

# Modellieren mit Raumbezug: Spezifikation dynamischer Topologien mit den Mitteln von Graphersetzungssystemen

Jochen Wittmann<sup>1</sup>

<sup>1</sup> Hochschule für Technik und Wirtschaft Berlin, Studiengang Umweltinformatik,  
Wilhelminenhofstraße 75A, 12459 Berlin, Germany, [wittmann@htw-berlin.de](mailto:wittmann@htw-berlin.de)

**Abstract.** Viele Anwendungen im Bereich der Umweltsimulation beschränken sich nicht auf die Dynamik von eindimensionalen Bestandsgrößen, sondern versuchen zusätzlich, die räumliche Dimension der untersuchten Objekte mit ihren dynamischen Veränderungen zu beschreiben. In dieser Situation zeigt der Artikel, dass es einer allgemeinen Spezifikationsebene bedarf, die es erlaubt, anwendungsnah und problemspezifisch die Dynamik von Objekten mit Raumbezug abzubilden und andererseits die Möglichkeit gibt, diese Spezifikation algorithmisch sauber in einem Simulationsprogramm abzuarbeiten. Dabei sollen insbesondere die aus den topologischen Eigenschaften der Objekte abgeleiteten semantische Konsistenzbedingungen eingehalten werden. Zu diesem Zweck werden die Möglichkeiten der Dynamik für die GIS-Primitive Punkt, Linie und Polygon klassifiziert. Anschließend wird der Ansatz der Graphgrammatiken aus dem Bereich der Formalen Sprachen auf die Probleme der Dynamikspezifikation von Topologien von raum-zeitlichen Objekten übertragen und das algorithmische Optimierungspotential für die Implementierung dieses Ansatzes aufgezeigt.

## 1 Motivation

Die Anforderungen an Analyse, Modellierung und Simulation von dynamischen Prozessen haben sich in den letzten Jahren grundlegend verändert. Einerseits durch eine zunehmende Verbreitung von Smartphones mit automatischer Positionsermittlung über GPS-Satelliten auf Seite der Datenerfassung, die selbst für den Consumer-Bereich zum Standard geworden ist, und andererseits mit dem freien, komfortablen und schnellen Zugriff auf geographisches Karten- und Bildmaterial z.B. durch das Web-GIS Google-Maps (Google Maps, 2019) oder OpenStreetMap (OpenStreetMap, 2019). Die Ansätze

zur Modellierung können nun geographisch differenziert mit hoher räumlicher Auflösung erfolgen. Von Seiten der Informatik wird dieser Trend durch die Konzepte der objektorientierten Programmiersprachen bzw. der individuenbasierten Modellierungstechniken unterstützt, die die Behandlung einer Vielzahl auch räumlich differenzierter Objekte bzw. Individuen auf relativ einfache und anschauliche Weise möglich machen (siehe z.B. Ortman, 1999) oder eine Einführung in die objektorientierte Programmierung bei (Balzert, 1999). Während die Erfassung und Speicherung von Raum-Zeit-Daten durch entsprechende objektorientierte Datenbankkonzepte im Wesentlichen gelöst ist, gestaltet sich die Spezifikation von dynamischen Modellen mit Raum- und Zeitbezug schwierig. Eine detaillierte Analyse der entsprechenden Ansätze aus dem Bereich der Modellspezifikation einerseits und den Geoinformationssystemen andererseits findet sich in vorausgehenden Arbeiten des Autors, z.B. in (Wittmann, 2019).

In diesem Beitrag sollen nun zunächst die GIS-Primitive Punkt, Linie und Polygon auf ihre potenziellen dynamischen Eigenschaften hin untersucht und klassifiziert werden. Anschließend wird der Ansatz der Graphgrammatiken, der aus dem Bereich der formalen Sprachen stammt, auf das die Dynamikspezifikation der Geo-Objekte übertragen. Am Ende steht eine Abschätzung des Potenzials dieses Ansatzes in Bezug auf laufzeit-technische und modellierungstechnische Aspekte.

## 2 Dynamikbeschreibung auf der Basis der GIS-Primitive

Nach der vorausgehenden Analyse soll nun ein konstruktiver Vorschlag entwickelt werden, wie Dynamik

von Geoobjekten beschrieben und algorithmisch behandelt werden kann. Im Folgenden Abschnitt wird für die Primitive des GIS Punkt, Linie, Polygon und Topologie klassifiziert und spezifiziert, wie sich Bewegung ausdrücken kann.

## 2.1 Dynamik von Geo-Objekten

Für jedes der Primitive soll hier klassifiziert werden, welche Möglichkeiten der dynamischen Veränderungen auftreten und wie diese im Modell zu beschreiben sind. Dabei fließen sowohl die Klassifikation von Yattaw (Yattaw, 1999) als auch die Klassifikation von Modellbeschreibungsmethoden aus (Wittmann, 2019) ein und werden zu einer Dynamikbeschreibung für Geoobjekte zusammengeführt. Abbildung 1 zeigt die Klassifikation im Überblick.

### Typ Point

Im einfachsten Fall liegt ein Punkt-Feature vor. Je nach Modellziel kann sich ein Punktobjekt entweder durch eine kontinuierliche Bewegung im Raum bewegen oder aber plötzliche, sprunghafte Positionsveränderungen durchführen. Im ersten Fall kann diese kontinuierliche Bewegung durch einen Bewegungsvektor, also die Angabe von Richtung und Geschwindigkeit, dargestellt werden, mathematisch lässt sich die Beschreibung auf eine Differentialgleichung zurückführen.

Im Fall der sprunghaften Positionsänderung muss man auf die Konzepte der diskreten Simulation zurückgreifen und die Positionsänderung als diskretes Ereignis interpretieren, dessen Ausführung ohne Zeitverzug (also eben sprunghaft) erfolgt und das durch eine wie auch immer geartete logische Bedingung ausgelöst wird.

Neben diesen Bewegungsformen eines zum aktuellen Zeitpunkt im System bzw. im Modell existierenden Punkt-Objektes sind dynamische Veränderungen in der Topologie möglich, die in diesem Fall das Erzeugen bzw. das Löschen eines Punkt-Objektes abbilden.

Beispiele sind offensichtlich: für den kontinuierlichen Fall ein kontinuierlich fahrendes Fahrzeug oder ein Vogel, der kontinuierlich im 3-D-Raum fliegt. Eine nur zu diskreten Zeitpunkten gemessene bzw. beobachtete Position eines Tieres bei Tierwanderungen oder aber alle Bewegungen zwischen Haltestellen eines ÖPNV-Verkehrsmittels, bei dem der eigentliche Fahrweg nicht wesentlich für den Modellzweck ist sondern nur die Info

wann sich das Fahrzeug laut Fahrplan an welcher Haltestelle befindet. Das Erscheinen und Verschwinden vom Punktobjekten bedarf wohl keiner weiteren Beispiele, es kann sowohl mit dem kontinuierlichen Fall als auch mit dem diskreten Fall kombiniert werden.

### Typ Line

Für Linienobjekte ist die Dynamik weiter zu differenzieren: Es kann sich um eine Bewegung des Linienobjektes als Ganzes handeln oder aber das Linienobjekt ändert seine Form nur in Teilen, indem sich nur eine Teilmenge der definierenden Punkte bewegt, die übrigen Stützstellen ihre Position jedoch beibehalten. Beide Varianten sind sowohl für den kontinuierlichen als auch für den diskreten Fall möglich. Hinzu kommen die Veränderungen in der Topologie, die sich durch die elementaren Methoden „Stützstelle hinzufügen“ und „Stützstelle entfernen“ abbilden lassen.

Beispiel für den kontinuierlichen Fall könnte die dynamische Entwicklung einer Küstenlinie sein, der diskrete Fall kann beispielsweise eine Absperrung oder einen Zaun abbilden, der einmal jährlich neu an geographische Gegebenheiten angepasst wird, beispielsweise als Schutz für Spaziergänger an die kontinuierliche Veränderung einer Steilküste.

Wird die Liniendynamik durch die Dynamik einer Teilmenge der Stützstellen beschrieben, so ist für die Modellierung bedeutsam, dass durch die für die Einzelpunkte getrennt spezifizierte Dynamik die Semantik und Topologie der Linie als Ganze nicht verletzt wird (z.B. die Eigenschaft kreuzungsfrei). Entsprechend muss bei der Beschreibung oder zumindest bei der anschließenden Abarbeitung der Beschreibung in der Simulation darauf geachtet werden, dass semantische Verstöße zu einer Fehlermeldung und zum Abbruch der Simulation führen.

### Typ Polygon

Für Objekte des Typs Polygon gelten analog dieselben Bemerkungen wie für Linienobjekte. Polygone können sich mit allen ihren Stützstellen synchron kontinuierlich oder plötzlich bewegen. Zeigt nur eine Teilmenge der Stützstellen des Polygons Dynamik, so kann diese wiederum durch einen Bewegungsvektor (kontinuierlich) oder durch ereignisartige, sprunghafte Veränderungen der Position der Stützstellen spezifiziert werden. Auch hier kann es durch die Einzeldynamik von Stützstellen zur Verletzung der topologischen Eigenschaften

des Polygons kommen. Für solche Fälle sind entsprechende Vorkehrungen bei Modellbeschreibung und Simulation zu treffen.

### Bemerkungen zur vorausgehenden Klassifikation

Bemerkung 1: Grundsätzlich ist die Modellierung über die Angabe eines Bewegungsvektors oder durch die Angabe eines diskreten Ereignisses möglich. Der kontinuierlich zu interpretierende Bewegungsvektor reicht zur Dynamikspezifikation aus und wird für physikalisch-naturwissenschaftliche Modelle bevorzugt. Er ermöglicht die Simulation mit Zeitschritten, die gegen Null gehen, also eine sehr hohe zeitliche Auflösung. Diese bedingt jedoch auch einen erheblichen Rechenzeitbedarf für so beschriebene Modelle. Daher ist die Alternative der diskreten Modelle zu bedenken, die zwar immer nur sprunghafte Änderungen des Systemzustands zulassen, in der Regel jedoch wesentlich weniger Rechenzeit beanspruchen. Im Fall der diskreten Ereignisse ist die Angabe einer logischen Bedingung (wann das Ereignis stattfindet.) und eines in beliebiger formalen Sprache zu formulierenden „Effekts“ des Ereignisses (Was zum Ereigniszeitpunkt geschieht.) notwendig.

Bemerkung 2: In der tabellarischen Übersicht der unterschiedlichen Dynamik-Spezifikationsvarianten fällt auf, dass eine kontinuierliche Spezifikation von Änderungen in der Topologie nicht vorgesehen ist. Dies ist durch die mengentheoretische Definition der Topologie begründet: ein Element einer Menge (Ecke oder Kante) ist entweder vorhanden oder nicht. Ein „Wachsen einer Kante muss demnach durch die Bewegung der die Kante definierenden Stützstellen nachgebildet werden.

Bemerkung 3: Beim Erzeugen von neuen Objekten (Punkten oder Stützstellen) wurde in der Beschreibung allein der Dynamikaspekt diskutiert und die Auswirkung auf die Topologie beschrieben. Selbstverständlich müssen beim Erzeugen auch sämtliche Attribute der neu erzeugten Objekte sinnvoll parametrisiert werden.

Bemerkung 4: In diesem Abschnitt sollten allein und ausschließlich die Spezifikationsmöglichkeiten für die dynamischen Beziehungen für Geo-Objekte klassifiziert werden. Die Frage, wie sich Betrag und Richtung eines Bewegungsvektors berechnen lassen und wie weit und wohin eine Stützstelle bei einem Ereignis springt, hängt von diversen anderen Größen, oder ganz allgemein formuliert, vom aktuellen Systemzustand ab.

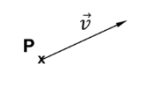

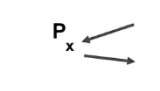
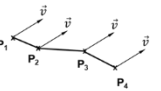
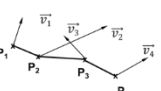
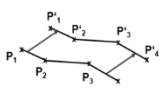
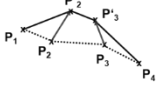
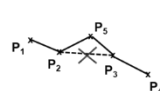
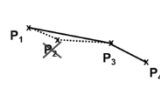
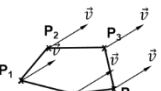
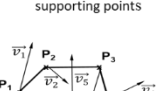
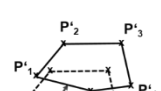
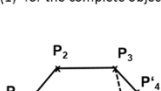
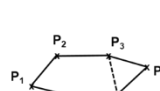
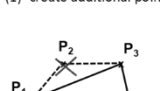
	continuous movement	sudden movement	topological change
point	 <p><math>\vec{v}</math> with velocity direction</p>	 <p>triggered by event condition</p>	 <p>create Point delete point</p>
line	 <p>(1) <math>\vec{v}</math> identical for all supporting points</p>  <p>(2) <math>\vec{v}_i</math> for supporting point i</p>	 <p>(1) for the complete line</p>  <p>(2) individually for a subset of supporting points</p>	 <p>(1) create additional point</p>  <p>(2) delete point of the line</p>
polygon	 <p>(1) <math>\vec{v}</math> identical for all supporting points</p>  <p>(2) <math>\vec{v}_i</math> for supporting point i</p>	 <p>(1) for the complete object</p>  <p>(2) individually for a subset of supporting points</p>	 <p>(1) create additional point</p>  <p>(2) delete point of the object</p>

Abbildung 1: Dynamikspezifikation für GIS-Primitive

Die entsprechenden topologischen Dynamikalternativen verhalten sich analog zu den zuvor besprochenen.

Beispiel für ein diskretes Bewegung eines Polygonobjektes ist das tägliche Umstellen eines Schafpferchs als Ganzes oder mit Anpassung der Pfähle an geographische Gegebenheiten. Die kontinuierliche Entwicklung eines Polygons bildet beispielsweise die Entwicklung eines Siedlungsgebietes. Bei diesem Beispiel werden die Stützstellen unterschiedliche Bewegungsmuster aufweisen. Die kontinuierliche Bewegung einer durch einen kontinuierlich fahrenden Traktor gezogenen Egge über ein Feld kann als Beispiel für eine kontinuierliche aber form-erhaltende Dynamik gelten.

### 3 Graphen Grammatiken als Ansatz zur Formalisierung von topologischen Veränderungen

Mit den Standardverfahren der kontinuierlichen und diskreten Simulation lassen sich eine Reihe der in Abbildung 1 klassifizierten Fälle behandeln. Allerdings ergeben sich Schwierigkeiten, wenn die Bewegung für Teile eines Objekts unterschiedlich erfolgt, weil durch diese Bewegung die Konsistenz der Topologie der Objekte gegebenenfalls verletzt wird (neue Schnittpunkte bei Linien entstehen, ein Polygon entwickelt sich zu zwei Teilpolygonen, ...). Daher erscheint es sinnvoll, die Spezifikation dynamischer Veränderungen nicht an die Beschreibung der Dynamik einzelner Bestimmungsstücke eines Objektes (z.B. einzelne Punkte als Stützstellen) zu binden, sondern die Dynamikspezifikation auf höherer Ebene für eine komplexere Situation anzugeben. Damit können, wie sich zeigen wird, Konsistenzprobleme an der Wurzel vermieden werden.

Zu diesem Zweck wird im Folgenden ein Ansatz von Schneider (Schneider, 2019) referiert, der aus dem Bereich der formalen Sprachen stammt und sich mit sogenannten Graphersetzungssystemen bzw. Graphgrammatiken auseinandersetzt. Die Definitionen und das Beispiel der folgenden zwei Abschnitte sind wörtlich mit nur kleinen Anpassungen aus dem genannten Skript von Schneider übernommen, das den Stand der Forschung kompetent und kompakt zusammenfasst. Die Übertragung auf die Situation der Geo-Objekte erfolgt dann im dritten Abschnitt dieses Kapitels.

#### 3.1 Definitionen

Die Definition der Graph-Grammatik ist motiviert durch die allgemein bekanntere Definition der Chomsky-Grammatik, die nicht auf Graphen, sondern auf Zeichenketten arbeitet:

Definition (Chomsky Grammar (Chomsky, 1959)):

A Chomsky grammar (phrase structure grammar) is a quadruple  $G = (T, N, P, S)$  where  $T$  and  $N$  are disjoint finite sets (alphabets),  $S$  is a distinguished element of  $N$ , and  $P$  is a finite subset of  $L^*NL^* \times L^*$  with  $L = T \cup N$ .

Die Elemente der Mengen  $T$  und  $N$  werden Terminalsymbole bzw Nicht-Terminalsymbole genannt.  $S$  ist das Startsymbol.  $P$  ist eine Menge von Produktionen bzw. Ersetzungsregeln. Statt der Tupelschreibweise  $(u, v)$  werden die Produktionen für gewöhnlich in der sogenannten Backus-Naur-Form als  $u ::= v$  angegeben.

Definition (Chomsky Language):

Jede Chomsky-Grammatik definiert eine Menge von Zeichenketten, die unter Verwendung der Produktionen und ausgehend vom Startsymbol abgeleitet werden können und ausschließlich aus Terminalsymbolen bestehen. Diese Menge wird Chomsky-Sprache zur Grammatik  $G$  genannt:

$$L(G) := \{w \mid w \in T^* \wedge S \xRightarrow{*G} w\}$$

In einem zweiten Schritt können wir Chomskys Ansatz zur Formalisierung des Begriffs einer Graphengrammatik verallgemeinern. Der Hauptpunkt ist, Produktionen anzuwenden, bis eine Art Normalform erreicht ist. Dazu hat Chomsky Terminalsymbole von nicht-terminalen unterschieden. Produktionen werden angewendet, bis die Zeichenkette keine nicht-terminalen Symbole mehr enthält. Wir können diese Idee leicht auf Graphen-Grammatiken übertragen: (hier wieder die wörtlichen Definitionen aus (Schneider, 2019):

Definition (Graph grammar):

A graph grammar is given by a quadruple  $\mathcal{G} = (L, T, P, S)$  with

$P$  being a finite set of graph productions in a category of labeled (hyper-)graphs using  $L$  as the labeling alphabet.

$T \subseteq L$  is called the terminal alphabet,

$S$  is the starting graph.

Definition (Graph language):

If  $\mathcal{G}$  is a graph grammar, then the set

$$\mathcal{L}(\mathcal{G}) := \{G \mid S \xRightarrow{*G} G \wedge \begin{aligned} l_{EG}[E_G] &\subseteq T_E \wedge \\ l_{VG}[V_G] &\subseteq T_V \end{aligned}\}$$

is called the language of  $\mathcal{G}$ .

Da diese Definitionen mathematisch anspruchsvoll sind, muss an dieser Stelle zur weiteren Erläuterung und Vertiefung auf die Originalquelle verwiesen werden. Die Bedeutung des Ansatzes soll im Folgenden zunächst durch ein etwas ausführlicheres Beispiel veranschaulicht werden, bevor die Übertragung auf die Situation der Geo-

Objekte erfolgt.

### 3.2 Beispiel Graphersetzung

Abbildung 2 zeigt die Situation am Beispiel der Relation "is\_mother". Die Knoten des gegebenen Graphen sind Personen von weiblichem (f), männlichen (m) oder beliebigem (x) Geschlecht. Die Knoten sind zur Identifikation mit hochgestellten ganzen Zahlen durchnummeriert. Zwischen den Knoten besteht die Relation "is\_mother", ebenfalls mit hochgestelltem Identifikator.

Als Produktion wird ein dreiteiliges Ersetzungsschema angegeben: Auf der linken Seite wird ein Ausschnitt eines Graphen als parametrisierte Ist-Situation beschrieben. In der Mitte steht der sogenannte Interface-Graph und auf der rechten Seite muss die Situation nach Anwendung der Produktion angegeben werden. Sinn der Produktion ist es, die Relation "sister\_of" zu ergänzen.

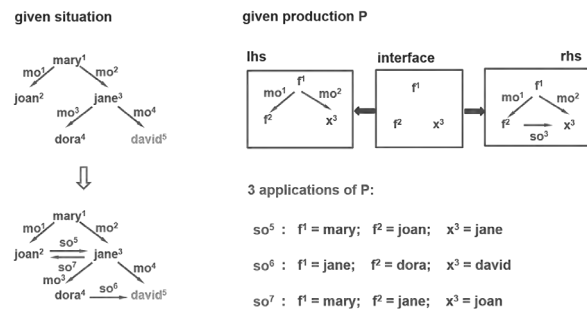


Abbildung 2: Beispiel für Graphersetzung (aus (Schneider, 2019) )

Das linke Objekt der Produktion muss nicht nur sicherstellen, dass beide Söhne die gleiche Mutter haben, sondern auch, dass die Person an der Quelle der neuen Kante weiblich ist. Wir stellen dies sicher, indem wir w für mary, joan, jane, dora und x für mary, joan, jane, dora, dora, david definieren. Dann haben wir drei Möglichkeiten, diese Produktion auf den gegebenen Graphen (so<sup>5</sup>, so<sup>6</sup> und so<sup>7</sup>) anzuwenden, wie in der Abbildung dargestellt.

Es ist zu beachten, dass durch die Spezifikation der Produktion sämtliche semantische Bedingungen automatisch eingehalten werden: passendes Geschlecht der beteiligten Personen, Beziehung zur gemeinsamen Mutter. Darüber hinaus wird bereits an dieser Stelle deutlich, dass das Finden geeigneter Passungen der linken Seite im aktuell gegebenen Graphen eine nicht-triviale Aufgabe darstellen kann.

### 3.3 Beispiele für eine Anwendung auf Geo-Objekte

Es kann aus Platzgründen das Potential dieses Ansatzes für die Beschreibung raum-zeitlicher Geo-Objekte nur angedeutet werden. Drei Beispiele sollen wenigstens illustrieren, wie komplexe Zusammenhänge durch geeignete Graphersetzungsgesetze in allgemeiner Form angegeben werden können. Topologisch lassen sich derartige Dynamiken als Wachstumsregeln für Graphen beschreiben. Abbildung 3 zeigt drei einfache Beispiele:

1. Stößt eine vorbestimmte Flugroute auf ein Hindernis, so soll dieses durch eine lokale Änderung der Route mit einer zusätzlichen Stützstelle umflogen werden.
2. Stellt sich in einem Netzwerk eine Punkt-zu-Punkt-Verbindung als stark nachgefragt, eine andere als schwach nachgefragt dar, so soll erstens eine Direktverbindung für die stark nachgefragte Strecke eingerichtet werden und zweitens die schwach nachgefragte Strecke gelöscht werden.
3. Im dritten Beispiel können die Objekte sowohl als Polygone als auch als Graphen interpretiert werden. In jedem Fall geht es darum, Wachstumsprozesse zu beschreiben. Eine Form, wie sie links des Pfeiles dargestellt ist, kann sich zu einer Form, wie sie rechts des Pfeiles dargestellt ist, verwandeln. Interpretiert als Polygone könnte auf diese Weise die Entwicklung eines Siedlungsgebietes modelliert werden. Als Graph könnte das Wachstum eines Versorgungsnetzes auf diese Weise beschrieben sein.

Die prinzipielle Vorgehensweise lässt sich dabei in 4 Schritten sehr einfach beschreiben:

1. Erstelle einen Satz von Produktionen, die die Dynamik der behandelten Objekte wiedergeben.
2. Finde in einem bestehenden Graphen eine Repräsentation der linken Seite einer Produktion und löse diese linke Seite als Teilgraphen heraus.
3. Führe in dem Teilgraphen, der durch die linke Seite gegeben ist, die Veränderung entsprechend der rechten Seite der Produktion aus.
4. Binde den veränderten Teilgraphen wieder in seinen Kontext des bestehenden Graphen ein.

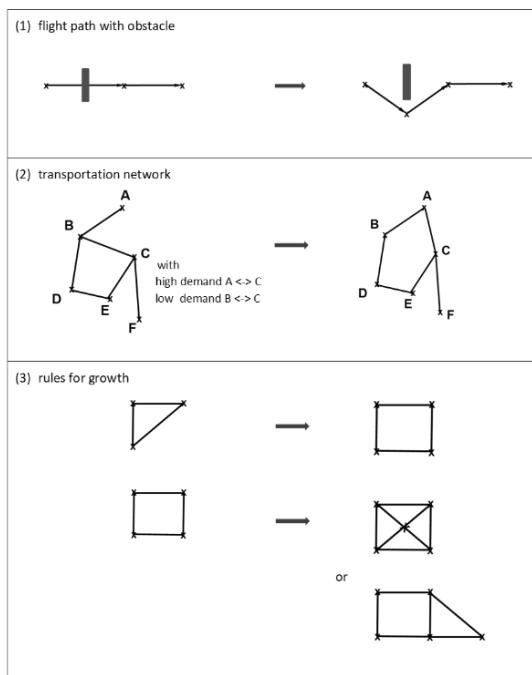


Abbildung 3: Beispiele zur Dynamik von Topologien

Allerdings stellen sich bei Detailbetrachtung einige Fragen, die die beschriebene Vorgehensweise verkomplizieren:

1. Wie weit muss die Übereinstimmung beim Vergleich der aktuellen Situation mit der linken Seite der Produktion gehen? Nur topologisch? Mit allen Attributen der beteiligten Knoten? ... Hier handelt es sich im Wesentlichen um ein Spezifikationsproblem, das durch den Formalismus der Graphgrammatiken gelöst werden kann. (Im Beispiel durch die Einbeziehung des Geschlechts der Knoten.)
2. Wie findet man alle Passungen im gegebenen Graphen? Dies ist ein Suchproblem, das näherer Betrachtung bedarf, durch die Formalisierung aber algorithmisch gut zu fassen ist.
3. Wie sind die Regeln (Produktionen) zu formulieren, um Konsistenz zu erhalten? Hierbei hilft ebenfalls der Formalismus, indem er Eigenschaften einer Produktionenmenge mathematisch ableitbar macht (analog zu den Eigenschaften der Chomsky-Grammatiken).

4. Wie ist das Zeitverhalten einer derartig spezifizierten Dynamik? Hier schlägt der Autor eine Interpretation gemäß dem Paradigma der diskreten Modelle vor, indem die Ausführung einer Produktion als Effekt eines diskreten Events behandelt wird.

## 4 Ansätze zur Optimierung der algorithmischen Behandlung der Dynamikspezifikation durch Graphgrammatiken

Vielmehr sollen hier zwei Ansatzpunkte zur Optimierung, die sich allein aus dem vorgestellten Simulationskonzept ergeben, als Themen für weitergehende Forschungsarbeiten andiskutiert werden.

Ein wesentlicher Unterschied zur Behandlung der Geo-Primitive im GIS liegt darin, dass sich ein Objekt nicht mit allen seinen bestimmenden Stützstellen gleich bewegen muss, sondern dass die Dynamik differenziert für jede individuelle Stützstelle spezifiziert werden kann. Das führt zu erheblichen Problemen bei Speicherung und Zugriff auf diese Objekte, weil ein effizienter Zugriff auch auf die einzelnen Stützstellen die Kapselung der höheren, zusammengesetzten Objekte (z.B. eines Polygons) verletzt. Hier sind Überlegungen notwendig, wie diese Zugriffe effizient ermöglicht werden können.

Neben diesen Aspekten der effizienten Speicherung soll hier aber besonders das Optimierungspotential betrachtet werden, das durch die Einführung der Graphersetzung entsteht. Im konzeptuellen Teil wurde zu diesem Schritt zwar die Spezifikation in Form der Ersetzungsregeln angegeben, die Teilschritte, die eine Implementierung umfassen muss und die im Wesentlichen für die Rechenzeiten verantwortlich sind, sind nicht aufgeführt worden. Es handelt sich dabei um die Schritte:

- Suche nach dem durch die linke Seite einer Ersetzungsregel gegebenen Mustern im aktuellen Modellzustand.
- Generieren der Menge mit erlaubten Ersetzungsvarianten.
- Variation der Position neuer Knoten im Raum.

Neben den offensichtlich benötigten Suchstrategien werden Methoden zum effizienten Aufspannen des Suchraums durch die systematische Generierung von Alternativen benötigt. Für beide Methoden müssen effektive und

effiziente Datenstrukturen gefunden werden. Für lokal begrenzte Suchen wären geeignete Hashing-Strategien in Betracht zu ziehen. Gefundene Alternativen müssen bewertet und einem Ranking unterzogen werden. Effiziente Zugriffe auf das Bewertungsmodell sind dazu notwendig. Die Bewertung ist durch entsprechende statistische Verfahren abzusichern (Konfidenzintervalle, ...). Dies wiederum legt es nahe, die Simulationsläufe für unterschiedliche Parametrisierungen durch Parallelisierung zu beschleunigen. Da ähnliche Probleme häufig auftreten werden (z.B. das Finden einer optimalen Position für einen neuen Knoten), ist es wahrscheinlich, dass selbstlernende Verfahren diesen Verfahrensschritt erheblich verkürzen können.

Bevor ein Satz von Ersetzungsregeln zum Einsatz kommt, erlauben die Erkenntnisse aus der Theorie der Graphgrammatiken syntaktische und semantische Tests auf Vollständigkeit und Widerspruchsfreiheit des Regelsatzes. Es ist denkbar und wünschenswert, diese Tests um Bedingungen zur Sicherstellung der korrekten Topologie und Topographie zu erweitern. Können derartige Aussagen bereits formal aus dem Regelsatz abgeleitet werden, werden topologische und topographische Fehler zur Laufzeit ausgeschlossen und damit die Menge der zu vergleichenden Alternativen von Anfang an reduziert.

## 5 Zusammenfassung

Ausgangspunkt für die Arbeit war die Beobachtung, dass im interdisziplinären Bereich der Modellierung und Simulation von räumlichen Objekten eine Spezifikationsebene fehlt, die einerseits beschreibend genug ist, um komplexe dynamische Veränderungen auch für die Nicht-Informatik-Experten aus den jeweiligen Anwendungsbereichen darzustellen und andererseits formal genug ist, um einer algorithmischen Behandlung im Sinne eines Simulationsalgorithmus zugänglich zu sein. Eine Analyse bestehender Modellierungsparadigmen und Softwaresysteme zeigt, dass die üblichen Modellierungstechniken wenig Unterstützung bei der Spezifikation der räumlichen Dynamik bieten. Obwohl die objektorientierten, einzelbasierten Ansätze auch zur Modellierung räumlicher Prozesse genutzt werden können, werden die Nutzer bei der Beobachtung räumlicher Konsistenzbedingungen auf die proprietären Lösungen zurückgreifen.

Basierend auf dieser Analyse wird der Formalismus der Graphensubstitutionssysteme auf die Anwendung im Bereich der räumlichen Modelle übertragen. Der Nutzen,

der dem nicht unerheblichen Aufwand durch Formalisierung entgegensteht, besteht vor allem in der sauberen algorithmischen Handhabung und Verarbeitung der Modelldynamik, die von Grafikproduktionen beschrieben wird. Konsistenzbedingungen als Folge des räumlichen Bezugs können auf der Metaebene der Produktionen berücksichtigt werden und vermeiden topologisch sinnlose Dynamik. Darüber hinaus erweist sich der Formalismus als vorteilhaft, wenn es darum geht, die algorithmische Verarbeitung der zahlreichen alternativen Möglichkeiten zur Entwicklung von Objekten im Raum beherrschbar zu machen und die Komplexität und Konsistenz der Lösungen zu optimieren.

Die Integration dieses Ansatzes in ein Simulationslaufzeitsystem wird in einem separaten Beitrag diskutiert. Die Idee ist es, die Ausführung eines von der Produktion vorgegebenen Ersatzes als diskretes Ereignis im Sinne einer diskreten Ereignissimulation zu behandeln. Darüber hinaus wird die Praktikabilität des Ansatzes und insbesondere die Eignung der vorgeschlagenen Dynamikbeschreibung für die Kommunikation mit Nicht-Informatikern in geeigneten Praxisprojekten aus dem Anwendungsbereich der Umweltinformatik getestet.

## 6 Literaturverzeichnis

- Brown, D., Riolo, R., Robinson, D., North, M., & Rand, W. (2005). Spatial Process and Data Models: Toward Integration of . *Journal of Geographical Systems* 7(1), S. 25-47.
- Dransch, D. (1997). *Computer-Animation in der Kartografie: Theorie und Praxis*. Heidelberg: Springer.
- ESRI. (25. 6 2018). *ArcGIS API for Python*. Von <https://developers.arcgis.com/python/> abgerufen
- Esri ArcMap. (25. 6 2018). *Esri ArcMap Online-Hilfe: Erstellen von Animationen in ArcGIS*. Von <https://desktop.arcgis.com/de/arcmap/10.3/map/animation/about-building-animations-in-arcgis.htm> abgerufen
- GAMA. (12. 3 2019). Von GAMA Platform: <https://gama-platform.github.io/> abgerufen
- Kimerling, A., Buckley, A., Muehrcke, P., & Muehrcke, J. (2016). Map Use, Chapter 8, Quantitative

- Thematic Maps. 206. Redlands California: ESRI Press.
- MARS. (04. 03 2019). *MARS-Group*. Von <https://mars-group.org/features/#gis> abgerufen
- O'Sullivan, D. P. (2013). *Spatial Simulation – Exploring Pattern and Process*. Chichester: Wiley.
- OGC. (25. 6 2018). *OGC® Open Geospatial APIs - White Paper OGC*. Von <http://docs.opengeospatial.org/wp/16-019r4/16-019r4.html> abgerufen
- Ortmann, J. (1999). *Ein allgemeiner individuenorientierter Ansatz zur Modellierung von Populationsdynamiken in Ökosystemen unter Einbeziehung der Mikro- und Makroebene*. Rostock: Dissertation am Fachbereich Informatik, Universität Rostock.
- Rechenberg, P., & Pomberger, G. (2006). *Informatik Handbuch, Stichwort Architektur von SW-Systemen*. München: Hanser.
- Schneider, H. (6. 3 2019). *Graph Transformations - An Introduction to the Categorical Approach, Vorlesungsunterlagen*. Von <https://www2.cs.fau.de/staff/schneider/gtbook/index.html> abgerufen
- Torrens, P., & Benenson, I. (2005). Geographic automata systems. *International Journal of Geographical Information Science* 19(4), S. 385-412.
- Uhrmacher, A. (2001). Dynamic Structures in Modeling and Simulation - A Reflective Approach. *ACM Transactions on Modeling and Simulation, Vol.11. No.2.*, S. 206-232.
- Wittmann. (2000). Simulationsmodell und Geographisches Informationssystem Kopplungsalternativen am praktischen Beispiel. In A. Cremers, & K. Greve, *Umweltinformatik 2000, 12.Internationales Symposium, Bonn, 2000* (S. 45-58). Marburg: Metropolis.
- Wittmann, J. (2017). *Simulation in Umwelt- und Geowissenschaften - Workshop Hamburg 2017*. Aachen: Shaker.
- Wittmann, J., & Thiel-Clemen, T. (2016). *Simulation in Umwelt- und Geowissenschaften - Workshop Hamburg 2016*. Aachen: Shaker.
- Yattaw, N. J. (1999). Conceptualizing Space and Time: A Classification of. *Cartography and Geographic Information Science*, 26:2, S. 85-98.
- Zeigler, B. (1990). *Object-Oriented Simulation with Hierarchical, Modular Models*. London: Academic Press.