maschine-Learning-Wettbewerben wie z.B. auf der Plattform Kaggle hat sich gezeigt, dass sich Gradient-Boost basierte Methoden sehr gut eignen um komplexe Probleme mit einer hohen Genauigkeit zu lösen. Auch neurale Netze (Deep Learning) setzen sich in der Praxis immer mehr durch. Beide Methoden setzen große Datenmengen voraus was beim praktischen Einsatz der Materialflusssimulation nicht immer gegeben bzw. praktikabel ist. Durch die Regulierungsverfahren beider Methoden ist es jedoch möglich, diese Verfahren auch auf Probleme mit einer „kleineren“ Anzahl an Datensätzen durchzuführen.

Wir untersuchen die Einsetzbarkeit dieser beiden Methoden an einem Beispiel aus der täglichen Praxis bei der Simulation von Produktions- und Logistiksystemen.

Als Gradient-Boost Verfahren verwenden wir XGBoost was aktuell (2018) zusammen mit LightGBM und CatBoost zu den leistungsstärksten Gradient-Boost Verfahren zählt.

Als Deep Learning Software wird Keras verwendet. Keras ist ein Backend für Tensor-Flow bzw. Theano mit sich sehr schnell neuronale Netze erstellen und trainieren lassen.

¹ Siehe VDI 3633

2 Vorteile einer Approximation

Durch die Approximation bzw. der Überführung eines Simulationsmodells in ein „maschinelles Lernen“ Modell generiert man aus dem Simulationsmodell eine mathematische Formel. Diese Formel kann dann zusammen mit einer Optimierung dazu verwendet werden um z.B. Ressourcen optimal auszuliegen (z.B. Puffer optimal zu dimensionieren oder die Geschwindigkeiten von Förderstrecken optimal zu bestimmen). Da das Modell als Formel vorliegt, können die Simulationsergebnisse auf „Knopfdruck“, ohne Simulationssoftware und zeitlich aufwändige Experimente, erzeugt werden.

Durch die Regulierungsverfahren von XGBoost und Keras kann die Stochastik die im Simulationsmodell Modell entsteht ggf. geglättet werden. Dadurch können Daten die für das maschinelle Lernen benötigt werden ggf. mit einer geringen Anzahl von Simulationsexperimenten bzw. Parametervarianz erzeugt werden.

Ein weiterer Vorteil ist das bessere Systemverständnis das aus den Daten erlernt wird und über die „feature importance“ Funktion der Lernverfahren extrahiert werden kann.

3 Anwendungsfall und Modell

Untersuchungsgegenstand ist ein Modell aus der Praxis. Das Modell bildet ein Produktionssystem mit 3 Prozessstationen ab. Der Transport zwischen den Prozessstationen erfolgt über ein fahrerloses Transport System (FTS). Die Bearbeitung an den Prozessstationen erfolgt jeweils auf dem Fahrzeug (FTF). Nach der Letzen Prozessstationen muss das Produkt erst aushärten bevor es vom FTF abgenommen werden kann. Während der Aushärtezeit verweilt das FTF in einem der drei Bahnhöfe. Der Kurs ist in zwei Bereiche mit jeweils unterschiedlicher FTF Fahrgeschwindigkeit eingeteilt.

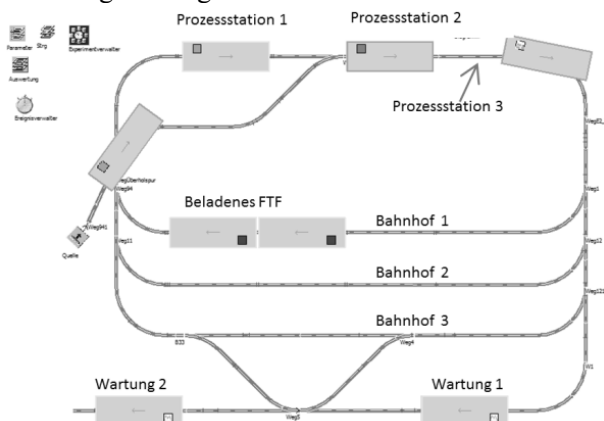


Abbildung 1 Modell in Siemens Plant Simulation

Als Ergebnissenkenngröße wird die Ausbringung pro Schicht verwendet. Um den Prozess optimal auszuliegen stehen folgende Prozessparameter zur Verfügung:

- Anzahl Fahrzeuge
- Prozesszeiten an den Stationen 1, 2 und 3
- Aushärtezeit
- 2 unterschiedliche Geschwindigkeiten

Das Materialflussmodell wurde mit Plant Simulation Version 13.0 der Firma Siemens erstellt.

Ziel ist es eine Berechnungsvorschrift zu entwickeln, die die Ausbringung pro Schicht in Abhängigkeit der Parameter: Anzahl Fahrzeuge, Prozesszeiten, Aushärtezeit und Geschwindigkeiten) in einen funktionalen Zusammenhang bringt.

$$Ausbringung = f(\#Fahrzeuge, t_{prozi1}, t_{prozi2}, t_{prozi3}, t_{härte}, v_1, v_2)$$

3.1 Datengenerierung

Die Parameter Konstellationen für die Experimente wurden über einen zweistufigen voll faktoriellen Versuchsplan erzeugt, der zusätzlich mit 5000 raumfüllenden Experimenten angereichert wurden.

Für das Training der Approximation stehen somit 5128 Datensätze zur Verfügung

	A	B	C	D	E	F	G
1	#FTF	Härtezeit	t1	t2	t3	v1	v2
218	9	3:32.2070	26.308	18.714	1:05.3000	0.3	0.119
219	9	3:10.0760	22.821	18.668	1:03.5100	0.368	0.149
220	13	3:31.3430	28.019	23.74	1:24.5370	0.273	0.131
221	12	3:30.9030	25.274	23.577	1:07.6760	0.367	0.155
222	6	3:54.6870	27.341	18.305	1:08.5960	0.34	0.209
223	13	3:24.8700	28.944	20.195	1:19.1670	0.319	0.211
224	11	3:34.3800	21.179	17.537	1:02.9630	0.4	0.16
225	7	3:45.1980	29.827	24.357	1:23.2320	0.292	0.176
226	9	3:33.0530	25.864	23.587	1:06.1680	0.421	0.136
227	13	3:45.0470	29.614	20.754	1:06.0920	0.303	0.246
228	8	3:00.3110	21.6	22.868	1:24.0280	0.443	0.181
229	13	3:03.5160	25.803	21.801	1:21.4830	0.278	0.138
230	12	3:42.5520	27.988	20.939	1:22.2020	0.415	0.153
231	13	3:31.0870	24.728	17.764	1:05.2520	0.447	0.128
232	7	3:20.2400	24.81	24.33	1:06.9280	0.358	0.227
233	13	3:45.0540	29.957	24.703	1:14.7730	0.315	0.239
234	7	3:22.8730	27.94	24.071	1:04.3100	0.339	0.169

Abbildung 2. Definition der Experimente für das Simulationsmodell

Die Experimente wurden auf einem DELL Inspiron 5759 mit 16 GB Ram und einem i7-6500U CPU mit 2.6 GHz Prozessor durchgeführt. Die Rechenzeit bei paralleler Nutzung aller CPU Kerne betrug 9h 35 min.

4 Ersatzmodelle

Für die Bestimmung der Ersatzmodelle stehen insgesamt 5128 Datensätze, die durch die Simulation generiert wurden zur Verfügung.

Von diesen Datensätzen werden 70% für die Bestimmung der Ersatzmodelle verwendet (Trainings-Daten). Die restlichen 30% werden verwendet um die Güte der Ersatzmodell zu überprüfen (Hold out-set). Zusätzlich wird mit den 70% Trainingsdaten eine 5 fache Kreuzvalidierung durchgeführt, die Aufschluss über die Güte des Ersatzmodells gibt.

Als Fehlermetrik wird „Mittlerer absoluter Fehler“

(mae) verwendet:
$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Zur Bestimmung der Modelle wurde Python 2.7 mit folgenden Bibliotheken verwenden:

- Scikit-learn (Lasso, Aufbereiten der Daten)
- XGBoost
- Keras 2.0 mit Theano als Backend

4.1 Lineare Regression - Lasso

Das einfachste Ersatzmodell, ist ein lineares Modell, bei dem die Ausbringung linear von den Eingangsvariablen abhängen (Lineare Regression). Dieses Modell verwenden wir als Referenz („Base-Line“) zum Vergleich mit den anderen Verfahren.

Da in Produktion und Logistik meist nicht lineare Zusammenhänge gegeben sind, werden zusätzliche die Quadrate und die Interaktionen zwischen den Eingangsvariablen als weitere Eingangsgrößen (sog. Feature Engineering) verwendet.

Da für die Ersatzmodellbildung die Regulierung sehr wichtig ist, verwenden wir hier zusätzlich die l1-Norm zur Regulierung (Lasso). Damit erhält man robuste und zuverlässige Modelle die gleichzeitig auch eine Eingangsvariablen Selektion durchführen, da die Koeffizienten nicht relevanter Variablen auf null geschätzt werden. Der Regulierungsparameter für die l1 Norm wurde über einen 5 fold cross validation gridsearch bestimmt.

Ergebnisse:

mae für die 30% Testdaten: 4.6986
 mae für 5-fold cross validation: 4.6620

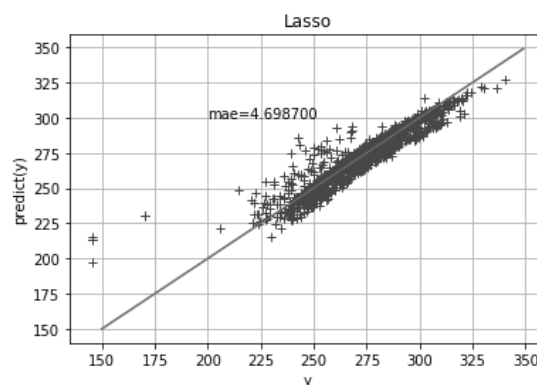


Abbildung 3. Lasso-Prognosegüte. Die Darstellung zeigt die Prognose der Ausbringung (predict(y)) vs. der aus der Simulation bestimmten Ausbringung (y) für die 30% Testdaten

Durch die l1 Norm werden die Koeffizienten nicht relevanter Variablen auf Null gesetzt. Im Beispiel verbleiben nur 17 der 36 Variablen Kombinationen im Modell.

Über das Ersatzmodell ist es nun möglich Zusammenhänge zwischen der Ausbringung und jeweils zwei Eingangsvariablen grafisch darzustellen. Im Beispiel wird der funktionale Zusammenhang zwischen der Ausbringung und Anzahl FTF sowie der Prozesszeit t3 dargestellt. Die anderen Eingangsvariablen wurde für die Darstellung konstant gehalten (t1=30, Härte Zeit=180, t2=26, v1=0.45 und v2=0.25)

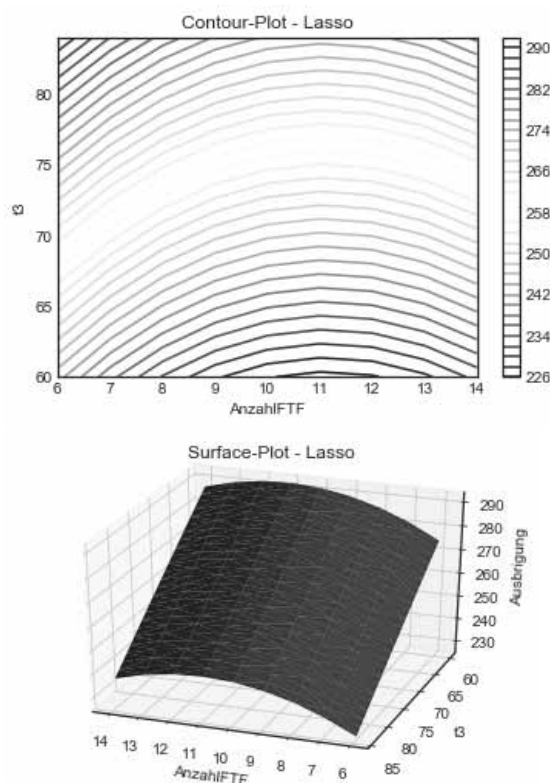


Abbildung 4 Lasso Contour und Surface plot

4.2 Gradient Boost - XGBoost

XGBoost ist ein weit verbreitetes und sehr beliebtes Tool in Kaggle-Wettbewerben und in der Industrie. In der Praxis hat sich gezeigt, dass sich mit diesem Tool sehr effizient auch hoch komplexe Probleme lösen lassen. Es ist ein hochflexibles und vielseitiges Werkzeug, das für Regressions-, Klassifikations- und Ranking-Probleme verwendet werden kann. Als Open-Source-Software ist es leicht zugänglich und kann über verschiedene Plattformen und Schnittstellen genutzt werden.

Der Name steht für eXtreme Gradient Boosting, und ist ein Teil einer größeren Sammlung von Open-Source-Bibliotheken, die von der Distributed Machine Learning Community (DMLC) entwickelt wurden. XGBoost verwendet eine regulierte Modell Formalisierung um „over-fitting“ zu verhindern. Gerade diese Eigenschaft ist wichtig für die Bildung von Ersatzmodellen, da wir aus verrauschten Daten (Störungen in der Simulation) die deterministischen Komponenten extrahieren wollen.

Ergebnisse:

mae für die 30% Testdaten: 0.8513
 mae für 5-fold cross validation: 0.8560

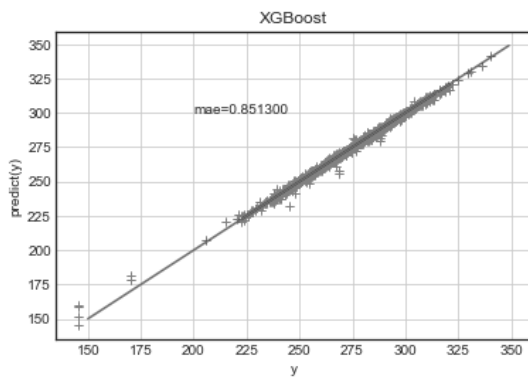


Abbildung 5 XGBoost-Prognosegüte. Die Darstellung zeigt die Prognose der Ausbringung (predict(y)) vs. der aus der Simulation bestimmten Ausbringung (y)

Die Approximation durch XGBoost ist sehr gut. Es gibt keine größeren Abweichungen in den Testdaten.

Visualisiert man den funktionalen Zusammenhang zwischen Ausbringung und Anzahl FTF sowie der Prozesszeit t3, stellt man fest, das XGBoost die Anzahl Fahrzeuge als diskret betrachtet. Klar ersichtlich ist, dass die Ausbringung für kleinere Prozesszeiten t3 steigt. Betrachtet man Anzahl FTF, so steigt die Ausbringung von 6 auf 7 FTF. Bei mehr als 8 FTF sinkt die Ausbringung wieder, da sich die FTF im Kurs gegenseitig blockieren.

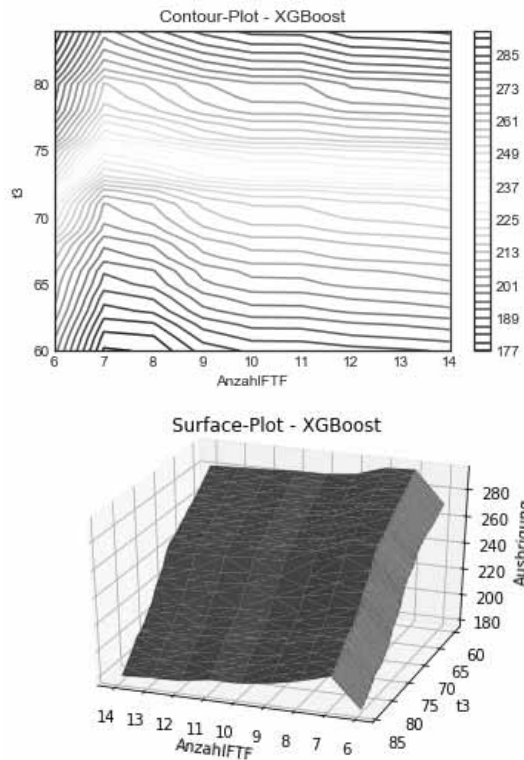


Abbildung 6 XGBoost Contour und Surface plot

Quasi ein Abfallprodukt der Gradient Boost Verfahren ist die Bestimmung der Relevanz der Eingabevariablen. Diese Information ist für den Planer sehr nützlich für die Auslegung der Anlage und der Prozessparameter. Man erkennt die Wirksamkeit der Stellschrauben, also welche Veränderung der Eingangsgrößen wirkt sich wie stark auf das Ergebnis aus.

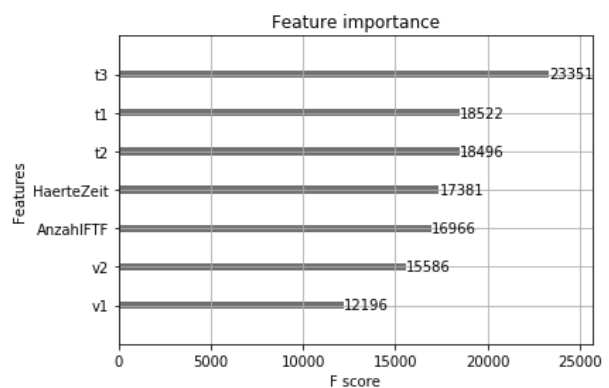


Abbildung 7 XGBoost. Relevanz der Eingabevariablen

4.3 Künstliches neuronales Netz - Keras

Deep learning ist ein Untergruppe der Machine Learning Methoden, die künstliche neuronale Netze verwenden. Das Wort „deep“ leitet sich von der Anzahl Schichten ab, im neuronalen Netz verwendet werden. Wobei das Wort „deep“ relativ ist. Vor 5 Jahren waren 10 Schichten ausreichend, heute wird ein Netz als

„deep“ bezeichnet wenn es hunderte von Schichten enthält.

Keras, ist eine High Level API, die mit vereinfachten Funktionen auf TensorFlow oder Theano aufbaut und die Implementierung von Standardmodellen schnell und einfach gestaltet.

Wenn man viele Schichten im Netz verwendet, steigt die Gefahr des „overfittings“, d.h. das Netz lernt auch das Rauschen in den Eingangsdaten mit. Man benötigt deshalb Regularisierungsverfahren, die beim Training des Modelles die Komplexität einschränken und somit das „overfitting“ verhindern.

Regularierungen können über Verfahren wie Dropout, BatchNormalization, Bagging, „early stopping“ und die Anzahl Schichten im Netz realisiert werden. Für das erstellen und trainieren eines Netzes benötigt man viel Zeit und Erfahrung, da das Verfahren sehr viele Freiheitsgrade aufweist. Z.B. die Wahl der Netztopologie, die Wahl der Aktivierungsfunktionen, Regularisierungsverfahren, Optimierungsalgorithmus und Optimierungsparameter.

Um die optimale Modelltopologie zu bestimmen wurden viele Netze mit einer unterschiedlichen Anzahl an Schichten und Anzahl Neuronen in den Schichten trainiert. Es wurde viele verschiedene Regularisierungs- und Optimierungsverfahren getestet. Die Besten Ergebnisse wurden mit einer Schicht und 50 Neuronen in dieser Schicht erzielt. Als Aktivierungsfunktion wurde „relu“ verwendet. Als Optimierungsverfahren erwies sich „Adam“ als das Beste.

Layer (type)	Output Shape	Param #
dense_99 (Dense)	(None, 50)	400
dense_100 (Dense)	(None, 1)	51
Total params: 451		
Trainable params: 451		
Non-trainable params: 0		

Abbildung 8 Topologie des Netzes

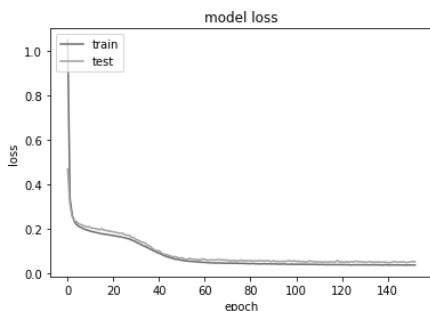


Abbildung 9 Güte der Approximation während des Trainings für Trainingsdaten und Testdaten

Die Anzahl Trainings-Epochen wurden durch „early stopping“ bestimmt. Da einzelne Netze sehr von der

zufälligen Parametrisierung der Gewichte abhängen (lokalen Minima beim trainieren des Netzes), wurden 10 Netze trainiert und die Prognosen über die 10 Netze gemittelt. Dieses Mitteln ist eine sehr gute Form der Regularung. Durch das Mitteln konnte der Fehler (mae) für die 30% Testdatensätze von 0.9 auf 0.61 reduziert werden.

Ergebnisse:

mae für die 30% Testdaten: 0.7023
 mae für 5-fold cross valdierung: 0.6179

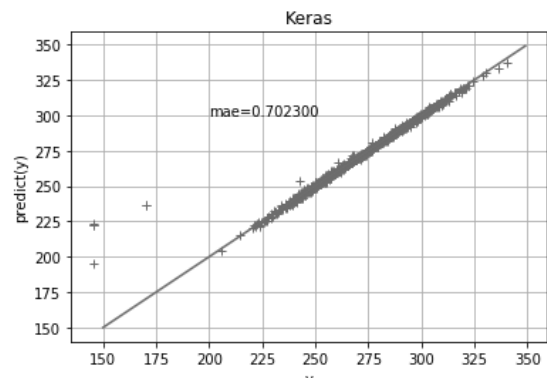


Abbildung 10 Keras-Prognosegüte. Die Darstellung zeigt die Prognose der Ausbringung (predict(y)) vs. der aus der Simulation bestimmten Ausbringung (y)

Die Approximation durch das neuronale Netz ist sehr gut. Nur bei den geringen Ausbringungen (147 und 174) gibt es größere Abweichungen die in der Praxis Probleme bereiten können. Die Abweichungen entstehen durch einen sprunghaften Abfall der Ausbringung bei t1 im Bereich von t1= 29 nach t1= 30. Regressionsfunktionen lernen solche Sprünge nur schwer. Verfahren die Entscheidungsbäume als Basisfunktionen verwenden, wie z.B. XGBoost, können diese Sprünge leichter und besser erlernen.

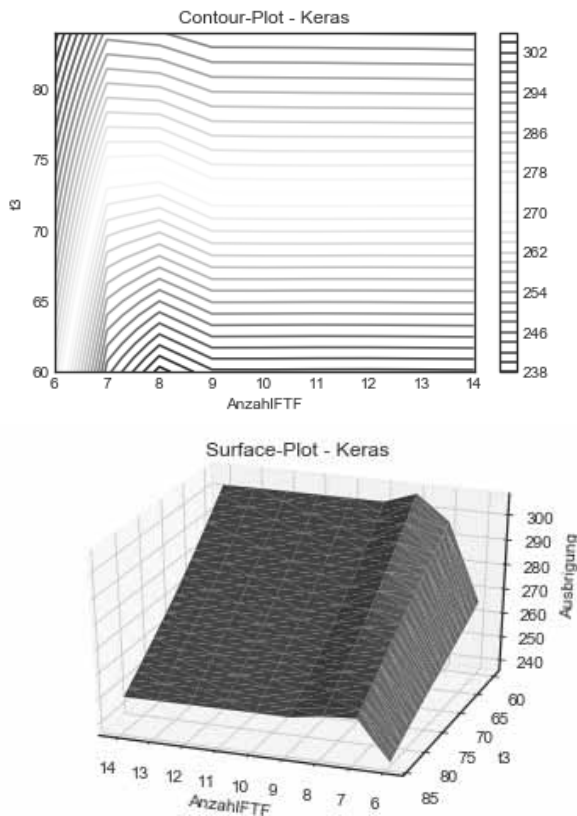


Abbildung 11 Keras Contour und Surface Plot

Die Approximation durch das neuronale Netz weist eine glattere Darstellung auf, als die Approximation durch XGBoost.

5 Zusammenfassung

Die aus den Simulationsdaten bestimmten neuronale Netze (Keras)- und Gradient Boost-Verfahren (XGBoost) -Ersatzmodelle weisen eine wesentlich höhere Genauigkeit auf als herkömmliche lineare Regressionsmodelle.

Damit wird gezeigt, dass sich der größere Aufwand in den Aufbau des Knowhows zur Beherrschung der verfahren und in die Erstellung der Ersatzmodelle lohnt.

mae	Lineare Regression Lasso	Gradient Boost XGBoost	Künstliches neuronales Netz Keras
30% Testdaten	4,6986	0,8513	0,7023
5-fold cross validation	4,6620	0,8560	0,6179

Abbildung 12 Vergleich der Ergebnisse

Darüber hinaus ist ein wesentlicher Faktor, dass beide Verfahren die verrauschten Experimentdaten glätten ohne die Daten zu "overfitten". Das Rauschen in den Eingangsdaten, hervorgerufen durch die Abbildung von stochastischen Prozessen in den Materialflussmodellen wie Beispielsweise Maschinenstörungen, Verfügbarkeit von Personalressourcen (Verteilzeiten) oder zufälligen Ankunftszeitpunkten von externen

Lieferungen etc. Simulationsstudien ohne die Abbildung dieser stochastischen Prozesse machen in der täglichen Praxis keinen Sinn.

Der Vorteil der Gradient Boost Verfahren liegt neben der besseren Approximation darin, dass sich die Relevanz der Eingangsgrößen „feature –importance“ sehr einfach und schnell bestimmen lässt. Oft wird im Planungsprozess die Frage gestellt wie sich Änderungen einzelner Prozessparameter auf das Ergebnis auswirken. Da zum Beispiel während der Planung des Materialflusssystems, die ja als Prämisse eine stabile Planung des eigentlichen Bearbeitungsprozesses voraussetzt, wesentliche Prozessparameter. Zum Beispiel ändern sich Bearbeitungszeiten noch einmal, da sich aus der Vor-Serie heraus neue Erkenntnisse ergeben haben. Mit einer genauen Kenntnis des Wirkungszusammenhangs der Eingangsparameter auf das Ergebnis lässt sich sofort ohne auf aufwändige Simulationsexperimente warten zu müssen, prüfen ob und wie die Planung davon betroffen ist.

Das Erzeugen von Ergebnissen „auf Knopfdruck“ ermöglicht, die Realisierung von Online Optimierungen auf Basis genauer Prozess und Materialflussmodellen im Operativen Betrieb des Produktions- oder Logistiksystems.

6 References

- [1] Averill M Law *Simulation Modeling and Analysis*. McGrawHill, 2017.
- [2] Antonio Gulli, Sujit Pal. *Deep Learning with Keras*. Packt, 2017.
- [3] Trevor Hastie, Robert Tibshirani, Jerome Friedman. *The Elements of Statistical learning*. Springer, 2001.

Implementing the Argesim C21 benchmark with Modelica components

Jan-Philipp Disselkamp, Peter Junglas, Alexander Niehüser, Phillip Schönfelder
PHWT Vechta/Diepholz
peter@peter-junglas.de

The Argesim C21 benchmark requires to investigate three different systems showing state events: a bouncing ball, an RLC circuit with a diode and a rotating pendulum with a free flight phase. These models can be easily implemented in Modelica by coding the complete state and event equations. But the simulation practitioner is usually working in a graphical environment using components from a library, writing explicit Modelica code at most for special components. We will show how to implement such systems in a component based style by providing switchable and event handling components, thereby coping with the limitations of the complicated Modelica event system as well as with some problems of the simulation environments.

1 Introduction

The modelling and simulation of hybrid systems combining a continuous dynamic evolution with state events is still a nontrivial task for most simulation environments. To compare the different approaches the Argesim C21 benchmark [1] defines three systems that show state events with event types of varying complexity: A bouncing ball model using either a simple state change event or a slightly more complex structure change event, an RLC circuit with a diode leading to a parameter or structural change, and a rotating pendulum with a complex structural change between its rotating and free flight phases.

Using Modelica [2] to implement the three models seems to be a simple task at first sight, since Modelica provides a rich set of event related features. In addition the recent synchronous elements [3] define a complementary set of event functions with different semantics. On the other hand Modelica's events are quite involved, sometimes leading to a counterintuitive behaviour [4]. And [5] even claims that Modelica "provides only very limited means" for the description of systems with variable structure, due to restrictions of the Modelica language, which are often related to its equation based foundation.

But Modelica is more than a modelling language: It provides a frame for the definition of components that can be combined with graphical tools to build

up a complex model. Since this is the preferred way how Modelica is used in the industrial practice, it would be highly desirable to have components that make the construction of the example models possible. Some appropriate components are already given by the Modelica Standard Library (MSL), in other cases one has to build them using explicit Modelica code.

In the following we will show several different approaches to implement the example models in a component based way. Since the implementations of Modelica often differ in subtle ways, especially with regard to their event behaviour, we will compare our models using the proprietary programs Dymola from Dassault Systemes and MapleSim from Maplesoft. Finally, the conclusions will address the question, to which extent Modelica is able to model structure changing systems.

2 Bouncing ball

The first example is a falling ball that bounces off the ground. The C21 benchmark contains two different contact models: The event contact describes a time-less bounce that changes the velocity immediately according to

$$v_{\text{after}} = -\mu v_{\text{before}},$$

where the coefficient μ describes the energy loss due to the bounce. The dynamic contact includes the deformation of the ball using a linear spring and damper model.

A simple approach that replicates the standard Simulink method, uses integrator blocks to reproduce the differential equation of the system directly. For the event contact one can utilize limit and reset integrators as shown in Fig. 1. All blocks come from the MSL

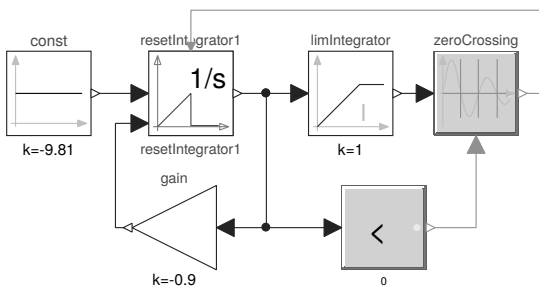


Figure 1: Bouncing ball with event contact (“Simulink”)

except the `ResetIntegrator`, which can easily be implemented with a `reinit` inside a `when-construct`. Since the MSL `zeroCrossing` block triggers on rising and falling input, it must be disabled for positive velocity. This is superfluous if one uses a selfmade `NegativeZeroCrossing` component instead.

To be more in line with the physical modelling spirit of Modelica one can use concepts and components of the `Mechanics.Translational` part of the MSL. One creates a component for each force acting on the falling mass, including a `Hardstop` component that is responsible for the bounce (cf. Fig. 2). Its equations

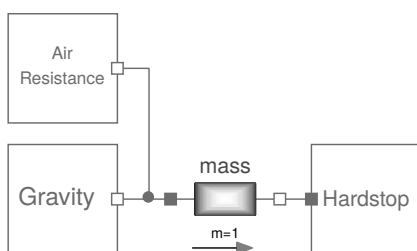


Figure 2: Bouncing ball with event contact (1d mechanics)

are based on the example in [2, pp. 96f]:

```
s = flange.s;
v = der(s);
when s <= 0 then
  reinit(v, -mu*pre(v));
end when;
```

```
flying = not (s <= 0 and v <= 0);
flange.f = if flying then 0 else -m*g;
```

The interesting part is the use of the additional `flying` variable instead of a simple `s <= 0`. This is necessary to cope with the numerical inaccuracies, which otherwise would lead to very small but negative values of `s` after an event and to the infamous fallthrough of the ball near the scattering singularity.

Problematic is the computation of the counterforce $-m \cdot g$ after the flying phase, which needs the mass of the falling ball. Therefore the `Hardstop` component is tightly connected to the `Mass` component and cannot be used as a universally applicable component by its own. A similar problem has already surfaced when trying to include a collision detection mechanism in the Modelica MultiBody Library [6] and has been listed in [5] as one of the obstacles for modelling of variable structure systems in Modelica.

The apparently more complicated dynamic contact can be easily handled in Modelica, since the 1d-mechanics library already contains a corresponding component `ElastoGap`. Actually its equations are a bit more complicated than the ones in the C21 benchmark in order to cope with unphysical effects of the simple equations. But it is a trivial matter to create a copy and strip it down to comply with C21.

3 RLC circuit with diode

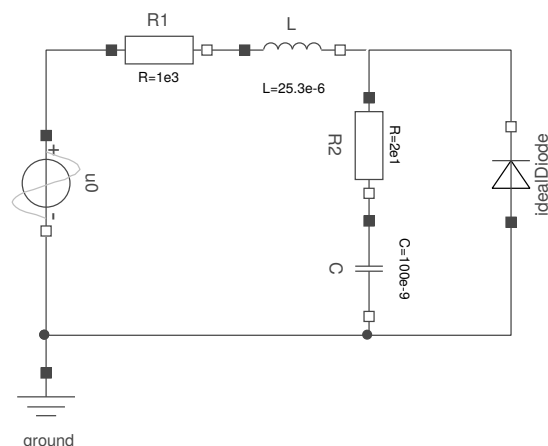


Figure 3: RLC circuit with diode

The second example is a simple RLC circuit with a diode, where different diode models are to be investigated. It is constructed easily with standard compo-

nents from the `Electrical.Analog` part of the MSL (cf. Fig. 3), using the `IdealDiode` for the shortcut diode of C21.

The MSL contains a diode component implementing the common Shockley characteristic, but it differs from the “Shockley diode” of C21, because it has a non-vanishing current in locking phase. Therefore one has to create a special component here. Using a partial model provided by the MSL this is a matter of a few lines:

```
model DIsh11 "diode model after Shockley"
  extends Analog.Interfaces.OnePort;
  parameter Real IS = 1e-8;
  parameter Real UT = 26e-3;
equation
  if v < 0 then
    i = 0;
  else
    v = UT*log(i/IS + 1);
  end if;
end DIsh11;
```

Alternatively the model `DIshu1` uses the inverse relation to define i as function of v in conducting phase.

C21 defines a “real-time” variant, where the algebraic equation for v in conducting phase is replaced by its derivative in order to reduce the DAE equations of the complete model to an ODE. In the component based model used here things are more complicated: If one uses the derivative equation for v

$$\text{der}(v) = (UT/IS)/(i/IS + 1)*\text{der}(i);$$

inside the diode model, the variable i changes from algebraic to state variable, when the diode switches from locking to conducting phase. To cope with this problem, we added another variant, where the equation in locking phase is derivated as well:

$$\text{der}(i) = 0;$$

Another diode variant defined in the benchmark replaces the exponential Shockley characteristic with a piecewise linear interpolated approximation. This can be implemented either graphically using a `LookupTable` component from the MSL together with a `VoltageSensor` and a generic `Diode` model or by implementing the linear interpolation directly with a Modelica function.

4 Rotating pendulum with free flight phase

The last example is a point mass with air resistance on a rope of fixed length. Its movement switches between swinging and free fall phases according to the direction of the force acting on the mass. Since its structure changes completely between states of different dimension, this is the most interesting and demanding system of the benchmark.

Even an entirely equation-based solution is challenging. In [1] three modelling concepts are presented: a hybrid decomposition with subsystems of different dimensions, a maximal state space approach and a DAE of high index that uses cartesian coordinates throughout, but changes the constraint equation. Of these the first one seems hopeless in Modelica: Although the variables that are used as states can change dynamically, their total number is fixed by an equal number of equations. The second method can be applied in a straightforward manner, leading to a working - but monolithic - solution. The third idea seems to be most promising, since DAEs of high index are commonplace in Modelica. But though a corresponding model has been formulated by the authors, none of the available simulation programs could cope with it.

Focusing on component based models, two different ideas have been followed, similar in spirit to the two different solution methods described in [7]: The first is a “local” method using switchable components for the structurally different phases, the other takes a top-down approach to switch globally between two different systems.

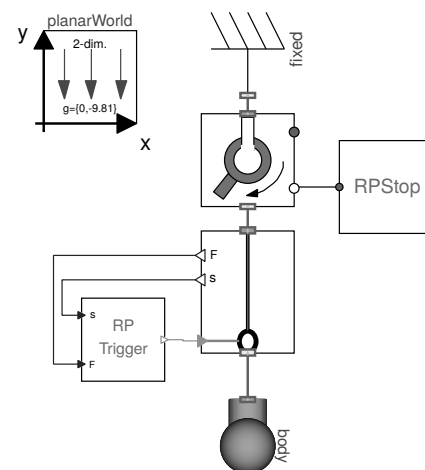


Figure 4: Rotating pendulum model (2d mechanics)

The first model uses graphical components as far as possible, coming from the MSL or from the free Planar Mechanical Library [8], which is used here instead of the Modelica MultiBody library, since it is much simpler and works identically in both simulation programs. The complete model consists mainly of a Revolute joint defining the swinging motion, a selfmade Rope together with a trigger and a Body for the point mass (cf. Fig. 4).

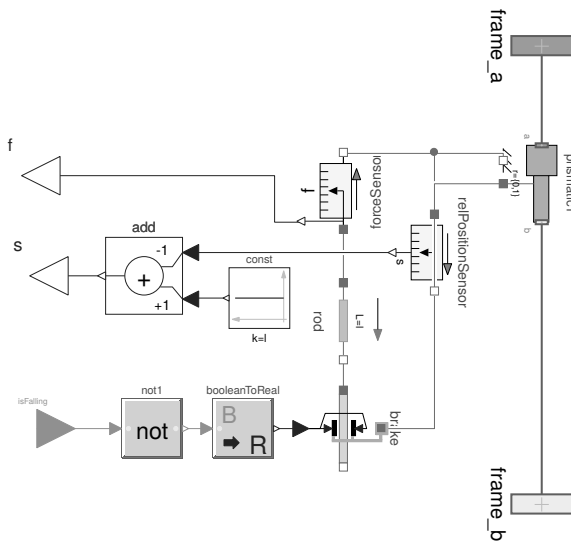


Figure 5: Rope component

The central idea is found in the implementation of the Rope (cf. Fig. 5): It is basically a Prismatic joint which is controlled by a rod of fixed length that is attached to a brake. According to the state of the brake the length of the joint is either fixed or freely variable, thus defining the swinging and falling states of the pendulum. The RPTrigger component uses a standard flipflop [4] to represent the state and changes it according to outward force and slack of the rope.

From a practical point of view the model works fine, but technically it is not a complete solution of the benchmark: The transition from falling to swinging state takes a short time due to the large, but finite force delivered by the brake. Presumably this could be fixed with an event-based brake, but this would destroy the charme of the model that consists almost entirely of standard components.

The second implementation takes a global approach: It consists of separate blocks for the two different system configurations and a SystemSwitch that alternatively activates one or the other system, depending on the state of the active system (cf. Fig. 6). The switch

contains the event functions h^F , h^S as defined in [1] and computes the initial state at a system change.

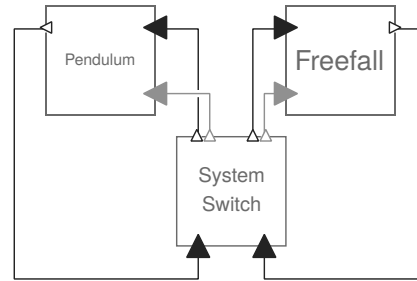


Figure 6: Rotating pendulum model (hybrid version)

To facilitate the implementation of the two systems a partial model SwitchableSystem has been defined that contains the state, the inputs and outputs and the triggering:

```

partial model SwitchableSystem
  parameter Integer N = 2
  parameter Real[N] s0 = {pi/4, 15};
  RealInput[N] newState;
  BooleanInput active;
  RealOutput[N] sOut;
  Real[N] state(start=s0, each fixed=true,
    each stateSelect=StateSelect.always);
equation
  when active then
    reinit(state, pre(newState));
  end when;
  sOut = if active then state else zeros(N);
end SwitchableSystem;
    
```

Then the implementation of a concrete system only needs the definition of the state equations, usually in form of an ODE. But even a graphical approach is possible: One starts with a new component that inherits from SwitchableSystem, thereby getting the necessary input and output connectors automatically. Inside the model one simply ignores these connectors and constructs the system graphically with components from the MSL or other libraries (cf. Fig. 7).

Finally one adds a few lines of explicit Modelica code to identify the (inherited) state variable with corresponding variables from the graphical model. For the pendulum this is as simple as

```

state[1] = pi/2 - revolute.phi;
state[2] = -revolute.w;
    
```

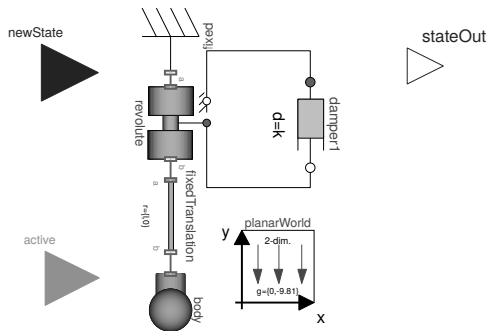


Figure 7: SwitchablePendulum component

The attribute `stateSelect=StateSelect.always` of `state` guarantees that these variables will be used by the solver as the actual states.

This model looks exactly like a hybrid decomposition (cf. Fig. 13 of [1]) and for the purpose of constructing the model it really is one: Both submodels can be created independently and almost in the same way as standalone systems. But formally this again is a maximal state space approach: The variables of an inactive system are simply ignored or their derivatives set to zero. In any case they always exist, enlarge the total state space and have to be computed always albeit trivially. On the other hand Modelica compilers routinely handle large systems with lots of trivial equations, so this should not be a large burden.

5 Differences between Dymola and MapleSim models

All models have been tested with Dymola 2018 and MapleSim 2017.3 under Kubuntu 16.04. MapleSim uses an internal (non-Modelica) format to save the models, but can import Modelica libraries. Therefore all components have been collected in a Modelica library, which is identical for both programs. To make this possible we couldn't use the Modelica MultiBody library, because MapleSim uses its own proprietary version, which is incompatible with the Modelica one. Fortunately the free Planar Mechanical Library [8] works in both programs and was sufficient here. Furthermore it allows to access internal variables, which was useful for some tests, but not possible with MapleSim's MultiBody library.

Except for the differences mentioned in the following and for minor numerical deviations all models

worked identically in both programs. Execution times were always small except for one notable example: The Shockley diode model `DIshi1`, which defines the characteristic as $v(i)$, was about 100 times slower than the $i(v)$ variant, when using MapleSim. This is probably due to a poor choice of the state variable.

Another problem showed up only in MapleSim: When running the hybrid version of the pendulum model using graphically constructed subsystems, the simulation stops immediately with the error message

```
'reinit' applied to an algebraic variable
'Main.switchablePendulum\_1.state[2]'(t),
will have no effect
```

Obviously the solver did not use the `state[2]` variable as a proper state variable. Since this is in conflict with the `stateSelect` attribute, it seems to be a bug.

Much more interesting in the present context is the behaviour of the different “real-time” diodes: The variants using derivated equations for both phases ran in both programs, whereas the versions with the equation $i = 0$ in locking phase worked only in MapleSim. Dymola stopped the simulation, when the first locking phase appeared, claiming to hit upon a singular linear system of equations. At this point the variable i changes from a state to an algebraic variable, which might be the root of the problem.

A very strange bug came up, when we tried to implement a pendulum that gets an additional kick (i. e. a sudden change in angular velocity) as requested in [1]: The first version contained several independent when-clauses with `reinit`s. Dymola claimed this to be non-deterministic and combining two of the `whens` with `elsewhen`, in fact, did the trick and worked properly. But not with MapleSim: It always stopped with an internal error. To make it work one had to delete the `else-branch` in

```
when {state[2] < 0, state[2] > 0} then
  if abs(psi) < psiEnd then
    awaitKick = true;
  else
    awaitKick = false;
  end if;
end when;
```

This is definitively a bug, but more interesting is what happened after this strange workaround: The defective model with independent `whens` runs properly,

but after the elsethen correction has been applied, it misses the kick. This strongly suggests some errors deep in the event system.

6 Conclusions

The basic question is: Can one implement structure-variable systems of different dimension in Modelica? This is a hotly debated topic, and the answers given in the literature are quite different, e. g.

- “There is until now no possibility implemented to make graphical model switching for subsystems with different state space dimension.” [7]
- “Even though ... the set of variables and the set of equations is fixed over time, it is the case that conditional equations in hybrid DAEs can be activated and deactivated. ...
... Thus the active part of the hybrid DAE can be structurally dynamic, i.e., at run-time change the number of active variables and equations in the DAE.” [9].

Even though the authors of the first quote are correct formally, the results presented here emphasize the point of the second one. At least Modelica’s event system is rich enough to provide working and interesting solutions to the tasks of the C21 benchmark, even if one concentrates on graphical modelling.

Nevertheless, the most straightforward way to implement the pendulum example - a DAE system using only cartesian coordinates - did *not* succeed in any program, though it seems to be correct formally. Whether this is a problem of Modelica itself or of the implementations, is an open question.

Which concepts work and which do not can depend on the concrete implementing software. There will always be differences because of bugs, but some of the problems seem to indicate that they are rather due to different interpretations of the exact semantics of the event system. For further progress on structure-variable systems in Modelica, this needs to be clarified in a precise manner.

Acknowledgements

The authors are thankful to the Maplesoft Support for providing the basic idea of the rope model.

References

- [1] A. Körner, F. Breiteneker. *State Events and Structural-dynamic Systems: Definition of AR-GESIM Benchmark C21*. Simulation Notes Europe, 26(2), pp. 117-122, 2016.
- [2] Modelica Association. *Modelica® - A Unified Object-Oriented Language for Systems Modeling - Language Specification Version 3.4, April 10, 2017*. Online: <https://modelica.org/documents/ModelicaSpec34.pdf> (called 2018-01-08).
- [3] H. Elmqvist, M. Otter, S. E. Mattsson. *Fundamentals of Synchronous Control in Modelica*. Proc. 9th Int. Modelica Conf., Munich, pp. 15-26, 2012.
- [4] P. Junglas. *Pitfalls using discrete event blocks in Simulink and Modelica*. Proc. ASIM Symp. STS/GMMS, Lippstadt, pp. 90-97, 2016.
- [5] D. Zimmer. *Enhancing Modelica towards Variable Structure Systems*. Simulation News Europe, 17(2), pp. 23–28, 2007.
- [6] M. Otter, H. Elmqvist, J. Díaz López. *Collision Handling for the Modelica MultiBody Library*. Proc. 4th Int. Modelica Conf., Hamburg, pp. 45-53, 2005.
- [7] G. Zauner, D. Leitner, F. Breiteneker. *Modeling Structural - Dynamics Systems in MODELICA/Dymola, MODELICA/Mosilab and AnyLogic*. Simulation News Europe, 17(2), pp. 49–54, 2007.
- [8] D. Zimmer. *A Planar Mechanical Library for Teaching Modelica*. Proc. 9th Int. Modelica Conf., Munich, pp. 681-690, 2012.
- [9] H. Lundvall, P. Fritzson, B. Bachmann. *Event Handling in the OpenModelica Compiler and Runtime System*. Technical report, PELAB, Dept. Computer Science, Linköping University, 2008.

Regressionsmethoden zur Ersatzmodellbildung und analytischen Beschreibung von Erdstoffmodellen

Christian Richter¹, Frank Will¹,

¹Stiftungsprofessur für Baumaschinen, Technische Universität Dresden
christian.richter1@tu-dresden.de

Die numerische Simulation von Erdstoffen oder granularen Stoffen erfolgt oft mithilfe der Diskreten Elemente Methode (DEM). Ein bisher ungelöstes Problem der DEM besteht in der effizienten und richtigen Bestimmung der Kontaktmodellparameter. Oft wird dabei auf Kalibrierungsverfahren zurückgegriffen, welche die Parameter solange variieren, bis die Simulationsergebnisse hinreichend genau mit der Realität übereinstimmen. Aufgrund sehr langer Rechenzeiten soll die Gesamtanzahl notwendiger Simulationen dabei so gering wie möglich gehalten werden. Eine Möglichkeit dies zu Erreichen ist der Aufbau von Ersatzmodellen. In der nachfolgenden Arbeit soll gezeigt werden, wie verschiedene Regressionstechniken zur Ersatzmodellbildung genutzt werden können. Dabei soll besonders auf die symbolische Regression eingegangen werden. Bei dieser wird unter Nutzung der genetischen Programmierung nach einer analytischen Funktion gesucht, welche die Ergebnisdaten möglichst genau abbildet.

1 Einleitung

Ein wichtiger Bestandteil zur Auslegung von Baumaschinen, wie Radlader oder Bagger, ist die richtige Abschätzung der während des Arbeitsprozesses entstehenden Kräfte und Momente. Diese werden überwiegend durch die Interaktion mit Erdstoffen (z.B. Kies, Sand, Lehm) verursacht. Für eine prospektive Analyse des Erdstoffverhaltens hat sich die Diskrete Elemente Methode (DEM) in den letzten Jahren immer mehr durchgesetzt. Bei dieser Methode handelt es sich um ein numerisches Berechnungsverfahren, welches zur Simulation von Schüttgütern eingesetzt werden kann.

Um ein realitätsgetreues Verhalten des simulierten Materials zu erhalten, müssen vor Beginn der eigentlichen Simulation die richtigen Modellparameter gewählt werden. Diese werden im Rahmen eines aufwändigen Kalibrierungsprozesses ermittelt. Hierzu werden zunächst eine oder mehrere charakteristische Größen des zu untersuchen Materials in Laborversuchen festgestellt. Anschließend wird der gleiche Versuch im Simulationsmodell abgebildet und die Modellparameter solange variiert bis die Simulationsergebnisse mit den real gemessenen Größen übereinstimmen. Aufgrund der sehr langen Rechenzeiten für DE-Rechnungen kann dieser

Prozess mehrere Stunden bis Wochen dauern. Bisher wird für jedes zu untersuchende Material eine neue Kalibrierung durchgeführt. Ersatzmodelle, welche die Zusammenhänge zwischen den Modellparametern und Zielgrößen abbilden, können diesen Aufwand maßgeblich zu reduzieren. Im Rahmen dieser Arbeit soll gezeigt werden, wie solche Modelle mithilfe unterschiedlicher Regressionsmethoden aufgebaut werden können.

2 Schüttgutsimulationen

2.1 Diskrete Elemente Methode

Die Diskrete Elemente Methode wurde erstmals von Cundall und Strack [1] vorgestellt. Mithilfe dieser Methode lassen sich die Bewegungen großer Mengen diskreter, interagierender Objekte untersuchen. Als vereinfachte Modellvorstellung werden diese Objekte dabei oftmals als ideale Kugeln abgebildet. Zur Berechnung der entstehenden Kontaktkräfte sich berührender Kugeln werden sogenannte Kontaktmodelle verwendet. Ein einfaches und oft angewendetes Kontaktmodell ist das lineare Kontaktmodell mit viskoser Dämpfung, welches in Abbildung 1 dargestellt ist. Neben zwei Feder-Dämpfer-Paaren in

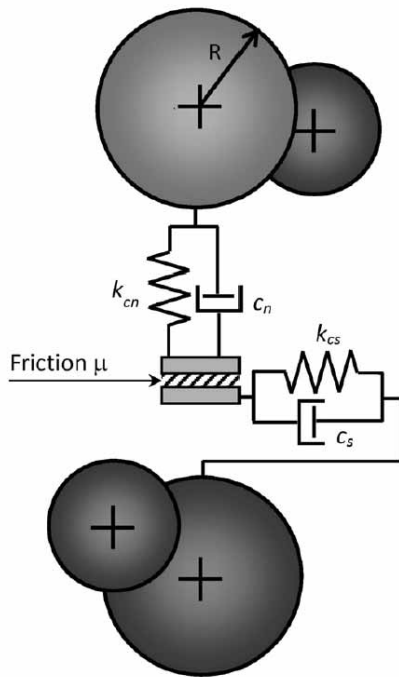


Abbildung 1: Lineares Kontakt Modell mit viskoser Dämpfung (Quelle: [2])

Normal- und Tangentialrichtung, verfügt das Modell über ein zusätzliches Element zur Beschreibung der Gleitreibung zwischen Partikeln und Oberflächen. Die abgebildeten Parameter k_{cn} , k_{cs} , c_n , c_s und μ werden als Kontaktmodell- oder Mikroparameter bezeichnet und haben wesentlichen Einfluss auf das Verhalten des simulierten Materials. Ein weiterer wichtiger Bestandteil vieler Kontaktmodelle, welcher in dieser Darstellung nicht abgebildet ist, aber trotzdem genannt werden soll, ist die die Rollreibung μ_r .

Die effiziente und eindeutige Bestimmung der richtigen Zahlenwerte für die einzelnen Parameter, in Abhängigkeit des zu untersuchenden Schüttguts, stellt eine der größten Herausforderungen und ein bis dato nicht vollständiges gelöstes Problem der DEM dar. Eine weitverbreitete Methoden zur Parametrierung stellt die Nutzung von Kalibrierungsverfahren dar. Im Rahmen von kleinskaligen Laborversuchen werden hierbei zunächst charakteristische Größen des zu untersuchenden Materials ermittelt. Anschließend werden die gleichen Versuche virtuell abgebildet und mehrere Simulationen durchgeführt. Dabei werden die Kontaktmodellparameter solange mithilfe statistischer Methoden oder von Optimierungsalgorithmen variiert, bis die Simulationsergebnisse weitestgehend mit den realen Messwerten übereinstimmen. Ein

Nachteil dieser Vorgehensweise ist, dass sie für jedes zu untersuchende Material neu durchgeführt werden muss.

2.2 Schüttwinkelversuch

Der statische Schütt- oder auch Böschungswinkel, welcher sich aufgrund der inneren Reibungseigenschaften des Materials ausbildet, gilt als charakteristische Größe für nicht-kohäsive Erdstoffe. Dieser kann mittels verschiedener Versuche gemessen werden, wobei sich je nach Versuchsanordnung ein anderer Winkel ergibt. Nachfolgend soll die Schüttwinkelmessung mittels Scherboxversuch betrachtet werden, welche in Abbildung 2 dargestellt ist. Hierzu wird eine rechteckige Box mit dem Schüttgut gefüllt und anschließend eine seitliche Klappe geöffnet, so dass das Material ausfließen kann. Am Ende wird der Winkel α gemessen.

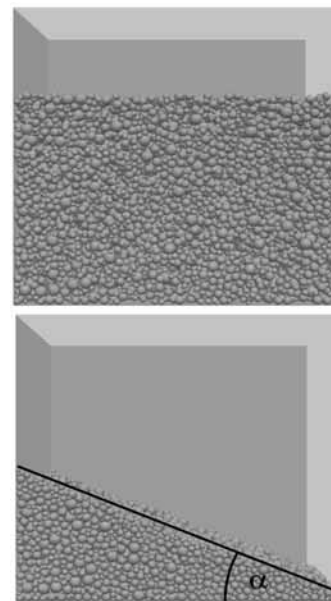


Abbildung 2: Bestimmung des Schüttwinkels α mittels Scherboxversuch; Oben: Box zu Versuchsbeginn; Unten: Versuchsende mit Schüttwinkel α

3 Ersatzmodellbildung

Eine Simulation stellt immer eine Abbildung der Form $\vec{x} \rightarrow \vec{y}$ dar, wobei $\vec{x} = \{x_1, x_2, \dots, x_N\}$ mit $N \in \mathbb{N}$ den Vektor der Simulations- und Modellparameter

darstellt und $\vec{y} = \{y_1, y_2, \dots, y_M\}$ mit $M \in \mathbb{N}$ den Vektor der Simulationsergebnisse.

Ersatzmodelle, welche in der Literatur oftmals auch als Metamodelle oder Surrogates bezeichnet werden, stellen ein vereinfachtes Modell dieser Abbildungsfunktion dar. Der Vorteil von Ersatzmodellen besteht darin, dass diese mit wesentlich weniger Rechenaufwand berechnet werden können.

Der generelle Ablauf einer Ersatzmodellbildung beginnt immer mit der Generierung von Trainingsdaten. Hierzu wird zunächst eine Parametermatrix $\mathbf{X} = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_I\}$, bestehend aus $I \in \mathbb{N}$ Parametervektoren, gebildet und durch Simulationen die zugehörige Ergebnismatrix $\mathbf{Y} = \{\vec{y}_1, \vec{y}_2, \dots, \vec{y}_I\}$ ermittelt. Die Anzahl und Definition der einzelnen Parametervektoren ist dabei meist das Ergebnis einer statistischen Versuchsplanung. An dieser Stelle sei darauf verwiesen, dass die meisten Ersatzmodelle nur den Zusammenhang zwischen den Parametern und einer Ergebnisgröße abbilden. Die nachfolgenden Betrachtungen gehen deshalb von nur einer Ergebnisgröße aus. Bei der von M Ergebnissen müssten demzufolge M Ersatzmodelle gebildet werden.

Anschließend beginnt die Trainingsphase, in der das entsprechende Ersatzmodell mit den Trainingsdaten trainiert bzw. an diese angepasst wird. Nachdem dieser Prozess abgeschlossen ist, kann das Ersatzmodell genutzt werden um die Ergebnisse \vec{y} eines unbekannt Parametervektors \vec{x}^* zu schätzen.

Mittlerweile existiert eine Vielzahl von Methoden und Verfahren, welche zur Ersatzmodellbildung eingesetzt werden können. Zu den bekanntesten Vertretern gehören Künstliche Neuronale Netze (KNN), Radiale Basis Funktionen (RBF), Kriging und Support Vector Machines (SVM). Ein Nachteil all dieser Methoden ist, dass die Anzahl der Daten, welche zum Training der Modelle benötigt wird, meist sehr groß ist. Eine Alternative, welche bereits mit weniger Daten auskommt, sind die Regressionsmethoden auf die nachfolgend näher eingegangen werden soll.

3.1 Lineare Regression

Gegeben sei ein Datensatz bestehend aus einer Parametermatrix $\mathbf{X} = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_I\}$ und einem zugehörigen Ergebnisvektor $\vec{y} = \{y_1, y_2, \dots, y_I\}$. Besteht jeder Parametervektor \vec{x}_i aus mehr als einer Variable so wird dies auch als multiple Regressionsanalyse bezeichnet. Bei der linearen Regression wird ein lin-

earer Zusammenhang zwischen den einzelnen Parametern und der Ergebnisgröße angenommen, welcher durch die folgende Gleichung beschrieben werden kann.

$$y_i = \beta_0 \mathbf{1} + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_N x_{iN} + \varepsilon_i \quad (1)$$

In Matrixschreibweise kann diese auch wie folgt notiert werden.

$$\vec{y} = \mathbf{X} \vec{\beta} + \vec{\varepsilon} \quad (2)$$

Dabei stellt ε einen Fehlerterm dar, welcher die Abweichungen vom Modell beinhaltet. Diese können aus stochastischen Effekten oder statistischen Unsicherheiten stammen. Ohne den Fehlerterm erhält man die folgende Schätzfunktion.

$$\hat{y}_i = \beta_0 \mathbf{1} + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_N x_{iN} \quad (3)$$

Ziel der linearen Regression ist es nun eine Schätzfunktion die Koeffizienten β_0 bis β_N so zu bestimmen, dass die Summe der quadratischen Abweichungen RSS (Residual Sum of Squares) 4, minimal wird.

$$RSS = \sum_{i=1}^I (y_i - \hat{y}_i)^2 \quad (4)$$

Zur Identifikation eines Minimums wird generell ein Gradientenabstiegsverfahren genutzt. Auf die genaue Bestimmung der Koeffizienten wird unter anderem in [3] näher eingegangen.

3.2 Polynomiale Regression

Bei der Polynomialen Regression wird der Zusammenhang zwischen den unabhängigen Variablen $x_0 \dots x_N$ und einer abhängigen Variablen y als Polynom vom Grad K abgebildet. Diese sind allgemein wie folgt definiert.

$$y_i = \beta_0 \mathbf{1} + \beta_1 b_1(\vec{x}_i) + \beta_2 b_2(\vec{x}_i)^2 + \dots + \beta_K b_K(\vec{x}_i)^K + \varepsilon_i \quad (5)$$

Dabei stellen $\vec{b}_1(\vec{x}_i)$ bis $\vec{b}_K(\vec{x}_i)$ die polynomialen Basisfunktionen dar. Für ein Polynom mit zwei Parametern a, b und einem Grad $K = 2$ wäre der Vektor $\vec{b} = \{a, b, a^2, b^2, ab\}$.

Bei genauer Betrachtung der Gleichungen 2 und 5 wird ersichtlich, dass die Polynomiale Regression in Wirklichkeit ebenfalls eine lineare Regression

darstellt. So kann durch die Substitution $x_{in} = b_k(\vec{x}_i)^k$ die Gleichung 5 in 2 überführt werden.

Von entscheidender Bedeutung für die Güte der Approximation eines Ersatzmodells, basierend auf polynomialer Regression, ist die Wahl des Grades K . Allgemein lässt sich sagen, je größer K , desto kleinere Unterschiede zwischen den einzelnen Datenpunkten können abgebildet werden. Allerdings besteht hierbei die Gefahr des sogenannten *overfitting*, was dazu führt dass neue Datenpunkte falsch berechnet werden. Bei zu kleinem K hingegen kann es zu einem *underfitting* kommen. Beide Varianten sind exemplarisch in Abbildung 3 dargestellt.

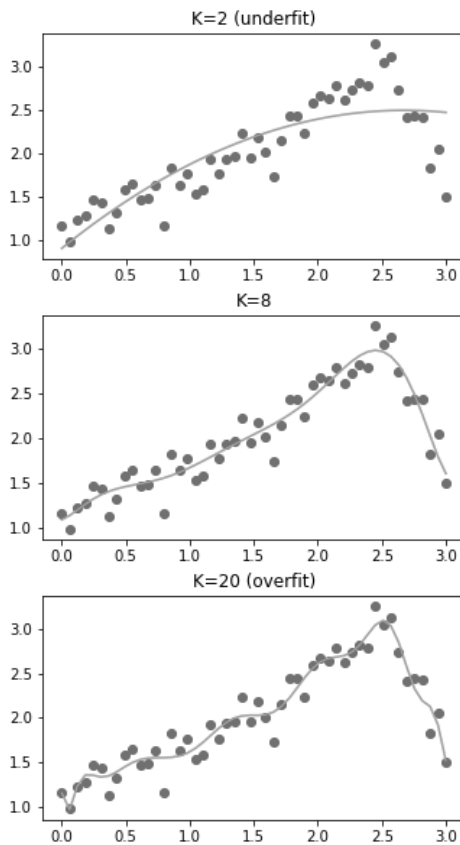


Abbildung 3: Auswirkungen des Grad K bei Polynomialer Regression

3.3 LASSO Regression

Die LASSO Regression (Least Absolute Shrinkage and Selection Operator) stellt eine Art Erweiterung oder auch Verbesserung der Polynomialen Regression dar. Hierbei werden die Linearkoeffizienten

$\beta_k b_k(\vec{x})$ mit geringem Einfluss identifiziert und eliminiert. Erreicht wird dieser Effekt durch eine L1-Regularisierung. Hierzu wird der zu minimierende Fehler RSS wie folgt erweitert.

$$RSS = \sum_{i=1}^I (y_i - \hat{y}_i)^2 + \lambda \sum_{n=0}^N |\beta_n| \quad (6)$$

Diese Erweiterung führt dazu, dass Terme mit kleinen β -Werten zu 0 werden. Über den Parameter λ kann dieser Effekt gesteuert werden.

3.4 Nichtlineare Regression

Bei den bisher betrachteten Regressionsmethoden kamen nur Polynomielle Basisfunktionen der Form $a \cdot x^k$ zum Einsatz. Viele funktionelle Zusammenhänge lassen sich damit nur schwer beschreiben, da sie einen periodischen Charakter besitzen bzw. einem logarithmischen oder exponentieller Trend unterliegen. In diesen Fällen kommt oftmals die nichtlineare Regression zum Einsatz. Hierbei wird, ähnlich wie bei den bisherigen Verfahren, versucht die Koeffizienten bzw. Parameter einer parametrischen Funktion so zu bestimmen, dass ein Fehler zwischen den Schätzwerten und den realen Daten minimiert wird. Die Lösung des Minimierungsproblems erfolgt meist iterativ. Mögliche Basisfunktionen sind zum Beispiel $a \cdot \sin(x)$, $a \cdot \cos(x)$, $a \cdot e^x$ oder $a \cdot \log(x)$. Neben der Verwendung anderer Basisfunktionen erlaubt die nichtlineare Regression auch die Verwendung anderer Operatoren zwischen den einzelnen Termen.

Ein Nachteil der nichtlinearen Regression ist der Umstand das *a priori* eine parametrische Funktion definiert werden muss, für welche die notwendigen Parameter bestimmt werden. Oftmals ist diese Funktion unbekannt, so dass der Benutzer diese durch Interpretation des bekannten Systemverhaltens und auf Basis seiner Erfahrungswerte definieren muss. Dies ist einer der Hauptgründe weshalb die nichtlineare Regression selten zur Anwendung kommt.

3.5 Symbolische Regression

Die symbolische Regression (SR) ist so etwas wie die „Königsklasse“ unter den Regressionstechniken. Das Ziel der symbolischen Regression ist es eine analytische Gleichung zu identifizieren, welche die gegebenen Daten möglichst genau abbildet. Das bedeutet,

im Gegensatz zur nichtlinearen Regression muss eine analytische Funktion nicht *a priori* definiert werden, sondern wird durch das Verfahren selbst ermittelt. Wie bereits in [4] und [5] gezeigt ist die symbolische Regression in der Lage analytisch Zusammenhänge in Datensätzen zu entdecken und so universelle Naturgesetze aufzustellen.

Um eine symbolische Regression durchzuführen existieren grundsätzlich zwei Ansätze. Der erste, welcher an dieser Stelle nur genannt werden soll, ist die *Proritized Grammar Enumeration* (PGE) [6]. Weiter verbreitet ist die Lösung mittels *Genetischer Programmierung* (GP).

Grundgedanke der genetischen Programmierung ist, dass Funktionen als Baumstrukturen dargestellt werden können. Die Operatoren und Basisfunktionen einer Gleichung bilden dabei die Knoten des Baumes und die Parameter und Variablen die Blätter. Ein Beispiel für eine solche Baumstruktur ist in Abbildung 4 dargestellt.

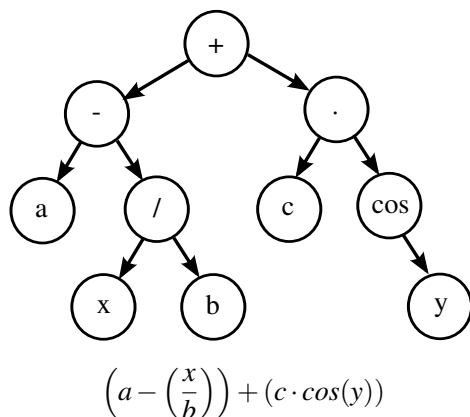


Abbildung 4: Darstellung einer Funktion als Baumstruktur

Zu Beginn einer symbolischen Regression werden zunächst J verschiedene Funktionsbäume zufällig generiert. Dabei dürfen nur Operatoren, Variablen oder Basisfunktionen aus einer zuvor definierten Menge verwendet werden. Die maximale Baumgröße, sowie die Länge bestimmter Äste kann über Parameter gesteuert werden. Nachdem die einzelnen Bäume erzeugt wurden, werden sie evaluiert. Das bedeutet jede Funktion wird auf die Eingangsdaten \mathbf{X} angewendet und die Ergebnisse \vec{y} mit den realen Werten \vec{y} verglichen. Mithilfe des ermittelten Fehlers kann jedem Baum ein Fitnesswert zugeordnet werden. Als Unterklasse der Evolutionären Algorithmen, sind die meisten Konzepte der genetischen Programmierung

an natürliche Evolutionsprozesse angelehnt. Das bedeutet, im weiteren Verlauf werden die einzelnen Bäume mittels Selektion, Rekombination und Mutation immer wieder verändert um den Fehler zwischen den berechneten Daten und den Referenzdaten zu verringern. Der Baum bzw. die Funktion mit dem besten Fitnesswert wird als Ersatzmodell verwendet.

4 Anwendungsbeispiel

Zur näheren Untersuchung der einzelnen Regressionstechniken wurde die oben beschriebene Scherbox-Simulation 100-mal durchgeführt. Dabei wurden der im Kontaktmodell definierte Reibungskoeffizient μ , sowie der Rollreibungskoeffizient μ_r variiert. Anschließend wurden die Daten in einen Trainings- und einen Testdatensatz (67%, 33%) aufgeteilt und jeweils ein Ersatzmodell mittels Polynomialer Regression ($K = 8$), LASSO Regression ($K = 8$) und symbolischer Regression aufgebaut. Die mittels SR identifizierte analytische Gleichung zur Beschreibung des Schüttwinkels α , in Abhängigkeit der Parameter, lautet wie folgt.

$$\alpha(\mu, \mu_r) = \frac{\mu_r^2(\mu^2 - 1) - 0.8\mu - 1.4\mu_r}{\mu\mu_r} + \mu_r^2(3\mu + \mu_r) + 2 \cdot \log(\mu_r) + 42.7 \quad (7)$$

Die drei Regressionsmodelle mit den Trainingsdaten sind in Abbildung 5 dargestellt. Es ist zu erkennen, dass die Polynomiale Regression besonders in den unbekannt Randbereichen des Parameterraumes ein sehr unstetiges Verhalten aufweist. Im Gegensatz dazu besitzt die Schätzfunktion des LASSO Modells in den Randbereichen einen äußerst "glatten" und gleichmäßigen Charakter. Dennoch ist auch hier das Extrapolationsverhalten unzureichend. Dies zeigt sich daran, dass im Parameterursprung bei $\mu = 0, \mu_r = 0$ ein Schüttwinkel von über 20° geschätzt wird. Dies kann nicht sein, da sich bei keinerlei Reibungsverhältnissen ein theoretischer Winkel von 0° einstellt. Das analytische Modell weist hier ein besseres Extrapolationsverhalten auf und ist nach subjektiver Einschätzung den anderen beiden Modellen überlegen.

Zur objektiven Bewertung der einzelnen Modelle wurde anhand der Testdaten der Determinationskoeffizient R^2 für jedes Modell ermittelt. Dieser berechnet sich gemäß Gleichung 8 und ist eine wichtige Kennzahl zur Abschätzung der Anpassungsgüte eines Regressionsmodells. Die Ergebnisse sind in Tabelle 1

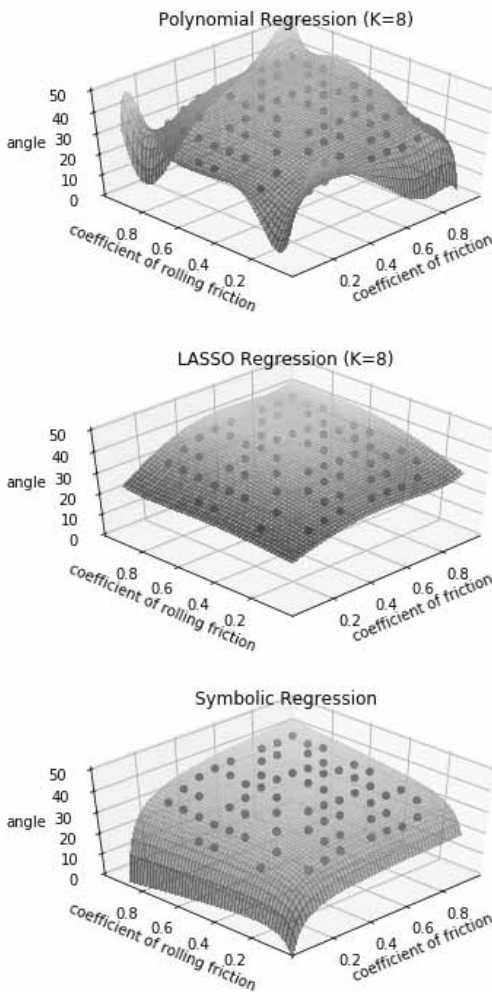


Abbildung 5: Ersatzmodelle zur Schüttwinkelberechnung

aufgelistet. Der Wert für das Polynomiale Modell ist dabei am geringsten, während die anderen beiden Werte in etwa gleich sind.

$$R^2 = \frac{\sum_{i=1}^I (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^I (y_i - \bar{y})^2} \quad (8)$$

5 Zusammenfassung

Der Einsatz von unterschiedlichen Regressionstechniken eignet sich zum Aufbau von Ersatzmodellen für rechenintensive Simulationen. So kann zum Beispiel der Kalibrierungsaufwand von Schüttgutsimulationen mittels DEM wesentlich reduziert werden. Ein bisher

Ersatzmodell	R^2
Polynomial Regression	0.9055
LASSO Regression	0.9757
Symbolic Regression	0.9744

Tabelle 1: Determinationskoeffizient R^2 der Ersatzmodelle

wenig untersuchte, aber vielversprechende Methode stellt die symbolische Regression dar. Diese ist in der Lage analytische Funktionen zu bilden, welche Simulationsingenieuren später als Hilfestellung bei der Modellparametrierung dienen können.

Gefördert durch

DFG Deutsche
Forschungsgemeinschaft

References

- [1] P. Cundall, O.D.L. Strack. *A discrete numerical model for granular assemblies*. geotechnique 29.1, pp. 47-65, 1979
- [2] C.J. Coetzee. *Calibration of the discrete element method and the effect of particle shape*. Powder Technology, p. 50-70, 2016
- [3] A. Nealen. *An As-Short-As-Possible Introduction to the Least Squares, Weighted Least Squares and Moving Least Squares Methods for Scattered Data Approximation and Interpolation*. Online: <http://www.nealen.com/projects>, 130(150), p. 25, 2004
- [4] M. Schmidt, H. Lipson. *Distilling Free-Form Natural Laws from Experimental Data*. Science Journal, 324, pp.81-85, 2009
- [5] M. Schmidt, H. Lipson. *Symbolic Regression of Implicit Equations*. Genetic Programming Theory and Practice VII, Springer US, pp. 73-85, 2010
- [6] A. Worm. *Prioritized Grammar Enumeration A novel method for symbolic regression*. Diss. State University of New York at Binghamton, 2016

Simulation der automatisierten Schaufelbefüllung eines Radladers mit Modelica

Volker Waurich, Prof. Frank Will
Stiftungsprofessur für Baumaschinen, TU Dresden
volker.waurich@tu-dresden.de

Die Entwicklung von Simulationsmodellen für automatisierte Erdbewegungsmaschinen umfasst sowohl das Maschinensystem, als auch das Modell des Prozesses. Während die Modellierungssprache Modelica ideal für die Beschreibung von mechatronischen Systemen geeignet ist, erfordert die Maschinen-Boden-Interaktion jedoch spezifische Lösungen, welche in Modelica über externe C-Schnittstellen realisiert werden können. Am Beispiel eines automatisierten Radladers, wird der Werkzeugeingriff beim Einfahren in ein Haufwerk simuliert. Die Maschine ist dabei mit Standard-Modelica Komponenten aufgebaut. Um eine effiziente Berechnung der Reaktionskräfte des Werkzeugeingriffs im Schüttgut zu erzielen, wurde das analytische Grabkraftmodell als externe C-Bibliothek umgesetzt. Des Weiteren ist die Erkennung einer dreidimensionalen Umgebung elementar für die Auslegung eines automatisierten Fahr- und Arbeitsprozesses. Solch eine Umgebungserkennung wird über eine Kopplung zu einer 3D-Spiele-Engine realisiert. Ein virtueller Sensor übermittelt Messgrößen an das Maschinenmodell und bildet so ein reales Umgebungssensoriksystem simulativ ab. Für die Pfadplanung bei der automatisierten Schaufelbefüllung wird ein zeitlich-diskreter Zustandsautomat genutzt, welcher mit dem neuem Modelica-Sprachstandard verfügbar wurde.

1 Einleitung

Der Einsatzbereich von mobilen Erdbewegungsmaschinen, wie beispielsweise Radladern, ist äußerst vielfältig. Neben Fahr- und Transportaufgaben werden vor allem Gewinnungsvorgänge von unterschiedlichen Schüttgütern realisiert. Ein effizienter Betrieb wird maßgeblich durch den Fahrzeugführer bestimmt. Daher bietet die Automation von einzelnen Arbeitsprozessen großes Potenzial zur Effizienz- und Produktivitätssteigerung.

1.1 Automation eines Radladers

Als Beispielanwendung wird die automatisierte Schaufelbefüllung mit einem Radlader aus einem Haufwerk mit homogenem, feinkörnigem Schüttgut betrachtet. Für einen effizienten Arbeitsvorgang sind diverse Faktoren zu beachten, wie Energieverbrauch, technologische Leistung (geladene Masse pro Zeit), Verschleiß und Arbeitssicherheit. Während eines Arbeitsspiels muss der Radlader in das Haufwerk einfahren, mit der Werkzeugschneide das Material lösen, die Schaufel befüllen und aus dem Haufen herausfahren.

Dabei wird der Vorschub über den Allradantrieb und eine Hub- sowie eine Kippbewegung der Schaufel über Hydraulikzylinder realisiert. Abbildung 1 zeigt die externen Lasten eines Radladers bei der Einfahrt in das Haufwerk. Die Hydraulikzylinder üben die Kräfte F_{tilt} und F_{lift} auf die Arbeitsausrüstung aus. Der Vorschub wird über das Antriebsdrehmoment T_{drive} am Rad erzeugt, welches sich auf dem Boden abstützt und die Gegenkraft F_{trac} kompensiert. Die Prozesslast F_{dig} ist eine zusammengesetzte Kraft aus Schnittkraft, Reibungskraft, Gewichtskraft und Kohäsionskraft [3].

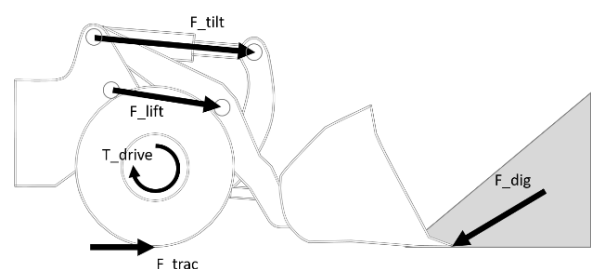


Abbildung 1 Externe Lasten am Radlader während einer Haufeneinfahrt

Entscheidend für einen energieeffizienten Prozess ist die Trajektorie durch das Schüttgut. Die Reibung am Werkzeug und im Erdstoff sowie die Bewegung des Schüttguts durch den Haufen sind maßgeblich für den Energieverbrauch. Je nach Schnittbewegung variiert die geladene Masse. Weiterhin ist zu betrachten, dass der Schlupf an den Rädern nicht zu groß wird um unnötigen Verschleiß zu vermeiden. Dementsprechend existieren verschiedene Strategien um die Schnittführung durch den Haufen zu gestalten [1]. Als guter Anhaltswert ist der „slicing cheese“ Ansatz gewählt worden. Dabei wird versucht mit einer konstanten Spandicke das Material von Einstechhöhe bis Aushubhöhe kontinuierlich aufzunehmen. Bei diesem Ansatz bleibt die Eindringtiefe konstant und die Schnittkraft möglichst gering. Zudem wird wenig Erde durch das Haufwerk transportiert.

1.2 Modellierungswerkzeug

Objektorientierte, gleichungsbasierte Modellierungssprachen wie Modelica [2] sind sehr gut geeignet um Multidomänenmodelle abzubilden. Auch die hydraulischen und mechanischen Subsysteme von mobilen Arbeitsmaschinen wie Radlader oder Bagger lassen sich hinreichend gut abbilden und simulieren. Um Funktionalität und Systemverhalten im konkreten Einsatzfall zu untersuchen, wird weiterhin ein Prozessmodell benötigt, welches auf das Maschinenmodell rückwirkt und eine Prozesssteuerung, welche in der Realität durch den Bediener erfolgt. Das Prozessmodell kann ebenfalls als kontinuierliches, gleichungsbasiertes Modell in Modelica formuliert werden. Zur Abbildung der Prozessstrategie wurde ein Zustandsautomat entwickelt, was seit dem Sprachstandard 3.3 auch in Modelica möglich ist. Als Simulationsumgebung wurde der freie Modelica-Compiler OpenModelica genutzt.

2 Modellkomponenten

Ziel der Simulation ist der Entwurf einer Prozesssteuerung für die automatisierte Schaufelbefüllung. Ausgangspunkt ist das domänenübergreifende Modell von Maschine und Prozess. Die Steuerung der Maschine wird von einer Regelstrategie übernommen. Diese erhält Informationen über den Maschinenzustand und über die Geometrie des Haufwerks und errechnet daraus die Vorgaben für die Aktorik. Virtuelle Sensoren zur Geschwindigkeitsermittlung sind einfach implementiert (Wobei zu beachten ist, dass die Messung einer Absolutgeschwindigkeit in Realität zusätzliche Messtechnik erfordert) und können ohne großen Auf-

wand der Prozesslogik verfügbar gemacht werden. Anders ist es bei der virtuellen Abbildung einer Haufwerkserkennung. Auf der realen Maschine soll ein 3D-Lidar-Sensor zur Erkennung der Haufwerkoberfläche eingesetzt werden. Dieser erzeugt große Datenmengen, deren Auswertung nicht trivial ist. Somit darf dieser Aspekt des Systementwurfs nicht vernachlässigt werden, sondern muss Teil des Modells sein. Bisher haben Modelica-Werkzeuge keine native Unterstützung zur komplexen 3D-Modellierung. Als Ausweg wurde eine detaillierte 3D-Szene in der Spieleentwicklungsumgebung Unity entworfen und ein virtueller Lidar-Sensor implementiert. Dadurch kann die Prozesssteuerung auf eine simulierte Umgebung zurückgreifen. Der Ablaufplan zur Prozesssteuerung selbst wurde ebenfalls in Modelica umgesetzt. **Abbildung 2** stellt die Komponenten des Simulationsmodells dar.

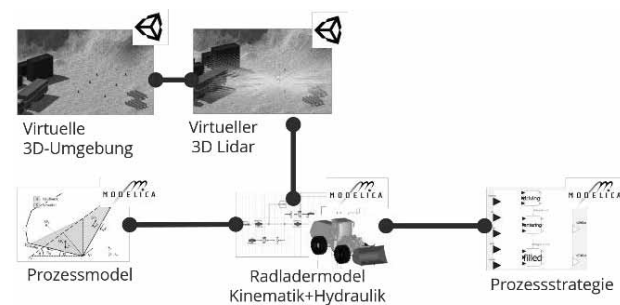


Abbildung 2 Modellkomponenten

3 Modellierung des Haufwerks

Im Gegensatz zum Maschinenmodell, bei dem Elemente der Modelica-Standardbibliothek sowie einer Hydraulikbibliothek grafisch verknüpft werden, wird das Erdstoffverhalten rein algorithmisch formuliert. Eine Anforderung ist, dass beim Einstechen in das Haufwerk Material in die Schaufel transportiert wird und diese geometrische Änderung während der Simulation bestehen bleibt. Aus der Literatur bekannte Grabkraftmodelle [3] sind zweidimensional, weswegen der Haufen ebenfalls in 2D modelliert wird, wobei die Haufwerksebene der Mittelebene der Schaufel entspricht. Die Oberflächenkontur des Haufens wird durch diskrete Punkte definiert, welche die Haufwerksabschnitte begrenzen. Sobald die Schneide der Schaufel einen neuen Haufwerksabschnitt unterhalb der Oberflächenkontur passiert, kann die Eindringtiefe aus der Differenz zwischen Schneidenposition und Oberflächenpunkt berechnet werden, um die Schnittkraft zu bestimmen. Verlässt die Schneide den Haufwerksabschnitt, wird der Oberflächenpunkt auf Schneidenposition verschoben und das überstrichene

Volumen der geladenen Masse zugeschlagen. **Abbildung 3** zeigt eine Visualisierung des Haufenmodells mit einer Schaufel im Eingriff. Die Markierung im Haufwerk stellt die Sollgrößen für den Schnittverlauf dar.

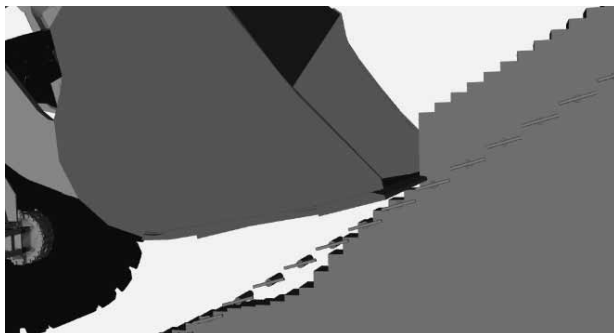


Abbildung 3 Diskretes Haufwerksmodell mit persistentem Volumen. Die Markierungen zeigen die Sollschnittkontur an.

Das persistente Volumenverhalten ist zeitlich diskret, was bedeutet, dass beim Eintritt in ein neues Haufen-segment ein state-event ausgelöst wird. State-events erfordern eine Nullstellensuche und sind somit nicht echtzeitfähig und performant. Eine kontinuierliche Modellierung unter Nutzung der Eindringtiefe und gleichzeitiger Verschiebung der Oberfläche ist nicht möglich, da Modelica solch ein Verhalten in der kontinuierlichen Zeitdomäne nicht erlaubt. Dies ist auch als „Online-Minimum Problem“ bekannt [4]. Als Alternative bietet sich die Modellierung als externe C-Funktion an. Dazu wird ein C-Objekt mit den Oberflächenpunkten angelegt. Das C-Objekt gibt die Schnitttiefe zurück und manipuliert die Haufwerkskontur ohne ein state-event auszulösen. Dies führt zu einer 2.5-fachen Laufzeitverbesserung im Vergleich zum zeitlich diskreten Ansatz. Als Einschränkung für den gewählten Ansatz bleibt jedoch, dass kein numerischer Zeitintegrator genutzt werden darf, welcher Zeitschritte verwirft, da invalide Zeitschritte nicht im externen C-Objekt erkannt werden können. Somit besteht nur noch die Möglichkeit Löser mit fester Schrittweite zu nutzen.

4 Virtuelle Umgebungssensoren

Um mit einem Simulationsmodell zu experimentieren, müssen Randbedingungen festgelegt werden. Für die Bewertung des Systemverhaltens in verschiedenen Betriebszuständen sind meist skalare Randbedingungen mit konstantem oder zeitlich variablen Zeitverhalten ausreichend. Für Simulation von automatisierten Fahrfunktionen oder autonomen Fahrzeugen sind die Randbedingungen jedoch komplexer. Die Erkennung der

Umgebung im dreidimensionalen Raum ist für automatisierte, mobile Maschinen zwingend erforderlich. Die Modellierung umfangreicher 3D-Umgebungen ist aber nur mit großem Aufwand in Modelica-Simulationssoftware möglich. Zur Kopplung von 3D-Umgebung und Maschinenmodell wurde deshalb eine Schnittstelle zur Spieleengine Unity implementiert. Unity bietet die Möglichkeit umfangreiche, dreidimensionale Umgebungen zu visualisieren. **Abbildung 4** stellt dar, wie diese Umgebungsinformationen in Modelica genutzt werden können.

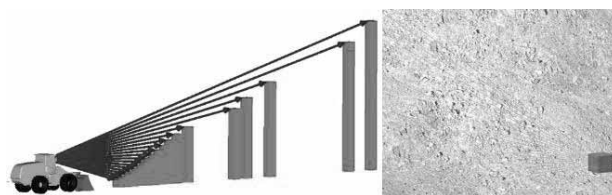


Abbildung 4 links: Modelica-Modell des Radladers mit gemessenen Haufwerkspunkten. Rechts: 3D-Szene in unity mit virtuellem Abstandssensor.

Der Datenaustausch wird über UDP-Verbindungen realisiert. Das Maschinenmodell übermittelt die aktuelle Pose des Sensorelements an Unity. Ein *gameObject* wird dementsprechend in der Szene bewegt. Dieses Game-Object ist in der Lage über sogenannte *raycast*-Funktionen, den Abstand zum Terrain in einer angegebenen Richtung zu ermitteln:

```
Physics.Raycast(startPosition, direction, out hit, maxRange)
```

Code 1. Raycast-Befehl in unity

Die Raycast-Messungen können über einen Winkelbereich wiederholt werden um so einen 2D- bzw. 3D-Lidar abzubilden. Die gemessenen Abstandswerte werden anschließend wieder über UDP-Nachrichten an das Modelica-Modell geschickt. Zur Visualisierung der erhaltenen Daten wurden in **Abbildung 4** die Strahlen mit blauen Pfeilen dargestellt und das zugehörige Höhenprofil mit roten Quadern. Der Messbereich des virtuellen Lidars lässt sich beliebig parametrisieren, wodurch auch reale Lidarsysteme virtuell abgebildet werden können. Am Beispiel eines Velodyne VLP16 [5] wurde ein virtueller Sensor implementiert, welcher die Messergebnisse im original Velodyne-Datenformat via UDP verschickt. Für Hardware-in-the-Loop Anwendungen kann somit ein authentisches Sensorsignal eingebunden werden, welches von einer virtuellen 3D-Umgebung erzeugt wird. **Abbildung 5** zeigt eine Unity-Szene im Echtzeit-Viewer VeloView[6].

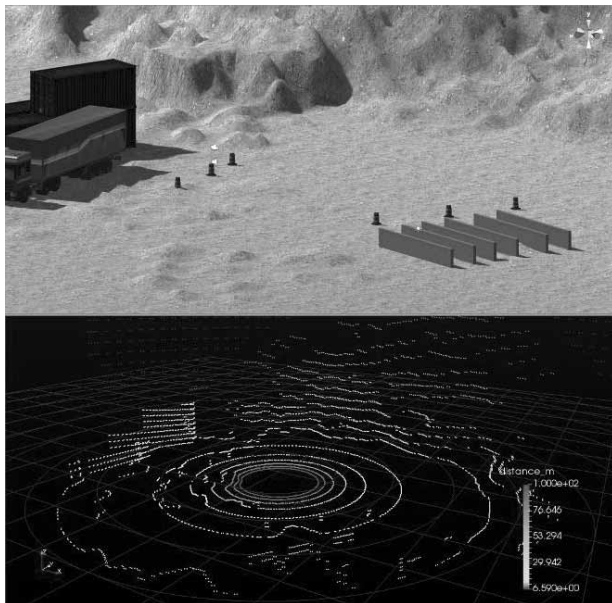


Abbildung 5 oben: 3D-Umgebung in unity mit virtuellem 3D-Lidar, unten: Visualisierung der virtuellen Sensordaten in VeloView

5 Prozessstrategie

Das Simulationsmodell ist in der Lage, die Maschine im Arbeitsprozess abzubilden und die Arbeitsumgebung zu erkennen. Unter diesen Voraussetzungen ist es möglich, einen Algorithmus zur automatisierten Schaufelbefüllung modellbasiert zu entwickeln. Für die Strukturierung dieser Prozessstrategie bietet sich ein Zustandsautomat an. Seit dem Modelica Sprachstandard 3.3. sind sowohl zeitlich getaktete Systeme als auch diskrete Zustandsautomaten mit der Modelicasprache nativ abbildbar. **Abbildung 6** zeigt wie ein Zustandsautomat im OpenModelica-Editor dargestellt wird.

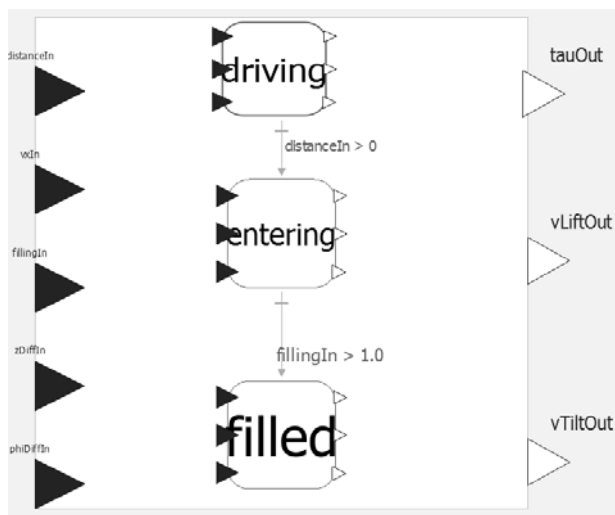


Abbildung 6 Graphische Darstellung eines Zustandsautomaten in Modelica

Zu sehen sind drei Zustände: driving, entering, filled, welche über Transitionen miteinander verbunden sind. Im driving-state beschleunigt der Radlader bis zu einer gewählten Geschwindigkeit. Die Schaufel befindet sich dabei auf dem Untergrund. Der Abstand zum Haufen wird kontinuierlich gemessen. Tritt die Schaufel in den Haufen ein, so wird die Schaufel entlang einer errechneten Schnittkontur geführt. Die überstrichene Haufenfläche gibt Anhaltswerte über den Füllgrad der Schaufel. Dieser Wert wird genutzt, um die Transition in den Zustand filled einzuleiten. Dabei wird die Schaufel angekippt und der Radlader fährt rückwärts aus dem Haufen. Damit ist der Ladevorgang abgeschlossen.

6 Fazit und Ausblick

Systemsimulation ist ein geeignetes Werkzeug um die Funktionsweise von Maschinen abzubilden und zu untersuchen. Für die Entwicklung von automatisierten Funktionen, Prozessinteraktion und Umgebungserkennung muss das Systemmodellsmodell um neue Funktionalitäten ergänzt werden. Dazu gehören beispielsweise Prozessmodelle als externe C-Implementierung und die Interaktion mit komplexen Umgebungsszenarien über Netzwerkverbindungen. Für die Modellierung von Prozessabläufen und Steuerlogik, erlaubt der aktuelle Modelica-Sprachstandard die Beschreibung von Zustandsautomaten.

Zukünftig müssen die Entwicklungswerkzeuge den Anforderungen bei der Entwicklung autonomer Maschinen nachkommen. Das umfasst die einfache Interaktion mit anderen Simulationsumgebungen (bspw. virtuelle Sensoren) und Hardware (HiL) über einsatznahe Protokolle (bspw. CAN, UDP). Mit definierten Schnittstellen und geeigneten Modellbibliotheken lässt sich so die Entwicklungszeit weiter verkürzen.

7 Fördervermerk

Diese Maßnahme wird mitfinanziert mit Steuermitteln auf Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushalts.

8 References

[1] Filla R. und Frank B. *Towards Finding the Optimal Bucket Filling Strategy through Simulation*, Proceedings The 15th Scandinavian International Conference on Fluid Power, SICFP'17, 2017

- [2] Modelica Association, *Modelica® -A Unified Object-Oriented Language for Systems Modeling Language Specification Version 3.4* Verfügbar unter: <https://www.modelica.org/documents/ModelicaSpec34.pdf>
[05.01.2018]
- [3] Kunze G. und Göhring H. und Jacob K. *Baummaschinen*, Vieweg Verlag, Deutschland, 2002
- [4] Beutlich T. Experiments and an FMI Idea on the Online-Minimum Problem, Verfügbar unter: http://www.modelisax.de/wp-content/uploads/2013/06/Modelica_FMI_Last.pdf,
[05.01.2018]
- [5] Velodyne: <http://velodynelidar.com/vlp-16.html>, [11.01.2018]
- [6] Kitware: <https://github.com/Kitware/VeloView>,
[11.01.2018]

Modeling and Simulation Based Development of Autonomy Features for Drones

Shihui Chen¹, Umut Durak^{1,2}, Sven Hartmann²

¹TU Clausthal, Institute of Informatics

²German Aerospace Center (DLR), Institute of Flight Systems

shihui.chen@tu-clausthal.de

Abstract: In the last decade, more and more aerial robotics researchers show interests in developing autonomy features for drones to solve problems in different areas. But the development of autonomy features is complex and labor intensive. Accordingly, Model-based design and simulation-based verification is becoming an industry standard in development of autonomous airborne systems. This we call modeling and simulation-based development. However, commercial model-based design and simulation tools and supporting testing environments require a considerable amount of investment. In order to provide a more economic and efficient solution, this paper investigates a pipeline for modeling and simulation-based development of autonomy features for drones using open source software and hardware stacks. In this context, a generic drone architecture is being designed based on open source hardware platforms, namely CC3D and Raspberry Pi. In the software stack, LibrePilot, an open source software suite to control multicopters is extended to support the designed architecture. The design of the autonomy features is developed using the model-based design in Scilab/Xcos. Xcos Re-useable and Customizable Code Generator is utilized for automatic code generation. The software stack will also include a generic plant model. The workflow starts from autonomy feature modeling and ends with flight testing through Model-in-the-Loop (MiL) testing, Software-in-the-Loop (SiL) testing, target deployment, Hardware-in-the-Loop (HiL) testing. The approach is demonstrated with a simple case study about an autonomous landing feature.

1 Introduction

1.1 Overview

An aircraft without a human pilot aboard is called an unmanned aerial vehicle (UAV), commonly known as a drone. A UAV is regarded as an essential part of an unmanned aircraft system (UAS). The other parts of the system are ground control system (GCS) and the communication system between the UAV and GCS. The usage of UAS has increased sharply in the recent years. Various researchers have developed different autonomy features of drones to solve problems in many fields, such as health care emergency response. Especially in some dangerous situations, UAS that incorporate a high level of autonomy has the ability to accomplish the missions more efficiently without risking lives.

The architecture of autonomous systems is very im-

portant. It is regarded as a method to structure the algorithms for creating functionalities. Figure 1 depicts the a general autonomy architecture [15] for UAS.

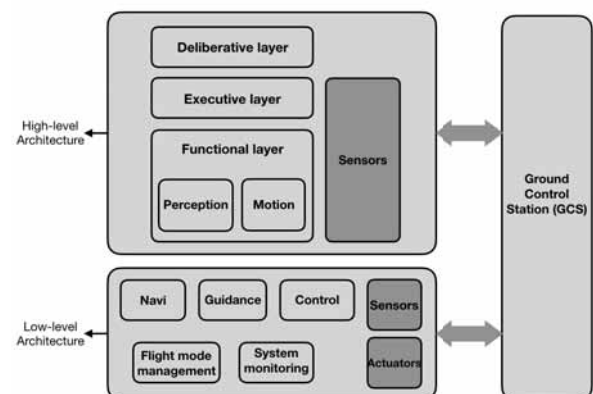


Figure 1: General autonomy architecture for UAS (Adapted from [15]).

The low-level architecture implements the basic functions like navigation and control algorithms that keep the UAV stable in the air and listen to commands from the high level. In this paper, the low-level architecture is realized using OpenPilot CC3D controller [5].

Clough [4] and Merz [13] have elaborated the difference among automatic, autonomous and intelligent systems. An automatic system will exactly do as the programmings say while an autonomous system has the capabilities to make decisions for achieving the missions. An intelligent system can do whatever an autonomous system does and it can produce the goals by its own motivations without any instructions and influence from the outside world. In this paper, system development towards autonomous drones will be discussed.

1.2 Modeling and Simulation-based Development

Model-based design [8] and simulation-based verification are becoming an industry standard in development of autonomous airborne systems. This we call modeling and simulation-based development (Figure 2).

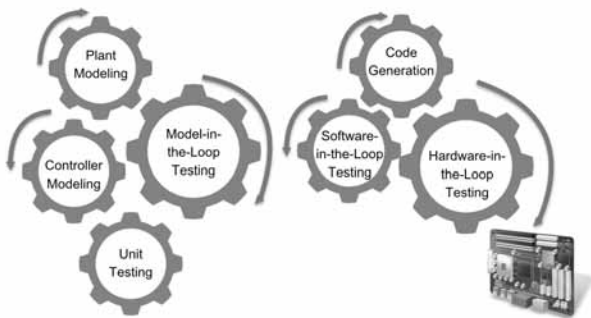


Figure 2: Modeling and simulation based development.

A plant [9] is often desired with a transfer function which indicates the relation between the input signals and the output signals of a system without feedback, commonly determined by physical properties of the system. Usually a plant model is identified by collecting and processing raw data from the real world. We could define the plant model by using mathematical equations or creating a block diagram model that implements known differential-algebraic equations governing plant dynamics. This is called plant modeling.

The mathematical model conceived from the plant is

applied to identify dynamic characteristics of the system. According to those characteristics, a control algorithm that can be executed under the condition which the physical processes are controllable is derived and a suitable controller is chosen. The controller has two levels, the supervisory control that determines the mode transition structure and the low-level control that decides the time-based inputs to the plant [12]. For UAS, low-level controller corresponds to low-level architecture and supervisory control is a part of high-level architecture.

To verify the system flexibility, we use simulation-based verification. Each unit and subsystem should be tested and finally achieve a Model in-the-Loop (MiL) testing. Then it leads to Code Generation (CG) [7]. Software in-the-Loop (SiL) testing follows CG to verify the generated code by checking its conformance to the model. Then it comes to the Hardware in-the-Loop (HiL) testing where the generate code is tested using the target hardware.

2 Autonomy Feature Development Pipeline

2.1 Architecture

This paper aims at proposing a process to enable developing complex autonomy features for drones by using open source software and hardware.

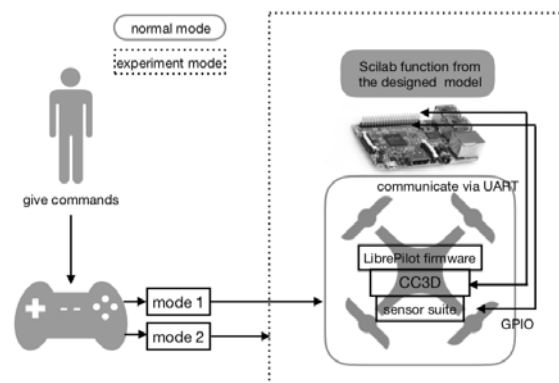


Figure 3: Testbed architecture.

In this paper, regarding their availability, accessibility, cost and flexibility, 250mm class racing drones

are used as testbeds. Figure 3 illustrates the testbed architecture. CC3D running the LibrePilot firmware is utilized as the Flight Controller (FC). In the normal mode, users can give command to the Remote Controller (RC) to control the drone. Once the flight mode is switched to experiment mode, Raspberry Pi 3, the target hardware platform of the testbed, will take charge of the controlling while the RC will be disabled. Scilab/Xcos will be utilized as the model-based design and simulation environment. Xcos Reusable and Customizable Code Generator¹ is used for generating Scilab scripts from Xcos model. For the use case, the generated Scilab script for the autonomous landing feature is deployed to Raspberry Pi 3. It gets data from ultrasonic distance sensor and CC3D controller, computes the next command and send it to CC3D controller as the new command. The communication between Raspberry Pi 3 and CC3D is physically established by Universal Asynchronous Receiver/Transmitter (UART) interface where Sci-Py [11] is used to execute Scilab scripts on target platform through. Related Python libraries are used for interface implementations. UART can control the series device that attached to the computer interface. It provides the computer with the RS-232C Data Terminal Equipment (DTE) interface so that it can "talk" to and exchange data with modems and other serial devices.

2.2 Open source Software and Hardware Stacks

Scilab/Xcos Scilab [3] is a open source software for numerical computation providing a powerful computing environment for engineering and scientific applications. It is a platform to be utilized for model simulation, loading, design, saving and compilation using a graphic editor called Xcos. Xcos has some core features like standards palettes and blocks, model building and modification, model customization and simulation. For some special requirement blocks which are not provided in the Xcos palette browser, users can create their own module by toolbox skeleton to achieve the specific goals.

LibrePilot LibrePilot² is an open source research project which focuses on research and development of software and hardware to be utilized for different

¹<https://forge.scilab.org/index.php/p/xcos-code-generator/>

²<https://www.librepilot.org>

applications like vehicle control and stabilization, unmanned autonomous vehicles and robotics.

LibrePilot includes hardware and software elements (Figure 4). In the hardware side, UAV is mainly controlled by a RC called transmitter. The transmitter has a paired receiver for signal receiving. This receiver also connects with the FC and directly control the actuators. The role of FC is to interpret the control command from RC and runs control algorithm and flight code on the aircraft. If FC is connected to PC where runs LibrePilot Ground Control Station (GCS), users are able to monitor and log flight telemetry data of their vehicle in a real-time environment. This is the software system of LibrePilot which includes GCS software and flight firmware. The flight firmware is implemented in C and C++ using the FreeRTOS [1] embedded real time operating system and typically runs on ARM architecture micro controllers. The communication between GCS and FC is implemented via UAVTalk protocol.

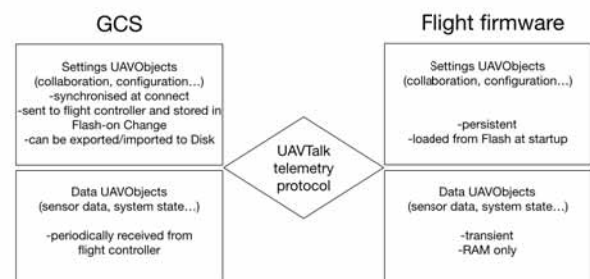


Figure 4: Elements of LibrePilot (adapted from [6]).

UAVTalk is a highly efficient, extremely flexible and completely open binary protocol designed specifically for communication with UAVs [6]. It implements the low level communication between the GCS and the autopilot. It acts as a transportation tool for the data structures defined by the UAVObjects, a data container written in XML format for all of the telemetry data. This protocol does not need to know the details of the data structure, its mission is to send byte arrays and routing received byte arrays to specified object for dealing with the data. For example, all of the RC commands are stored in an UAVObject called ManualControlCommand. Meanwhile, the states of the vehicle can be easily accessed from UAVObjects including accelerate states and attitude states. This is also the way to establish the communication between CC3D and Raspberry Pi 3.

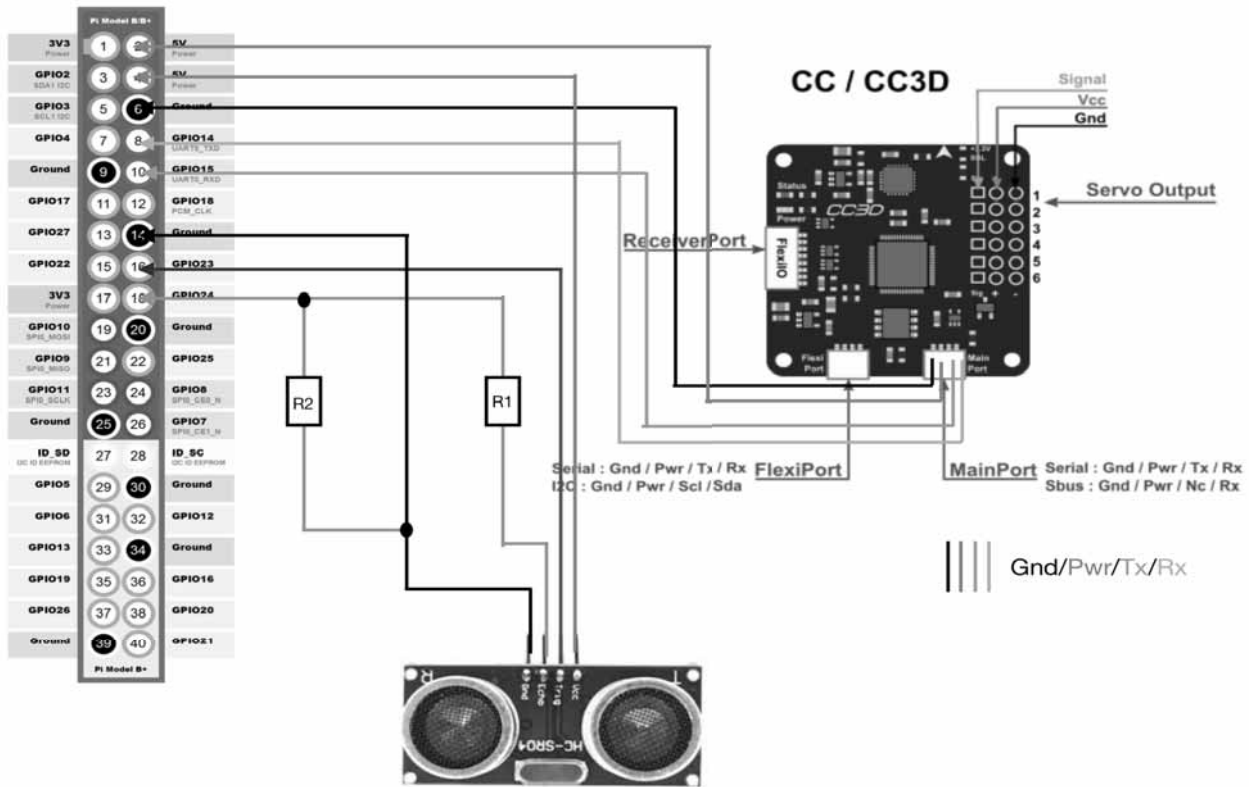


Figure 5: Hardware connection diagram.

Raspberry Pi Integration Raspberry Pi [14], series of small single-board computers, could be equipped with operating system. Raspberry Pi 3 Model B is the third generation of Raspberry Pi family. It has the quad core 64bit CPU that has the best performance. To physically connect Raspberry Pi and CC3D, we use the main port of CC3D which can be configured as a serial port and GPIO pin module of Raspberry Pi as it shown in Figure 5. Moreover, To achieve an autonomous system, we utilize an ultrasonic distance sensor to measure distance between the drone and the at ground. An ultrasonic distance sensor transmit from and receive an ultrasonic wave with a single ultrasonic transmitting and receive element to measure proximate distances such as vehicle floor heights or distances to obstacles or pedestrians approaching relative to a vehicle [10]. For the testbed, a low-cost sensor called HC-SR04 is selected. However, The ECHO pin of the sensor is rated at 5V while the GPIO input pins are rated as 3.3V. Therefore two resistors are added to protect the GPIO module. To access the telemetry data in Python, UAVTalk protocol is applied. For instance, below is a code excerpt to get attitude state of the drone.

```

self.objMan.AttitudeState.metadata.telemetryUpdateMode =
    UAVMetaDataObject.UpdateMode.PERIODIC
self.objMan.AttitudeState.metadata.telemetryUpdatePeriod.value = 50
self.objMan.AttitudeState.metadata.updated()
...
Yaw = self.objMan.AttitudeState.Yaw.value
Pitch = self.objMan.AttitudeState.Pitch.value
Roll = self.objMan.AttitudeState.Roll.value
    
```

Until here, all of the open source hardwares are integrated together for data communication between the Raspberry Pi 3 and the CC3D.

3 Demonstration

3.1 Workflow

To demonstrate the pipeline that promotes modeling and simulation-based development using open source software and hardware stacks, we developed an autonomous landing controller.

A generic Scilab/Xcos quadcopter model called Generic Quadcopter Simulation (GQS) that employs a proportional-derivative flight controller as a low-level architecture is used as a plant model. GQS model is based on [2]. This generic model can be tailored using

parameters to represent a specific platform. The high level architecture is designed in a model-based fashion using Scilab/Xcos. MiL testing is to optimize and verify the controller design for the autonomy feature.

Xcos Re-useable and Customizable Code Generator³ performs as a mean of generating code for the autonomous landing model and to evaluate how good the generated code functions are, a SiL simulation will be tested. For HiL testing, Raspberry Pi 3 executes the code that automatically generated from code generator, A second Raspberry Pi 3 is used as a real-time simulation computer target that enables an UART communication between the plant model and the controller. Finally, to verify the autonomy feature, all of the hardware stacks is assembled on the drone and flight testing is conducted.

Figure 6 explains the modeling and simulation based workflow applied:

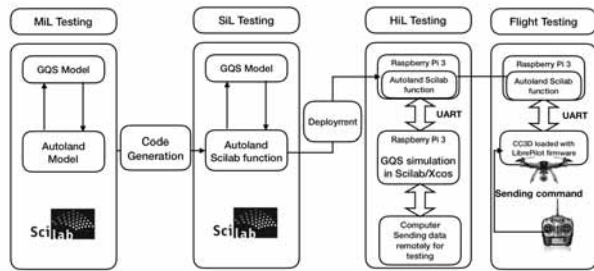


Figure 6: Demonstration workflow.

3.2 Model in-the-Loop Testing

In order to verify the autonomous landing controller, a MiL simulation (Figure 7) is conducted.

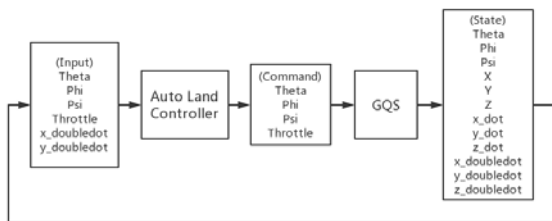


Figure 7: MiL simulation.

We prepared eight scenarios to test that in different situations this system will be working. Figure 8 shows

³<https://forge.scilab.org/index.php/p/xcos-code-generator/>

the results for one of the scenarios.

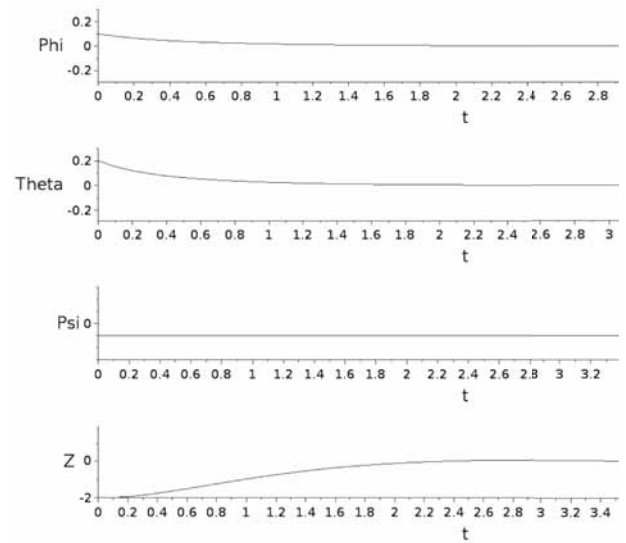


Figure 8: MiL simulation result.

The system can land for all the eight scenarios that means the designed autonomous landing controller is working properly.

3.3 Software in-the-Loop Testing

After MiL testing, the next step for evaluation is to generate source code out of autonomous landing controller model. As mentioned before, we use Xcos Re-useable and Customizable Code Generator for generating Scilab scripts. The auto generated code is then reintroduced in MiL schema. The same eight scenarios are executed and the results are compared with the MiL results.

3.4 Hardware in-the-Loop Testing

The HiL testing is essential a further step in testing the autonomous landing feature.

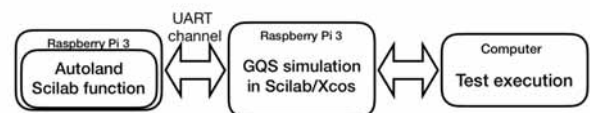


Figure 9: HiL simulation process.

Figure 9 depicts the test setup. The middle Raspberry

Pi 3 runs the GQS simulates the drone with controller while the target Raspberry Pi 3 on the left side will execute the Auto Land Scilab function for further verification. It is quite convenient for two Raspberry Pis to establish UART communication by GPIO using RX and TX pins. To achieve UART between Pi and Xcos model, a ATOM toolbox named serial Xcos IO module⁴ is used. This module provides a Xcos block to interface real hardware platform for a Xcos simulation via serial ports. It can also be applied into HiL simulation. Originally, the block supports Arduino and provides the bidirectional way to receive C structure input signal from an embedded system and then send back a C structure output signal to the embedded system. On the other hand, to execute autonomous landing Scilab function in Raspberry Pi, a python module called Scilab2Py⁵, a mean to seamlessly call Scilab functions and scripts from Python is applied. The result for HiL simulation for the same eight scenarios matched with the MiL results.

3.5 Flight Testing

Figure 10 describes the execution logic of auto generated Scilab script on the target platform for achieving the autonomous landing.

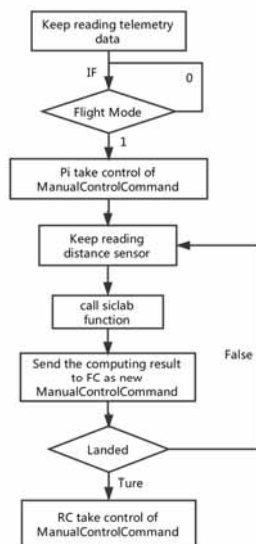


Figure 10: Target execution logic.

Raspberry Pi is listening to the flight mode from FC,

⁴<https://atoms.scilab.org/toolboxes/wgserialxcosio>

⁵<https://pypi.python.org/pypi/scilab2py>

once the flight mode changes to experiment mode, Pi will take control of the drone and receive the required system states which are sent to Scilab function for computing the movement of autonomous landing. When landed, RC will take back the control of FC.

Once the code is running on the testbed platform, the drone is ready to fly (Figure 11).

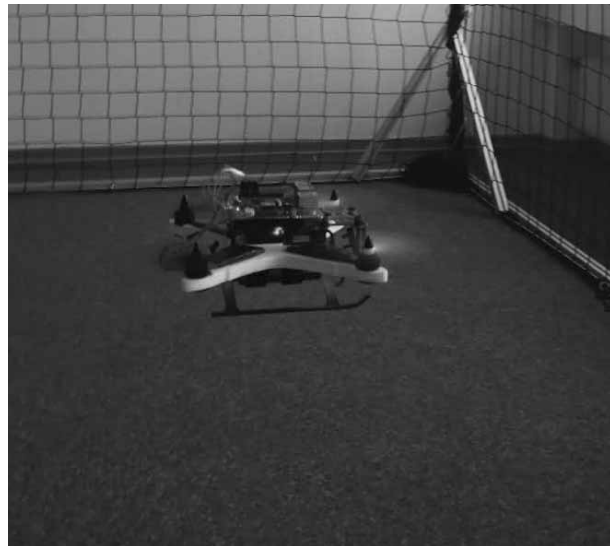


Figure 11: Flight testing.

4 Conclusion

Since the usage of drones is sharply growing while the modeling and simulation based development gets popular in many fields as well, it is quite meaningful to investigate a methodology to combine those two.

In this paper, we utilize the free open source software stacks including Scilab/Xcos which serves as a model design and simulation environment and LibrePilot. We also use open source hardware stacks like Raspberry Pi, HC-SR04 sensor and CC3D. They are sold with low price tags in the market and easy to acquire. The result shows that the pipeline is able to be used for simple autonomy feature design. However, since of all the resources are open source and low-end, the whole system does not perform perfectly. For example, the sensor can be easily broken so that the sensed data is not correct. For the future work, we will be using more reliable products with this pipeline to expect better performance.

After building the pipeline for the Simulation and Model-based development of autonomy features for

drones using only open source software and hardware, more students who have interests in aerospace domain can take this pipeline as an guidance for developing their own autonomy features.

References

- [1] Richard Barry. *Using the FreeRTOS real time kernel: a practical guide*. Real Time Engineers, 2010.
- [2] Samir Bouabdallah. *Design and control of quadrotors with application to autonomous flying*. PhD thesis, 2007.
- [3] Stephen La Vern Campbell, Jean-Philippe Chancelier, and Ramine Nikoukhah. *Modeling and simulation in Scilab/Scicos*. Springer, 2006.
- [4] Bruce T Clough. Metrics, schmetrics! how the heck do you determine a uav’s autonomy anyway. Technical report, Air Force Research Lab, 2002.
- [5] LibrePilot/OpenPilot community. Coptercontrol/cc3d/atom hardware setup, 2016.
- [6] LibrePilot/OpenPilot community. Librepilot documentation, 2017.
- [7] Tom Erkkinen. Model style guidelines for flight code generation. In *AIAA Modeling and Simulation Technologies Conference*, 2005.
- [8] Tom Erkkinen and Bill Potter. Model-based design for do-178b with qualified tools. In *AIAA Modeling and Simulation Technologies Conference and Exhibit*, 2009.
- [9] Gene F Franklin, J David Powell, Abbas Emami-Naeini, and J David Powell. *Feedback control of dynamic systems*, volume 3. Addison-Wesley Reading, MA, 1994.
- [10] Makoto Iwabuchi and Sinichi Ohzawa. Ultrasonic distance sensor, April 17 1990. US Patent 4,918,672.
- [11] Eric Jones, Travis Oliphant, and Pearu Peterson. {SciPy}: open source scientific tools for {Python}. 2014.
- [12] Edward Ashford Lee and Sanjit A Seshia. *Introduction to embedded systems: A cyber-physical systems approach*. MIT Press, 2016.
- [13] Torsten Merz. Building a system for autonomous aerial robotics research. In *Proc. of the IFAC Symp. on Intelligent Autonomous Vehicles*, 2004.
- [14] Raspberry Pi—Teach. learn, and make with raspberry pi. *Raspberry Pi*, 2016.
- [15] Antidio Viguria. Autonomy architectures. In *Encyclopedia of Aerospace Engineering. 1–14*, 2016.

Analyse, Simulation und Optimale Steuerung eines mathematischen Dengue-Fieber Modells mit Impfung

Gaby Albrecht¹, Anne Fischer^{2,0}, Kurt Chudej^{1,2}

¹Lehrstuhl für Wissenschaftliches Rechnen, Universität Bayreuth

²Lehrstuhl für Ingenieurmathematik, Universität Bayreuth

kurt.chudej@uni-bayreuth.de

Dengue Fieber wird durch den Stich einer Asiatischen Tigermücke übertragen. Eine Mensch zu Mensch Infektion ist nicht möglich. Die Krankheit tritt (bisher) hauptsächlich in einem Streifen rund um den Äquator auf und ist für jährlich ca. 390 Millionen Krankheitsfälle verantwortlich. Bisher sind alle in Deutschland bekannten Krankheitsfälle durch Auslands-(flug-)reisen eingeschleppt. Leider breitet sich die Asiatische Tigermücke seit einigen Jahren (wieder) in Europa und insbesondere auch in Süddeutschland aus. Selbsterhaltende Populationen der Asiatischen Tigermücke sind im Jahre 2017 aus Freiburg i.Br., Heidelberg und Sinnsheim (Baden-Württemberg) bekannt. Erwachsene Mücken wurden u.a. auch zwischen Rosenheim und Kufstein an der Inntalautobahn und in der Nähe des Containerbahnhofs in Regensburg gefunden. Die Tigermücken werden u.a. durch Altreifen-Importe, Lkw-Frachten auf Autobahn und Schiene und Wohnwägen nach Deutschland eingeschleppt. Wir berechnen charakteristische Größen eines mathematischen Dengue-Fieber Modells und präsentieren numerische Simulationsergebnisse.

1 Dengue-Fieber Modell mit Impfung

Seit einigen Jahren sind Funde erwachsener Tigermücken, bzw. noch schlimmer eigenständige Tigermückenpopulationen, in einigen Orten und Städten Süddeutschlands bekannt [4]. Diese unglückliche Szenerie wird durch die Klimaänderung begünstigt [3]. Zudem sind 708 Erkrankungen in Deutschland im Jahr 2015 beobachtet worden, die wohl alle durch Fernreisen eingeschleppt wurden [1]. Dies motiviert die vorliegende Forschung an mechanistischen Kompartimentmodellen für Denguefieber.

Diese Arbeit beschäftigt sich mit einem mathematischen Modell zur Bekämpfung von Denguefieber durch eine zukünftige Impfung. Da Dengue über einen Vektor, die Asiatische Tigermücke, übertragen wird, und eine Mensch zu Mensch Übertragung nicht möglich ist, müssen sowohl die Mücken als auch die Menschen ins Kompartimentmodell aufgenommen wer-

den. Wir betrachten ein nicht-perfektes Massenimpfungsmodell mit abnehmender Immunität, das am Ende der PhD Arbeit [7] von Rodrigues eingeführt wird, für das aber keine theoretische Analyse mehr vorgenommen wird. Das hier besprochene Modell verallgemeinert auch die Massenimpfungsmodelle, die in [8] theoretisch analysiert werden.

Die Mückenpopulation wird unterteilt in die aquatischen Mücken A_m , die erwachsenen infizierbaren Mücken S_m und die erwachsenen infizierenden Mücken I_m . Für die Menschen wird ein SVIR-Modell benutzt: Infizierbare Menschen S_h , geimpfte Menschen V_h , infizierende Menschen I_h und geheilte und immune Menschen R_h . Das mathematische Modell (imperfect random mass vaccination with waning immunity) enthält verschiedene wichtige Charakteristika eines (zukünftigen) Impfstoffs für eine Massenimpfung (von Erwachsenen): Wichtigste Größe ist die Impfrate ψ ; zunächst eine Konstante, später eine optimal zu steuernde Zeitfunktion. Wir nehmen an, das ein Anteil σ der geimpften Menschen bei Kontakt mit einer infizierenden Mücke mit der gleichen Wahrscheinlichkeit erkranken kann wie die infizierbaren Men-

⁰Neue Anschrift: Josef Witt Gruppe, Weiden

sehen. Zudem berücksichtigen wir eine Abnahmrage θ der durch die Impfung erlangten Immunität.

Weitere Parameter im Modell sind: Gesamtbevölkerung N_h , durchschnittliche Stechrate B , Infektionsrate von Mücken auf Menschen β_{mh} , Infektionsrate von Menschen auf Mücken β_{hm} , durchschnittliche Lebenserwartung von Menschen $1/\mu_h$, durchschnittliche Lebenserwartung von Mücken $1/\mu_m$, durchschnittliche Erkrankungsdauer von Menschen $1/\eta_h$, Rate mit der Larven zu ausgewachsenen Mücken reifen η_A , Anzahl an Eiern pro Brutplatz φ , natürliche Todesrate der Mückenlarven μ_A , Anzahl weiblicher Mücken pro Mensch m , Anzahl Mückenlarven pro Mensch k .

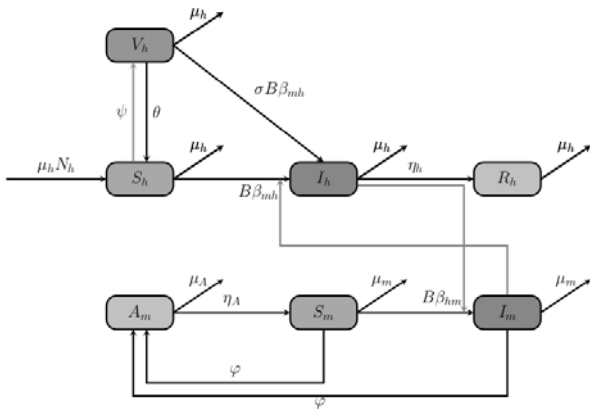


Abb. 1: Kompartimentmodell mit Impfung für die menschliche Population sowie die Mückenpopulation.

Die zum SVIR-ASI-Kompartimentmodell (siehe Abb. 1) gehörigen Differentialgleichungen für die menschliche Population lauten:

$$\begin{aligned}\dot{S}_h &= \mu_h N_h + \theta V_h - (B\beta_{mh} \frac{I_m}{N_h} + \psi + \mu_h) S_h, \\ \dot{V}_h &= \psi S_h - (\theta + \sigma B\beta_{mh} \frac{I_m}{N_h} + \mu_h) V_h, \\ \dot{I}_h &= B\beta_{mh} \frac{I_m}{N_h} (S_h + \sigma V_h) - (\eta_h + \mu_h) I_h, \\ \dot{R}_h &= \eta_h I_h - \mu_h R_h.\end{aligned}\quad (1)$$

Für die Mückenpopulation erhält man:

$$\begin{aligned}\dot{A}_m &= \varphi (1 - \frac{A_m}{kN_h}) (S_m + I_m) - (\eta_A + \mu_A) A_m, \\ \dot{S}_m &= \eta_A A_m - (B\beta_{hm} \frac{I_h}{N_h} + \mu_m) S_m, \\ \dot{I}_m &= B\beta_{hm} \frac{I_h}{N_h} S_m - \mu_m I_m.\end{aligned}\quad (2)$$

2 Theoretische Analyse

Das Differentialgleichungssystem (1), (2) besitzt als Invariante $N_h \equiv S_h(t) + V_h(t) + I_h(t) + R_h(t)$.

Die Analyse des math. Modells wird in der abgeschlossenen Menge

$$\Omega = \{(S_h, V_h, I_h, R_h | A_m, S_m, I_m) \in \mathbb{R}_+^7 \mid A_m \leq kN_h, \\ S_m + I_m \leq mN_h, S_h + V_h + I_h + R_h \leq N_h\}$$

durchgeführt. Alle Koordinaten haben nichtnegative Werte und die Summe der Anzahlen in den Kompartimenten, die die menschliche Population betreffen, ist kleinergleich als die angenommene Gesamtpopulation. Auch die Anzahl der ausgewachsenen und nicht ausgewachsenen Mücken ist, bezogen auf die menschliche Gesamtpopulation, beschränkt.

Hilfssatz 1 [5]: Ω ist eine positiv invariante Menge für das Differentialgleichungssystem (1), (2).

Beweis: Das DGL-System kann umgeschrieben werden zu $\dot{X} = A(X)X + F$ mit $X = (S_h, V_h, I_h, R_h | A_m, S_m, I_m)^\top$, $F = (N_h, 0, 0, 0 | 0, 0, 0)^\top$ und der Blockdiagonalmatrix $A(X) = \begin{pmatrix} A_h & O \\ O & A_m \end{pmatrix}$ mit den Teilmatrizen

$$A_h = \begin{pmatrix} -B\beta_{mh} \frac{I_m}{N_h} - \psi - \mu_h & \theta & 0 & 0 \\ \psi & -\sigma B\beta_{mh} \frac{I_m}{N_h} - \theta - \mu_h & 0 & 0 \\ B\beta_{mh} \frac{I_m}{N_h} & \sigma B\beta_{mh} \frac{I_m}{N_h} & -\eta_h - \mu_h & 0 \\ 0 & 0 & \eta_h & -\mu_h \end{pmatrix},$$

$$A_m = \begin{pmatrix} -\varphi \frac{S_m + I_m}{kN_h} - \eta_A - \mu_A & \varphi & \varphi \\ \eta_A & -B\beta_{hm} \frac{I_h}{N_h} - \mu_m & 0 \\ 0 & B\beta_{hm} \frac{I_h}{N_h} & -\mu_m \end{pmatrix}.$$

Die Matrix $A(X)$ ist offensichtlich eine Metzler-Matrix. Außerdem ist $F \geq 0$. Damit ist das DGL-System in Ω positiv invariant nach [6, 2]. \square

Daraus folgt, dass für jeden Anfangswert aus Ω eine eindeutige Lösung für alle nichtnegativen Zeiten in Ω existiert.

Als nächstes werden die Gleichgewichtspunkte des Differentialgleichungssystems berechnet. Dazu sind die polynomialen Gleichungssysteme

$$0 = \dot{A}_m = \dot{S}_m = \dot{I}_m \quad (3)$$

$$0 = \dot{S}_h = \dot{V}_h = \dot{I}_h = \dot{R}_h \quad (4)$$

zu lösen. Die Abkürzungen

$$\mathcal{M} = \varphi \eta_A - \mu_m (\mu_A + \eta_A), \quad (5)$$

$$\xi = \varphi \mu_m^2 (\eta_h + \mu_h), \quad \chi = kB^2 \beta_{hm} \beta_{mh} \mathcal{M}. \quad (6)$$

erweisen sich als nützlich.

Zu Vergleichszwecken wird zunächst ein vereinfachtes Modell ohne Impfung betrachtet, d.h. wir setzen die Impfrate $\psi \equiv 0$ und benutzen den Anfangswert $V_h(0) = 0$. Daraus ergibt sich dann sofort $V_h(t) \equiv 0$. Damit vereinfacht sich das Modell (1), (2) zu dem in [7] analysierten SIR-ASI Modell.

Satz 2 [7, 9, 5]: Man erhält in Ω für das Differentialgleichungssystem (1), (2) mit $\psi \equiv 0$ (keine Impfung) und $V_h \equiv 0$ die Gleichgewichtspunkte:

- $\bar{E}_1 = (N_h, 0, 0, 0 | 0, 0, 0)$
- $\bar{E}_2 = \left(N_h, 0, 0, 0 \mid \frac{kN_h \mathcal{M}}{\varphi \eta_A}, \frac{kN_h \mathcal{M}}{\varphi \mu_m}, 0 \right)$
- $\bar{E}_3 = (S_h^{**}, 0, I_h^{**}, R_h^{**} \mid A_m^{**}, S_m^{**}, I_m^{**})$ mit

$$S_h^{**} = \frac{\varphi N_h (\mu_h B \beta_{hm} + \mu_m (\eta_h + \mu_h)) \mu_m}{B \beta_{hm} (kB \beta_{mh} \mathcal{M} + \varphi \mu_h \mu_m)}$$

$$I_h^{**} = \frac{\mu_h N_h (\chi - \xi)}{(\eta_h + \mu_h) B \beta_{hm} (kB \beta_{mh} \mathcal{M} + \varphi \mu_h \mu_m)}$$

$$R_h^{**} = N_h - (S_h^{**} + I_h^{**})$$

$$A_m^{**} = \frac{kN_h \mathcal{M}}{\varphi \eta_A}$$

$$S_m^{**} = \frac{N_h (\eta_h + \mu_h) (kB \beta_{mh} \mathcal{M} + \varphi \mu_h \mu_m)}{B \beta_{mh} \varphi (B \beta_{hm} \mu_h + \mu_m (\eta_h + \mu_h))}$$

$$I_m^{**} = \frac{\mu_h N_h (\chi - \xi)}{B \beta_{mh} (\varphi B \beta_{hm} \mu_h \mu_m + \xi)}$$

\bar{E}_1 ist der triviale Gleichgewichtspunkt. \bar{E}_2 ist der krankheitsfreie Gleichgewichtspunkt (DFE = disease free equilibrium). \bar{E}_3 ist der endemische Gleichgewichtspunkt.

Abhängig vom Vorzeichen von \mathcal{M} existieren in Ω bis zu drei Gleichgewichtspunkte:

- falls $\mathcal{M} \leq 0$ existiert nur der triviale Gleichgewichtspunkt \bar{E}_1 .
- falls $\mathcal{M} > 0$ und $\xi \geq \chi$ existieren die beiden Gleichgewichtspunkte \bar{E}_1 und \bar{E}_2 .
- falls $\mathcal{M} > 0$ und $\xi < \chi$ existieren drei Gleichgewichtspunkte \bar{E}_1 , \bar{E}_2 und \bar{E}_3 .

Beweis: Ein ausführlicher Beweis findet sich in [9], siehe auch [7]. \square

Jetzt berechnen wir die Gleichgewichtspunkte des Differentialgleichungsmodells (1), (2) mit Impfung. Dies erweist sich als wesentlich anspruchsvoller.

Eine brute-force Maple Berechnung [5] liefert die folgenden drei Gleichgewichtspunkte in Ω

- $E_1 = \left(\frac{N_h(\mu_h + \theta)}{\mu_h + \theta + \psi}, \frac{N_h \psi}{\mu_h + \theta + \psi}, 0, 0 \mid 0, 0, 0 \right)$,
- $E_2 = \left(\frac{N_h(\mu_h + \theta)}{\mu_h + \theta + \psi}, \frac{N_h \psi}{\mu_h + \theta + \psi}, 0, 0 \mid \frac{kN_h \mathcal{M}}{\varphi \eta_A}, \frac{kN_h \mathcal{M}}{\varphi \mu_m}, 0 \right)$,
- $E_3 = (S_h^*, V_h^*, I_h^*, R_h^* \mid A_m^*, S_m^*, I_m^*)$,

wobei das Maple Ergebnis für E_3 über mehrere Seiten geht. Die Komponente A_m^* stimmt mit dem Ergebnis A_m^{**} von Satz 2 überein. Auch $R_h^* = N_h - (S_h^* + V_h^* + I_h^*)$ ist vorab bekannt.

Ein Vergleich mit dem vorhergehenden Satz 2, der die Situation ohne Impfung beschreibt, zeigt, dass sich beim trivialen Gleichgewichtspunkt \bar{E}_1, E_1 und beim krankheitsfreien Gleichgewichtspunkt \bar{E}_2, E_2 jetzt eine neue Aufteilung bzgl. (S_h, V_h) von in Anteilen $(1, 0)$ auf $\left(\frac{\mu_h + \theta}{\mu_h + \theta + \psi}, \frac{\psi}{\mu_h + \theta + \psi} \right)$ ergibt. Die übrigen Kompartimente ändern sich beim trivialen und krankheitsfreien Gleichgewichtspunkt nicht.

Wir wünschen uns jetzt biologisch interpretierbare Formeln der Komponenten von E_3 . Deshalb zerlegen wir das polynomiale Gleichungssystem (3), (4) für die Gleichgewichtspunkte in zwei Teilsysteme. Wir führen die Abkürzungen

$$B_m := B \beta_{hm} I_h^* / N_h, \quad B_h := B \beta_{mh} I_m^* / N_h \quad (7)$$

ein. Dann lösen wir zunächst $\dot{A}_m = \dot{S}_m = \dot{I}_m = 0$ und erhalten

$$A_m^* = 0 \text{ oder } A_m^* = \frac{kN_h \mathcal{M}}{\varphi \eta_A} =: \tilde{A}_m \quad (8)$$

und weiter in Abhängigkeit von A_m^*

$$S_m^* = A_m^* \frac{\eta_A}{\mu_m} \frac{\mu_m}{\mu_m + B_m}, \quad (9)$$

$$I_m^* = A_m^* \frac{\eta_A}{\mu_m} \frac{B_m}{\mu_m + B_m}. \quad (10)$$

Die Anzahl A_m^* der aquatischen Mücken im Gleichgewichtspunkt ist unabhängig von den Parametern der menschlichen Population mit Ausnahme von N_h . Der

uninteressante Fall $A_m^* = 0$ führt eindeutig zum trivialen Gleichgewichtspunkt E_1 . Sei deshalb im folgenden $A_m^* = \tilde{A}_m$. Durch hinschauen erkennt man $S_m^* + I_m^* = \tilde{A}_m \frac{\eta_A}{\mu_m} =: \tilde{M}_m$, so dass die Anzahl \tilde{M}_m der erwachsenen Mücken im Gleichgewichtspunkt unabhängig von den Parametern der menschlichen Population mit Ausnahme von N_h ist. Die prozentuale Aufteilung der erwachsenen Mücken auf (gesunde) infizierbare und (erkrankte) infizierende Mücken hängt dagegen von der Anzahl der (erkrankten) infizierenden Menschen I_h^* ab.

Jetzt lösen wir $\dot{S}_h = \dot{V}_h = \dot{I}_h = 0$ und erhalten:

$$S_h^* = \frac{\mu_h N_h (\sigma B_h + \theta + \mu_h)}{(B_h + \psi + \mu_h)(\sigma B_h + \theta + \mu_h) - \theta \psi} \quad (11)$$

$$V_h^* = \frac{\mu_h N_h \psi}{(B_h + \psi + \mu_h)(\sigma B_h + \theta + \mu_h) - \theta \psi} \quad (12)$$

Damit ergibt sich:

$$I_h^* = \frac{\mu_h N_h [(\sigma B_h + \theta + \mu_h) + \sigma \psi]}{(B_h + \psi + \mu_h)(\sigma B_h + \theta + \mu_h) - \theta \psi} \frac{B_h}{\eta_h + \mu_h} \quad (13)$$

Die Kopplung zwischen dem ASI-Modell und dem SVIR-Modell ist über die nicht-linearen Gleichungen (10), (13) und (7) gegeben. Wenn man (7) jeweils in (10), (13) einsetzt, dann ergibt sich nach Multiplikation mit den Nennern ein Gleichungssystem mit einem quadratischen und einem kubischen Polynom in I_m^*, I_h^* . Im generischen Fall erwartet man mit dem Satz von Bezout höchstens sechs Lösungen. Für die Referenzwerte der Parameter (siehe Abschnitt 3) erhält man numerisch sechs verschiedene Lösungen, von denen drei in Ω liegen.

Die obigen Ergebnisse lassen sich zusammen mit verfeinerten Maple Rechnungen wie folgt zusammenfassen:

Satz 3: Das Differentialgleichungssystem (1), (2) besitzt höchstens drei Gleichgewichtspunkte, die in Ω liegen. Davon sind höchstens zwei Punkte krankheitsfreie Equilibria und höchstens ein Punkt ist ein endemisches Equilibrium.

- $E_1 = \left(\frac{N_h(\mu_h + \theta)}{\mu_h + \theta + \psi}, \frac{N_h \psi}{\mu_h + \theta + \psi}, 0, 0 \mid 0, 0, 0 \right)$
- $E_2 = \left(\frac{N_h(\mu_h + \theta)}{\mu_h + \theta + \psi}, \frac{N_h \psi}{\mu_h + \theta + \psi}, 0, 0 \mid \tilde{A}_m, \tilde{M}_m, 0 \right)$
- $E_3 = (S_h^*, V_h^*, I_h^*, R_h^* \mid \tilde{A}_m, S_m^*, I_m^*)$

Für $\mathcal{M} \leq 0$ existiert nur der triviale Gleichgewichtspunkt E_1 . Die Komponenten von E_2 ergeben sich als Lösung von (10), (13) und (7).

Satz 4 [5]: Die Basisreproduktionszahl \mathcal{R}_0 des Differentialgleichungssystems (1), (2) erfüllt:

$$\mathcal{R}_0^2 = \underbrace{B^2 \beta_{hm} \beta_{mh}}_{\text{Stiche}} \underbrace{\frac{1}{\mu_m} \frac{\tilde{M}_m}{N_h}}_{\text{Mücken}} \underbrace{\frac{1}{\eta_h + \mu_h} \frac{\mu_h + \theta + \sigma \psi}{\mu_h + \theta + \psi}}_{\text{Menschen}}$$

Die ersten Terme beschreiben die Stichintensität, dann folgen Terme abhängig von den Mückenparametern, insbesondere \tilde{M}_m/N_h . Dann folgen Terme mit den menschlichen Parametern incl. den Impfparametern.

Beweis: Dazu werden nach [10] die beiden Matrizen

$$F = \begin{pmatrix} 0 & B \beta_{mh} \frac{S_h + \sigma V_h}{N_h} \\ B \beta_{hm} \frac{S_m}{N_h} & 0 \end{pmatrix}, V = \begin{pmatrix} \eta_h + \mu_h & 0 \\ 0 & \mu_m \end{pmatrix}$$

betrachtet. Die Basisreproduktionszahl ergibt sich dann als $\mathcal{R}_0 = \rho(FV^{-1})$, wobei der krankheitsfreie Gleichgewichtspunkt (DFE) E_2 einzusetzen ist. \square

Satz 5 [10, 5]: Der krankheitsfreie Gleichgewichtspunkt (DFE) E_2 ist lokal asymptotisch stabil, wenn $\mathcal{R}_0 < 1$ ist, aber instabil wenn $\mathcal{R}_0 > 1$ gilt.

Was ist zielführend für das schwache Ziel „Verkleinerung der Basisreproduktionszahl \mathcal{R}_0 “ ?

Eine höhere Effizienz des Impfstoffs ($\sigma \rightarrow 0$) verkleinert die Basisreproduktionszahl \mathcal{R}_0 . Ein umschreiben von $\frac{\mu_h + \theta + \sigma \psi}{\mu_h + \theta + \psi} = 1 - \frac{[1 - \sigma] \psi}{\mu_h + \theta + \psi} \in (0, 1]$ zeigt, das eine kleinere Immunitätsabnahmerate θ ebenfalls die Basisreproduktionszahl \mathcal{R}_0 verkleinert.

Eine Verkleinerung von $B^2 \beta_{hm} \beta_{mh}$ verringert die Basisreproduktionszahl \mathcal{R}_0 ; erreichbar etwa durch die Verwendung von Moskitonetzen. Auch eine Verringerung der (menschgemachten) Brutplätze (k kleiner) verringert die Basisreproduktionszahl \mathcal{R}_0 . Eine (künstliche) Verringerung der durchschnittlichen Lebenserwartung der erwachsenen Mücken $1/\mu_m$ (durch Insektizide), siehe [7], [5], verringert ebenfalls die Basisreproduktionszahl \mathcal{R}_0 .

Was ist (evtl. alleine nicht) zielführend für das starke Ziel „ $\mathcal{R}_0 < 1$ “ ?

Eine alleinige Erhöhung der Effizienz des Impfstoffs ist evtl. nicht hinreichend, da nur $\frac{\mu_h + \theta + \sigma \psi}{\mu_h + \theta + \psi} \xrightarrow{\sigma \rightarrow 0} \frac{\mu_h + \theta}{\mu_h + \theta + \psi} = A > 0$ gilt.

Eine alleinige Verkleinerung der Immunitätsabnahmerate θ ist evtl. auch nicht hinreichend, da nur $\frac{\mu_h + \theta + \sigma \psi}{\mu_h + \theta + \psi} \xrightarrow{\theta \rightarrow 0} \frac{\mu_h + \sigma \psi}{\mu_h + \psi} = B > 0$ gilt.

3 Numerische Simulation

Wir benutzen die folgenden Referenzwerte für die Parameter [7, 5]: $B = 0.8$, $\beta_{mh} = \beta_{hm} = 0.375$, $\mu_h = \frac{1}{80 \cdot 365}$, $\eta_h = \frac{1}{3}$, $\mu_m = 0.1$, $\varphi = 6$, $\mu_A = 0.25$, $\eta_A = 0.08$, $m = 3$, $k = 3$. Wir wählen $N_h = 383\,000$ (Bevölkerung Landkreis Freiburg und Emmendingen). Wir wählen für die Effizienz des Impfstoffs 80%, d.h. $\sigma = 0.2$ und für die Abnahmerate der Immunität $\theta = 0.05$.

Bemerkung 6: Damit ergeben sich sechs Gleichgewichtspunkte:

- $E_1 \doteq (273625, 109375, 0, 0 | 0, 0, 0)$
trivialer Gleichgewichtspunkt $\in \Omega$,
- $E_2 \doteq (273\,625, 109\,375, 0, 0 | 1\,070\,006, 856\,005, 0)$
krankheitsfreier Gleichgewichtspunkt (DFE) $\in \Omega$,
- $E_3 \doteq (58\,793, 23\,486, 31, 300\,690 | 1\,070\,006, 855\,798, 207)$
endemischer Gleichgewichtspunkt $\in \Omega$,
- $E_4 \doteq (-89\,269\,707, 446\,574\,759, -36\,666, -356\,885\,373 | 1\,070\,006, 1\,200\,911, -344\,906) \notin \Omega$,
- $E_5 \doteq (888\,080\,666, 355\,052\,102, -127\,667, -1\,242\,622\,222 | 0, 57, -57) \notin \Omega$,
- $E_6 \doteq (310\,586\,313, 1\,553\,719\,245, -127\,667, -1\,242\,622\,222 | 0, 344\,906, -344\,906) \notin \Omega$.

Die Basisreproduktionszahl ergibt sich zu $\mathcal{R}_0 \doteq 2.46 > 1$. Damit ist der krankheitsfreie Gleichgewichtspunkt (DFE) E_2 instabil, vgl. Satz 5.

Hilfssatz 7: Der triviale Gleichgewichtspunkt E_1 ist für die Referenzparameter instabil. Der endemische Gleichgewichtspunkt E_3 ist für die Referenzparameter lokal asymptotisch stabil.

Beweis: Berechne numerisch die Eigenwerte der Jacobimatrix von (1), (2) ausgewertet am Gleichgewichtspunkt E_1 bzw. E_3 . □

Als Anfangswert für die infizierende Population wurde ein Prozentsatz von 0.01% der Gesamtpopulation gewählt, was einer Anzahl von $I_{h0} = 38$ entspricht. Zudem setzen wir $\psi \equiv 0$ (keine Impfung) und $V_h \equiv 0$ (keine Geimpften). Es ergeben sich die Anfangswerte:

$$\begin{aligned} I_{h0} &= 38, R_{h0} = V_{h0} = 0, S_{h0} = N_h - I_{h0}, \\ A_{m0} &= kN_h, S_{m0} = mN_h, I_{m0} = 0. \end{aligned} \quad (14)$$

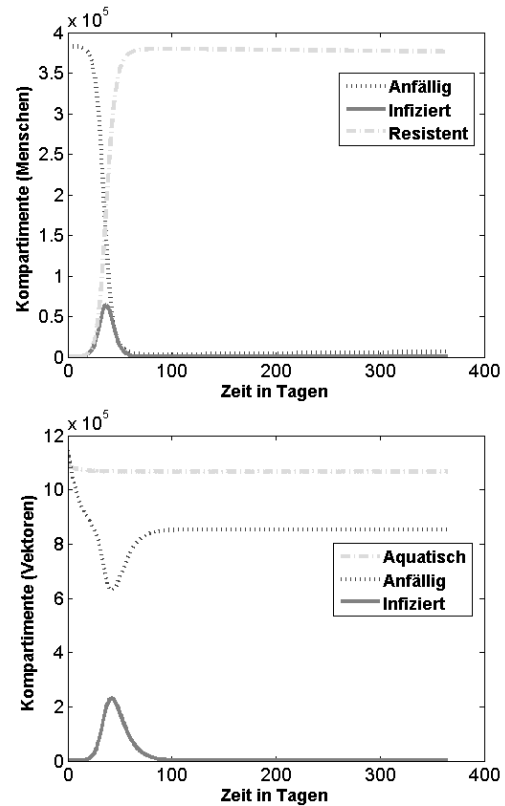


Abb. 2: Numerische Simulation eines Krankheitsausbruches in einer komplett anfälligen Population [5].

Auf dem Zeithorizont von 365 Tagen ergeben sich für die Lösung des Anfangswertproblems die Ergebnisse in Abb. 2. In Abb. 2 erkennt man eine sehr schnelle Konvergenz von $A_m \rightarrow \bar{A}_m \doteq 1\,070\,006$ und $S_m + I_m \rightarrow \bar{M}_m \doteq 856\,005$. Ausserdem sieht man, dass das Einschleppen einiger mit Dengue infizierter Menschen beim Vorhandensein einer Tiger-Mückenpopulation bereits ausreicht, um das Virus in beiden Populationen zu verbreiten. 30 bis 50 Tage nachdem die Infizierenden Menschen das erste Mal mit der anfälligen Population in Kontakt gekommen sind, kommt es zum Krankheitsausbruch. Hierbei erfolgt der Ausbruch in beiden Populationen nur leicht zeitversetzt. Die Spitze der Kurve erreicht für die menschliche Population einen Wert von 64 000, es erkranken also sehr viele Individuen gleichzeitig. Nach einem steilen Anstieg fällt die Kurve ähnlich steil wieder ab. Ein Ausbruch erfolgt also relativ rasch und zeitnah innerhalb der ersten zwei Monate nach Beobachtung der ersten Infektionen. Die Anzahl resistenter Individuen

steigt in dem Maße an, wie infizierte Individuen genesen, bis die gesamte Population immun ist. Am Ende des Betrachtungszeitraums beginnt die Anzahl resistenter Menschen langsam wieder abzunehmen und die Anzahl Anfälliger wieder zuzunehmen. Nach einem kurzen anfänglichen Abfall bleibt, bedingt durch das angenommene logistische Wachstum, die Anzahl der nicht ausgewachsenen Moskitos A_m konstant. Da Moskitos keine Resistenzen aufbauen können, zeigt der Verlauf der Kurve anfälliger Vektoren zum Zeitpunkt des Krankheitsausbruchs den gespiegelten Verlauf zu dem der Infizierenden.

4 Optimale Steuerung der Impfung

Wir betrachten jetzt ein optimales Steuerungsproblem: Gesucht ist eine Funktion $\psi(t) \in [0, 1]$, so dass das Zielfunktional $J = \int_0^{t_f} \left[\gamma_1 \cdot \left(\frac{I_h(t)}{N_h} \right)^2 + \gamma_2 \cdot \psi(t)^2 \right] dt$ minimiert wird unter der Nebenbedingung, dass das Differentialgleichungssystem (1), (2) und die Anfangsbedingungen erfüllt sind.

Für drei verschiedene Situationen wurde nun eine optimale Impfstrategie errechnet. In Fall A wird ein allgemeines Szenario betrachtet, wobei die Gewichte auf ungefähr denselben Wert gesetzt wurden. Es kann wieder als Vergleichsszenario zu den anderen beiden drastischeren Situationen gesehen werden. Eine kostspielige Behandlung von Erkrankten und damit ein sehr hoher Wert von γ_1 wird in Szenario B berücksichtigt. In Szenario C wird der Fall untersucht, dass der Preis des Impfstoffes im Vergleich zu den Behandlungskosten für Infizierende sehr hoch ist (γ_2 groß).

Wie aus Abb. 3 ersichtlich ist, ändert sich die Anzahl an infizierten Individuen abhängig vom Gewicht γ_1 . In Fall B hat der Anteil an Infizierten den größten wirtschaftlichen Einfluss und deshalb ist in diesem Fall, verglichen mit den beiden anderen Szenarien, die Anzahl an Erkrankten am niedrigsten. Vergleicht man die absoluten Anzahlen während des Ausbruchs mit denen aus der numerischen Simulation für das Grundmodell ohne Kontrollen oder Impfung, so stellt sich heraus, dass der Einsatz eines Impfstoffes merkliche Auswirkungen hat. Im Gegensatz zu den 64 000 Erkrankten (Abb. 2), die ohne Impfung auftreten, erreicht die Kurve in Fall C trotz teurem Impf-

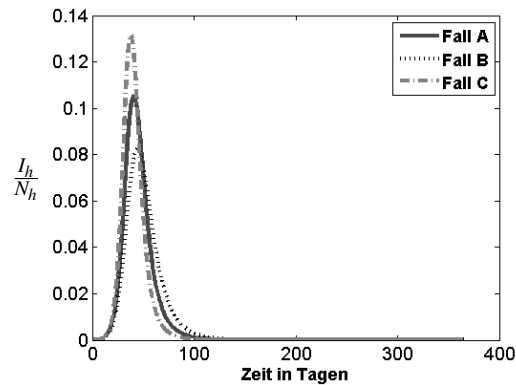


Abb. 3: Anteil infizierter Individuen in den drei Fällen [5].

stoff einen Wert von 13.15%, was ca. 50 000 Infizierten entspricht. Wird großer Wert darauf gelegt, dass möglichst wenige Menschen erkranken, also betrachtet man Fall B, so werden 8.26%, also ca. 32 000 Individuen infiziert. Die Zahlen können also um die Hälfte gesenkt werden, wenn der Fokus auf einer Verringerung der Infizierten liegt.

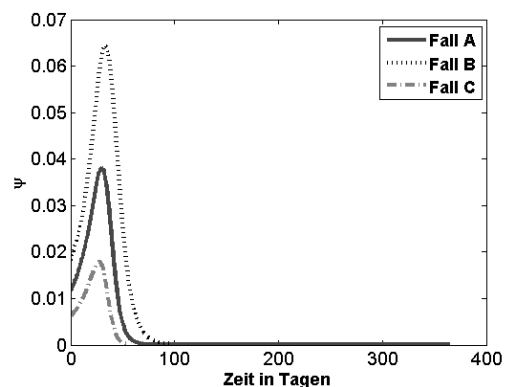


Abb. 4: Massenimpfung $\psi(t)$ in den drei Fällen [5].

In Abb. 4 sind die optimalen Impfstrategien angegeben und man sieht sofort, dass der Prozentsatz des verwendeten Impfstoffes für Erwachsene am höchsten in Fall B (6.44%) ist, wenn man möglichst wenig Erkrankte erreichen möchte, und am niedrigsten in Fall C (1.79%), in dem das Serum der kostenintensivste Einfluss ist. Nimmt man alle Gewichte als gleichgroß an, so liegen erwartungsgemäß die Kurven der Massenimpfung zwischen denen der beiden extremen Fälle. Der Verlauf der Kurven ist steil, was einen rasch

verlaufenden Ausbruch und eine schnelle Durchführung der Impfung impliziert, und nachdem die ersten Maßnahmen ergriffen wurden, ist keine weitere Handlung nötig. In Fall B ist eine etwas längere Durchführung der Massenimpfung nötig.

5 Fazit und Ausblick

Bereits leichte Verallgemeinerungen bekannter Kompartimentmodelle für gefährliche Krankheiten mit Impfungen führen bei der mathematischen theoretischen Analyse zu erheblichem Mehraufwand. In diesem Paper lag der Fokus auf der theoretischen Analyse des Kompartimentmodells – so weit wie möglich ohne Bezug zu speziellen Parameterwerten. Weitere Arbeiten werden sich u.a. mit der Anpassung der Parameter an die Situation in Süddeutschland beschäftigen.

Literatur

- [1] Zur Situation bei wichtigen Infektionskrankheiten – Reiseassoziierte Krankheiten 2015. *Epidemiologisches Bulletin – Robert Koch-Institut*, (39):431–446, 2016.
- [2] A. Abate, A. Tiwari, and S. Sastry. Box invariance in biologically-inspired dynamical systems. *Automatica*, 45(7):1601–1610, 2009.
- [3] N. Becker, A. Krüger, C. Kuhn, A. Plenge-Bönig, S.M. Thomas, J. Schmidt-Chanasit, and E. Tannich. Stechmücken als Überträger exotischer Krankheitserreger in Deutschland. *Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz*, 57(5):531, 2014.
- [4] N. Becker, S. Schön, A. Klein, I. Ferstl, A. Kizgin, E. Tannich, C. Kuhn, B. Pluskota, and A. Jöst. First mass development of *Aedes albopictus* (Diptera: Culicidae)—its surveillance and control in Germany. *Parasitology Research*, 116(3):847–858, 2017.
- [5] A. Fischer. Optimale Impfstrategien für Dengue-Fieber. Masterarbeit, Universität Bayreuth, 2016.
- [6] W. Mitkowski. Dynamical properties of Metzler systems. *Bulletin of the Polish Academy of Sciences, Technical Sciences*, 56(4):309–312, 2008.
- [7] H.S. Rodrigues. *Optimal Control and Numerical Optimization Applied to Epidemiological Models*. PhD, Universidade de Aveiro, Portugal, 2012.
- [8] H.S. Rodrigues, M.T.T. Monteiro, and D.F.M. Torres. Vaccination models and optimal control strategies to dengue. *Mathematical Biosciences*, 247:1–12, 2014.
- [9] C. Schreppel. Bekämpfung des Denguefiebers als Optimalsteuerungsproblem. Masterarbeit, Universität Bayreuth, 2015.
- [10] P. van den Driessche and J. Watmough. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences*, 180(1–2):29–48, 2002.

Ein Ansatz zur objektorientierten Modellierung und Simulation technischer Systeme mit dem Physolator

Dirk Eisenbiegler
Universität Furtwangen
dirk.eisenbiegler@hs-furtwangen.de

In dem Beitrag soll dargestellt werden, wie mit Hilfe des Simulationsframeworks Physolators technische Systeme in einer modularen, objektorientierten Weise modelliert und anschließend simuliert werden können.

Einführung

Technische Systeme sind modular aufgebaut. Sie setzen sich in hierarchischer Weise aus Komponenten und Teilkomponenten zusammen. In diesem Beitrag soll ein objektorientierter Ansatz zur Modellierung technischer Systeme auf der Basis des Simulationsframeworks Physolator beschrieben werden. Bei diesem Ansatz werden zunächst technische Komponenten und deren physikalisches Verhalten beschrieben. Anschließend werden diese technischen Komponenten zu technischen Systemen zusammengefügt. Dabei kommen objektorientierte Konzepte zum Einsatz: Klasse, Instanz, Vererbung. Zwischen den technischen Komponenten gibt es Wechselwirkungen. Die technischen Komponenten beeinflussen ihr physikalisches Verhalten gegenseitig. Das technische System ist mehrstufige Hierarchie aus Komponenten und Teilkomponenten. In dieser Hierarchie sollen Querverweise verwendet werden, um diese Beziehungen zwischen den technischen Komponenten zu beschreiben.

Physolator

Die Arbeit basiert auf dem Simulationsframework Physolator. Informationen zum Physolator findet man auf der Webseite *physolator.com*. Die Programmierung erfolgt durchgängig mit Java. Der Physolator selbst besteht aus Java-Programmcode und auch alle physikalische Modelle werden mit Java beschrieben. Der Physolator unterstützt eine modulare, objektorientierte Modellierung mit modernen Konzepten. Der Physolator verwendet eingebaute ODE-Solver, um die in den Physolator geladenen physikalischen Systeme zu simulieren.

Baukasten für technische Systeme

In dem Beitrag soll die Vorgehensweise am Beispiel eines Krans beschrieben werden. Ziel der in diesem

Beitrag beschriebenen Vorgehensweise ist es jedoch nicht, ein einzelnes technisches System zu modellieren, sondern einen Baukasten für einen bestimmten technischen Bereich zur Verfügung zu stellen. Der Baukasten soll die technischen Komponenten für diesen Bereich (im Beispiel: Kräne) enthalten sowie eine generische Grafikkomponente zur Visualisierung derartiger Systeme.

Lehre

Auf der Webseite *physolator.com* gibt es bereits eine Aufgabensammlung, die in der Lehre eingesetzt werden kann, um Studenten und Schüler mit unterschiedlichen Voraussetzungen an das Thema Simulation heranzuführen. Diese Aufgabensammlung soll um Aufgaben zu technischen Systemen erweitert werden.

Verweise

- [1] D. Eisenbiegler, "Objektorientierte Modellierung und Simulation physikalischer Systeme mit dem Physolator", BoD Norderstedt, 2015.
- [2] D. Eisenbiegler, "The Software Architecture of the Physolator – a Physical Simulation Framework", MSAM 2015, Atlantis Press, pp. 61-64.
- [3] D. Eisenbiegler, "A Generic Particle Modeling Library for Fluid Simulation", AMSM 2016, Atlantis Press.
- [4] D. Eisenbiegler, "An Object Oriented Library for Acoustics Simulation Based on the Physolator Simulation Framework", CMSAM 2016, DEStech Publications.

Model-driven development and simulation of Integrated Modular Avionics (IMA) architectures

Björn Annighöfer¹

¹Institute of Aircraft Systems, University of Stuttgart

bjoern.annighoef@ils.uni-stuttgart.de

Presented is model-driven development method for avionics systems comprising of a domain-specific model, mathematical optimization, and an attached network simulation. For Integrated Modular Avionics (IMA) the degree of freedom in choosing the system architecture is so high that determining the optimum by hand is hardly possible for large aircraft. A domain-specific model was created with the Eclipse Modeling Framework (EMF) holding system requirements and architecture variants, such that it can automatically be validated and evaluated. Moreover, combinatorial optimization is used to determine optimal architectures by algorithm for single and multiple objectives. Optimization on civil aircraft and a space launcher revealed improvements of up to 30% in single design objectives. Moreover, the architecture model can automatically be converted in configuration stubs and an AFDX network simulation.

1 Introduction

Integrated Modular Avionics (IMA) are state-of-the-art for large civil and military aircraft. The concept of IMA is that computing, memory, and IO resources are shared between several safety-critical and non-critical system functions. System functions are, for instance, cabin pressure control and landing gear retraction. Those are loaded as segregated software partitions. The major portion of the avionics system's hardware is standardized. Computing and IO modules are configured in software to fulfil their purpose in multiple system functions.

IMA reduced the hardware, cost, weight, and space [1, 2]. Two challenges arising are first design freedom and second the massive number of configuration parameters. Considering design, questions like, what is the lightest architecture, how many modules do I need, or what is the optimal IO distribution per module, can no more be answered optimally by hand. Considering configuration, current IMA systems require so many parameters that the process became inefficient and error-prone [3, 4, 5].

To improve the situation, a domain-specific model was developed for computer-aided design of avionics architectures. It holds the complete architectures, but in addition the system requirements, like functions, resource needs, and safety constraints. Architectures shall be automatically validated, evaluated and compared. It is generic in terms that no precise avionics technology or function architectures are predefined.

Requiring only a minimum mandatory information makes it applicable to the design phase. Nevertheless, it can be automatically converted to mathematical optimization problems. For instance, function assignment, routing, and module sizing. Moreover, the model is used to derive configuration stubs or simulations automatically. All is implemented in a seamless model-driven IMA design method depicted in **Figure 1**.

The remainder of this article is organized as follows. Section 2 introduces the domain-specific avionic architecture model. Section 3 explains a multi-objective optimization approach and section 4 shows an example of an AFDX simulation derived from the model. The article ends with a conclusion and outlook.

2 Avionics modeling

A standardized avionics system is a distributed computing platform, which provides computing resources and I/Os to aircraft system functions running as software. Since the number of different module types is kept as small as possible in order to reduce development costs, there is usually a large number of equal hardware modules, which could technically host the same functions. Whether a certain module should host a function or not, often depends on the position, the distance to required sensors and actuators, and safety consideration. Overall, there is a high degree of freedom, e.g. the dimension of modules, the installation locations, the assignment of functions, the network topology, and the routing of signals.

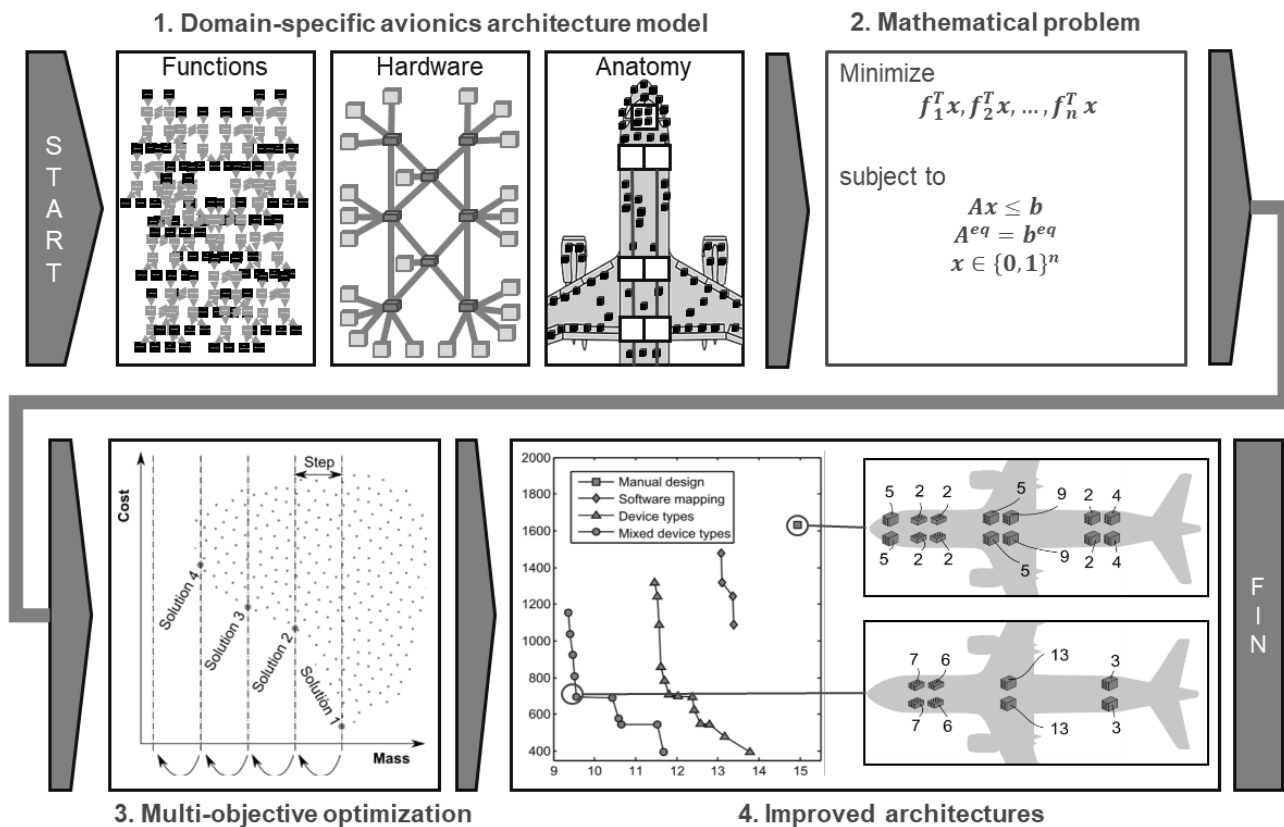


Figure 1. Workflow of the model-driven IMA architecture design method

A domain-specific model was developed especially for the purpose of avionics system design. Therefore, it generically captures the capabilities and resources of hardware, without requiring a certain module type or technology. Moreover, it generically covers the software as atomic building blocks, so called tasks, signals, and their resource requirements. In addition, segregation, symetrie, location, and power constraints can be attached to single, tasks or task groups.

The first instance of an avionics architecture model [6] was used in several in air and space research programs. At the end of 2017, a second generation of the domain-specific model has been finalized [7], which is more, considering bus systems and hardware. It has been made available as Open Source as the **Open Avionics Architecture Model¹ (OAAM)**.

OAAM is designed in nine almost independent layers. This matches the concurrent development process of IMA systems, which is distributed over multiple parties. The nine layers are Library, Scenarios, Systems

Functions, Hardware, Anatomy, Capabilities, Restrictions, Mapping.

The top-level class diagram of OAAM is depicted in **Figure 2**. The four main layers are Function, Hardware, Anatomy, and Mapping.

The **Functions layer** holds all tasks to be assigned to the avionics system and the signals that must be routed. In addition, it includes timing and safety constraints.

The **Hardware layer** allows modelling device instances and interconnection topologies without physical dimensions.

Within the **Anatomy layer**, the installation locations and cable routes of the aircraft including the length and positions are modeled. It is a graph-like representation of a simplified 3-dimensional construction plan.

¹ www.oaam.de

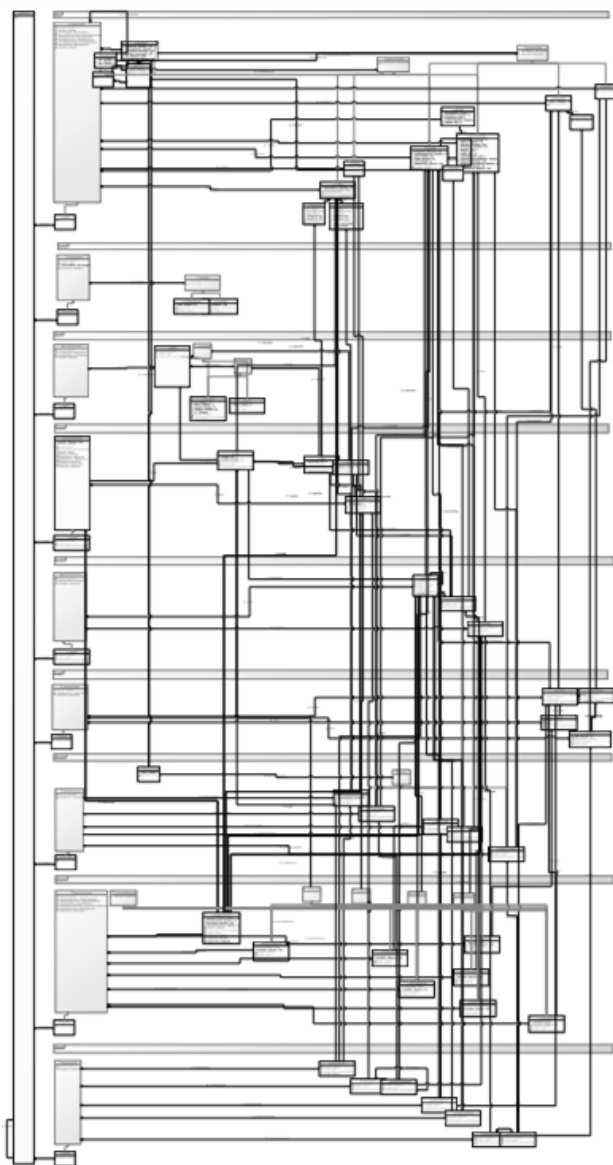


Figure 2. Top-level view on all packages of the OAAM meta-model

In the **Mappings layer** assignment objects can be created, which assign task to devices, devices to installation locations, as well as signals and cables. The basic constraint determining if an assignment is valid or not, is a linear **resource provisioning and resource consumption model**. Each task or device K_i requires a set of certain resources r^{K_i} in a certain amount $r_j^{K_i} \in \mathbb{R}_+$, i.e.

$$r^{K_i} = (r_1^{K_i}, \dots, r_n^{K_i}). \quad (1)$$

A resource is an abstract unit of what has to be provided to the task or device and is consumable, e.g. computational power, memory or space in installation

location. Devices and locations provide resources. An assignment to D_j is valid as long as the available resources r^{D_j} are not exceeded, i.e.

$$\sum_{K_i \in K} r^{K_i} \leq r^{D_j}. \quad (2)$$

This must hold for all assignments of the architecture. In addition, constraints on device, locations, power sources, areas, symmetries, and co-location control what are valid mappings. Multiple mapping variants of the same elements can be created to represent different architecture variants. For each variant the validity and design objectives are individually be calculated.

Technically OAAM is realized with the ECORE **meta-modeling** language of the Eclipse Modeling Framework (EMF)², which allows to define formal UML-like meta-models and automatically derive the implementation, persistence layer, and edit tools. Moreover, extensions exist for the verification and evaluation of EMF derived domain-specific models. OAAM models are edited, validated, and evaluated within a specialized Eclipse instance. In addition, an interface to MATLAB was developed.

3 Avionics architecture optimization

The IMA systems of current aircraft have more than 4000 tasks and peripheral as well as more than 50 devices and thousands of possible installation locations and cable routings. The pure number of elements prevents that the design engineer is able to derive the optimal dimensioning, installation, and software assignment by hand. Even if he would do, he would not be able to prove it. Moreover, the optimality of an avionics system is not a unique property. There are multiple design objectives desired in the design process. Simple examples are cost and mass, which shall both be minimal. Objectives that are more complex are installation cost or maintenance effort. There is usually not a single architecture optimizing all objectives, but objectives are partially contradictory, such that the best trade-off is desired.

In order to automated and prove the optimality of recurring design tasks, a link between the architecture model and combinatorial optimization was developed. Overall, **eight generic optimization routines** for avionics architecture were developed as depicted in **Figure 3**. Optimization routines are classified in 1-level ,

² www.eclipse.org/modeling/emf/

2-level, and 3-level assignments depending on how many different element types are assigned in parallel.

The most basic optimization routine is **function assignment**, which assigns a set of functions to a given set of devices. The optimization objectives are, for instance, the device weight (i.e. use a few devices as possible) or the wire weight (i.e. put tasks as close as possible to the related sensors and actuators). In addition, all defined constraints for the tasks must hold.

A more complex routine is **device type optimization**, which starts from an empty topology and selects the optimal number, installation, and dimensioning of devices, while assigning tasks.

3-level assignment is the highest level of automation. It derives devices and network, as well as task assignment and signal routing in a single step. It has the highest degree of freedom and it was show in [8] that in general this leads to the highest optimization potential. However, also the complexity of the optimization problem rises, such that this is only feasible to calculate single systems with no more than 50 tasks. 1-level and 2-level assignments are feasible for scenarios with 4000 and more tasks. Calculation times are between several hours and four weeks.

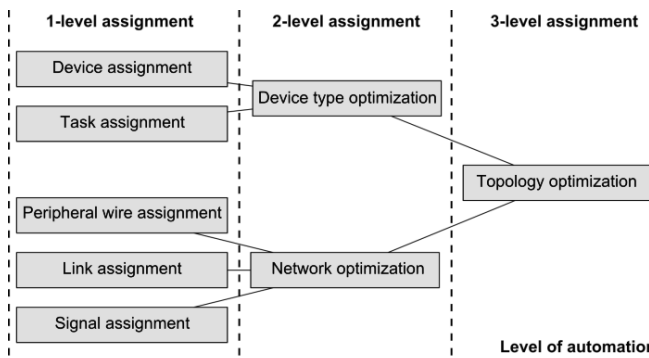


Figure 3. Eight generic optimization routines for avionics architecture optimization

Technically, all optimization routines are implemented as multi-objective binary programs (BP). A multi-objective BP searches for a binary solution vector x such that

$$f_1^T x, f_2^T x, \dots, f_n^T x \quad (3)$$

are minimized subject to

$$\begin{aligned} Ax &\leq b \\ A^{eq}x &= b^{eq} \\ x &\in \{0,1\}^n. \end{aligned} \quad (4)$$

x encodes - depending on the optimization routine - either task assignments, signal routes, or hardware topologies. The cost vectors f_1, \dots, f_n allows for linear objectives. With the addition of auxiliary variables, also non-linearities can be included. Linear inequalities A and equalities A^{eq} constrain the resource consumption, enforce segregation, and ensure a unique mapping of every object. For more information on the optimization problem formulations please refer to [9, 10, 11, 12, 13].

Information from the domain-specific model are automatically converted to the mathematical optimization problems. The conversion and the invocation of combinatorial optimization are implemented in MATLAB. Commercial-of-the-shelf solvers³ for Mixed Integer Linear Programming (MILP) are integrated to efficiently solve BPs. The solution is converted back in meaningful model information. In case of multiple-contradicting objectives, a custom iterative multi-objective solver calculates the so-called Pareto optimum [14], i.e. the set of best possible trade-off solutions. See an example of a Pareto optimum in **Figure 1** on the lower right.

Several of the optimization routines were used to derive an optimal avionics system for the ARIANE 5 space launcher [15, 16] and for deriving some general scaling laws for avionics architecture design [17]. In all application, up to 30% improvements in single objectives between the manually derived architectures and the optimized architectures were found.

4 AFDX simulation

The avionics architecture model presented above is static. For design, validation, and evaluation static properties and capacities are assumed that have sufficient safety margins such that the architecture should also be valid during operation. For instance, during architecture desing the CPU is modeled as a static resource, i.e. the percentage of CPU load consumed by each task vs. 100% available CPU. For sure, however, within the real system, most tasks are periodic and must be scheduled and the schedule must be valid in

³ IBM CPLEX or GUROBI

terms of the individual deadlines. This can be considered by assuming, for instance, a maximum load of 70% during design. Nevertheless, real schedules have to be determined and their correctness has to be proven.

Another example is the network. During desing and optimization, bandwidth is assumed as a single consumable resource. Each signal has a bandwidth portion it consumes. However, in the real systems, network messages have to be scheduled and maximum transfer delays have to be proven. Typically, this happens with network calcus or network simulations.

A common network for IMA systems is **Avionics Full Duplex Switched Ethernet (AFDX)**. AFDX is an asynchronous switched network based on Ethernet. AFDX messages address virtual links (VL) instead of targete devices. VLs are preconfigured and static communication routes in the network. Each VL is assigned a maximum bandwidth (Bandwidth Allocation Gap - BAG). If too large or too many messages are send, those are dropped by the switches. An AFDX network and the corresponding message routing is assumed to be valid if for each VL a valid BAG can be found and if under worst case conditions the maximum delay and jitter requirments for each signal are met. It is common to validate this in simulations.

Worst case conditions, delay, and jitter measurements cannot be made in the OAAM model. However, a simulation framework for AFDX was developed within MATLAB/SIMULINK. The simulation framework provides basic components for switches, as well as the AFDX send and receive interfaces of avionics devices. In addition, it is able to run-on a rapid prototyping system, which is able to output the virtual communication on a real AFDX interface. All components of the simulation framework can be used to create virtual AFDX networks by hand and validating the timing of all messages up to an accuracy of 1 ms. More commen is, however, to derive the AFDX simulation model automatically from the OAAM static architecture, which – if a mapping was calculated – includes all necessary information. **Figure 4** shows an example of an Airbus A380-like AFDX network completely derived from an architecture model. It was possible to run the simulation in realtime on a rapid prototyping system.

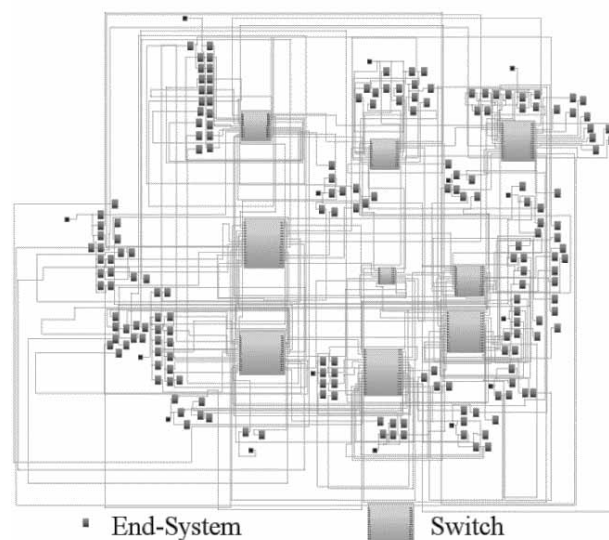


Figure 4. An A380-like AFDX system simulated in MATLAB/SIMULINK

The detailed description of the AFDX simulation and the results can be found in [18].

5 Conclusion

Current avionics systems are generic resource-sharing distributed computing systems. The most common representatives are Integrated Modular Avionics (IMA). A domain-specific model for avionics architectures and system function requirements enables computer-aided design of avionics systems including automated validation, evaluation, and optimization. Eclipse EMF and MILP solvers are used as technologies. Optimization of real aircraft and the ARIANE 5 space launcher revealed improvments of up to 30% in single objectives as weight, compared to the manual desing. The applicability of the model during system development is extended by the possibility to derive dynamic simulations, which was shown for a MATLAB/SIMULINK AFDX simulation.

In future works, it is intended to extend the capabilities of the model-driven avionics architecture tool chain by additional optimization routines. Moreover, it is worked on a self-configuring avionics system, which utilizes OAAM as its online knowledge base.

6 References

- [1] J. W. Ramsey, "Integrated modular avionics: Less is more - approaches to IMA will save weight, improve reliability of A380 and B787 avionics," *Avionics magazine*, 2007. [Online]. Available: <http://www.aviationtoday.com/av/categories/commercial/8420.html>
- [2] C. Wilkinson, "IMA aircraft improvements," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 20, no. 9, pp. 11–17, September 2005.
- [3] H. Butz, "Open integrated modular avionic (IMA): State of the art and future development road map at airbus deutschland," in *Proceedings of the 1st International Workshop on Aircraft System Technologies*, March 2007, pp. 211–222.
- [4] C. Watkins and R. Walter, "Transitioning from federated avionics architectures to integrated modular avionics," in *Digital Avionics Systems Conference, 2007. DASC '07. IEEE/AIAA 26th*, oct. 2007, pp. 2.A.1–1 –2.A.1–10.
- [5] A. Wilson and T. Preyssler, "Incremental certification and integrated modular avionics," in *27th Digital Avionics Systems Conference*, 2008.
- [6] B. Annighöfer, E. Kleemann, and F. Thielecke, "Model-based development of integrated modular avionics architectures on aircraft-level," in *Deutscher Luft- und Raumfahrtkongress, Bremen 27. - 29. Sept. 2011*, no. 1395. Bremen: Deutsche Gesellschaft für Luft- und Raumfahrt, September 2011.
- [7] B. Annighöfer, M. Riedlinger, and O. Marquardt, "How to tell configuration-free integrated modular avionics what to do?!" in *36th Digital Avionics System Conference*, Saint Petersburg, FL , USA, September 2017.
- [8] B. Annighöfer and F. Thielecke, "A systems architecting framework for optimal distributed integrated modular avionics architectures," *CEAS Aeronautical Journal*, pp. 1–12, 2015.
- [9] B. Annighöfer, "Model-based architecting and optimization of distributed integrated modular avionics," Dissertation, Hamburg University of Technology, March 2015. ISBN: 978-3-8440-3420-2
- [10] B. Annighöfer, E. Kleemann, and F. Thielecke, "Automated selection, sizing, and mapping of integrated modular avionics modules," in *32st Digital Avionics System Conference*, Syracuse, NY, USA, October 2013.
- [11] B. Annighöfer and F. Thielecke, "Supporting the design of distributed integrated modular avionics systems with binary programming," in *Deutscher Luft- und Raumfahrtkongress, Berlin 10. - 12. Sept. 2012*. Berlin: Deutsche Gesellschaft für Luft- und Raumfahrt, September 2012
- [12] —, "Multi-objective mapping optimization for distributed modular integrated avionics," in *31st Digital Avionics System Conference*, Williamsburg, VA, USA, October 2012.
- [13] —, "Network topology optimization for distributed integrated modular avionics," in *33rd Digital Avionics System Conference*, Colorado Springs, CO, USA, October 2014.
- [14] M. Ehrgott, *Multicriteria Optimization*, 2, Ed. Springer Berlin Heidelberg, 2005.
- [15] B. Annighöfer, C. Nil, J. Sebald, and F. Thielecke, "Structured and symmetric ima architecture optimization: Use case Ariane launcher," in *34th DASC*, 2015.
- [16] B. Annighöfer, Çelen Nil, J. Sebald, and F. Thielecke, "Ariane-5-based studies on optimal integrated modular avionics architectures for future launchers," in *6th EUCASS*, 2015.
- [17] B. Annighöfer, V. Posternak, and F. Thielecke, "Empirical investigations on avionics scaling laws," in *35th Digital Avionics System Conference*, Sacramento, CA , USA, September 2016.
- [18] B. Annighöfer, H. Ihle, and F. Thielecke, "An easy-to-use real-time AFDX simulation framework," in *35th Digital Avionics System Conference*, Sacramento, CA , USA, September 2016.

Variability Modeling for Engineering Applications

Hendrik Folkerts¹, Thorsten Pawletta¹, Christina Deatcu¹, Umut Durak²

¹University of Applied Sciences Wismar, Research Group Computational Engineering and Automation

²German Aerospace Center (DLR)

hendrik.folkerts@cea-wismar.de

The System Entity Structure (SES) is a high level approach for variability modeling, particularly in simulation engineering. The SES is under continuous development. In this context, an enhanced framework is introduced that supports dynamic variability evolution using the SES approach and connects the SES to a model base (MB), which allows building executable models. However, the main focus of this paper is to show how to use the SES to model complex engineering system configurations by the example of a test bench for valves. Specifying the SES, it is clarified that the system configuration can be decomposed into generalized design patterns. The design patterns were identified during the development of our MATLAB-based SES toolbox for construction and pruning of SES trees and were employed for testing and verification of the respective functionality.

1 Introduction

Variability modeling is an approach to describe more than one system configuration derived from one underlying configuration. The basic system configuration is often referred to as product line [1]. According to [2] the most popular technique of modeling commonality and variability of products in software engineering is feature modeling. Feature models were introduced by Kang in [3] and subsequently extended and used in various ways. An important further development of variability modeling has been the notion of variability in time, known as binding time in product line engineering [4]. Variability can be realized from design time to runtime.

An approach to variability modeling in simulation engineering is the System Entity Structure (SES) introduced by Zeigler in [5]. The objective was to describe a set of system configurations for a family of systems. In the early nineties the idea of combining the SES with a model base (MB) in order to generate an executable model led to the SES/MB approach [6]. An SES is represented by a tree structure, which describes a set of modular, hierarchical system structures, defines references to basic models in a MB and specifies various parameter settings for the referenced basic models. The classical SES/MB framework only allows static modeling. To allow dynamic variability modeling and to keep a lean SES tree, the SES/MB framework was extended by several features, such as an interface to the SES and a procedural knowledge representation [7]. In the research group Computational Engineering and Automation (CEA) at the University of Applied Sciences Wismar a prototype of a full SES/MB-based modeling and simulation

infrastructure has been developed and implemented within MATLAB/Simulink.

In this paper a short introduction into the extended SES/MB framework is given. It is demonstrated how the framework can be used to simulate a configurable test bench for valves. The main goal is to show that the whole SES can be constructed with elementary design patterns. These in turn can be classified in the categories of the feature relations in feature models.

2 Background

According to [8], the SES is an ontology, a language with syntax and semantics to represent declarative knowledge. It is particularly suitable for describing system configurations for different application domains. An SES is represented by a directed tree structure. Objects are represented by nodes which are connected by edges. There are four node types with different properties describing the objects and their relations. Furthermore, there are axioms for defining the SES correctly. Since an SES describes a number of system configurations, the SES tree needs to be pruned to get one particular configuration, which is called Pruned Entity Structure (PES).

The classic SES theory was extended by several researchers over the last decades. Some of these extensions, which are introduced in [7, 9], will be used in this presentation, too.

2.1 Node Types

Among the four node types, there are two groups, the entity nodes and the descriptive nodes. Entity nodes describe objects of the real or the imaginary world.

The root and the leaves of an SES tree are always entity nodes. Relations between the entity nodes are specified by descriptive nodes.

Descriptive nodes are the genus for aspect nodes, specialization nodes and multi-aspect nodes. Aspect nodes describe how entity nodes can be disassembled in partial entities whereas the taxonomy of an entity is described by specialization nodes. Multi-aspect nodes are a special case of an aspect node with all children being of the same kind.

Each node or edge can have attached variables, also called attributes. For entity nodes, the variables represent properties of the respective object whereas the variables at descriptive nodes specify relations between their parent node and children nodes or decisions for the pruning process. With the extended procedural knowledge representation as described in [7, 9], values of attached variables can be assigned dynamically.

2.2 Axioms

The semantics of the SES are defined by axioms. The types of the nodes have to follow the axiom alternating mode. Every entity node has to be followed by a descriptive node, and vice versa. A strict hierarchy is needed. In every path of the tree, a name of a node may occur only once. If nodes in different paths have the same name, they need to have the same variables and isomorphic partial trees. This is called uniformity. Nodes on the same level of hierarchy, called sibling nodes, have to be valid brothers, meaning that sibling nodes must not have the same name. The axiom of attached variables implies that a node must not have variables of the same name. The axiom of inheritance implies that, during pruning, the parent and the child of a specialization combine their attributes. If parent and child have the same attributes, the parent attributes are overwritten with the child attributes and their values.

2.3 Extended SES/MB Infrastructure

The SES describing a set of system designs has been associated with the idea of model generation of modular, hierarchical systems from the very beginning (Zeigler, Praehofer, and Kim 2000) which led to the SES/MB approach. Each system design is defined by its system structure and parameter configuration in the SES. The core assets of all system variants are specified as a set of configurable basic models, which are organized in a Model Base (MB). The classic

SES/MB framework defines a set of transformation methods for generating executable simulation models, but automated model generation is not provided. To allow automated generation and execution of models, the SES/MB approach has been extended [7, 9, 10]. These extensions make the SES/MB approach more pragmatic for implementation and to be used in a simulation infrastructure. Figure 1 depicts the extended SES/MB infrastructure consisting of the SES/MB framework, an Execution Unit, and an Experiment Control. Although the SES/MB approach and its extensions are usually considered in connection with the generation of simulation models, they are generally applicable to modular-hierarchical structured software systems.

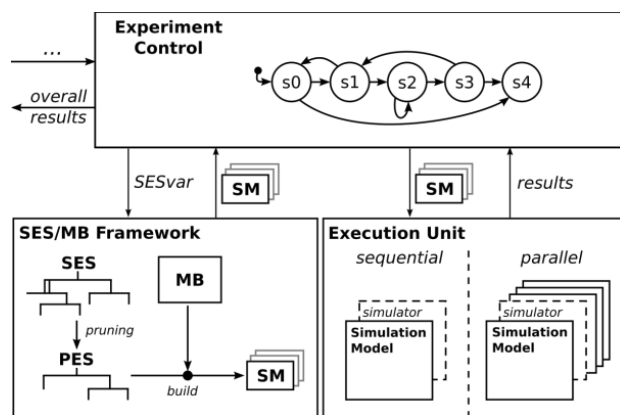


Figure 1. Extended SES/MB-based infrastructure

Operations

On the SES, a merge operation is defined allowing two or more SES to be combined. This allows the quick reuse of a once defined SES. The essential operation on the SES is the pruning method. To extract one particular system structure and configuration, the SES needs to be trimmed to a PES. During the pruning process, decisions have to be taken at descriptive nodes. Therefore, rules need to be defined at aspect, multi-aspect and specialization nodes. The specialization rule (specrule) associated with a specialization node determines which child entity shall be selected. Aspect rules (aspectrule) associated with aspect or multi-aspect nodes on the same hierarchy level determine which of the siblings is to be chosen. Furthermore, cross-tree relations can be expressed by selection constraints. Selection constraints can be used to select a certain entity based on decisions taken anywhere else in the SES tree. Next to the pruning method, another transformation method is the build method. With the help of the build method, an executable model can be built from a PES and basic

models organized in an MB. The basic models are specific for a simulation software. Therefore, the build method needs to match to the simulator used.

Execution Unit and Experiment Control

For automated and reactive processing of SES models, an execution unit and an overall experiment control unit are added to the framework, as depicted in Figure 1. For automatic generation of different PES, leading to different simulation models, an interface to the SES is needed. This interface can be established by global variables of the SES, called SES Variables (SESvar), which can affect the decisions taken in descriptive nodes during pruning. Thus, a particular system configuration derived from an SES depends on the current settings of the SES variables. The value range of SES variables can be limited by defining semantic conditions, which are checked before pruning to exclude certain system configurations. By assigning values to the SES variables, the experiment control determines the order and system configurations of executable simulation models (SM) to generate from the SES with the pruning and build operations. Different variants of the executable simulation models are generated hereby. The experiment control then transmits the SM to the execution unit. The execution unit links the generated simulation model to the simulator, executes a simulation run and, finally, sends the results back to the experiment control. The results, in turn, can influence the decision of the experiment control on how to assign the SES variables next.

Special Attributes

Combining basic models from the MB leads to the creation of coupled models. In order to describe the structure of the executable model, some nodes need to define couplings. Couplings are properties of descriptive nodes of the type aspect and multi-aspect and consist of pairs of entity names and port names. They describe causal or acausal relationships. Furthermore, for a multi-aspect node, a special variable, numRep, has to be defined representing the number of children to generate when pruning this node. To specify the basic model from the MB an entity node refers to, the mb-attribute is introduced. This special attribute is permitted just for leaf nodes. Finally, for some cases, it is necessary to define priorities among descriptive nodes on the same level of hierarchy in the priority attribute. All values of attributes can be

defined by constants or set via SES variables or SES Functions.

SES Functions

The concept of SES Functions (SESfcn) has been introduced to specify complex variability within node attributes with minimal effort and to keep a lean SES tree. Typical examples include the definition of varying coupling relations, varying port numbers of systems or the definition of variable parameter configurations in attributes. During pruning, SES functions are evaluated, often with SES variables as input parameters. For effective coding of SES functions, the implicit attributes parent and children are introduced for each SES node. They encode the parent and children node names, respectively.

2.4 Software Tools for the Extended SES/MB Infrastructure

In the Research Group CEA, a prototype tool for the SES/MB infrastructure was developed, The SES Toolbox for Matlab/Simulink [10]. Currently, SES trees can be defined via a graphical user interface and a concrete variant can be extracted by pruning. The toolbox supports the modeler with plausibility test during SES construction, graphical representation of the SES, automatic generation of HTML documentation, and other features. The pruning process can be started from the graphical user interface and, in addition, is implemented to function automatically. Automatic pruning is necessary when using an SES constructed with the toolbox together with the experiment control. Furthermore, there is a prototype Matlab function implementing a build method for the simulation software Simulink, including SimEvents and Simscape, the MatlabDEVs toolbox [11], and for Modelica models. The SES is linked to the appropriate MB with the special mb-attribute of the leaf entity nodes.

Another software tool based on Python3/PyQt5 is under development. The aim is to be more independent from a computing environment and to support a greater number of simulators for building executable simulation models.

3 An Engineering Application Specified with the Extended SES/MB Framework

For development and testing of the pruning algorithm built in the software tools, design patterns were

found. Design patterns are rather small SES fragments with a certain behavior when pruned. They can be used to compose complex SES. The design patterns identified can be classified according to the feature relations in feature models. In the context of feature modeling, four kinds of features are used: (i) mandatory features (logical AND), (ii) alternative features (logical XOR), (iii) optional features and, (iv) OR-features (logical OR). In an SES mandatory and alternative features can be expressed in several ways [12]. In order to be able to describe optional features an extension of the SES is used, the NONE element [8]. A NONE element as a leaf entity node means that, if the NONE branch is selected, the entity is not included at all. OR features can only be expressed by the combination of a decomposition with two or more specializations. Each specialization needs to have one NONE element as child.

In this section, an engineering example describing a test bench for valves is given, here called industrial plant (*industrialPlant*) for generalization. The SES of the whole plant structure and the corresponding process controls is depicted in Figure 2. However, the several coupling attributes are not specified in detail. Since different valve types shall be tested, there is a need for a variability in structure for different test procedures. The example shall demonstrate how a complex SES tree can be composed of design patterns, how the SES can be pruned to extract one variant in the form of a PES, and how an executable

model can be built from the PES. The feature relations, except for the OR relation, can be found in the SES. Because of complexity the SES is split up into several SES to increase clarity and to demonstrate modular SES-based modeling.

3.1 Main SES

The plant is a composition of the plant structure and the necessary process control strategies. Firstly, the plant structure (*plantStructure*) shall be explained. The plant consists of the plant parts for gas (*gas*), liquid (*liquid*), the electrical power supply (*electricalEnergy*) and finally, there are *controllers* to drive the plant. The *gas* and *liquid* entities are decomposed further whereas the electrical energy (*electricalEnergy*) is a merge point to the SES in Figure 6. Merge points are written in bold letters in Figure 2. As described, merge points allow to combine two SES, which are specified separately. A merge point is characterized by the same entity node name in both SES, which shall be combined. This implies, that both SES are defined completely and fulfill all axioms.

The mandatory relations described by the aspect node *plantStructureDec* represent the simplest design pattern for pruning, since the resulting PES is identical to the SES. We call it design **pattern #1**. The design patterns are numbered according to the classification found in [12]. However, in the following pruning is mentioned for all design patterns. One resulting PES representing one test bench configuration is discussed

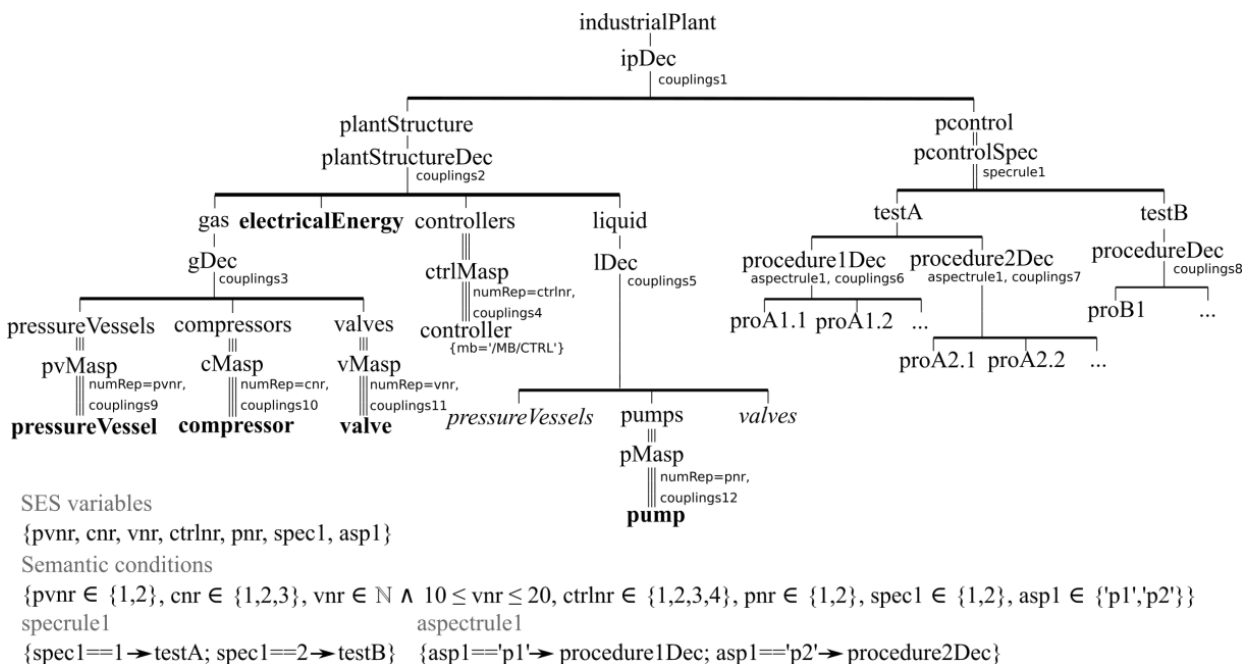


Figure 2. Main SES of the test bench for valves (without detailed coupling definitions)

in section 3.8.

Pressure vessels (*pressureVessels*) as well as *compressors*, *valves*, *pumps* and *controllers* are entities to subsume a varying number of identical children. They are the parents of a multiple aspect, so during pruning a number of their respective children is created. The entity *controllers* for example consists of a number of the entity *controller*. The number is defined in the variable *numRep*. This is another mandatory feature referring to feature models. It is referred to as design **pattern #2**. Pressure vessel (*pressureVessel*), *compressor*, *valve*, and *pump* are merge points to SES specified later.

As described before, one SES axiom urges nodes in different paths with the same name to have the same variables and isomorphic partial trees. Therefore the entities *pressureVessels* and *valves* in the branch for the liquid (written italic in Figure 2) do not need to be specified since they are specified for the partial tree defining the gas.

Before listing the sub SES for merge points, the process control part in the main SES shall be explained. In a derived variant of the SES, either *testA* or *testB* can be part of the PES to specify a particular system configuration. This is an alternative selection, called design **pattern #4**. Therefore, the entities *testA* and *testB* are specializations of the process control that is represented by the entity *pcontrol*. For decision, which test to execute, a specialization rule (specrule) has to be defined at the specialization node *pcontrol-Spec*. During pruning the specrule is interpreted and the selected *testA* or *testB* entity is united with the parent entity *pcontrol*.

If *testA* is selected, there are two possible main procedures which can be taken alternatively. The decision is taken by evaluation of aspectrule1 at *procedure1Dec* and *procedure2Dec*. This design pattern for an alternative selection is referred to as **pattern #5**. The decision taken leads to a number of elementary procedures (e.g. *proA1.1...proA1.n*) specific for the selected superordinate main procedure. For *testB* there is only one main procedure leading to a number of elementary procedures as explained in design **pattern #1**.

The SES in Figure 2 demonstrates the use of SES variables (SESvars) for dynamical generation of different system configurations. SESvars can contain any type of value. At nodes needing a decision during pruning, the corresponding attribute or rule can be

defined based on an SESvar, so each value can be assigned dynamically. The value range of every SESvar should be restricted using semantic conditions. A semantic condition needs to evaluate to a logical value and can contain more than one logical expression. SESvars are assigned to the *numRep* attributes at the multi-aspect nodes. There are the SESvars *spec* and *asp* used in the specrule and the aspectrule respectively. If the SESvar *spec* equals 1, *testA* is taken, if *spec* equals 2, *testB* is selected during pruning. The SESvar *asp* can take one of two possible strings as value. The evaluation of the aspectrule defines that, if *asp* equals the string *p1*, *procedure1Dec* is selected, if *asp* equals the string *p2*, *procedure2Dec* is taken.

The aspect and multi-aspect nodes need the special attribute couplings as seen in Figure 2. They describe the relations of entity nodes. An example for a coupling definition is given in section 3.9. In this context the usage of SES functions is illustrated as well.

3.2 Sub SES: Pressure Vessel

In Figure 3 the pressure vessel (*pressureVessel*) is specified. Every pressure vessel has the two attributes volume and maxPressure. Creating the PES depending on the specrule defined, either *steel* or *aluminum* is taken as material, which is united with the *pressureVessel* entity. The mb-attribute of the leaf entities refers to a basic model in the MB as depicted in Figure 8. During pruning attributes are inherited to the parent, which is renamed and has three attributes afterwards. This is another example for the design **pattern #4** with the addition of attribute inheritance. As in the main SES the decision at the specialization is taken based on the specrule and the current values of SES variables.

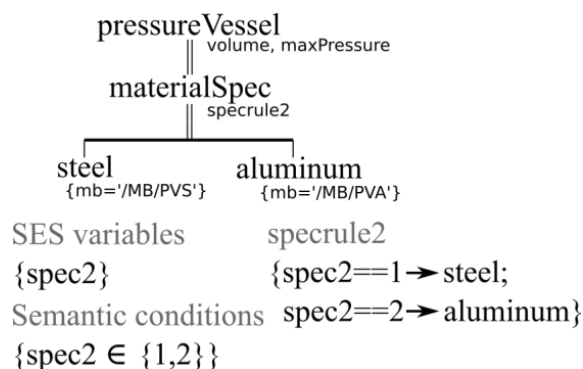


Figure 3. SES of a pressure vessel

3.3 Sub SES: Compressor

The details of the compressor can be seen in Figure 4. A *compressor* is composed of the *drive*, the *cooling* and a *controller*. The *drive* is composed of the *motor* and a *transmission*. This is an SES construction as described in design **pattern #1**. The motor can be specified as a three-phase asynchronous motor or an ac motor, whereas these refer to a model in the MB with their mb-attribute. During pruning, this part is resolved as described for the pressure vessel in design **pattern #4**. The cooling is an optional element. Since it is a specialization with a NONE element, the pruning process resolves the cooling either to water cooling, air cooling or no cooling at all. The optional tree section is referred to as design **pattern #8**.

3.4 Sub SES: Valve

Valves are needed for the plant parts for gases as well as for liquids and are only defined in the partial SES defining the gas as described before. A valve is of a type and needs to have a drive. The specialization siblings *typeSpec* and *driveSpec* in Figure 5 can be classified as mandatory feature and are referred to as design **pattern #3**. The entity *electricalDrive* is specialized into *acMotor* and *dcMotor* according to design **pattern #4**. During the pruning process all specializations are resolved, attributes are inherited and the entity *valve* is renamed with the selected type, drive and, if the *electricalDrive* is selected, the motor type. Two specialization nodes in one path as seen in this example is a combined design pattern referred to

as design **pattern #10** and is showing attribute inheritance among several layers. In section 3.8 the inheritance of names and attributes is explained in detail. The *controlValve* is composed of the *cvController* and the *cvActuator*. Seen from the top entity *valve*, this is another combined design pattern consisting of a specialization with a succeeding aspect. We call it design **pattern #11**. It shows that during pruning the specialization *typeSpec* is resolved according to design **pattern #4** and the aspect *cvDec* is not changed like in design **pattern #1**. The dotted line in Figure 5 is a selection constraint for cross-tree relations, which forces the use of a *pneumacticalDrive*, if a *viceValve* is selected. It may be noticed, that in Figure 5 not all leaf nodes have an mb-attribute. Either the leaf nodes in the branch of the *typeSpec* entity or the leaf nodes in the branch of the *driveSpec* are allowed to have mb-attributes. The other branch can only configure the basic model, which the mb-attribute refers to, by evaluating the attributes. There is no rule to specify, which specialization to prune first at specialization siblings. During pruning identical attributes are overwritten in parent nodes, so would be the mb-attribute.

3.5 Sub SES: Electrical Energy

The SES describing the electrical energy is shown in Figure 6. The electrical energy (*electricalEnergy*) can be generated with a number of generators specified in the variable numRep or the electrical grid (*gridDec*) is used. The grid is decomposed into a power plant

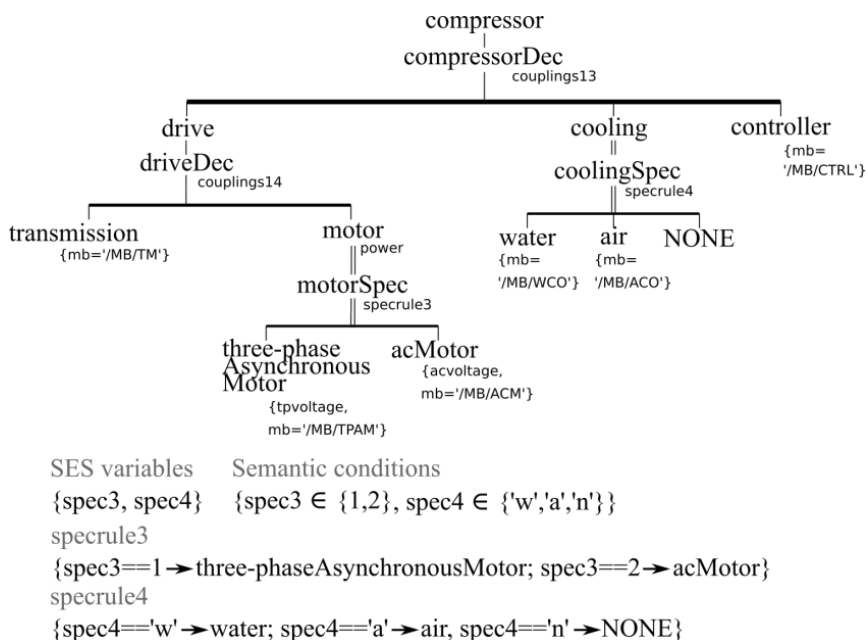
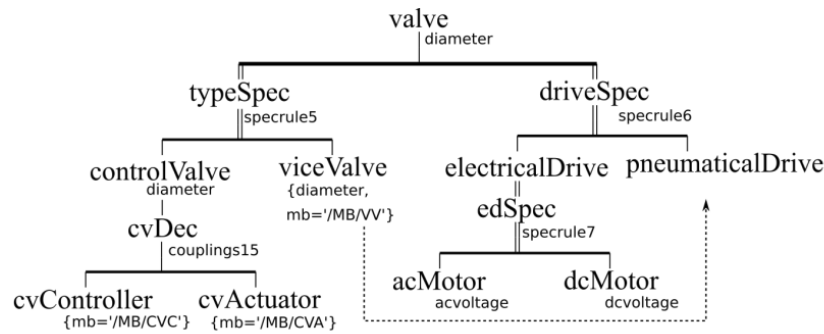


Figure 4. SES of a compressor



SES variables Semantic conditions
 {spec5, spec6, spec7} {spec5 ∈ {1,2}, spec6 ∈ {1,2}, spec7 ∈ {1,2}}
 specrule5 specrule6 specrule7
 {spec5==1 → controlValve; {spec6==1 → electricalDrive; {spec7==1 → acMotor;
 spec5==2 → viceValve}; spec6==2 → pneumacticalDrive} spec7==2 → dcMotor}

Figure 5. SES of a valve

(powerPlant) and the transmission net (transmission-Net). The decision, whether the generator or the grid is selected, is taken in aspectrule2. This alternative structure created of siblings of aspect and multi-aspect nodes is called design **pattern #7**.

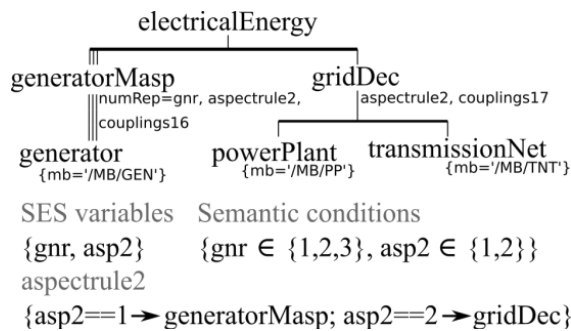


Figure 6. SES of the electrical energy

3.6 Sub SES: Pump

The *pump* can be built with the SES depicted in Figure 7. It is decomposed into the *motor*, the *wheel* and a *controller* just like design **pattern #1**. The motor does not need to be specified since it is already defined in the SES of the compressor (see Figure 4).

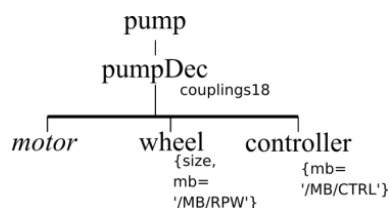


Figure 7. SES of a pump

3.7 Model Base

For model generation a library containing the configurable basic models specified in the mb-attribute of

leaf nodes is needed. A possible model base (MB) for this example is given in Figure 8. The MB contains the dynamic models for simulation of the respective devices. Every model needs input and output ports for building coupling relations. A basic model in the MB can be composed of a number of other basic models as depicted for the *TNT* model.

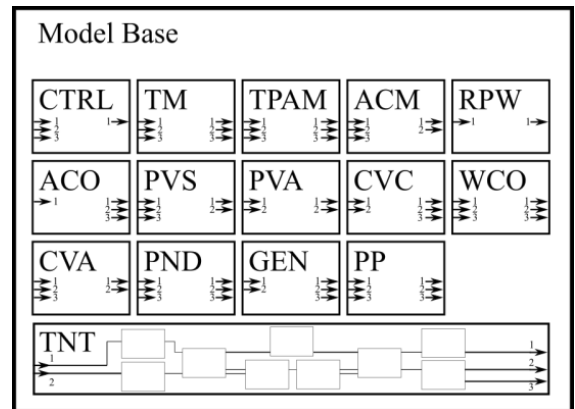


Figure 8. Model Base for this example

3.8 Deriving a PES

Since an SES specifies a number of different system configurations, the SES has to be transferred to a PES by pruning in order to get a particular system configuration. For the configurable test bench for valves one possible PES is depicted in Figure 9. Before pruning, values are assigned to the SESvars and checked using the semantic conditions. The process of resolving all necessary decisions is described in the following.

First, it shall be described how to derive a particular variant of the main SES. Since the resulting PES for

pruning **pattern #1** is identical to the SES, the composition of the plant (*industrialPlant*) as well as the composition of the plant structure (*plantStructure*), the gas (*gas*) and the liquid (*liquid*) are part of the PES. The multi-aspect nodes *pvMasp*, *cMasp*, *vMasp*, *ctrlMasp*, and *pMasp* are resolved following design **pattern #2** by converting them in aspect nodes with the name extension *Dec* and a number of children defined in *numRep* is created. The number of each child is appended to the nodename of the child.

For the electrical energy (*electricalEnergy*) the selection for *generatorMasp* is taken in the *aspectrule2* (design **pattern #7**) and the respective children (*generator*) are created as in design **pattern #2**.

The pressure vessel is pruned according to design **pattern #4**. As material of the pressure vessel (*pressureVessel*) aluminum is selected in the *specrule*. The name of the selected entity (*aluminum*) is united with the parent of the specialization (*pressureVessel*) to *aluminum_pressureVessel*, attributes of the children are inherited to the renamed parent and the specialization node *materialSpec* is deleted.

The compressor drive (*drive*) can be pruned using design **pattern #1** and design **pattern #4** as before. The cooling is a specialization evaluating to *NONE* according to design **pattern #8**. The *NONE* node is depicted for understanding, but has no relevance for

the succeeding model generation.

For generating the PES of a valve both specialization siblings have to be resolved, the resulting names are connected and the attributes are inherited (design **pattern #3**). It depends on the pruning algorithm used, which of both specializations is pruned first. There is no explicit rule. In this example the pruning is started with the specialization *typeSpec*. The type *controlValve* is selected and an intermediate parent entity *controlValve_valve1* can be assumed. The value of the attribute *diameter* in the parent node is replaced with the value of the same attribute in the child according to design **pattern #4**. The tree beginning with *cvDec* is a composition, which is not changed during pruning (design **pattern #1**).

Since *electricalDrive* is selected in *driveSpec*, both of the specializations in one path (design **pattern #10**) *driveSpec* and *edSpec* need to be resolved like in design **pattern #4**. The entity node *dcMotor* is selected, the attribute *dcvoltage* is moved to the top entity which is renamed to *dcMotor_electricalDrive_controlValve_valve1* according to the selections.

The entities *pressureVessels* and *valves* in the branch for the liquid (written italic in Figure 9) are specified for the partial tree defining the gas and do not need to be specified again. That implies, in this design the

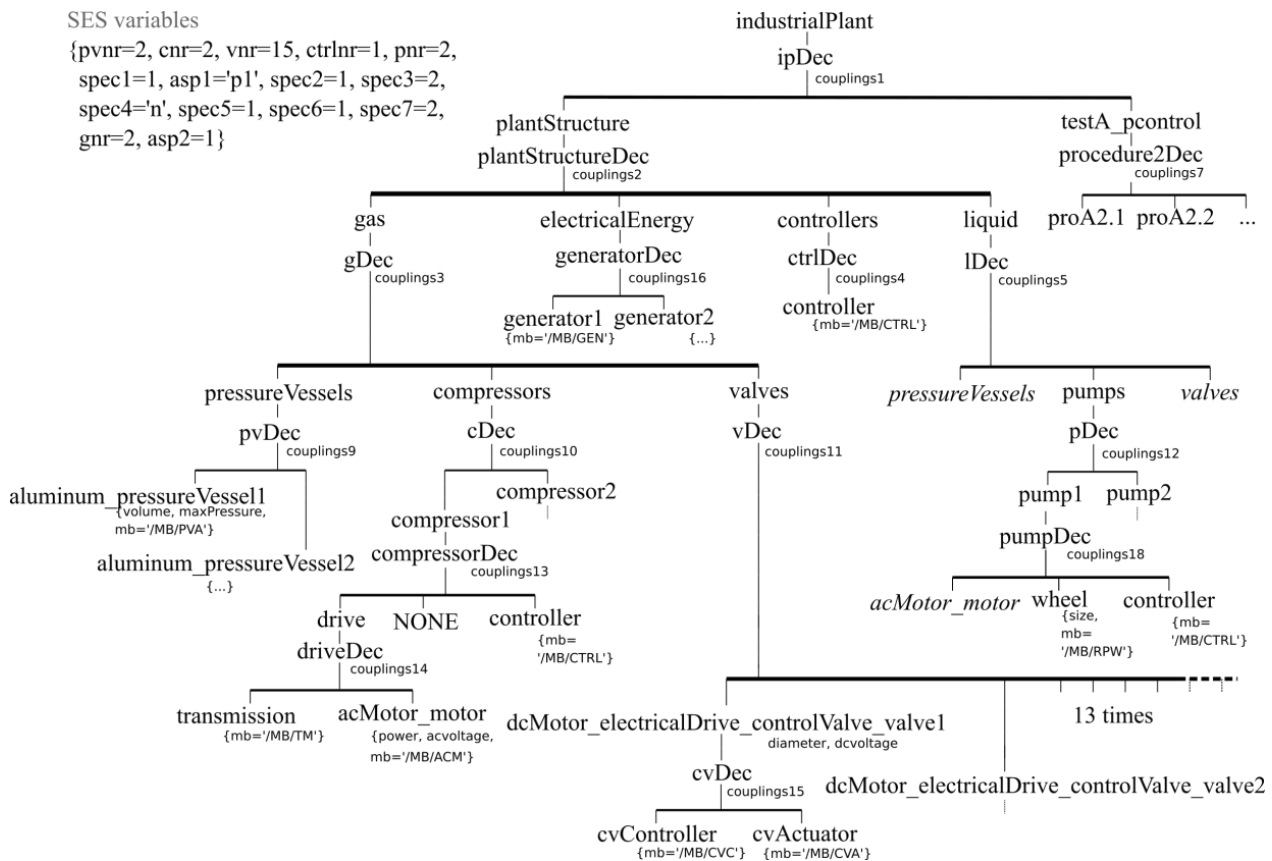


Figure 9. Example of a PES describing one particular system configuration

pressure vessels and the valves have be of the same type and number and have to have the same properties for both plant parts.

A pump is a composition of a motor, a *wheel* and a *controller* and is pruned according to design **pattern #1**. However, since the motor of the pump is the same entity as of the compressor because of the same name in the SES, in the PES both nodes need to have the same name as well.

The process control (*pcontrol*) is derived by evaluating a specrule like in design **pattern #4** and an aspectrule for aspect siblings (design **pattern #5**). The entity node *testA* is selected and united with *pcontrol*. Finally the aspect node *procedure2Dec* is selected, which is the child of the renamed *testA_pcontrol*.

Summarized in this configuration the industrial plant has four pressure vessels, two compressors, thirty valves, two pumps, two generators and one controller for the plant. Each device of these device groups has the same structure and is of the same type. However, in the case of the motors of the compressors and pumps it makes sense to relax the axiom of uniformity, so that different values can be assigned to the

power attribute. The relevance to relax the uniformity attribute is discussed in [7] comprehensively.

Since the PES is a decision-free tree structure, a particular executable model can be generated based on its information and using basic models from the MB.

3.9 Building the Executable Model

For the generation of the executable model, the couplings attributes are especially important, as discussed in section 2.3. Examples on how couplings are specified and resolved are depicted in Figure 10 according to the SES part of a drive of a compressor in Figure 4.

In the SES couplings often have to be defined dynamically, since the SES has to describe several configuration variants. During pruning, a particular configuration with static couplings is derived, so that an executable model can be built. Dynamic specification can be done using SESfcns as illustrated in pseudo-code in Figure 10. In the example, the attribute couplings14 in the SES has to specify the coupling relations for both types of basic motor models with a different number of ports. In the PES the concrete

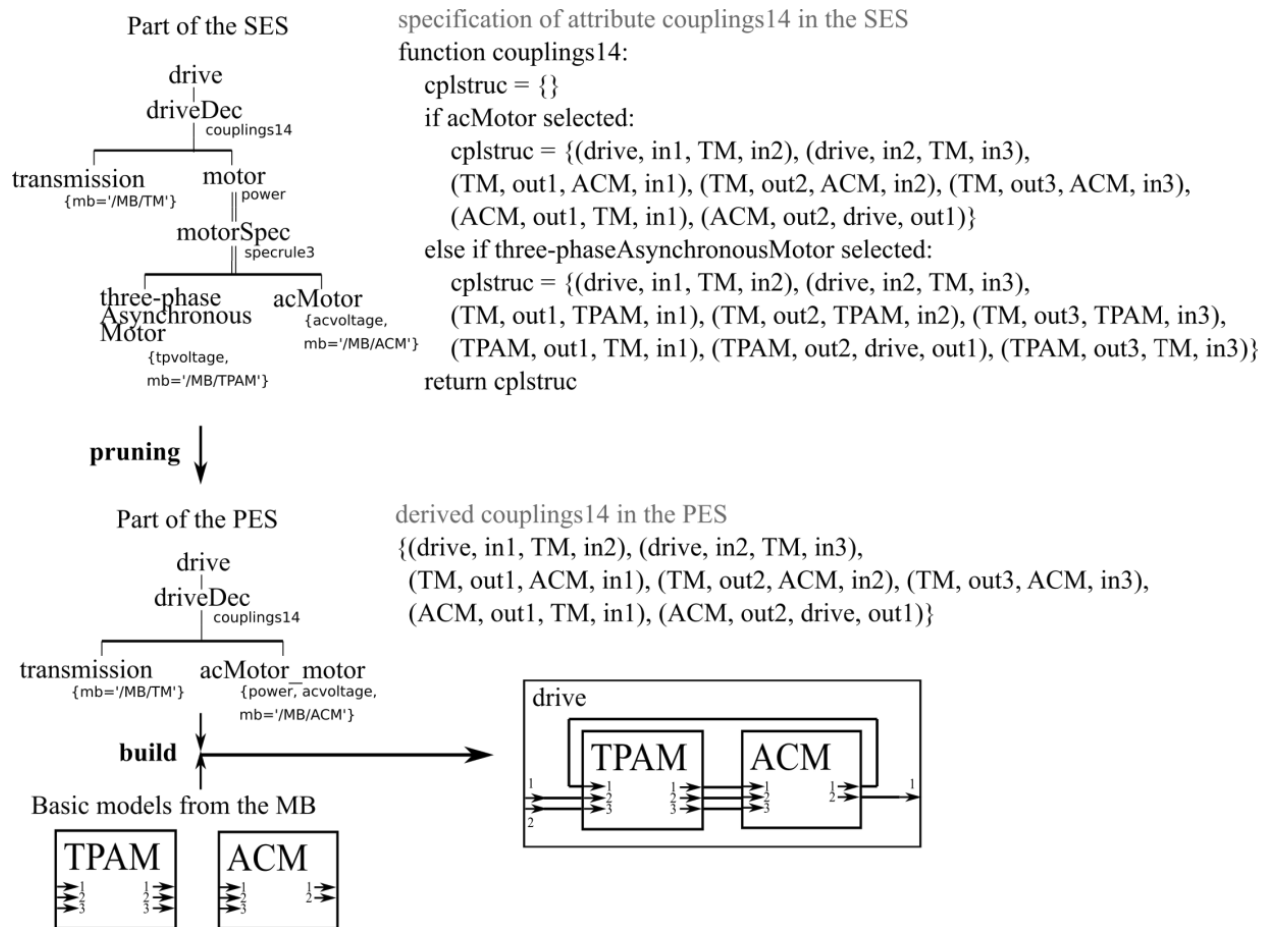


Figure 10. Specification of dynamic couplings in an SES, derivation of static couplings in the PES and building an executable model

static couplings for the selected basic models are derived, in this case for the basic model ACM.

Particularly, when defining the couplings for a multiple-aspect node or for nodes referring to basic models, whose port numbers vary depending on the selected configuration, defining the couplings via SESfens is best practice.

4 Conclusion

This paper discusses, how to use the extended SES/MB framework in order to describe and build a set of simulation models of a complex configurable engineering application by the example of a test bench for valves. It is shown that previously found design patterns are appropriate to specify a complex engineering problem. Such engineering problems can be modeled using the Software Tools for the Extended SES/MB Infrastructure presented and can be built for different simulators.

5 References

- [1] O. Alt. *Variantenmanagement*. In: Modellbasierte Systementwicklung mit SysML, Hanser, Germany, pp. 139-144, 2012.
- [2] K. C. Kang and H. Lee. *Variability Modeling*. In: Systems and Software Variability Management, Editors: R. Capilla, J. Bosch, and K. C. Kang, Springer, Germany, pp. 25-42, 2013.
- [3] K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Nowak, and A. S. Peterson. *Feature-Oriented Domain Analysis (FODA) Feasibility Study*. Technical Report, Software Engineering Institute Carnegie Mellon University, USA, 1990.
- [4] R. Capilla and J. Bosch. *Binding Time and Evolution*. In: Systems and Software Variability Management, Editors: R. Capilla, J. Bosch, and K. C. Kang, Springer, Germany, pp. 57-73, 2013.
- [5] B. P. Zeigler. *Multifaceted Modelling and Discrete Event Simulation*. Academic Press, UK, 1984.
- [6] B. P. Zeigler, T. G. Kim, and H. Praehofer. *Theory of Modeling and Simulation*. Academic Press, USA, 2000.
- [7] T. Pawletta, A. Schmidt, B. P. Zeigler, and U. Durak. *Extended Variability Modeling Using System Entity Structure Ontology within MATLAB/Simulink*. Proceedings of the 2016 Spring Simulation Multi-Conference, Pasadena/CA, USA, pp. 62-69, 2016.
- [8] B. P. Zeigler, and P. E. Hammonds. *Modeling and Simulation-Based Data Engineering*. Academic Press, USA, 2007.
- [9] A. Schmidt, U. Durak, and T. Pawletta. *Model-Based Testing Methodology Using System Entity Structures for MATLAB/Simulink Models*. SIMULATION: Transactions of The Society for Modeling and Simulation International. 92(8), pp. 729-746, 2016.
- [10] RG CEA. *The SES Toolbox for MATLAB / Simulink Website*. http://www.cea-wismar.de/tbx/SES_Tbx/sesToolboxMain.html, Accessed Jan. 15, 2018.
- [11] RG CEA. *The PDEVs Toolbox for MATLAB Website*. http://www.cea-wismar.de/tbx/DEVs_Tbx/MatlabDEVs_Tbx.html, Accessed Jan. 15, 2018.
- [12] C. Deatcu, H. Folkerts, T. Pawletta, and U. Durak. *Design Patterns for Variability Modeling Using SES Ontology*. Paper submitted for Spring Simulation Multi-Conference, Baltimore/MD, USA, 2018.

Experimental Adaptation of a Training Simulator for Manual Welding Processes towards the Teach-In of Welding Robots

Benjamin Knoke¹, Mareike Voskuhl², Marcel Tebbe², Markus Häveker³, Christian Gorltdt¹, Klaus-Dieter Thoben¹

¹ BIBA – Bremer Institut für Produktion und Logistik GmbH at the University of Bremen

² University of Bremen

³ Daimler AG

kno@biba.uni-bremen.de

The programming of welding robots in the automotive industry is mostly performed through the programming by demonstration technique. During such teach-in processes, the robots are manually guided along their desired path. This path is saved as a program that can later be accessed to perform the taught tasks within the production line.

The programming by demonstration technique is usually applied by dedicated programmers with limited welding experience. This results in one of the main causes for delays during the startup of new production lines.

The paper describes an attempt to provide an assistance during the teach-in process by experimental adaptation of training simulator components. A mounted camera system continuously monitors the relative position and orientation of the welding torch and workpiece. The simulation contains images of metal textures and a welding seam that are computed in real-time and that are displayed using augmented reality technology. Feedback on deviations from the ideal position, orientation, and movement is displayed during the process and can also be analysed afterwards.

The experimental application is strongly limited by the restriction to a given set of basic workpieces. However, the visual feedback was considered by the programmers to be very helpful and the application showed great potential that could be used for dedicated industrial products.

1 Introduction

Automatisation technology within the automotive sector strongly relies on welding robots [1]. Due to the ongoing trend towards a „digital factory“, this is not expected to change within the foreseeable future [2], [3]

The setup of the welding robots is usually performed by a qualified programmer in multiple iterations. The body parts that are used during setup are usually scrapped. Nowadays, the welding programmes are often prepared in CAD environments or during other forms of simulations, but this still does not lead to acceptable quality levels without conventional testing. As current car models inherit up to 600 complex welding seams, this results in considerable costs.

The presented work focuses on the experimental adaptation of a welding simulator, which is usually used within manual welding training exercises, to guide the programming of welding robots. This simu-

lator visually tracks the relative positions and orientations of welding gear and workpiece to continuously provide feedback to the user. This paper investigates the application of the generated feedback information to guide the teach-in process of welding robots.

2 State of the Technology

2.1 Programming of Welding Robots

The most prevalent methods for the programming of welding robots include the generation of programmes based on CAD models, where the path is defined within a CAD environment [4], and the programming by demonstration technique. The latter is commonly used within the automotive sector [5]. During such teach-in processes, the robots are manually guided along their desired path. This path is saved as a program that can later be accessed to perform the taught tasks within the production line.

In industrial practice, the programming is usually applied by dedicated programmers with limited welding experience. However, some complex welding seams require careful tuning of the welding parameters [6]. This results in one of the major causes for delays during the startup of new production lines.

A similar approach to the presented method has been proposed by Ni et al. (2017). It generates 3D models of scanned workpieces and employs a haptic input device to define the welding path. The programming is supported by an AR interface as well as haptic force feedback. This approach is designed for scenarios in which the users do not have physical access to the workpieces and the feedback is limited to the positioning of the welding seam, not the generated quality [7].

2.2 Training Simulators for Manual Welding

The current key simulation technologies in welding are Virtual Reality and Augmented Reality. During the past decade these technologies were applied to create welding simulators and made the transition towards practical application [8], [9].

These simulators rely on optic measurement to capture the position and movement of a welding torch or electrode [10]. The measured characteristics include stick out, work angle, travel angle, travel path, and travel speed. The available welding simulators differ greatly in physical and functional fidelity, as well as in price.



Figure 1. Soldamatic welding simulator

The applied simulator system is the Soldamatic training simulator for welders (Fig. 1), which focusses on manual welding processes and is a commercial product sold by Seabery Soluciones. SL [12]. The hardware consists of a computing unit that is embedded in a welding equipment casing; a welder's helmet that holds cameras, speakers and an integrated display; attachable hand-held devices for MIG/MAG, elec-

trodes, and GTAW welding processes; a stand; and multiple standard work pieces that are made out of plastic and are printed with reference markers [12].



Figure 2. Soldamatic evaluation screen

A mounted camera system continuously monitors the relative position and orientation of the welding torch and workpiece. The simulation contains images of metal textures and a welding seam that are computed in real-time and that are displayed using augmented reality technology. Feedback on deviations from the ideal position, orientation, and movement is displayed during the process and can also be analysed afterwards (Fig. 2).

3 Methodology

The paper describes an attempt to provide an assistance during the teach-in process by experimental adaptation of training simulator components. The experiment uses the MIG/MAG simulation in combination with a T-joint workpiece.

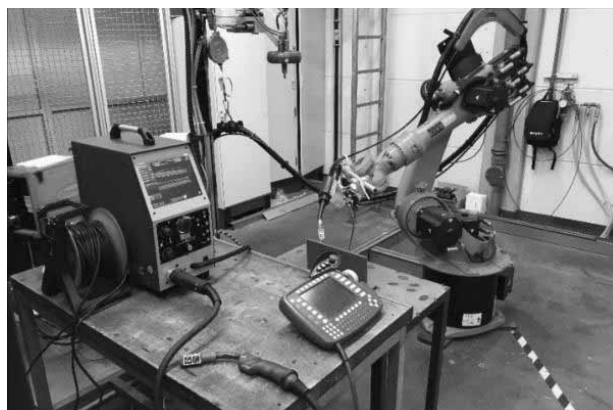


Figure 3. Experimental setup

The main components are the Soldamatic welding simulator as well as a KUKA KR15 industrial robot

with six axes and a payload of 15kg. The complete assembly is depicted in figure 3. The welding helmet of the simulator has been disassembled, as only the camera system was required. Additionally, a light attachment has been constructed to mount the camera system on the industrial robot (figure 4). The artificial welding gun for the robot has been created with a 3D printer to resemble the simulator's MIG/MAG gun, as it is required to hold the reference markers in their original position.

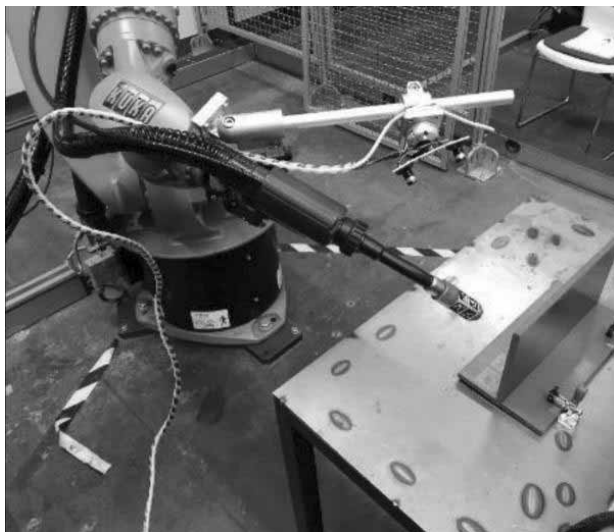


Figure 4. Camera mount and custom welding torch in position over the workpiece

The simulation parameters are listed in table 1. The experiment has been run in a configuration for beginners, as this offers the most guidance during the process. The simulations' feedback has been applied to tune the position, orientation, and movement of the welding robot. The robot has been programmed with an average speed of 1,5mm/s and five reference points have been defined along the weld. Once the calibration has been optimized according to the simulation, the exercise was performed multiple times.

parameter	value
difficulty	beginner
material	steel
workpiece thickness	10mm
welding additive diameter	1mm
seam type	single-layer fillet weld
protective gas	argon mixed gas
current	23A
voltage	30V

Table 1. Selected simulation parameters

The final evaluation has been performed by industrial experts, who included welding trainees, welding trainers, and maintenance staff.

4 Findings and Limitations

Once the programming of the welding robot had been tuned according to the feedback of the simulation, the repetition of the experiment constantly led to positive results, which indicates a certain degree of consistency.

The industrial experts stated the application could be used to display and correct position and orientation of the welding torch relative to the workpiece. The ideal configuration of these parameters was considered to be similar to the programming made based on the simulation. However, the velocity that is demanded by the simulation, is estimated to be too low for industrial practice and may ultimately lead to the destruction of the workpiece due to burning through.

The practical usability of the simulations in production was rated as high by the maintenance employee. Even trained welding experts benefit in his opinion from the guidance, since the programming of a welding robot, in contrast to manual welding, cannot be performed intuitively. The technology also provides the opportunity to have an initial teach-in done by a robot expert, so that a welding expert is only required to assist at the end of the teach-in process for additional fine-tuning operations.

It was also highlighted that the working position of the programmer and his vision on the task are improved by the simulation. Only in rare moments, a detailed look at the exact position of the nozzle must be taken. The adjustment of angle and distance can be done on the monitor of the simulation device. This could be particularly helpful in the case of inaccessible or unergonomic seam positions.

Overall, the experts were sceptical as to whether the simulation device can reproduce the very complex processes involved in industrial welds. Necessary changes in the speed with varying wall thicknesses, material changes or elaborate curves are initially not provided in the system.

5 Conclusion and Outlook

The experimental application of the training simulator towards the teach-in of welding robots has been performed for a sample T-joint workpiece. The industrial experts reported the feedback through Augmented

Reality a considerable benefit in finding the correct position and orientation of the welding torch. This benefit would be especially relevant in settings with limited access or visibility.

However, the experimental application is strongly limited by restriction to a given set of basic workpieces. The introduction of additional workpieces to the Soldamatic simulation has already been performed in few cases, but requires considerable effort and interaction with the developer. The approach described by Ni et al. (2017) employs a scanning module that creates point clouds of the workpiece [7] and could be a possible approach to overcome this barrier. A method to define the seam positions would be required as well, which should be flexible enough to follow the outline of complex seams and allow for changes in material thickness.

Additional challenges were imposed by the optical sensors, which had to be positioned at an average distance of 300 mm to the workpiece. The recognition was impeded by vibrations of the system that were caused by robot movement. A dedicated application should be designed rather compact to allow for welds that are difficult to access.

6 Acknowledgement

The authors would like to thank the Mercedes-Benz plant in Bremen as well as the Daimler Zukunftswerkstatt for the opportunity and support necessary to realize the experiments, as well as the German Federal Ministry of Education and Research (BMBF) for their support in the project "MESA - Media Use in Welding Education" (grant number 01PD14016).

7 References

- [1] T. Bauernhansl, M. Ten Hompel & B. Vogel-Heuser. *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien und Migration*. Springer, Vieweg, Wiesbaden. 2014.
- [2] D. Hömberg, C. Landry, M. Skutella & W. A. Welz. *Automatic Reconfiguration of Robotic Welding Cells*. Mathindustry 27, pp. 183-203, 2017
- [3] A. Lüder & N. Schmidt. *Challenges of Mechanical Engineering of Production Systems: An Automation System Engineering View*. Mathindustry 27, pp. 93-114, 2017.
- [4] L. A. Ferreira, Y. L. Figueira, I. F. Iglesias, & M. Á. Souto. *Offline CAD-based Robot Programming and Welding Parametrization of a Flexible and Adaptive Robotic Cell Using Enriched CAD/CAM System for Shipbuilding*. Procedia Manufacturing 11, pp. 215-223, 2017.
- [5] P. Kah, M. Shrestha, E. Hiltunen & J. Martikainen. *Robotic arc welding sensors and programming in industrial applications*. International Journal of Mechanical and Materials Engineering 10(13), 2015.
- [6] C. M. Horváth, P. Korondi & T. Thomessen. *Robotized multi-pass Tungsten Inner Gas welding of Francis hydro power turbines*. Proceedings of the 26th International Symposium on Industrial Electronics (ISIE), pp. 1759-1765, 2017.
- [7] D. Ni, A. W. W. Yew, S. K. Ong & A. Y. C. Nee, *Haptic and visual augmented reality interface for programming welding robots*. Advances in Manufacturing 5(3), pp. 191-198, 2017.
- [8] K. Fast, T. Gifford & R. Yancey. *Virtual training for welding*. Proceedings of the 3rd IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004) in Arlington, pp. 298-299, 2004.
- [9] S. White, M. Prachyabrued, D. Baghi, A. Aglawe, D. Reiners, C. Borst & T. Chambers. *Virtual welder trainer*. Proceedings of the IEEE Virtual Reality Conference 2009 in Lafayette, pp. 303-303, 2009.
- [10] M. L. L. Okimoto, P. C. Okimoto & C. E. Goldbach. *User experience in augmented reality applied to the welding education*. Procedia Manufacturing 3, pp. 6223-6227, 2015.
- [11] B. Knoke & K. D. Thoben. *Integration of Simulation-based Training for Welders*. Simulation Notes Europe 27(1), pp. 37-44, 2017.
- [12] Seabery Soluciones. *Soldamatic educational augmented reality: la tecnología educativa más avanzada y competitiva para la formación de soldadores, 100% española*. Soldadura y tecnologías de union 23(130), pp. 58-60, 2012.

Maschinelles Lernen beim visuellen HMI in der Testautomatisierung vernetzter Fahrzeugfunktionen bei Volkswagen

Robert Buchta, Adrian Heller

Prüfinfrastruktur zur Elektronikintegration und -absicherung auf System- und Gesamtfahrzeugebene,
Volkswagen AG, Wolfsburg
robert.buchta@volkswagen.de

In diesem Beitrag wird aufgezeigt, wie das maschinelle Lernen in der Auswertung des visuellen HMI bei der testfallfreien Testautomatisierung unterstützt. Dabei werden die einzelnen HMI-Elemente von Analysealgorithmen ausgewertet. Um möglichst viele Szenarien mit zahlreichen Situationen auswerten zu können, unterstützt eine Klassifikation der HMI-Elemente erheblich. Im Beitrag wird das Vorgehen mittels merkmalsbasierender Verfahren zur Bildauswertung vorgestellt und diskutiert. Dabei wird auf die Detektion und Deskription der Merkmale entsprechend der vorhandenen Anforderungen eingegangen. Weiterhin wird die Erstellung eines Bag of Visual Words vorgestellt, der als Grundlage für die Klassifikation dient. Abschließend wird noch die Verwendung der Klassifikation in der testfallfreien Testautomatisierung skizziert.

1 Einleitung und Motivation

Die Variantenvielfalt sowohl technischer Art als auch Ländervarianten des HMI (engl. Human Machine Interface) in Kombination mit agiler Softwareentwicklung stellt hohe Ansprüche an die Testautomatisierung. Erhöht wird die Komplexität zusätzlich durch das rasante Wachstum neuer Funktionen, die durch eine Onlineverbindung des Fahrzeugs mit dem Internet ermöglicht werden.

Die agile Softwareentwicklung mit ihren kurzen Zyklen und damit häufigen Ergebnissen garantiert stetig wachsende Funktionshöbe für das HMI des Gesamtfahrzeugs. Um diesen schnellen Intervallen in der Testautomatisierung gerecht zu werden, setzt Volkswagen verstärkt auf neuartige Testverfahren und Auswertemethoden für das HMI des Gesamtfahrzeugs.

Am Beispiel der Fahrzeugklimatisierung fokussiert dieser Beitrag die Methodik zur Auswertung des visuellen HMI in der Testautomatisierung. Hierzu werden merkmalsbildende Verfahren der Bildauswertung verwendet. Neben den Anforderungen für den Anwendungszweck fokussiert dieser Beitrag das Vorgehen zur Detektion und Deskription der Merkmale und die anschließende Klassifikation der HMI-Elemente. Die Klassifikation unterstützt erheblich die Auswertung des Verhaltens bei testfallfreien, virtuellen Straßenfahrten, die automatisiert durchgeführt werden.

2 Anforderungen an die visuelle Auswertung des HMI in der Testautomatisierung

In der Testautomatisierung werden für die Fahrzeugfunktionen die HMI-Elemente überprüft. In den Gesamtbildern der Anzeige-Bedieneinheit des Infotainmentsystems und des Kombiinstrumentes wird das Vorhandensein der richtigen HMI-Elemente überprüft. Bislang erfolgte eine Bildanalyse zur Messung der Ähnlichkeit von Objekten bzw. Bildern, für das automatisierte, testfallfreie Testen kommt nun eine Objekterkennung und Klassifikation von Objekten dazu.

Folgend werden die Anforderungen an die visuelle Auswertung des HMI erläutert, um in den späteren Kapiteln die Wahl der verschiedenen Methoden nachvollziehen zu können.

Die Bildregistrierung bzw. Objektlokalisierung spielt eine weniger wichtige Rolle. Zwar wird bislang beim Template Matching die präzise Position ermittelt, jedoch aber auch nur mit dem Ziel das höchste Ähnlichkeitsmaß, zur Überprüfung der Übereinstimmung, zu ermitteln. Die Region of Interest (RoI) ist für eine erfolgreiche Klassifikation wichtig, jedoch muss für den angedachten Anwendungszweck nicht die exakte Position eines Objektes im Gesamtbild ermittelt werden. Vielmehr soll das Vorhandensein eines Objektes im Gesamtbild mit hoher Zuverlässigkeit ermittelt werden. Hierbei wird der Ansatz verfolgt die Merkmale eines Objektes im Gesamtbild wiederzufinden

unter der Bedingung, dass die Summe der Merkmale eines jeden interessierenden Objekts von allen anderen Objekten sicher unterschieden werden kann. Dabei variieren die Merkmale in gewissen Grenzen, da sich das HMI in einer sehr dynamischen Entwicklung mit entsprechend häufigen Softwarelieferungen befindet.

Die Variationen der Merkmale sind zurückzuführen auf eine stetige evolutionäre Entwicklung der HMI-Elemente über den Entwicklungsprozess hinweg. Die HMI-Elemente reifen bis zum Ende des Entwicklungsprozesses. Weiterhin gibt es geringfügige Translationen, Rotationen und Skalierungen der HMI-Elemente über die Softwarelieferungen hinweg. Robuste Detektorverfahren, die in der Testautomatisierung ein sicheres und zuverlässiges Erkennen von Objekten sicherstellen können, sollten für die angedachte Anwendung folgende Eigenschaften erfüllen [1]:

- Reproduzierbarkeit,
- Einzigartigkeit,
- Stabilität.

Zur Reproduzierbarkeit sollten die Schlüsselpunkte der korrespondierenden Merkmale des Objektbildes und des Referenzbildes gleich sein. Dies stellt die Unabhängigkeit der Punkte gegenüber Rauschen und Blickwinkeländerung sicher. Die Umgebung der Schlüsselpunkte variiert genug, um die jeweiligen Punkte hervorzuheben und somit zwischen den Merkmalen unterscheiden zu können (Einzigartigkeit). Die Punkte sollten einzigartige Eigenschaften haben, welche sie invariant gegenüber Rotation in mehreren Achsen, Translation in mehreren Richtungen, Unschärfe oder Skalierungsänderungen macht (Stabilität). Insbesondere stärkere Skalierungsänderungen resultieren aus den unterschiedlichen Technikvarianten der Infotainmentgeräte.

Das Zeitverhalten muss ebenfalls betrachtet werden. Der gesamte Auswerteprozess muss mit einem langsamen dynamischen Prozess Schritthalten können. Es geht hierbei um Antwortzeiten von bis zu mehreren Zehntelsekunden, die in der Testautomatisierung eingehalten werden müssen.

Schlussendlich müssen die Methoden für die Detektion, Deskription und Klassifikation mit diesen Variationen des sich entwickelnden HMIs umgehen können, um zuverlässig die Objekte präzisieren zu können.

Zur Organisation der Arbeit fördert Transparenz in den unterschiedlichen Teilaufgaben eine allgemeine Akzeptanz bei den vielen beteiligten Mitarbeitern in der Funktionsentwicklung der Domänen. Ebenso unterstützt Transparenz bei der Wahrung des Überblicks bei der stetig wachsenden Anzahl zu klassifizierender Objekte. Und eine Aufgabenteilung lässt Freiraum, um langfristig Teilaufgaben durch andere Methoden ersetzen zu können. Diese zusätzlichen Anforderungen begrüßen ein Vorgehen der Aufgabenteilung bei der Detektion, Deskription und Klassifikation. Stark integrative Lösungen, die diese Teilaufgaben integrativ und teils wenig transparent und nachvollziehbar durchführen, wie beispielsweise Convolutional Neural Networks (CNN) sind wegen der organisatorischen Anforderungen weniger im Fokus.

3 Merkmalsdetektion und -deskription

Für die Klassifikation ist eine Extraktion der Merkmale von entscheidender Bedeutung, um eine hinreichende Unterscheidung der Objektkategorien sicherzustellen. Für das systematische, aufgabengeteilte Vorgehen erfolgte zur Auswahl eines Detektors und Deskriptors eine Analyse und Bewertung der Methoden MSER, SURF und HOG. Als Referenz dient das bislang zur Bildregistrierung eingesetzte Verfahren der Kreuzkorrelation in der Testautomatisierung. Anhand von 30 Bilderpaaren, wovon 15 den Gutfall (Prüfung auf i.O.) und 15 den Schlechtfall (Prüfung auf n.i.O.) darstellen, wurden folgende 3 Varianten zur Merkmalsextraktion und -beschreibung miteinander verglichen:

- 1) MSER-Detektor/ SURF-Deskriptor/ SSIM-Index
- 2) HOG-Deskriptor/ SSIM-Index
- 3) Normierte Kreuzkorrelation (als Referenz)

Da für die Merkmalsmethode lediglich Grauwertbilder genutzt werden können, muss die Information der Farbe auf anderem Wege berücksichtigt werden. Die empirischen Analyseergebnisse haben gezeigt, dass das Untersuchen der einzelnen Farbkanäle mittels merkmalsbildender Verfahren schlechtere Ergebnisse erzielt hat, als die SSIM Methode [1]. Folglich wird der SSIM-Index als Methodik zusätzlich berücksichtigt.

Bei Variante 1) wird die Methode MSER (engl. Maximally stable extremal Regions) als Merkmalsdetek-

tor verwendet und der SURF-Deskriptor (engl. Speeded up robust Features) zur Beschreibung der detektierten Merkmale. MSER wurde zur Detektion ausgewählt, da dieser Detektor besonders bei Gebieten gleichbleibender Intensität effektiv wird [2]. Diese Regionen haben keine vorgeschriebene Kontur oder Form. Zudem ist diese Methodik invariant gegenüber affinen Transformationen, d.h. Translation, Rotation sowie Skalierung der Region beeinflussen somit nicht das Detektionsergebnis. MSER wurde zur Analyse ausgewählt, da die Eigenschaften dieser Methodik die Anforderungen an die visuelle Auswertung des HMIs erfüllen. Als effektive Methode zur Beschreibung der Umgebung der Merkmale wurde der SURF-Deskriptor gewählt [1]. Neben einem ähnlichen Ablauf im Vergleich zur SIFT-Methode, wurde bei SURF der Rechenaufwand optimiert, was hinsichtlich der Anforderungen an das Zeitverhalten sinnvoll ist.

Als Variante 2) wurde HOG (engl. Histogram of oriented Gradients) zur Extraktion von Merkmalen verwendet. Ursprünglich wurde HOG zur Detektion von Personen in Bildern entwickelt [3], jedoch zeigte sich bei der Analyse der Verfahren, dass es auch gute Ergebnisse in der Detektion und Beschreibung von Objekten liefert [1]. HOG beruht auf der Beschreibung von Objekten und Szenerien durch die Verteilung von Helligkeitsgradienten im Bild. Damit ist dieses Verfahren nicht auf Merkmale angewiesen und erfordert nicht zwingend einen separaten Detektor.

Das Ähnlichkeitsmaß wird bei den merkmalsbildenden Verfahren aus einer Summe der gewichteten Ergebnisse von SURF bzw. HOG und SSIM bestimmt. Beim Einfluss von SSIM wird das Maximum der drei gleich gewichteten Farbkanäle herangezogen.

Bei der Bewertung der Verfahren mit den 15 Gutprüfungen (Prüfung auf i.O.) und den 15 Schlechtprüfungen (Prüfung auf n.i.O.) zeigte sich besonders bei Variante 1) ein hoher Abstand zwischen den i.O.- und n.i.O.-Ergebnissen. Im produktiven Einsatz gewährleistet diese Variante, bei Vorgabe eines konstanten Sollwerts für das Ähnlichkeitsmaß, die höchste Sicherheit für ein richtiges Prüfungsergebnis.

Die Abstände bei Variante 2) und 3) fallen geringer aus. Bei Variante 3) fällt jedoch bei den Gutprüfungen ein stets sehr hohes Ähnlichkeitsmaß mit geringer Streuung auf. Folglich besteht auch hier eine hohe Sicherheit, um ein richtiges Prüfungsergebnis zu erhalten.

Betrachtet man die Rechenzeit, so ist sowohl bei jedem einzelnen Testpaar, als auch bei der Gesamtzeit aller Testpaare ein Zeitbedarf von Faktor 4 bei Variante 2) im Vergleich zur Referenz (Variante 3)) und ein Faktor von 5 bei Variante 1) aufgetreten. Dies signalisiert den erhöhten Rechenaufwand bei den merkmalsbildenden Verfahren. Die erhöhte Rechenzeit ist jedoch bislang nicht störend bei der Anwendung.

4 Klassifikation

Wegen der in den Anforderungen bewusst geforderten Aufgabenteilung wurden integrative Verfahren wie Convolutional Neural Networks (CNN) bei der Wahl der Methodik ausgeschlossen. Mit dem Erfolg der in Kapitel 3 beschriebenen Detektoren und Deskriptoren wurde für eine systematische Vorgehensweise die Methodik Bag of Visual Words verwendet, da sie als Basis auf den bewährten Detektoren und Deskriptoren anknüpft und die Handhabung der Merkmale für eine anschließende Klassifikation systematisch ergänzt.

Nachfolgend werden die Teilaufgaben des maschinellen Lernens aufgelistet und erläutert:

- Erstellen von Objektkategorien mit Test- und Validierungsdatensatz,
- Detektion und Deskription von Merkmalen,
- Erstellen eines Bag of Visual Words,
- Training und Validierung des Klassifikators (kNN bzw. multiclass linear SVM),
- Region of Interest bzw. Objektlokalisierung mittels Sliding Window,
- Produktiver Einsatz des Klassifikators

Zunächst wurden die Objektkategorien anhand des Testfokus definiert und hinreichende Test- und Validierungsdatensätze, die die relevanten technischen Varianten des HMIs abdecken, erstellt. In diesem Beitrag werden exemplarisch 11 Objektkategorien des Klima-HMI betrachtet. Hierbei handelt es sich um die Luftausströmerpfeile und einige Buttons für Klimafunktionen. In Abbildung 1 sind diese exemplarischen Objektkategorien enthalten. Wichtig beim Trainingsdatensatz ist, möglichst alle Variationen der Objekte für den Gutfall zu berücksichtigen. Ebenso sollten beim Validierungsdatensatz realistische Schlechtfälle der Objekte betrachtet werden. Der

Erfahrungsschatz vorhandener Objekte ist hierbei ausschlaggebend.

Details zur Detektion und Deskription der Merkmale können Kapitel 3 entnommen werden. Die gewählten Verfahren zur Extraktion der Merkmale dienen als Basis zur Erstellung des Bag of Visual Words. In den Objektkategorien des vorliegenden Klimabeispiels wurden einige Tausend Merkmale ermittelt. Diese vergleichsweise geringe Anzahl resultiert aus den kleinen Bildern der Objektkategorien, die lediglich Ausschnitte aus einem HMI-Bild darstellen und damit relativ wenig Inhalt haben.

Der Großteil der am stärksten ausgeprägten Merkmale über alle Objektkategorien hinweg bildet die Basis für das „visuelle Vokabular“ des Bag of Visual Words. Mittels K-Means Clustering wurden die Merkmale als „visuelles Vokabular“ quantifiziert. Mit 2000 Clustern ermöglicht der Bag of Visual Words eine hinreichende Unterscheidung der einzelnen Objektkategorien für das exemplarische Klima-HMI.

Mit dem erstellten Bag of Visual Words wird ein Klassifikator trainiert und anschließend validiert. Hierbei wurden mit einer linearen Support Vector Machine (SVM) als Multiklassen-Klassifizierer gute Resultate erzielt, die sich in der Wahrheitsmatrix für den Validierungsdatensatz wiederfinden (Tabelle 1). Der Mittelwert der Diagonalen beträgt dabei 97 %, womit im produktiven Einsatz eine erfolgreiche Klassifikation sichergestellt sein sollte.

	Kat. 1	Kat. 2	Kat. 3	Kat. 4	Kat. 5	Kat. 6	Kat. 7	Kat. 8	Kat. 9	Kat. 10	Kat. 11
Kat. 1	1										
Kat. 2		1									
Kat. 3			1								
Kat. 4				1							
Kat. 5					1						
Kat. 6						1					
Kat. 7							1				
Kat. 8	0,17							0,83			
Kat. 9									0,83	0,17	
Kat. 10										1	
Kat. 11											1

Tabelle 1. Wahrheitsmatrix exemplarisch für das Klima-HMI

Die Objektkategorien sind Ausschnitte eines gesamten HMI-Bilds. Zur erfolgreichen Klassifikation spielt die Region of Interest (RoI) noch eine wichtige Rolle. Jedoch ist für den anvisierten Anwendungsfall eine exakte Objektlokalisierung nicht von Interesse. Zur Berücksichtigung der Region of Interest werden 2 Ansätze parallel verfolgt. Einerseits wird das Gesamtbild in Bereiche mit Statusanzeigen und Buttons zerlegt, wo dann per Sliding Window-Verfahren die

einzelnen Objektkategorien ermittelt werden. Andererseits werden für die restlichen Objektkategorien bereits beim Erstellen der Datensätze (für den Bag of Visual Words) die relative Lage und relative Größe des Objekts im Gesamtbild vermerkt, so dass bei der produktiven Klassifizierung direkt dieser vermerkte Bildbereich ausgewertet wird. Mit diesem Vorgehen wird der Zeitaufwand für die Gesamtauswertung geringgehalten. Dies hilft insbesondere in zeitkritischen Anwendungen.

Abbildung 1 zeigt exemplarisch die Bereiche für die Region of Interest. Rahmen mit durchgezogenen Linien markieren die Buttonleisten (links, rechts und unten), wo mittels Sliding Window die Lokalisation der Buttons erfolgt. Weiterhin zeigen Rahmen mit gestrichelten Linien exemplarisch die Objekte der Objektkategorien an.

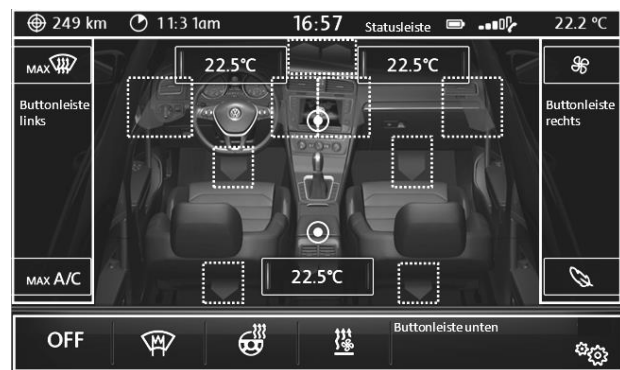


Abbildung 1. Exemplarisches Gesamtbild mit Region of Interest der Buttons (eingerahmt) und beispielhafte Objekte (gestrichelt eingerahmt)

5 Testfallfreie Testautomatisierung

Als Erweiterung für das automatisierte Testen anhand von Testfällen wird zunehmend auch ein testfallfreies Vorgehen mit Szenarien in der Testautomatisierung angewandt. Hierbei werden virtuelle Straßenfahrten durchgeführt, bei denen ganzheitliche Fahrermodelle das Gesamtfahrzeug in verschiedenen Szenarien bedienen. Mitlaufende Analysealgorithmen untersuchen die gegenwärtige Szene und die dabei auftretenden klassifizierten Objekte. Ergebnis der Testautomatisierung ist eine Auflistung der durchgeführten Handlungen des holistischen Fahrermodells. Dabei als auffällig bewertete Situationen werden von Fachexperten analysiert.

Ein Beispiel aus dem Bereich Klimatisierung wäre eine schnelle Autobahnfahrt im Hochsommer, wo bei Veränderung der Klimaeinstellungen das HMI holis-

tisch ausgewertet wird. Die Klimaeinstellungen erfolgen dabei durch das Fahrermodell, welches für die Szene parametrisiert wird. Zur Auswertung der Stimuli sind den Analysealgorithmen für die einzelnen Situationen passende und unpassende HMI-Objekte zugeordnet.

6 Literatur

- [1] A. Heller. *Optimierung der Bildauswertung (des HMI) in der Testautomatisierung*. Masterarbeit, Hochschule Ostfalia, Wolfenbüttel, 2017.
- [2] J. Matas, O. Chum, M. Urban und T. Pajdla. *Robust Wide Baseline Stereo from Maximally Stable Extremal Regions*. *Image and Vision Computing* 22, S. 761–767, 2004.
- [3] N. Dalal, B. Triggs. *Histograms of oriented gradients for human detection*. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR*, in San Diego, USA, S. 886-893, 2005.

MATLAB unterstützte Auslegung von Signalleitungen

Katrin Reblinsky¹

¹VWA-Hochschule für berufsbegleitendes Studium

Katrin.reblinsky@vwa-hochschule.de

Die leitergebundene Übertragung von elektrischen Signalen zwischen einzelnen Komponenten ist wesentlicher Bestandteil von allen modernen informationsverarbeitenden Systemen. Hohe Datenraten erfordern die Berücksichtigung von Wellenphänomenen und damit den Einsatz der Leitungstheorie. Eine bedeutende Kenngröße ist das Rückflußdämpfungsmaß $\alpha = -20 \text{ dB} \cdot \lg |\Gamma|$ des Reflexionsfaktors Γ . Sein Frequenzgang $\alpha(f)$ bestimmt maßgeblich das Übertragungsverhalten der Leitung. Sein Verlauf $\alpha(t)$ entlang eines Leiters läßt sich gut in Beziehung setzen zu elektrischen Materialeigenschaften und mechanischen Geometrien des Leiters. Übliches Vorgehen zur Charakterisierung und Optimierung eines Leitungssystems ist die mehrfache aufwändige Vermessung mit vektoriiellen Netzwerkanalysatoren und Zeitbereichsreflektometern sowie die Berechnung mit EM-Feldsimulatoren. Am Beispiel eines koaxialen HF-Steckverbinders wird gezeigt, wie sich mit MATLAB aus einer vorliegenden diskreten Zeitbereichskurve mit genügender Güte der Einfluß von konstruktiven Kenngrößen auf den Frequenzgang bestimmen läßt. Benötigt werden dazu nur Standardbibliotheken, Grundregeln der Leitungstheorie sowie ein qualitatives Modell von geometrischen Entwurfsparametern, die auch nicht spezialisierten Entwicklungsingenieuren zugänglich sind.

1 Einleitung

Moderne Informationstechnik basiert vornehmlich auf der Übertragung von Signalen in Form elektrischer Spannungspulse über Leitungen. Diese bestehen in der Regel aus einem metallischen Hin- und Rückleitern, die durch nicht leitendes Material isoliert sind. Typische Beispiele sind Streifenleitungen auf Leiterplatten oder koaxiale Verbindungsleitungen.

Die Datenübertragungsraten heutiger Systeme bedingen zeitlich kurze und schnell wechselnde Spannungspulse, die wiederum zu Effekten wie der Reflexion von Spannungspulsen an Diskontinuitäten in den Leitern oder der nicht resistiven Dämpfung im Isolationsmaterial führen. Diese Effekte lassen sich im Rahmen der Leitungstheorie der Elektrotechnik als Wellenphänomene verstehen [1].

Die Leitungstheorie ermöglicht, eine realen Übertragungsleitung stückweise durch idealisierte, homogene 2-Leiter-Systeme zu beschreiben, aus denen sich linearen partiellen Differentialgleichung für Strom und Spannung als Funktion von Zeit und Ort gewinnen

lassen.

Zur Charakterisierung von informationstechnischen Übertragungsleitungen hat sich die harmonische Analyse auf Basis von Sinussignalen bewährt. Eine wichtige Kenngröße ist der komplexe Leitungswellenwiderstand Z_L , der für einen homogene Leitungsbereich das Verhältnis von sich in gleicher Richtung ausbreitenden Strom- und Spannungswellen wiedergibt.

Durchläuft eine Strom- bzw. Spannungswelle einen Bereich mit sprunghafter Änderung der Leitungsimpedanz, so wird ein Teil der Welle unvermeidlich reflektiert und überlagert sich an dieser Stelle mit der einlaufenden Ursprungswelle. In informationstechnischen Kommunikationssystemen ist dies von großer praktischer Relevanz für die ungestörte Übermittlung von elektrischen Signalen und muss bei der Auslegung von Leitungsgeometrien und der Wahl von Isolationsmaterialien zwingend berücksichtigt werden.

In diesem Beitrag wird gezeigt, wie vorhandene Messdaten oder Simulationsergebnisse aus einer Zeitbere-

ichsanalyse eines koaxialen HF-Verbinders genutzt werden können, um unter Verwendung von Matlab mit einfachen qualitativen Modellen die Auswirkungen konstruktiver Änderungen auf die Hochfrequenzeigenschaften abschätzen zu können. Dies ermöglicht gerade weniger spezialisierten Entwicklungsingenieuren der Produktentwicklung, Erfahrungswerte für den Aufbau solcher Systeme zu entwickeln, ohne durch die darauf spezialisierten HF-Fachabteilungen technisch aufwändige Muster simulieren oder vermessen zu lassen.

2 Signalanalyse auf Übertragungsleitungen

Aus der Hochfrequenztechnik sind zwei etablierte Vorgehensweise bekannt, um die ungestörte Übermittlung von elektrischen Signalen, die sogenannte *Signal Integrity*, sicherzustellen [5].

Zum einen handelt es sich um die Zeitbereichsanalyse. Hier wird das betrachtete System mit einem zeitlich sehr kurzen Prüfpuls beaufschlagt, dessen Durchlauf durch die Leitergeometrie eine Aussage über seine elektrischen Ausbreitungseigenschaften ermöglicht. In der Praxis hat sich die Zeitbereichsimpulsreflektometrie durchgesetzt, besser bekannt unter der englischen Bezeichnung Time Domain Reflectometry (TDR) [6]. Hierbei wird am Eingang des Systems das Verhältnis von reflektiertem Spannungspuls zum stufenförmigen Prüfpuls in Abhängigkeit der verstrichenen Zeit ermittelt

$$\Gamma(t) = \frac{U_{\text{reflektiert}}(t)}{U_{\text{Prüf}}} \quad (1)$$

Dies ermöglicht die Bestimmung der Rückflussdämpfung und einer der Impedanz entlang des Leitersystems

$$\alpha(t) = -20 \text{ dB} \cdot \lg |\Gamma| \quad \text{mit} \quad |\Gamma(t)| = \frac{Z_L - Z_0}{Z_L + Z_0} \quad (2)$$

wobei die Zeitvariable t hier als Ausbreitungszeit entlang des Leitersystems zu verstehen ist.

Zu anderen handelt es sich um die Frequenzbereichsanalyse mit einem Netzwerkanalysator (NWA). Hierbei wird das Messobjekt als elektrisches N -Tor betrachtet. Der NWA beaufschlagt das Messobjekt an

einem Tor i mit einem sinusförmigen Eingangssignal und ermittelt an einem Tor j das Verhältnis von diesem Signal zum jeweiligen Ausgangssignal. Das Ergebnis sind die komplexen, frequenzabhängigen Streuparameter S_{ji} [9].

Von praktischer Bedeutung ist dabei der S-Parameter S_{11} , der sich als Frequenzgang des Reflexionsfaktors interpretieren läßt:

$$\alpha(f) = -20 \text{ dB} \cdot \lg |S_{11}(f)| \quad (3)$$

Der technisch gewünschte sehr niedriger Reflexionsfaktor erfordert ein hohes Rückflussdämpfungsmaß von mindestens 30dB in modernen Kommunikationssystemen wie WLAN oder Mobilfunk.

Grundsätzlich läßt sich aus dem Verlauf des Impedanzprofils entlang der Leitergeometrie der Streuparameter S_{11} am Eingang der Leitung ermitteln [4]. Dabei sind für komplexe Leitergeometrien allerdings technisch aufwändige digitale Signalverarbeitungen notwendig, um fehlerfreie Ergebnisse zu erhalten [2]. Für hochsymmetrische Strukturen wie koaxiale Filter wurde bereits gezeigt, dass sich das Vorgehen grundsätzlich deutlich vereinfachen läßt [3]. Dieser Ansatz wird hier aufgegriffen und weitergeführt.

Dabei wird die im TDR gemessene komplexe Reflexionsfaktor $\Gamma(t)$ als zeitliche Sprungantwort des Messobjekts angenommen. Seine Ableitung ergibt eine Impulsantwort $h_{\text{TDR}}(t)$. Wie in [3] gezeigt wurde, läßt sich diese bei geeigneter TDR Messung in guter Näherung durch eine diskrete Fourier-Transformation in die Übertragungsfunktion $S_{11} = H_{\text{TDR}}(f)$ überführen.

In dieser Arbeit wird die Annahme gemacht, dass alleine $|\Gamma(t)|$ als diskrete TDR-Messung eines beidseitigen, aber elektrisch kurzen Systems vorliegt, wie dies z.B. für einen koaxialen HF-Verbinders der Fall ist. Durch numerische Differenzenbildung mit dem `diff` Befehl von MATLAB wird die Impulsantwort berechnet. Danach erfolgt ein beidseitiges *zero-padding* und eine gewöhnliche diskrete Fourier-Transformation auf dem vorhandenen Abstraster der TDR-Messung. Abschliessend wird die Rückflussdämpfung gemäß Gleichung (3) berechnet und dargestellt.

3 Anwendung auf einen koaxialen HF Verbinder

Eine koaxiale HF-Verbindung besteht aus zwei konzentrischen Leitungen, die durch eine Isolierstoffstütze auf Abstand gehalten werden, siehe Abb. 1.

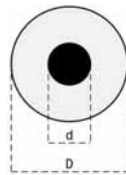


Abbildung 1: Querschnitt einer idealen koaxialen HF-Verbindung

Der hochsymmetrische Aufbau führt dazu, dass bei vorgegebenem Dielektrikum der charakteristische Wellenwiderstand nur noch von der Querschnittsgeometrie bestimmt ist und dort nur vom Verhältnis des Innendurchmessers des Außenleiters zum Außendurchmesser des Innenleiters abhängt:

$$Z_L \approx \frac{60\Omega}{\sqrt{\epsilon_r}} \ln\left(\frac{D}{d}\right) \quad (4)$$

Verbindungsleitungen für Hochfrequenzanwendungen sind aus diesem Grund vorzugsweise koaxial ausgeführt und in der Messtechnik mit einer Referenzimpedanz von $Z_L = 50\Omega$ fest etabliert.

Für praktische Zwecke werden allerdings Verbindungselemente wie HF-Steckverbinder benötigt. Diese Bauelemente enthalten typischerweise [7] Durchmessersprünge der Innen- und Außenleiter zur mechanischen Fixierung sowie zum Übergang von genormten Steckgesicht mit Luftdielektrikum, vergleiche Abb. 2.

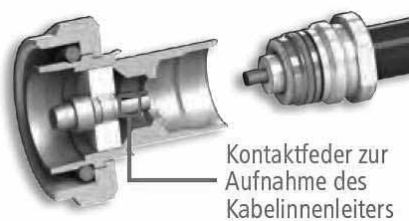


Abbildung 2: HF-Steckverbinder mit $d=7\text{mm}$ und $D=16\text{mm}$ im Steckgesicht (Serie 7-16) nach IEC 60169-4 (Quelle: Telegärtner Karl Gärtner GmbH).

Ohne Gegenmaßnahmen führt dies zu einer sprunghaften Änderung von Z_L und damit zu einer unerwünschten Erhöhung von Γ . Bereits in den Anfängen der koaxialen HF-Technik wurde empirisch ermittelt, dass sich dieser Impedanzsprung mildern lässt, wenn eine konstruktiv notwendige Änderung von d mit einer räumlich leicht versetzten Änderung von D kombiniert wird und umgekehrt. Konstruktive Gestaltungselemente sind z.B. Anfasungen wie in Abb. 2. Theoretisch lässt sich dies als Kompensation durch ein LC-Glied deuten [8].

Abbildung ?? zeigt ein mit dem TDR gemessenes Impedanzprofil eines HF-Steckverbinders ähnlich zu Abb. 2. Peaks oberhalb der Referenzimpedanz $Z_L = 50\Omega$ entsprechen einem induktivem Serienglied im Zwei-Leiter-Modell und Peaks unterhalb von Z_L entsprechen einem kapazitiv wirkendem Parallelglied [4]. Der Bereich zwischen den beiden roten Markierungen entspricht dem Steckgesicht des Steckverbinders.

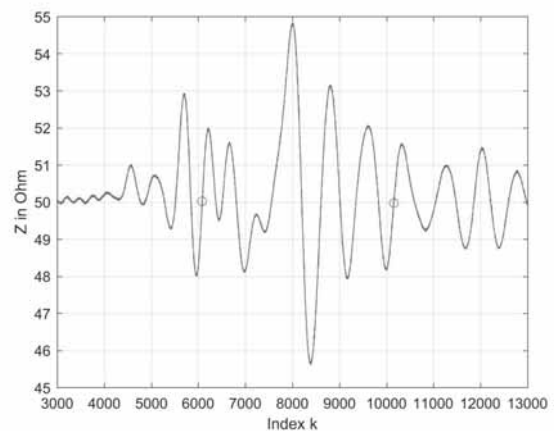


Abbildung 3: TDR-Messwerte eines HF-Steckverbinders

Wie im vorherigen Abschnitt beschrieben wurden für mehrere solche Messungen mit den Funktionen der Standardbibliothek von MATLAB die näherungsweise Rückflussdämpfungen berechnet. Es ergeben sich schon ohne Renormierung qualitativ gute Übereinstimmungen mit den parallel durchgeführten Messungen der Rückflussdämpfung mittels eines NWA's, siehe Abb. 4. Die errechneten Signale sind erstaunlich unempfindlich gegen Rauschen und Anzahl der Abtastpunkte. Untersuchungen mit erweiterter digitaler Signalvereinbarungsketten wie in [5] beschrieben zeigte nur unwesentlich besserer

Ergebnisse.

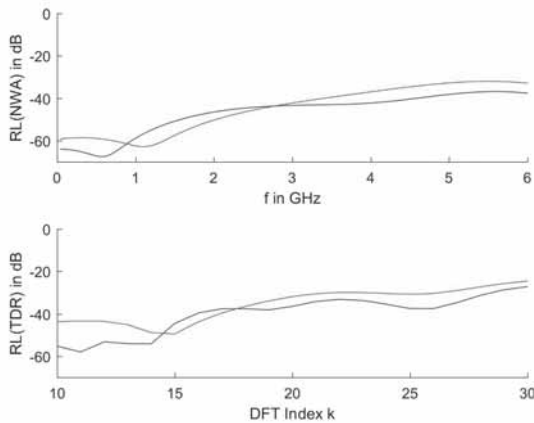


Abbildung 4: Rückflussdämpfung aus NWA-Messung im Vergleich zur Berechnung aus TDR-Messungen.

Bei der mechanischen Auslegung von coaxialen HF-Steckverbindern stellt der Übergang vom genormten Steckgesicht mit dem Dielektrikum Luft zum nicht normativ festgelegten Bereich mit festen Dielektrikum als Isolierstoffstütze die größte Herausforderung dar. An dieser Stelle zeigen sich im Impedanzprofil regelmäßig hohe induktiv-kapazitive Peaks. Solange deren Längsausdehnung klein im Vergleich zur Wellenlänge ist, kompensieren sich beide Bereiche näherungsweise. In Abb. 5 wurde nun mittels einfachster MATLAB-Array-Operationen der kapazitive Bereich um 1% (rote Kurve), 2% (grüne Kurve) erhöht und um 1% (schwarze Kurve) erniedrigt. Es zeigt sich, dass dies schon gravierende Auswirkungen auf den Frequenzgang der Rückflussdämpfung hat. Eine solche Änderung entspräche eine mechanischen Anfasung in der Größenordnung von 0,1mm.

Wird jedoch gleichzeitig die kapazitive und die induktive Fehlstelle um jeweils 1% reduziert, so ergibt diese Simulation auf Basis von Messdaten eine deutlich verbesserte Rückflussdämpfung genau so, wie es aus der Leitungstheorie zu erwarten ist.

4 Zusammenfassung

Es wurde untersucht, inwieweit sich ohne vertiefte Kenntnisse der digitalen Signalverarbeitung aus rou-

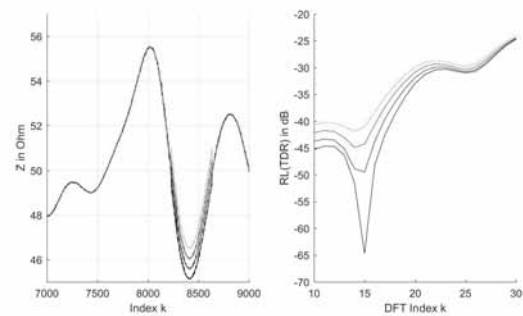


Abbildung 5: Simulierte kapazitive Änderungen am Impedanzprofil und ihre Auswirkungen auf die Rückflussdämpfung.

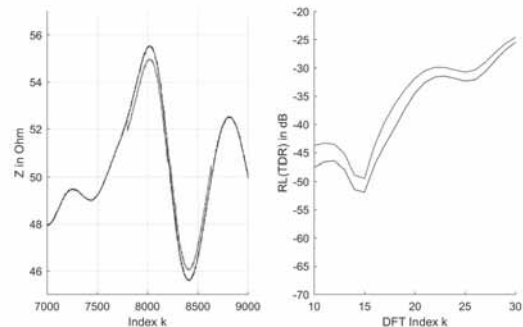


Abbildung 6: Simulierte induktive und kapazitive Änderungen am Impedanzprofil und die Auswirkungen auf die Rückflussdämpfung

tinemäßig aufbereiteten Messdaten der Zeitbereichsreflektometrie mittels MATLAB Vorhersagen über den Frequenzgang der Rückflussdämpfung machen läßt. Als Beispielsystem wurden dazu coaxiale HF-Steckverbinder herangezogen. Diese Systeme haben den Vorteil, dass sie sich nahezu ideal verhalten und die elektrischen Referenzebenen sich sehr nahe an die Messobjekte heranführen lassen.

Unter diesen Umständen können viele der üblicherweise notwendigen zusätzlichen Kalibrationsschritte entfallen und die Rückflussdämpfung läßt sich in der Tat fast Lehrbuchhaft berechnen. Dies hat schon einige praktische Relevanz, ermöglicht es doch Entwicklungsingenieuren, auch ohne längere Einarbeitung oder ausführlichere Absicherungsrechnungen schnell qualitativ vernünftige Berechnungen zu mechanischen Änderungen durchzuführen, z.B. zu Einflussanalysen.

Besonders bemerkenswert ist, dass im Gegensatz

zur Vorarbeit in [3] keine größeren Anstrengungen zur Glättung der Daten unternommen werden mußten oder einer Zwischenabtastung mit der Chirp-z-Transformation benötigt wurde.

References

- [1] E. Herter, W. Lörcher, *Nachrichtentechnik*. Hanser Verlag, Deutschland, 2004.
- [2] Pupalaikis et al., *TIME DOMAIN REFLECTION STEP TO S-PARAMETER CONVERSION* Patent No. US 8,706,433 B2, 22.4.2014.
- [3] I. Nobre, J. Nicolini, J. Garcia und M. Mosso, *Signal Processing Technique for Time Domain Measurement of S-Parameters*. J. Microw. Optoelectron. Electromagn. Appl., Seite 105-114, 2016, Band 15, Nr.2, 2016.
- [4] O. Ostwald, *Time Domain Measurements using Vector Network Analyzer ZVR*. Rohde & Schwarz, Application Note 1EZ44_0E, 1998.
- [5] D. Smolyansky, S. Corey, M. Resso, *Choosing the Right Signal Integrity Tools for Infini-Band Measurements*. DesignCon 2002, Paper HP7. Auch verfügbar als *Choosing Signal Integrity Measurement Tools: Time or Frequency Domain?*, TDA, Application Note, 2002.
- [6] D. Dahlmeyer, *Theorie der Zeitbereichsreflektometrie* elektronik industrie, Teil 1 auf Seite 60 in Heft 2 und Teil 2 auf Seite 50 in Heft 3, 2001.
- [7] *HUBER+SUHNER RF CONNECTOR GUIDE* HUBER+SUHNER AG, Document no. 648116, 4th edition, 2007.
- [8] J.R. Whinnery und H.W. Jamieson, *Equivalent Circuits for discontinuities in coaxial transmission lines*. Proc. Inst. Radio Engrs., Seite 98, 1944.
- [9] *S-Parameter Techniques* Hewlett Packard Company, Test and Measurement Application Note 95-1, 1997.

Dieser Tagungsband enthält die Beiträge des ASIM/GI Fachgruppen-Workshops „Simulation technischer Systeme“ (STS) sowie „Grundlagen und Methoden in Modellbildung und Simulation“ (GMMS), der am 8. und 9. März 2018 an der Hochschule Heilbronn stattfindet.

Der ASIM-Workshop ist ein Forum zur Diskussion methodischer Ansätze sowie praktischer Anwendungen auf den Gebieten der Simulation, der Modellbildung und des maschinellen Lernens. In guter Tradition fördert der Workshop in einer offenen Atmosphäre den Informations- und Erfahrungsaustausch zwischen Fachleuten aus Hochschulen, Forschungseinrichtungen und Industrieunternehmen. Die Beiträge sind sowohl anwendungsorientiert als auch grundlagenorientiert. Die Vorstellung von Ideenskizzen oder noch laufenden Projekten sind hier willkommen. Somit bietet der ASIM Workshop eine interessante Plattform für Wissenschaft und industriepraktische Anwendungen.