

Kombinierte Simulation

logistikrelevanter Verkehrszusammenhänge

Richard Pump, Matthias Baumann, Jannis-Tore Bellok, Volker Ahlers, Arne Koschel

Hochschule Hannover
Fakultät IV, Abteilung Informatik
Ricklinger Stadtweg 120, 30459 Hannover
{Richard.Pump | Volker.Ahlers | Arne.Koschel}@hs-hannover.de
{Matthias.Dietrich | Jannis-Tore.Bellok}@stud.hs-hannover.de

Zusammenfassung:

Im Rahmen des BMBF-Forschungsprojekts USEfUL: Untersuchungs-, Simulations- und Evaluations-Tool für Urbane Logistik wird ein webbasiertes Tool zur Entscheidungsunterstützung entwickelt, welches es kommunalen Akteuren aus Politik und Verwaltung ermöglicht, die Auswirkungen verschiedener Ansätze zur Urbanen Logistik (speziell dem innerstädtischen Liefer- und Wirtschaftsverkehr) zu evaluieren. Für die Entscheidungsunterstützung notwendige Daten werden über Simulationen logistikrelevanter Verkehrszusammenhänge erhoben. Dazu werden die beiden Simulationstools AnyLogic und MATSim über eine spezielle Datenbankschnittstelle miteinander verbunden. MATSim stellt Daten einer Städtepopulation zur Verfügung, wohingegen Anylogic die Simulation von konkreten Agenten im kleineren Rahmen ermöglicht. Die Kombination des proprietären Simulationstools AnyLogic und des Open-Source-Tools MATSim bietet einige technische Herausforderungen. Insbesondere die unterschiedlichen Simulationsansätze der Tools verhindern die klassische Co-Simulation mit beiden Tools. Daher wird auf einen Austausch von Agentenplänen über die Datenbank zurückgegriffen, um Agentenverhalten aus einem Simulationstool in das andere zu transferieren.

1 Einleitung

Um den Herausforderungen des weiter stark zunehmenden innerstädtischen Lieferverkehrs zu begegnen wurden in den vergangenen Jahren verschiedene Lösungsansätze entwickelt und – in der Regel in räumlich und zeitlich begrenztem Rahmen – in Pilotversuchen erprobt. Die Bandbreite der Konzepte reicht von individuellen und gemeinsamen Paketboxen über stationäre oder mobile Mikrodepots, von denen die Zustellung auf der letzten Meile per Lastenfahrrad erfolgen kann, White-Label-Depots, die von mehreren Lieferdiensten zur effizienteren Verteilung gemeinsam genutzt werden, bis hin zur Zustellung mit Hilfe von Drohnen und Lieferrobotern (Gronald & Posset 2015, Van Audenhove et al. 2015, Sonnenberg et al. 2019).

Die Initiative Urbane Logistik Hannover hat sich als Konsortium aus Wissenschaft, Industrie, Dienstleistung, Politik und Verwaltung das Ziel gesetzt, den Großraum Hannover als Modellregion für Urbane Logistik zu etablieren (Urbane Logistik Hannover). Unter Leitung der Landeshauptstadt Hannover verfolgt das Forschungsprojekt USEfUL (Untersuchungs-, Simulations- und Evaluations-Tool für

Urbane Logistik) das Ziel, ein webbasiertes Tool zur Entscheidungsunterstützung für Maßnahmen zur Urbanen Logistik zu entwickeln. Die primäre Zielgruppe sind Entscheidungsträgerinnen und Entscheidungsträger auf kommunaler oder politischer Ebene. Interessierte Bürgerinnen und Bürger sowie Expertinnen und Experten aus den Bereichen Stadt- und Verkehrsplanung sollen sekundär bedient werden.

Als Grundlage der Entscheidungsunterstützung dienen Simulationsergebnisse von Logistikprozessen im Hinblick auf verschiedene Kenngrößen (Emissionen, Lärmbelastung, Verkehrsdichte, Bequemlichkeit für Verbraucher etc.) für ausgewählte Stadtteile Hannovers. Durch eine breite und typische Auswahl dieser Pilotquartiere (Innenstadt, innenstadtnahes Wohn- und Einkaufsviertel, Wohn- und Gewerbeviertel im Randbereich der Stadt) soll gewährleistet werden, dass sich die Simulationsergebnisse auf andere Stadtteile Hannovers und andere Städte übertragen lassen; das Tool zur Entscheidungsunterstützung wird eine Auswahl der städtischen Situation anhand zentraler Parameter ermöglichen (Einwohnerzahl und -dichte, Anteil Wohn-/Gewerbefläche etc.). Um eine realitätsnahe Modellierung bis auf die Ebene einzelner Wohngebäude und Gewerbebetriebe zu ermöglichen, werden Daten aus verschiedenen Quellen zusammengeführt (Feuerhake et al. 2018).

Grundlage des Tools zur Entscheidungsunterstützung sind Simulationen verschiedener Konzepte zur Verbesserung urbaner Logistikprozesse. Dabei kommen zwei verschiedene Ansätze zur Anwendung: Auf makroskopischer Ebene der gesamten Stadt werden mit Hilfe des Simulationstools MATSim Verkehrsflüsse simuliert, die sich anhand variabler Parameter ergeben. Ein Beispiel stellt die Untersuchung der Auswirkungen des Einsatzes autonomer Car-Sharing-Flotten auf den Verkehrsfluss dar (Bienzeisler 2017). Auf mikroskopischer Ebene einzelner Stadtviertel (Pilotquartiere) werden mit Hilfe des Simulationstools AnyLogic Logistikkonzepte bis hin zur Belieferung einzelner Haushalte simuliert. Hier sei beispielhaft die Untersuchung der Auswirkungen der Nutzung von E-Grocery auf den Individual- und Lieferverkehr genannt (Jordan et al. 2019). Beide Simulationstools basieren auf agentenbasierter Modellierung.

Einen ähnlichen Ansatz zur Entwicklung eines Tools zur Entscheidungsunterstützung für allgemeine Maßnahmen zur Verkehrssteuerung verfolgt das Projekt Urban Transport Roadmaps (De Stasio et al. 2016, Urban Transport Roadmaps). Als Grundlage der Simulation dient dort jedoch ein globales Modell. Das Projekt USEfUL grenzt sich davon durch den Fokus auf urbane Logistikprozesse und den Einsatz von Simulationen auf mikroskopischer und makroskopischer Ebene für eine reale Stadtsituation ab.

In diesem Paper wird beschrieben, wie die makroskopischen und mikroskopischen Simulationen in MATSim bzw. AnyLogic miteinander kombiniert werden. Nach einer kurzen Beschreibung der Simulationstools wird auf den Datenaustausch mit Hilfe einer eigens entwickelten Datenbankschnittstelle und auf Fragen der Datensicherheit eingegangen. Eine Zusammenfassung und ein Ausblick auf zukünftige Arbeiten schließen das Paper ab.

2 Problemstellung: Kombinierte Simulation mit MatSim und AnyLogic

Im folgenden Kapitel werden die beiden Simulationstools vorgestellt, welche im Projekt USEfUL zum Einsatz kommen. Dabei steht das technische Design im Vordergrund, welches die Kombination der beiden Tools maßgeblich beeinflusst.

2.1 MATSim

Das erste im Projekt eingesetzte Tool ist MATSim (Horni et al. 2016, MATSim). Das Framework „Multi-Agent Transport Simulation“ ist ein in Java implementiertes Open-Source Framework zur Simulation verkehrstechnischer Zusammenhänge. Grundlage von MATSim sind Agentenpläne, welche über einen evolutionären Optimierungsalgorithmus iterativ verbessert werden, bis jeder Agent einen Plan mit einem möglichst hohen Fitnesswert besitzt.

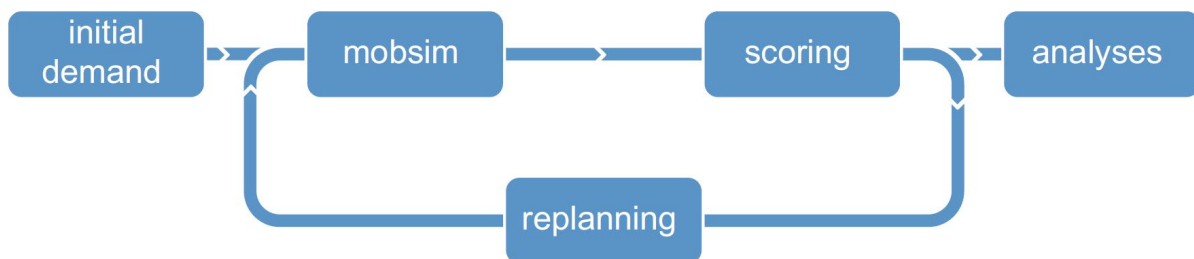


Abbildung 1. Der Arbeitsablauf von MATSim, auch MATSim-Zyklus genannt (Horni et al. 2016).

In Abbildung 1 ist der Ablauf aller MATSim-Simulationen abgebildet. Vor Beginn der Simulation wird der Initialbedarf erstellt. Dieser enthält sowohl eine Menge von Agenten mit Tagesplänen als auch das Netzwerk, in welchem sich die Agenten bewegen. Jeder Agent besitzt mehrere Tagespläne, bestehend aus einer Abfolge von Aktivitäten und Wegen zwischen den Aktivitäten innerhalb des Netzwerks.

Tabelle 1 zeigt ein Beispiel der Tagespläne des Simulationsmodelles. Der Agent beginnt seinen Tag zuhause mit einer Aktivität home, die bis 07:16 Uhr andauert, bevor dieser mittels des Autos zur Arbeitsaktivität (work) aufbricht. Gegen 15:38 Uhr fährt der Agent wieder mit dem Auto nach Hause. In den Tagesplänen können beliebige weitere Aktivitäten eingefügt werden.

Der Initialbedarf wird nach Erstellung in der sogenannten mobsim (mobility simulation) simuliert. Dabei folgt jeder Agent seinem Plan, unabhängig von Verspätungen oder dem durch andere Agenten entstehenden Verkehr. In jedem mobsim-Zyklus wird ein einzelner Tag simuliert. Weiterhin werden die durch Verkehr entstandenen Probleme wie erhöhtes Verkehrsaufkommen oder Verspätungen festgehalten.

Tabelle 1: Beispieltagesplan aus einer MATSim-Population.

```

<person id="BG1_42 Wegfolge_1">
  <plan selected="yes">
    <activity end_time="07:16:37" y="5806782.79" x="545427.02"
      link="11001" type="home"> </activity>
    <leg mode="car"> </leg>
    <activity end_time="15:38:26" y="5802126.38" x="550207.82"
      link="44113" type="work"> </activity>
    <leg mode="car"> </leg>
    <activity end_time="17:05:27" y="5806782.79" x="545427.02"
      link="11001" type="home"> </activity>
  </plan>
</person>

```

Nach Abschluss der mobsim erfolgt die Bewertung der Agentenpläne. Für jeden Plan wird der ursprünglich geplante Tagesablauf mit dem tatsächlichen Tagesablauf verglichen und anschließend der Tagesplan bewertet. Ein Plan, welcher z.B. zu vielen Verspätungen führt, wird dabei schlechter bewertet als einer, der den pünktlichen Aktivitätsstart erlaubt. Im letzten Schritt der Co-Evolution der Agentenpläne wird für jeden Agenten ein neuer Plan ausgewählt. Beim Großteil der Agenten wird der am besten bewertete Plan aus dem aktuellen Durchlauf als neuer Plan festgelegt, während ein kleiner Teil der Agenten den besten Plan modifiziert und auswählt. Der Prozess aus mobsim, Bewertung und Umplanung wird für eine feste Anzahl an Iterationen wiederholt, bevor die Ergebnisse in der Analysephase ausgewertet werden. Für das Projekt USEfUL wurde im ersten Schritt nach einer umfangreichen Datenerhebung ein möglichst genauer Initialbedarf generiert, um die aktuelle Verkehrslage in Hannover zu simulieren. Das Ergebnis der Simulation der aktuellen Verkehrslage bildet die Basis für alle weiteren Simulationen in AnyLogic und MATSim.

2.2 AnyLogic

AnyLogic ist eine auf der Programmiersprache Java basierende Software zum Erstellen und Ausführen von Simulationen (Borshchev 2013, Grigoryev 2012, AnyLogic). Die Software wird in einer freien Variante für den Lehrbereich zur Verfügung gestellt. Eine kostenpflichtige Variante bietet mehr Funktionalität. Die Software verfolgt drei verschiedene Ansätze zur Modellierung der Simulationen:

1. Systemdynamische Modellierung
2. Agentenbasierte Modellierung
3. Ereignisorientierte Modellierung

Die Methoden können in einer Simulation auch kombiniert werden.

2.2.1 Systemdynamische Modellierung

Mit der Methode der Systemdynamik sollen komplexe Systeme, z. B. eines Unternehmens, modelliert werden. Anhand der Modelle sollen Verhaltensweisen erklärt werden. Dabei wird untersucht, wie man das System an welchen Stellen beeinflussen kann (Borshchev 2013). Folgende Kriterien werden in der Systemdynamik berücksichtigt: „kausale Feedbackbeziehungen (feedback und feedback loops), Wirkungsverzögerungen, Bestandsgrößen (stocks), Flussgrößen (flows) und Nichtlinearitäten“ (Grösser).

2.2.2 Agentenbasierte Modellierung

Die Agentenbasierte Modellierung fokussiert sich auf das Verhalten einzelner Agenten. Dies können z.B. ein Lieferfahrzeug oder ein Haushalt sein. Die Agenten haben eine festgelegte Menge an Verhaltensweisen und können untereinander interagieren. Wenn ein Lieferfahrzeug zu einem Haushalt fährt und dort klingelt, wird das weitere Verhalten dadurch definiert, ob jemand im Haushalt anwesend ist oder nicht. Agentenbasierte Modelle werden zudem von Umgebungsvariablen beeinflusst (Borshchev 2013).

2.2.3 Ereignisorientierte Modellierung

Bei der ereignisorientierten Modellierung stehen verschiedene Zustände sowie die Auslöser für Zustandsübergänge im Fokus. So kann z.B. in einer Online-Shopping-Simulation eine Bestellung eines Kunden den Zustand eines Lieferfahrzeugs von „Parken“ in „Ausliefern“ ändern. Bei der Methodik kommt es nicht auf die physikalisch korrekte Simulation des Fahrzeugs an, sondern auf die verschiedenen Zustände und die möglichen Zustandsübergänge zwischen diesen, ähnlich einem endlichen Automaten (Borshchev 2013).

2.2.4 Verwendete Modellierungstechniken und Ablauf

In unseren AnyLogic-Simulationen werden sowohl agentenbasierte als auch ereignisorientierte Modellierungen eingesetzt. Eine Simulation soll einen Tagesablauf mit im Voraus gesetzten Parametern durchlaufen. Zum Beispiel kann eingestellt werden welcher Anteil der Population jeweils online Waren bestellt oder mit dem Auto einkaufen fährt. Die Daten der Agenten, z.B. die Anzahl der ausliefernden LKWs und deren gefahrene Strecke oder die Anzahl nicht zustellbarer Pakete, werden aufgezeichnet und am Ende der Simulation grafisch aufbereitet dargestellt. Außerdem können die Daten in Excel- oder CSV-Dateien exportiert werden. Die Haushalte und deren Positionen auf der Karte können in einer AnyLogic-eigenen Datenbank gespeichert werden. Der Simulationsablauf wird in einem Fenster grafisch dargestellt, indem die verschiedenen Agenten auf einer GIS Karte angezeigt und entsprechend ihren Verhaltensmustern animiert werden.

2.3 Co-Simulation von MATSim und AnyLogic

Zur Kombination der beiden Tools könnte auch Co-Simulation zum Einsatz kommen. Bei dieser könnten kleine Bereiche der Stadt in hoher Auflösung mittels AnyLogic simuliert werden, während die Simulation der restlichen Stadt in MATSim stattfindet.

Dieses Modell erfordert einen hohen technischen Aufwand, da wiederverwendbare Schnittstellen in AnyLogic und MATSim erforderlich sind, um beide Tools zu kombinieren. In USEFUL konnten diese Anforderungen aus organisatorischen Gründen nicht umgesetzt werden.

Weiterhin erfordert die Simulation eines kurzen Zeitraumes in AnyLogic viel Rechenzeit, wodurch ein synchron laufendes MATSim-Modell unverhältnismäßig stark verlangsamt werden würde.

Aus diesen Gründen werden die Tools verbunden, indem die Ergebnis-Daten der MATSim-Simulation zuerst abgespeichert und anschließend AnyLogic auf Abruf zur Verfügung gestellt werden.

3 Design

Im folgenden Abschnitt wird auf den Kommunikationsweg von MATSim zu AnyLogic eingegangen. Dabei werden Lösungen zu den Problemen auf konzeptioneller Ebene beschrieben.

3.1 Ausgabeformat MATSim

AnyLogic nutzt die Agentenpläne von MATSim, um das Verhalten der Agenten in den Simulationen anzupassen und entsprechende Szenarien durchspielen zu können. Diese Agentenpläne werden von MATSim als XML-Datei bereitgestellt, welche eine Größe von mehr als einem Gigabyte annehmen kann.

3.2 Eingabeformat AnyLogic

Zum Einlesen von Daten unterstützt AnyLogic Excel-Dateien sowie eine eingebaute Datenbank. Es gibt zudem weitere Möglichkeiten durch Java, die allerdings manuell implementiert werden müssen. Somit existiert die Möglichkeit die XML-Datei bei jedem Programmstart auszulesen. Allerdings wird dadurch der Programmstart deutlich verlängert, da die XML-Datei mehr als ein Gigabyte beanspruchen kann. Um einen dynamischen Zugriff auf relevante Daten und zugleich schnelle Startzeiten zu ermöglichen, kann eine externe Datenbank angebunden werden. Die Verbindung zu der extern bereitgestellten Datenbank kann beispielsweise mit Hilfe des JDBC-Treibers erfolgen.

3.3 Verbindungsmittel Datenbank

Für die Verbindung von MATSim und AnyLogic wird eine externe Datenbank verwendet. Dazu werden die Daten aus MATSim zunächst ausgelesen und in die Datenbank gespeichert. Dieser Vorgang muss nur durchgeführt werden, wenn sich die Agentenpläne verändert haben und ein neues Szenario besteht. Anschließend kann AnyLogic zu jedem Zeitpunkt die Pläne aus der Datenbank abrufen, ohne dass diese vollständig in die Simulation geladen werden müssen. Durch diesen Lösungsweg kann schnell zwischen einzelnen Szenarien bzw. Agentenplänen gewechselt werden. Zudem gestaltet sich die Entwicklung in den Teams für AnyLogic und MATSim einfacher, da diese im Projekt USEful an unterschiedlichen Orten arbeiten und so weniger auf genaue Absprachen angewiesen sind. Somit wird ein Webservice als Schnittstelle zwischen der PostgreSQL-Datenbank und AnyLogic verwendet, sodass eine losere Kopplung zwischen den Entwicklungsteams erreicht wird.

4 Lösungsansatz

Im Folgenden wird der Lösungsansatz noch einmal genauer vorgestellt. Hierbei wird auf technische Einzelheiten eingegangen und auch ein weiterer Blick auf die Lösung gerichtet.

4.1 Gesamtarchitektur

Die Architektur unserer Lösung ist stark von organisatorischen Abläufen geprägt. Das Simulationsteam des Projektes besteht aus Mitarbeitern dreier Hochschulen, welche räumlich getrennt voneinander arbeiten.

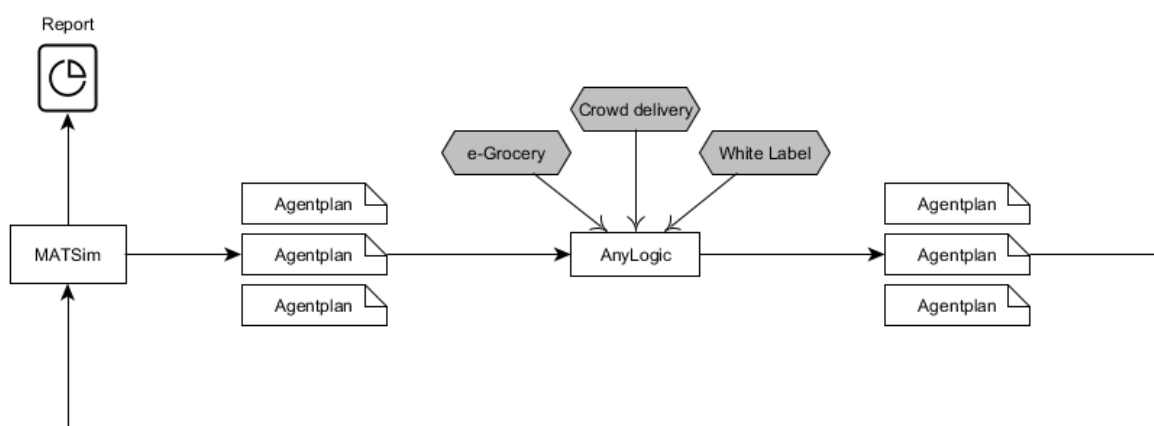


Abbildung 2. Ablauf der kombinierten Simulation.

Abbildung 2 zeigt den Ablauf der kombinierten Simulation. Zuerst werden mittels MATSim Agentenpläne erstellt, welche den realen Verkehr möglichst genau abbilden. Diese bilden anschließend die Grundlage für die Simulation des logistikrelevanten

Verhaltens in AnyLogic. So können verschiedene Logistikkonzepte auf derselben Basis evaluiert werden. Anschließend werden die Agentenpläne anhand des Logistikkonzepts verändert, zum Beispiel werden Shopping-Aktivitäten durch Online-Bestellungen ersetzt. Die veränderten und möglicherweise erweiterten Pläne werden zum Schluss noch einmal in MATSim simuliert, um entstehende Makroeffekte zu erfassen, bevor ein Bericht erstellt werden kann.

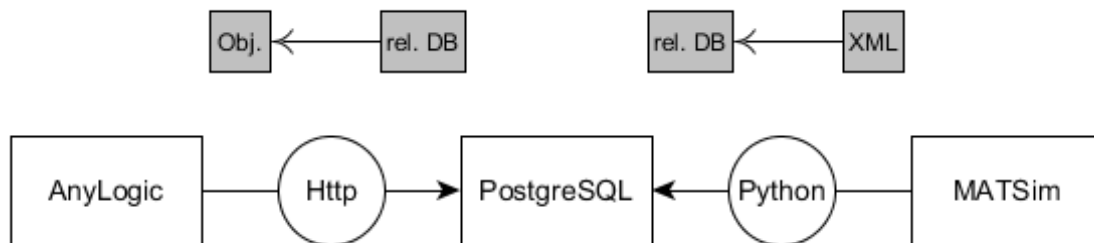


Abbildung 3. Übersicht über die Gesamtarchitektur.

Die Architektur ähnelt dem Arbeitsprozess und unterstützt primär die Kommunikation zwischen den beiden Tools. Abbildung 3 zeigt eine Übersicht über die Gesamtarchitektur. AnyLogic übernimmt die Simulation von mikroskopischen Zusammenhängen, wie z.B. dem Verhaltensmodell der Agenten. Aus Performanzgründen wird dabei die Simulation auf einen kleinen Teilbereich der Stadt begrenzt; dabei wird auch kein Normalverkehr abgebildet. Die umfassende Verkehrssimulation der gesamten Stadt wird in MATSim durchgeführt. Dazu werden mikroskopische Zusammenhänge ignoriert und auf die o.g. Agentenpläne zurückgegriffen. Die PostgreSQL-Datenbank verbindet die beiden Tools, indem Populationsinformationen über diese ausgetauscht werden. Die Datenbank enthält dabei immer den aktuellen gemeinsamen Stand, der die Basis für alle Evaluationen bildet.

Auf dem Weg von MATSim zu AnyLogic finden dabei zwei Umwandlungen des Datenformats statt. Zuerst werden die Agentenpläne durch das o.g. Python-Skript von einem XML-Format in das o.g. PostgreSQL-spezifische Objektformat transformiert, bevor die AnyLogic-Schnittstelle aus den PostgreSQL-Objekten Java-Objekte erstellt.

4.2 Datensicherheit

Die synthetische Grundpopulation basiert auf teilweise nichtöffentlichen Daten, welche dem Projekt durch Partner zur Verfügung gestellt wurden. Um eine unbeabsichtigte Veröffentlichung der Daten zu vermeiden erfolgt der Datenaustausch über verschlüsselte Verbindungen. Der Datenbankserver wird durch das Rechenzentrum der Leibniz-Universität vor Ort betrieben und ist durch eine sehr restriktive Firewall geschützt, welche nur den Zugriff aus den Hochschulnetzwerken erlaubt.

Zusätzlich wird die Datenbank durch eine Nutzerverwaltung geschützt, die den Zugriff auf die Schemata einschränkt. Dabei haben nur wenige Accounts überhaupt Zugriff auf die synthetische Grundpopulation. Die Grundpopulation kann nur durch einen speziellen Account geschrieben werden, der ausschließlich dafür eingesetzt wird. Für allgemeine Anfragen kommt ein weiterer Account zum Einsatz, der lediglich Lesezugriff auf die Grundpopulation besitzt, sodass diese nicht versehentlich geändert werden kann.

Durch diese Maßnahmen werden sowohl die Vertraulichkeit als auch die Integrität der Daten geschützt.

4.3 Offene Herausforderungen

Das bisherige Design erlaubt den automatischen Austausch der Population zwischen MATSim und AnyLogic, erschwert aber die manuelle Manipulation der Population. So ist das Ändern eines einzelnen Plans mit mehreren komplexen Datenbankabfragen oder einer Änderung der XML-Datei und einem neuen Datenbankimport verbunden.

In Zukunft soll ein Tool geschaffen werden, welches komplexe Anfragen an die Agentenpopulation sowie Änderungen an dieser ermöglicht. Das Tool soll spezielle Filter bereitstellen, welche es ermöglichen die Tagespläne der Agenten nach speziellen Eigenschaften auszuwählen. Zusätzlich sollen spezielle Szenariotabellen angelegt werden, welche Änderungen an der Population speichern, sodass nicht die Grundpopulation mehrfach gespeichert werden muss, sondern nur die Differenz zwischen Szenariopopulation und Grundpopulation.

Dieses Vorgehen unterstützt andere Akteure der Projektgruppe, die weniger informationstechnischen Hintergrund besitzen.

4.4 Konvertierungsskript

Das Konvertierungsskript dient dazu, die Ausgabedaten von MATSim in die Datenbank zu laden, sodass AnyLogic auf diese durch die Datenbank zugreifen kann. Das Datenbankschema basiert auf der DTD der XML-Datei, damit bei Veränderungen das Skript und die Datenbank schnell angepasst werden können (siehe Abbildung 4).

Das Python-Skript besteht aus zwei Modulen: Ein Modul liest die XML-Datei aus und speichert die Daten in Form von Objekten zur Verfügung (siehe Tabelle 2). Das zweite Modul stellt eine Verbindung zur Datenbank her und persistiert die Objekte des ersten Moduls.

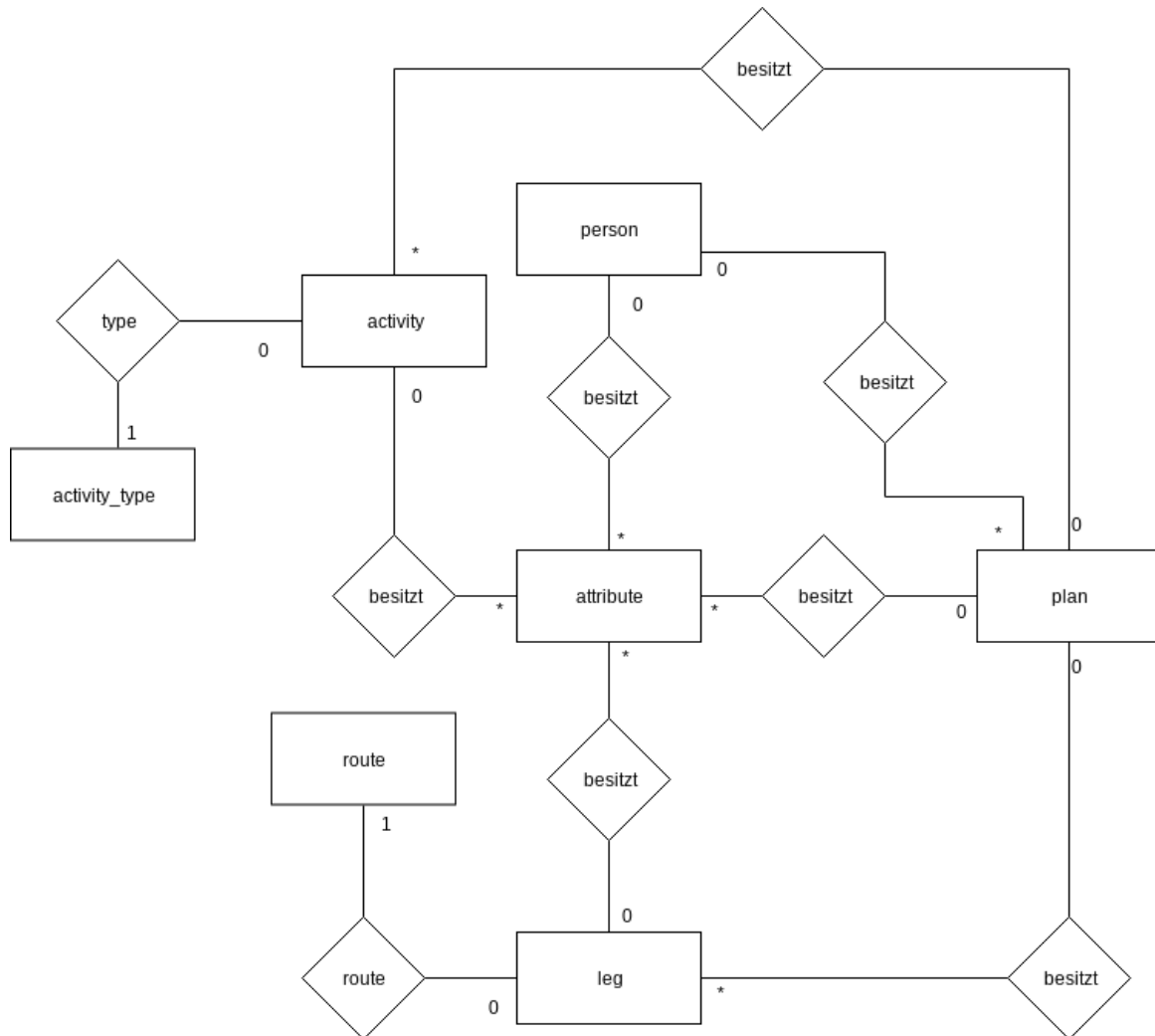


Abbildung 4. ER-Diagramm der Datenbank zur Verbindung der Simulationstools.

Tabelle 2. Ausschnitt aus dem zweiten Modul des Kovertierungsskripts zur Abbildung der Daten.

```
# Klassen schematisch nach der DTD aufgebaut
class Population:
    # values
    attribute = []
    person = []

    def __init__(self):
        self.desc = ""
        self.attribute = []
        self.person = []
```

5 Zusammenfassung

In diesem Paper wurde die Kombination der Simulationstools MATSim und AnyLogic für die Simulation urbaner Logistikprozesse beschrieben. MATSim dient dabei der Simulation von Verkehrsströmen auf makroskopischer Ebene der gesamten Stadt, während es AnyLogic ermöglicht, Logistikprozesse auf mikroskopischer Ebene ausgewählter Stadtteile bis hin zu einzelnen Straßen und Häusern zu simulieren. Ziel ist die Entwicklung eines Tools zur Entscheidungsunterstützung für Maßnahmen zur Steuerung urbaner Logistikprozesse.

Da eine klassische Co-Simulation aufgrund der verschiedenen Simulationsansätze und unterschiedlichen Laufzeiten nicht möglich ist, erfolgt die Verknüpfung der beiden Simulationstools durch den Austausch von Agentenplänen mit Hilfe einer eigens entwickelten Datenbankschnittstelle. Der Einsatz einer Datenbank (anstelle eines proprietären Dateiformats) erlaubt eine einheitliche und flexible Sicht auf die Daten, etwa für Analyse Zwecke. Bei Bedarf können die Daten zudem einfach angepasst werden.

Die nächsten Schritte bestehen in der Modellierung weiterer Logistikkonzepte sowie der Entwicklung einer grafischen Benutzerschnittstelle für die Simulation inklusive eines Editors für Szenarien. Parallel dazu wird in enger Abstimmung mit den späteren Benutzerinnen und Benutzern das Web-Tool zur Entscheidungsunterstützung entwickelt. Strukturell wird zudem der Einsatz von Microservices für die Kombination verschiedener Simulationsansätze als „Microsimulations“ erforscht (Pump et al. 2019).

Danksagung

Das Projekt USEful wird durch das Bundesministerium für Bildung und Forschung (BMBF) gefördert (FKZ 03SF0547). Wir danken unseren Kolleginnen und Kollegen der beteiligten Projektpartner, insbesondere den Simulationsteams der Abteilungen Maschinenbau und Wirtschaftsinformatik der Hochschule Hannover (AnyLogic) sowie des Instituts für Verkehr und Stadtbauwesen der Technischen Universität Braunschweig (MATSim), für die gute Zusammenarbeit.

Literaturangaben

- AnyLogic: Simulation Modeling Software Tools. <https://www.anylogic.com/>, letzter Zugriff 07.06.2019.
- Bienzeisler, L. (2017): Verkehrliche Auswirkungen einer autonomen Carsharingflotte. *ATZ Automobiltechnische Zeitschrift* **119** (2017) 66–71.
- Boshchev, A. (2013): *The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6*. AnyLogic North America, Chicago.
- De Stasio, C.; Fiorello, D.; Fermi, F.; Martino, A.; Hitchcock, G.; Kollamthodi, S. (2016): On-line tool for the assessment of sustainable urban transport policies. *Transportation Research Procedia* **14** (2016) 3189–3198.
- Feuerhake, U.; Wage, O.; Sester, M.; Tempelmeier, N.; Nejd, W.; Demidova, E. (2018): Identification of similarities and prediction of unknown features in an urban street network. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **XLII-4** (2018) 185–192.

- Grigoryev, I. (2012): AnyLogic 6 in Three Days. AnyLogic North America, Chicago.
- Gronald, M.; Posset, M. (2015): Best Practice Toolbox. Katalog von nationalen und internationalen Referenzprojekten für Güterverkehr und Logistik in Städten. Klima und Energiefonds, Wien. <https://www.smartcities.at/assets/01-Foerderungen/SUL/Best-Practice-Toolbox-v1.0.pdf>, letzter Zugriff 07.06.2019.
- Grösser, S.: System Dynamics. Gabler Wirtschaftslexikon. <https://wirtschaftslexikon.gabler.de/definition/system-dynamics-47445>, letzter Zugriff 07.06.2019.
- Horni, A.; Nagel, K.; Axhausen, K. (2016): Multi-Agent Transport Simulation MATSim. Ubiquity Press, London.
- Jordan, Y.; von Viebahn, C.; Haller, S. (2019): Konzeptionierung von Use Cases für die urbane Logistik in einer Schweizer Smart City mittels Design Thinking. In Tokarski, K.O.; Schellinger, J.; Berchtold, P. (Hg., 2019): Nachhaltige Unternehmensführung, 391–420. Springer Fachmedien, Wiesbaden.
- MATSim: Multi-Agent Transport Simulation. <https://www.matsim.org/>, letzter Zugriff 07.06.2019.
- Pump, R.; Koschel, A.; Ahlers, V. (2019): Applying microservice principles to simulation tools. In Hausotter, A (Hg.): SERVICE COMPUTATION 2019: The Eleventh International Conference on Advanced Service Computing, 6–9. IARIA.
- Sonneberg, M.; Leyerer, M.; Kleinschmidt, A.; Knigge, F.; Breitner, M.H. (2019): Autonomous unmanned ground vehicles for urban logistics: optimization of last mile delivery operations. In: Proceedings of the 52nd Hawaii International Conference on System Sciences (2019), 1538–1547.
- Urbane Logistik Hannover. <https://www.hannover.de/Urbane-Logistik-Hannover>, letzter Zugriff 07.06.2019.
- Urban Transport Roadmaps. <http://www.urban-transport-roadmaps.eu/>, letzter Zugriff 07.06.2019.
- Van Audenhove, F.-J.; De Jongh, S.; Durance, M. (2015): Urban Logistics. How to Unlock Value from Last Mile Delivery for Cities, Transporters and Retailers. Arthur D. Little. https://www.adlittle.com/sites/default/files/viewpoints/ADL_Urban_Logistics.pdf, letzter Zugriff 07.06.2019.