

# Implementation of Public Transportation in the MATSim Simulation Environment

**Author:** Katharina Mehlstäubler  
**Supervisors:** Carlos Llorca Garcia  
Nico Kühnel  
Gabriel Tilg  
**Chair:** Professorship for Modelling Spatial Mobility  
**Submission Date:** 23.07.2019

# Abstract

MATSim it is one of the leading large-scale and fully-integrated activity-based simulation frameworks. It also supports the explicit simulation of public transportation in a multi-modal setting. That is, it is capable of modelling the concurrent use of network links by all modes. This thesis explores different implementation methods of public transit in the MATSim simulation framework based on a scenario of the district of Wunsiedel in Germany. This encompasses establishing the transport network of the area, including the local public transit, modelling a synthetic population as well as analyzing the outcomes of the simulation. The first method simulates public transit with a fully-multimodal schedule and network. The second method simulates public transit using a separate network from the private car users. The third method of implementation does not explicitly simulate public transit and public transit users will be teleported. The networks and transit schedules were created using open-source data. The sensitivities of each public transit implementation are explored through simulating various population sample sizes, as well as scenarios with artificially induced congestion conditions. These are then presented and analysed based on model runtime, modal statistics, travel times and spatial correlations. Results present that the implementation methods can be assorted according to their capability to reflect model elements. The multimodal method offers the highest level of realism as it can consider spatial and temporal elements of public transit and also the effects of the traffic state of the entire network.

# **Acknowledgement**

I would like to express my gratitude to my supervisors Nico Kühnel and Carlos Llorca Garcia for their advice and support throughout this thesis.

As this thesis is the final step of my Master's program in Environmental Engineering at the Technical University of Munich, I would like to also thank the teachers, mentors and friends who have enriched my studies along the way.

Thank you to my parents for all the opportunities, encouragements and unconditional support they have provided me.

Finally, a special thanks to Basti and Phoebe for their love.

# List of Figures

Figure 1 Trips, Tours and Activities (Ortúzar and Willumsen 2011).....	9
Figure 2 MATSim Cycle (Horni, Nagel et al. 2016).....	11
Figure 3 Typical event sequence of a car trip in MATSim (Horni, Nagel et al. 2016)...	14
Figure 4 Traffic flow model in MATSim.....	15
Figure 5 Plan modifications by transit router (Rieser 2010).....	20
Figure 6 Example <code>transitSchedule.xml</code> file showing the hierarchical data structure of bus lines defined in MATSim.....	22
Figure 7 Entity-Relationship (ER) model of transit schedule data structure (Poletti 2016).....	23
Figure 8 PT2MATSim data sources and workflow (Poletti 2016) .....	25
Figure 9 Example <code>transitVehicles.xml</code> file (Rieser 2016) .....	26
Figure 10 District of Wunsiedel.....	28
Figure 11 Modal share in rural areas in the state of Bavaria, adapted from (Bundesministeriums für Verkehr Bau und Stadtentwicklung 2008). .....	29
Figure 12 Total traffic in the district of Wunsiedel by trip purpose (Jürgens and Fritsch 2017).....	30
Figure 13 Fully multimodal network of Wunsiedel .....	39
Figure 14 Wunsiedel network with public transit routes modelled on a separate network .....	40
Figure 15 Average iteration runtimes by population sample size.....	45
Figure 16 Average iteration runtimes by storage and flow capacity factors of the network. * Factors according to 6.4.1 .....	46
Figure 17 Average travel time by mode, for different population size samples.....	47
Figure 18 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof .....	48
Figure 19 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof .....	48

Figure 20 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof	49
Figure 21 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof	49
Figure 22 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof	50
Figure 23 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof	50
Figure 24 Multimodal network scenario	51
Figure 25 Separate network scenario	52
Figure 26 Teleportation scenario	53
Figure 27 Close ups of multimodal and separate network scenarios, respectively	54
Figure 28 Multimodal network scenario with normal traffic conditions	55
Figure 29 Separate network scenario with normal traffic conditions	55
Figure 30 Teleportation scenario with normal traffic conditions	55
Figure 31 Multimodal network scenario with flow capacity factor = 0.5, storage capacity factor=0.6	55
Figure 32 Separate network scenario with flow capacity factor = 0.5, storage capacity factor=0.6	55
Figure 33 Teleportation scenario with flow capacity factor = 0.5, storage capacity factor=0.6	55
Figure 34 Multimodal network scenario with flow capacity factor = 0.3, storage capacity factor=0.4	56
Figure 35 Separate network scenario with flow capacity factor = 0.3, storage capacity factor=0.4	56
Figure 36 Teleportation scenario with flow capacity factor = 0.3, storage capacity factor=0.4	56
Figure 37 Average travel times by storage and flow capacity factors of the network. * Factors according to 6.4.1	57

Figure 38 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof at normal traffic conditions.....	58
Figure 39 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof at normal traffic conditions.....	58
Figure 40 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.5, storage capacity = 0.6 .....	59
Figure 41 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.5, storage capacity = 0.6 .....	59
Figure 42 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.3, storage capacity = 0.4 .....	60
Figure 43 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.3, storage capacity = 0.4 .....	60
Figure 44 Score statistics for multimodal network scenario. ....	61
Figure 45 Score statistics for separate network scenario. ....	61
Figure 46 Score statistics for teleportation scenario. ....	61
Figure 47 Score statistics for multimodal network scenario. ....	61
Figure 48 Score statistics for separate network scenario. ....	61
Figure 49 Score statistics for teleportation scenario. ....	61
Figure 50 Score statistics for multimodal network scenario. ....	62
Figure 51 Score statistics for separate network scenario. ....	62
Figure 52 Score statistics for teleportation scenario. ....	62

# List of Tables

Table 1 Person events in MATSim.....	13
Table 2 Vehicle events in MATSim .....	13
Table 3 Activity types .....	31
Table 4 Number of Commuters leaving district Wunsiedel, aggregated by the municipalities of their place of residence .....	32
Table 5 Number of commuters entering district Wunsiedel, aggregated by the municipalities of their place of work.....	33
Table 6 Age distribution of the population in the district of Wunseidel (Bayerisches Landesamt für Statistik 2018).....	35
Table 7 Analysis Scenarios including population sample sizes, bus PCE, flow capacity factor and storage capacity factors.....	43

# List of Abbreviations

AGS	<i>Amtlicher Gemeindeschlüssel</i>
DTA	Dynamic Traffic Assignment
GTFS	<i>General Transit Feed Specification</i>
HAFAS	<i>HaCon-Fahrplan-Auskunfts-System</i>
MATSim	<i>Multi-Agent Transport Simulation</i>
OSM	<i>Open Street Map</i>
PCE	<i>Passenger Car Equivalent</i>
pt	<i>Public Transit or Public Transportation</i>
QSim	<i>Queue Simulation</i>
MobiDig	<i>Mobilität digital Hochfranken</i>
VGF	<i>Verkehrsgemeinschaft Fichtelgebirge</i>
XML	<i>Extensible Markup Language</i>

# Table of Contents

<b>Abstract .....</b>	<b>2</b>
<b>Acknowledgement .....</b>	<b>3</b>
<b>List Of Figures.....</b>	<b>4</b>
<b>List Of Tables .....</b>	<b>7</b>
<b>List Of Abbreviations.....</b>	<b>8</b>
<b>1. Introduction .....</b>	<b>1</b>
1.1. MOBIDIG .....	2
<b>2. Literature Research .....</b>	<b>3</b>
2.1.1. <i>Transit Simulation</i> .....	4
2.1.2. <i>Multimodal</i> .....	5
<b>3. Multi-Agent Transportation Simulation .....</b>	<b>8</b>
3.1. ACTIVITY BASED MODELLING .....	8
3.2. AGENT-BASED MODELLING AND SIMULATION .....	9
3.3. MATSIM .....	10
3.3.1. <i>Traffic Flow Model</i> .....	14
<b>4. Public Transit Simulation .....</b>	<b>16</b>
4.1. DEFINITIONS .....	16
4.2. DATA FORMATS.....	16
4.2.1. <i>Gtfs (General Transit Feed Specification)</i> .....	16
4.2.2. <i>Hafas (Hacon Fahrplan-Auskunfts-Systeme)</i> .....	17
4.2.3. <i>Osm (Openstreetmap)</i> .....	17
4.3. PUBLIC TRANSIT IN MATSIM .....	18
4.3.1. <i>Design Of Agent-Based Transit Simulation</i> .....	18
4.3.2. <i>Public Transit Data In Matsim</i> .....	21
<b>5. Methodology .....</b>	<b>27</b>
5.1. WUNSIEDEL IM FICHTELGEBIRGE.....	28
5.1.1. <i>Region</i> .....	28
5.1.2. <i>Transportation</i> .....	29
5.2. DATA PREPARATION .....	30
5.2.1. <i>Network</i> .....	30
5.2.2. <i>Population</i> .....	31
5.2.3. <i>Configuration</i> .....	36
5.2.4. <i>Public Transit</i> .....	36
<b>6. Analysis Scenarios.....</b>	<b>39</b>
6.1. MULTIMODAL.....	39
6.2. SEPARATE NETWORK.....	40
6.3. TELEPORTATION.....	41
6.4. ANALYSIS .....	41
6.4.1. <i>Scalability</i> .....	42
6.4.2. <i>Performance</i> .....	43
6.4.3. <i>Spatial Representation</i> .....	44
6.4.4. <i>Capacity</i> .....	44
<b>7. Results .....</b>	<b>45</b>
7.1. PERFORMANCE .....	45
7.2. SCALABILITY .....	47

7.2.1. <i>1% Population Sample</i> .....	48
7.2.2. <i>10% Population Sample</i> .....	49
7.2.3. <i>100% Population Sample</i> .....	50
7.3. SPATIAL REPRESENTATION .....	51
7.4. CAPACITY .....	55
7.4.1. <i>Mode Statistics</i> .....	55
7.4.2. <i>Travel Time</i> .....	57
7.4.3. <i>Route-Time Diagrams</i> .....	58
7.4.4. <i>Score Statistics</i> .....	61
<b>8. Discussion Of Results</b> .....	<b>63</b>
8.1. PERFORMANCE .....	63
8.2. SCALABILITY .....	63
8.3. SPATIAL REPRESENTATION .....	64
8.4. CAPACITY .....	64
8.4.1. <i>Flow Capacity Factor = 0.5, Storage Capacity Factor = 0.6</i> .....	64
8.4.2. <i>Flow Capacity Factor = 0.3, Storage Capacity Factor = 0.4</i> .....	65
<b>9. Conclusion</b> .....	<b>67</b>
<b>Bibliography</b> .....	<b>70</b>
<b>Statement Of Independent Work</b> .....	<b>75</b>
<b>Appendix A</b> .....	<b>76</b>

# 1. Introduction

The growth rate of the global population together with environmental concerns require increasingly creative and complex solutions for transportation systems. Public transit plays an essential role in sustainable transportation development as it not only provides basic mobility regardless of car ownership, age or income, but is also accredited to reducing congestion and urban sprawl as well as preserving open spaces, amongst others (Cervero, Murphy et al. 2004) .

Optimization of public transportation services together with the emergence of new mobility solutions such as on-demand concepts or Mobility-as-a-Service (MaaS) schemes require fine-grained representation of travel behaviour on an individual level as well as a better understanding of system-wide transportation dynamics.

Transportation models and simulations are able to support planning and decision-making processes through forecasting impacts of alternative scenarios (Castiglione, Bradley et al. 2015). Given the emerging challenges in transportation management, modelling tools must be able to keep up and represent multimodal interactions as well the dynamic demand responses of travel behaviour on the level of individual persons.

MATSim is an open-source, fully integrated agent-based simulation tool designed to model the travel behaviour of a synthetic population on a network during one simulated day. It also supports the explicit simulation of public transportation in a multimodal setting.

The thesis at hand analyses three different implementation methods for public transportation in the MATSim environment and presents their performance, based on different criteria such as travel times, capacity and model runtime, amongst others. The first method will simulate public transit with a fully-multimodal schedule and network. The second method simulates public transit using a separate network from the private car users. The third method of implementation will not be explicitly simulating public transit and public transit users will be teleported. The aim of this is to reveal the sensitivities offered by each method with respect to varying modelling aspect. This has been

realized through developing MATSim scenarios, including public transportation services, for the district Wunsiedel im Fichtelgebirge in Germany.

The underlying idea for this thesis developed during research on another project, Mobilität digital Hochfranken (MobiDig). MobiDig was carried out by the Faculty of Traffic Engineering and Control at the University of Munich. Details of this project are described in the following subchapter.

The thesis will commence by presenting a literature review followed by some background information on the MATSim software and its underlying theories, concepts and structures. In the following chapter, the varying implementations and extensions of public transit in MATSim are presented. Common transit schedule data formats are described. Chapter 5 describes the methodology employed for this thesis including the development of three scenarios representing public transit implementations in MATSim. This includes detailed descriptions on the data preparation. Following this, results are presented and discussed. The final chapter presents some concluding remarks.

## 1.1. MobiDig

‘Mobilität digital Hochfranken’, MobiDig for short, is a program funded by the Federal Ministry of Transport and Digital Infrastructure in Germany. Its aim is to develop and assess innovative transportation options in rural areas, as a result of applying digital data resources. With the general goal of making rural mobility more economical, environmentally friendly and overall attractive, a mobility model for the region of High Franconia, as a representative prototype, is being developed (Bundesministerium für Verkehr und digitale Infrastruktur 2017).

## 2. Literature Research

Travel demand models provide a systematic framework for evaluating transportation demand changes based on varying input assumptions. Given this, they are able to support decision making processes through forecasting impacts of alternative transportation and land use scenarios (Castiglione, Bradley et al. 2015).

For decades, the classic four-step modelling process has been the universal approach in transportation forecasting. This process encompasses a sequence of four sub-models: trip generation, trip distribution, modal split and traffic assignment (Ortúzar and Willumsen 2011). The use of trips as the unit of analysis, together with the sequential nature and underlying concept that spatial, demographic and temporal data is handled in a strongly aggregated way, amongst others, has led to the collective acknowledgement that this modelling approach is inherently unsuitable for varying modelling aspects. Specifically, the fundamental methods employed in the aggregated modelling process translate to a limited capability to capture the behavioural complexity of traveller responses (Rasouli and Timmermans 2014).

Current and emerging mobility developments addressing environmental concerns, demographic changes and advancements in technology, amongst many others, require a fine-grained and realistic representation of traffic. The prerequisite of realizing this is fundamentally rooted in the understanding that travel is a derived demand and arises from engagement in activities dispersed in space and time. Based on this, models which handle travel demand using an activity-based approach have been subject of ample research in recent decades. These models aim to anticipate the number, sequence and type of activities of all individuals, over a certain time period, and subject to a set of spatial, temporal and resource constraints. Rasouli and Timmermans (2014) present a comprehensive overview on the ample developments and theories in this field. The resulting temporally and spatially disaggregate description of travel demand has to be assigned to a network. For this, travel demand is typically described in time-dependent origin-destination (O-D) matrices. As this does not match well to the static and steady-state approaches of the four-step modelling approaches, time-dependent or dynamic traffic assignment has to be realized.

Dynamic Traffic Assignment (DTA) models deal with the route choice and traffic flow components of transport modelling. Likewise, DTA models have been subject to much attention in research and literature in recent times. Peeta and Ziliaskopoulos (2001) provide an overview on the multiple principles and explorations. In a further development, Balmer (2007) explains how due to the underlying conceptual and mathematical complexity of the DTA process, together with the inherent incompatibility of O-D matrices, simulation of the traffic assignment step, together with a fully agent-based approach, presents a more practical solution.

Simulation and modelling of traffic assignment, or traffic flow, is generally categorized by its level-of-detail into macroscopic, mesoscopic and microscopic. Macroscopic models employ aggregated flows to describe traffic. Microscopic models, on the other hand, describe individual traffic entities such as vehicles or drivers and their interactions. Mesoscopic models fall somewhere in between as they describe activities and interactions in rather low levels, based on small groups of traffic entities. Generally, there is a trade-off between resolution and computational efficiency as the more detailed microscopic models require immense amounts of processing times (Hoogendoorn and Bovy 2001).

Fully agent-based transport models derive travel demand from activity-based processes and apply microscopic, entirely dynamic, traffic simulations of every agent's individual demand based on constraints presented by the transport network and its attributes (Balmer, Axhausen et al. 2006). TRANSIMS (Smith, Beckman et al. 1995) was the first large-scale, integrated agent-based transport simulator. Since then, agent-based models have continuously been developed to incorporate varying degrees of mode, time, destination and activity-scheduling processes. Some of these models include SimAGENT (Bhat, Paleti et al. 2013) and MATSim (Horni, Nagel et al. 2016).

### **2.1.1. Transit Simulation**

Modelling route choice and traffic assignment for public transit is in many ways more complicated than private transportation. Basic modelling elements must be considered differently and a more complex set of factors must be integrated. Ortúzar and Willumsen (2011) define some of the issues in public-transit route choice and assignment as follows. To begin, the underlying network of public transit must be defined differently, it represents sections of bus or rail services which run between stops.

Moreover, link capacities must consider rather the capacity of each unit (transit vehicle) and its corresponding frequency. Also travel time must incorporate an in-vehicle component, waiting times, access and egress times as well as walking to and from transit service stops. As movements of passengers now become the focal point, transfer links between varying services, modes of public transit (bus, tram, subway, etc.) and also public and private transportation has to be considered. Moreover, monetary costs are perceived by public transit passengers differently and the vast and complex fare schemes such as flat fares, zonal fares, fares by distance, to name a few, play a significant role. Altogether, the generalised cost of travelling for public transit increases in complexity due to the plethora of influences perceived by public transit users.

Nuzzolo and Crisalli (2009) present the theories and developments of modelling transit based on the schedule-based modelling approach. This encompasses the representation of individual vehicle trips of scheduled services, in which all of its components such as demand, supply, path choice and assignment account for the explicit elements of a timetable. For most operational public transport models, typically route choice and assignment is realised through adapting classic equilibrium-based assignment models to work with transit. This is the case in most major commercial software packages such as EMME(INRO 2019), VISUM (PTV Group 2019), TransCAD (Caliper 2019) and OMNISTRANS (DAT.Mobility 2019), amongst others.

Microsimulation approaches of public transit employ dynamic processes in order represent behavioural responses based on information provision. MILATRAS (Wahba and Shalaby 2008) uses a learning model which develops a set of path choice probabilities over a number of iterations, together with an explicit vehicle simulation which simulates the movements of transit vehicles along routes based on their planned schedules. However, MILATRAS handles only transit traffic. MATSim (Rieser, Nagel et al. 2016) is based on a similar conceptual framework and is able to integrate both private traffic and public transit in an integrated way.

### **2.1.2. Multimodal**

Based on higher levels and frequencies of congestions experienced in many urban regions nowadays, Geroliminis, Zheng et al. (2014)'s work recommends that grasping the impacts of multiple modes of transportation sharing and competing for limited road infrastructure on a network level, and its effects on the overall performance of large-

scale urban systems, are essential. Based on this, their work develops an extended macroscopic fundamental diagram (3D-MFD) to include the influences of public transit vehicles and passengers in order to adjust and control the effects of traffic flows and modal share.

In the same interest, the Swiss Traffic Engineers and Traffic Experts Association (SVE 2017) conducted a research project, NetCap (Menendez, Ortigosa et al. 2016), at the Institute of Transportation Planning and Systems of the ETH Zürich. The aim of the project was to develop a unified methodology to evaluate capacities of multimodal networks based on the 3D-MFD, on the example of the inner city of Zurich. Within the framework of the NetCap project, a MATSim model of the city of Zürich was employed as part of the analysis and a shortcoming in available multimodal network data for modelling applications was addressed. Previous traffic simulations for the city of Zurich, modelled public transit and car traffic on separate road networks. Through this, buses would not be affected by car congestions and vice versa, cars would not be hindered by stopping buses, for example. In order to evaluate the effects of the concurrent use of road infrastructure by all modes, Bösch and Ciari (2015) developed a tool to generate fully-multimodal networks for the simulation platform MATSim, based on open-source data. In their work, a software package was developed to generate a fully-multimodal network and transit schedule for the MATSim simulation framework based on OpenStreetMap data and HAFAS transit data. Poletti (2016) developed a further package which offers a more comprehensive solution including also the conversion of GTFS, HAFAS and OSM data to generate fully-multimodal networks and mapped transit schedules for the use in the MATSim simulation framework.

MATSim, with its integrated multi-agent, activity-based modelling framework presents a promising solution for public transportation analysis requiring dynamic demand responses on an individual level. Several active fields of research to further optimize its application exist. MATSim possesses the capabilities of modelling the concurrent use of public transit and private vehicle traffic on the same network, yet these remained mostly unexploited in the past in practice and literature (Horni, Nagel et al. 2016). The work of Bösch and Ciari (2015) and Poletti (2016) have contributed to the issues of fully-multimodal network and schedule data limitations. Moreover, Ben-Dor, Dmitrieva et al. (2017) offer a calibration of MATSim to address scalability of public transit

implementations. Rieser, Métrailler et al. (2018) present related work on optimizing the public transit router employed in MATSim and further steps to heighten the realism and efficiency of public transportation.

### 3. Multi-Agent Transportation Simulation

MATSim is a transport simulation framework which is agent-based and uses activity-based modelling concepts for its underlying processes. The following provides a general insight on the fundamentals of activity- as well as agent-based modelling. Given this, the simulation structure of MATSim as well as its core elements are presented.

#### 3.1. Activity Based Modelling

Activity-based models adopt the principle that travel, and the demand for travel, is resultant from the need to pursue activities at different locations. That is, travel does not represent an end in itself but rather a means to link successive activities that are separated in time and space (Ortúzar and Willumsen 2011).

The following explanations provide insight on the core ideas of the activity-based modelling approach (Ortúzar and Willumsen 2011):

- An activity is an uninterrupted interaction with a service, person or the physical environment within the same socio-spatial environment.
- A stage is an uninterrupted movement using one mode of transport. It includes any waiting times directly before and during the movement.
- A trip is a continuous sequence of stages between two activities. A trip can have one or multiple stages.
- A tour is a succession of trips starting and ending at the same location. A trip chain is defined similarly, the difference being that the sequence of trips does not have to end at the same location.
- The trip purpose is distinguished by the most relevant activity at either ends of the trip.



**Figure 1 Trips, Tours and Activities (Ortúzar and Willumsen 2011)**

Moreover, in order to include the notion of constraints influencing activity participation, following concepts should be further be considered (Axhausen 2000):

- Scheduling encompasses the choice of time, duration, location and access mode for the chosen activity.
- Individuals are constraint in their scheduling by resources available to them.
- Individuals are constraint in their scheduling by their need to be available to others at particular times or locations.
- Individuals are constraint in their scheduling by their longer-term commitments to their household members, residential locations and their work places.

### 3.2. Agent-Based Modelling and Simulation

Agent-based modelling and simulation refers to the modelling of complex systems using a bottom-up approach, where the actions as well as interactions of autonomous individuals or agents, are observed in order to assess the behaviour of the system as a whole. Macal and North (2010) describe an agent as follows:

- An agent is a self-contained, modular and a uniquely identifiable individual. This is represented through attributes.
- An agent is autonomous and self-directed. It has behaviours which are applied to its decisions and actions, governed by information detected by the agent. The behaviour is often described through simple rules.

- An agent has a state, which consists of a set or subset of its attributes, containing essential parameters associated with its current situation.
- An agent is social. Agents have dynamic interactions with other agents, which influence their behaviour.
- An agent may be adaptive. It can have the ability to learn and adapt its behaviours based on its accumulated experiences.
- An agent may be goal oriented, which influences its behaviour.
- Agents may be heterogeneous. Agent simulation can reflect a full range of agent diversity across a population.

A simulation is understood as the dynamic and time-dependent modelling of processes. The simulation using agents reveals the emergent behaviour of the system altogether, based on the definition of the local behaviour policies.

Innovations in database technology and computation powers have facilitated the use of large-scale microsimulation models across many fields of science. Likewise, this agent-based simulation approach has proven itself useful for simulating traffic. As traditional traffic assignment methods have grown increasingly complex and mathematically often unfeasible, agent-based simulation of traffic are proving effective in many transportation planning matters (Rieser 2010). The simulation of agents pursuing a chain of activities throughout a day is precisely the underlying premise of MATSim.

### 3.3. MATSim

The MATSim software was created as a collaborative effort between the Swiss Federal Institute of Technology in Zurich (ETH) and the Technical University of Berlin (TU Berlin). It is open-sourced and implemented in Java. Through the combination of a detailed behavioural model and simplified traffic flow model, MATSim's advantage lies in its ability to simulate large-scale scenarios within practical processing efforts. This section offers a short overview of MATSim by describing the structural design and the main components. More detailed information is provided in the MATSim handbook *The Multi-Agent Transport Simulation MATSim* (Horni, Nagel et al. 2016).

All travellers in MATSim are modeled as individual agents. An agent is part of a synthetic population, which represents the real population of the city or region in question. Agents maneuver according to plans which describe an activity schedule of a single day. The agent's plans are fed to the simulation and yield transportation outcomes for the whole network.

The outcomes act as a form of feedback for the plans and route choices of the agents. That is, throughout the simulation, agents try to optimise their day plans by iteratively adapting their activity and travel arrangements, whilst at the same time interacting and competing with all other agents on the transportation infrastructure.

One scenario in MATSim consists of multiple iterations, each iteration consists of the traffic flow simulation and learning process as represented by the following cycle in Figure 2. The number of iterations can be individually configured, but should be repeated until the average population performance stabilizes

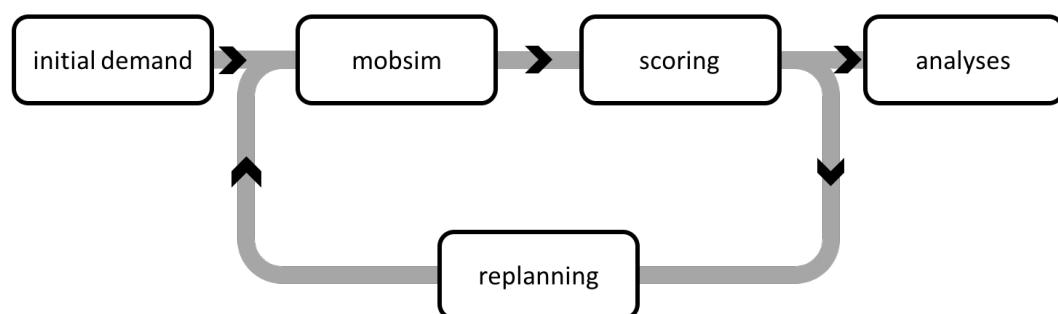


Figure 2 MATSim Cycle (Horni, Nagel et al. 2016)

To begin, an *initial demand* representing the mobility behaviour of the population is needed. It is described via all of the agents and their chain of activities, contained in so-called plans, for a full day. Plans are effectively lists comprised of *activities*, as well as the trips to reach those activities, called *legs*. Further core data required for the initial demand is the network, which is the physical infrastructure, as well as various configuration parameters.

*Mobsim* (see Figure 2) is the mobility simulation, that is, the simulation of the traffic flow in MATSim. The default traffic flow model of the mobsim, QSim, uses queue simulation. This will be described in further detail in section 3.3.1. In this step, all of the agent's day plans are executed simultaneously on the network, which generates the synthetic reality.

In order for the agents to evaluate the ‘performance’ of their plans, actual scores for each executed plan are generated in the *scoring* step. This is done using a custom utility function, the Charypar-Nagel-Function. In this function, utility and penalties are appointed and accumulated throughout the day of an agent. Positive utility is attributed to performing activities, whilst travelling produces negative utility. Moreover, penalties are also given in such cases of late arrivals, early departures or waiting times, amongst others.

The *replanning* step in MATSim encapsulates the learning and adaptation of the agents. This is achieved through the following three methods:

1. Choice set reduction and plans removal: should an agent exceed the maximum number of plans or have bad plans, these are removed.
2. Choice set extension, innovation: for some of the agents (typically 10%) a plan is selected, copied, modified and used for the following iterations. This supports the generation of good plans.
3. Choice set: all of the other agents choose and select between their plans.

Every action of the MATSim simulation generates a so-called event, which is recorded for analysis as an output component of each simulation. Events are marked with a timestamp, a type, and additional attributes describing the actions. This includes vehicle or link id’s, an activity type or other data. They are typically created by the mobsim, however MATSim extensions such as public transit or car sharing, amongst others, can create additional events. Figure 3 depicts a typical event sequence for a car trip of an agent. Events can be processed via a code infrastructure called *Event-Handlers*. Moreover, the events can be further processed or visualized in special software applications such as Via (Simunto 2018) for further analyses. The following two tables list the standard person and vehicle events in MATSim.

**Table 1 Person events in MATSim**

	<b>Event Type</b>	<b>Class Name</b>
Agent starts an activity	actstart	ActivityStartEvent
Agent ends an activity	actend	ActivityEndEvent
Agent starts a trip	departure	PersonDepartureEvent
Agent ends a trip	arrival	PersonArrivalEvent
Agent enters a vehicle	PersonEntersVehicle	PersonEntersVehicleEvent
Agent leaves a vehicle	PersonLeavesVehicle	PersonLeavesVehicleEvent
Agent gets stuck in simulation	stuckAndAbort	PersonStuckEvent

**Table 2 Vehicle events in MATSim**

	<b>Event Type</b>	<b>Class Name</b>
Vehicle enters traffic (at departure)	vehicle enters traffic	VehicleEntersTrafficEvent
Vehicle leaves traffic (at arrival)	Vehicle leaves traffic	VehicleLeavesTrafficEvent
Vehicle enters a link	entered a link	LinkEnterEvent
Vehicle leaves a link	left link	LinkLeaveEvent
Vehicle gets stuck in traffic	vehicle aborts	VehicleAbortsEvent
PT vehicle arrives at a stop	VehicleArrivesAtFacility	VehicleArrivesAtFacilityEvent
PT vehicle departs at a stop	VehicleDepartsAtFacility	VehicleDepartsAtFacilityEvent

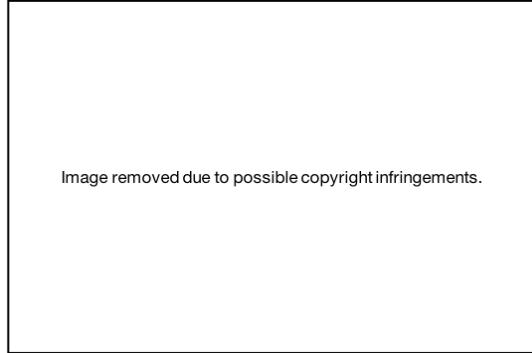


**Figure 3 Typical event sequence of a car trip in MATSim (Horni, Nagel et al. 2016).**

### 3.3.1. Traffic Flow Model

The underlying traffic flow model implemented in MATSim has been designed to be as straightforward as possible, in order for the model to remain modest with regards to computational efforts. QSim, short for queue simulation, represents the default traffic flow model in MATSim and realizes the physical simulation onto the network. As the name implies, this model is based on a spatial waiting queues concept. As seen in Figure 4, this approach essentially represents links as first-in-first-out (FIFO) queues. Two additional parameters, the storage capacity and the flow capacity, describe the functionality of this model. The storage capacity expresses the number of vehicles which fit on a network link. This factor is derived, implicitly, by taking the link length, divided by the vehicle length and then multiplied by the number of lanes. The flow capacity expresses the outflow capacity of a link, that is, the maximum number of vehicles that are able to leave a link in a given time unit. The parameter is explicitly defined as an individual attribute of the link. Additionally, the following criteria must also be met in order for a vehicle to advance to the next link:

- The vehicle must be at the head of the queue.
- The vehicle must remain on the link for a certain time, which corresponds to the free flow travel time.
- The flow capacity of the current link must allow the vehicle to move forward for this particular time step.
- The storage capacity of the following link must not be saturated.



**Figure 4 Traffic flow model in MATSim**

QSim does not specify an inflow capacity for a link. This has the consequence that congestions do not form at the beginning of merging links, but rather at the end of the low capacity link (Horni, Nagel et al. 2016).

## 4. Public Transit Simulation

When working with public transit data, various terms and expressions can be found describing the elements of public transit systems. A brief section providing basic definitions of the key elements is offered. Following this, common data formats for the exchange of transit schedule data are described. The underlying concepts and design of public transit in the MATSim framework are then presented in more detail.

### 4.1. Definitions

Public transit systems are comprised of transit lines. Transit lines offer regular connections between two given end points. They are typically given an identifier in the form a number, letter, colour, or combination of. Transit vehicles travel between the end points of a given line along transit routes. Routes can offer additional points for passengers of the transit lines to enter or exit the transit vehicles at set stop locations. Transit lines can have one or multiple transit routes, typically one transit route for each direction. Transit schedules describe how the transit vehicles operate on the route. That is, a schedule defines at what time the transit vehicles depart or arrive at certain stop locations via departure times (Rieser 2010).

### 4.2. Data Formats

#### 4.2.1. GTFS (General Transit Feed Specification)

The General Transit Feed Specification, GTFS, defines a common digital format for public transportation schedules and related geographic information. Originally developed as a side project from Google in 2005 (formerly named *Google Transit Feed Specification*), GTFS was established to let public transit agencies publish their transit data in a format which can be universally used by developers to create applications (Google 2016). GTFS feeds have since been published by several hundred public transit agencies worldwide (see (TransitLand 2019)).

A GTFS feed is composed of various comma separated files, saved as text files in a ZIP folder. Each one of the files contains a specific aspect of transit information such as stop locations, timetables, routes, as well as others. In its minimal form, including

only the required files, GTFS feeds are able to model one or more transit operator's schedule along with the location of transit stops. Additional files may optionally be included which may enable more detailed specifications of the transit services such as fares, transfer times or service exceptions.

#### **4.2.2. HAFAS (HaCon Fahrplan-Auskunfts-Systeme)**

HaCon Fahrplan-Auskunfts-Systems, or HAFAS (HaCon 2019), is the public transit data format used by the German railway agency, Deutsche Bahn (DB), along with many other transit agencies in central Europe (Österreichische Bundesbahnen (ÖBB) and Schweizerische Bundesbahnen (SBB)). As Rieser (2010) notes, public information on their exchange format, the HAFAS "Rohdatenformat" (raw data format), is limited. The data is stored in ASCII text files and contains information about operators, stop sequences and times, vehicles, schedules and other information. No naming conventions or hierachal structure of the schedule data exists.

#### **4.2.3. OSM (OpenStreetMap)**

OpenStreetMap (OSM) is a free and editable map database of the world. It has been built by a community of volunteers and is released under an open-content license (OpenStreetMap Wiki 2017). Zilske, Neumann et al. (2011) explain the significance of OSM as a data source for generating traffic simulation scenarios. Typically, gathering the necessary data for the creation of networks for the use in traffic simulations has proven to be cumbersome, due to the various agencies and sources involved. The data is many times delivered in non-standard formats and under proprietary terms, which entails additional processing efforts. OSM is able to offer a high level of information in one source.

OSM uses the elements nodes, ways and relations to represent its data, which are coded in XML (Extensible Markup Language) format. A node describes a point in space via longitude and latitude coordinates. Ways are polylines, connecting two or more nodes, which describe all line elements on a map such as roads and rivers but also for example boundary lines of buildings or forests. Relations are data structures used to group nodes and ways into logical entities such as buildings, forests, administrative boundaries or even bus routes. OSM additionally utilizes tags to assign particular meanings to elements. This can be applied to all of the available elements: nodes, ways and relations. Through this, OSM offers the possibility to define spatial features of

public transit systems, such as the exact paths the vehicles travel on or stop locations (OpenStreetMap Wiki 2017). Multiple methods to tag and define a public transit route, over various levels of details, exist. However, tagging practices of public transit and therefore the subsequent data, is very inconsistent. Many times parts of a route are missing or logical connections linking transit stop locations to the appropriate route or street are lacking. This has made the fully automatic generation of entire public transit schedules solely using OSM data unfeasible, for now (Poletti 2016).

### 4.3. Public Transit in MATSim

In MATSim's first release, cars were the only mode available for simulation. The integration of transit, due to its significance even when considering non-transit modes, had always been an objective and so MATSim has since been extended to include multi-modal options. In a first development, so-called 'teleportation' of modes other than car was implemented. Through this, agents are removed from one location and simply placed at their destination at a later point in time, according to an estimated travel time. This approach is still available and by default, any non- "main mode" in MATSim is handled this way. In a further development, and today's standard, the simulation of public transit is carried out in MATSim's mobility simulation, QSim. Public transit vehicles serve stops along a fixed route, according to a given schedule (Rieser 2016). The following section will present the underlying concepts of integrating public transit in an agent-based simulation environment, MATSim, based on the PhD Thesis of Marcel Rieser (2010) *Adding Transit to an Agent-Based Transportation Simulation*. Following this, the required data and files for simulating public transit in MATSim are described.

#### 4.3.1. Design of Agent-Based Transit Simulation

Above ground public transit vehicles such as buses and trams interact with private traffic on a network. This can influence traffic in a number of ways, as buses can be stuck in traffic, leading to delays in their schedule, or cause traffic for private cars if their stops are on the road. This necessitates the implementation of multimodal network capabilities in which these transit vehicles interact with private car traffic on the same network in the simulation environment.

Previously, the mobsim had only either handled the movement of agents who drive themselves, or agents which were teleported. Two new types of agents participating in

the traffic flow must now be considered: transit vehicle drivers and passengers. This requires the consideration of new interactions. Passengers must be able to wait for the transit vehicles at stop locations. Additionally, they must be able to amend travel legs with new legs which represent movements such as walking from one stop location to another stop locations of a connecting transit line or from the stop location to their activity location. Transit drivers must be identified by the mobsim as to permit them to handle stops. Also, this entails that these stop handling procedures (letting passengers enter and exit the transit vehicles) may cause the transit vehicle to block the street which would lead to the interference of the traffic flow on the links.

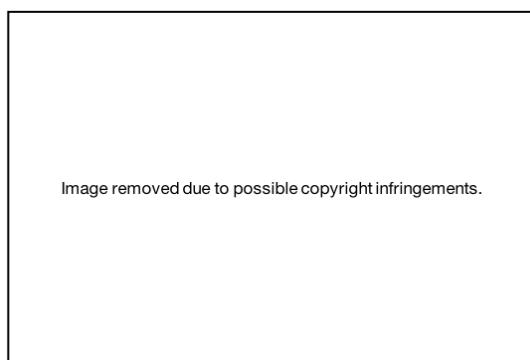
In its implementation, this has led to the incorporation of a data structure which keeps track of agents waiting at stops. With respect to the queue model employed in MATSim (see 3.3.1) the identification of transit drivers leads to the addition of delay values. These are conveyed by the transit drivers in the form of a time value and represent passengers alighting or boarding the transit vehicle. In this case, the number would be reflective of the number of passengers. Moreover, fixed times for opening and closing doors, waiting at the stop due to early arrivals, as well as other aspect may also be integrated in this delay value.

A delay value of zero leads to the transit vehicle being handled by the mobsim as usual. In the event of a delay being greater than zero, the transit vehicle remains on the link, for the amount of time of the delay, and two cases must further be considered. That is, whether the vehicle is blocking the link or not, which is determined by the physical geometry of the stop, and is stored as an attribute of the stop facility. In the case of the transit vehicle not blocking the link, the vehicle is removed from the link and inserted into a separate transit vehicle list. This allows the other traffic to proceed as usual. Once the time of the delay has been reached, the transit vehicle is placed back to the front of link's regular queue. In the event of a transit vehicle blocking the link, the vehicle is left on the link's regular queue for the amount of time of the delay. This has the effect that it is also blocking and subsequently delaying the other vehicles behind it.

A special transit router is required which calculates and assigns suitable routes, given the available public transit offerings, to the agents. This router is of relevance during the initial demand and replanning phases of the MATSim simulation cycle. This task of finding the best connection, given a fixed set of parameters is complex. The current

implemented transit router uses a modified version of the Dijkstra (1959) shortest path algorithm. It generates a graph, based on the transit schedule, which represents a logical topology of the transit route and interchange facilities. Shortcoming of this simplified approach have been discussed by Rieser, Métrailler et al. (2018). This includes bad performance rates as the graphs often become huge, requiring large amounts of memory and subsequently resulting in slow query times. This has the effect of making the replanning phase lengthy whenever public transit is used. Moreover, a lack of behavioural details through this router have been brought up. This is for example seen in the fact that the same minimal transfer times are applied to all transfers. Also, the varying preferences and price sensitivities of passengers cannot be reflected. Recent research on public transit routing has produced several faster and more suitable algorithms, which are presented in the work of Rieser, Métrailler et al. (2018).

As described in 3.3, plans are essentially lists of alternating activities and legs. The leg portion includes an attribute stating the transportation mode. In the past, it also included one route object which gave details on the route the agent had to take in order to get from one activity to the next. This approach is unfortunately not compatible when including multiple stages of a trip. In order to realize different stages, every stage is converted into a leg, which includes the route of the corresponding mode. Between these legs a new activity is inserted of the type *transit interaction*, with a duration of zero. The alternating set of legs and activities, representing the different stages, is now inserted in lieu of the one leg. The following diagram in Figure 5 presents a visualization of this concept.



**Figure 5 Plan modifications by transit router (Rieser 2010).**

Mode choice is implemented in MATSim via a mode choice replanning module, *ChangeMode*, which generates mode innovation. The module chooses a random transport mode from a configurable list of modes, excluding the present mode. This newly chosen mode is then applied to the legs of the plan. This facilitates agents trying out different modes without having to store a plan for each available mode. Through this module, only one mode is applied to all of the legs of the plan. A further module *ChangeSingleTripMode*, enables multiple modes in one plan (Rieser, Nagel et al. 2016).

The central data structure for the simulation of public transit in MATSim is provided by the transit schedule. Details on the elements and structure, along with further files and data preparations necessary for extending a MATSim simulation scenario to include the explicit simulation of public transit is presented in detail in the following subchapter.

#### 4.3.2. Public Transit Data in MATSim

Public transit in the MATSim simulation framework is based on two input files, `transitSchedule.xml` and `transitVehicles.xml`. The following shows an example of a simplified, but complete `transitSchedule.xml` file.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE transitSchedule SYSTEM
"http://www.matsim.org/files/dtd/transitSchedule_v2.dtd">

<transitSchedule>
    <transitStops>
        <stopFacility id="1" x="990.0" y="0.0" linkRefId="1"
                      name="A-City" isBlocking="true"/>
        <stopFacility id="2" x="1000.0" y="10.0" linkRefId="2"
                      name="B-Town" isBlocking="true"/>
        <stopFacility id="3" x="0.0" y="100.0" linkRefId="3"
                      name="C-Village" isBlocking=false"/>
    </transitStops>
    <transitLine id="Green Line">
        <transitRoute id="1">
            <description>comments</description>
            <transportMode>bus</transportMode>
            <routeProfile>
                <stop refId="1" departureOffset="00:00:00"
                      awaitDeparture="true"/>
                <stop refId="2" arrivalOffset="00:01:30"
                      departureOffset="00:02:00"
                      awaitDeparture="true"/>
                <stop refId="3" arrivalOffset="00:05:00"
                      awaitDeparture="true"/>
            </routeProfile>
            <route>
                <link refId="1"/>
                <link refId="2"/>
                <link refId="3"/>
            </route>
            <departures>
                <departure id="1" departureTime="07:00:00" vehicleRefId="10"/>
                <departure id="2" departureTime="07:30:00" vehicleRefId="16"/>
                <departure id="3" departureTime="08:00:00" vehicleRefId="32"/>
                <departure id="4" departureTime="08:15:00" vehicleRefId="9"/>
            </departures>
        </transitRoute>
    </transitLine>
</transitSchedule>

```

Figure 6 Example transitSchedule.xml file showing the hierarchical data structure of bus lines defined in MATSim

The transitSchedule.xml file can be considered in two sections. In the first section of this file, the physical infrastructure of the public transit service is described (transitStops). Every stop facility (stopFacility) served by the public transit provider is listed here. These are the locations where passengers can alight and board transit vehicles. They are described by coordinates, an identifier and a reference to a network link. Additionally, a name as well as further attribute describing whether the transit vehicle blocks other cars on the link whilst servicing this stop can be specified (isBlocking). In the second section, transit lines are defined (transitLine). These correspond to the definition of *transit lines* as stated in section 4.1. They are also described via an identifier and can be comprised of one or more route.

A transitRoute outlines one specific route of a transit line, including its temporal profile. Typically, transit lines are made up of two transit routes, one for each direction. However, this is not always the case as some public transit providers may offer varying

services to make up for peak hour demands or other factors. Transit routes are in a first section described by a detailed `routeProfile`. The route profile provides a list of the serviced transit route stops (`stop`). These in turn, reference one of the stop facilities defined in the first section of the file. Moreover, departure and arrival offsets are assigned to each stop defining a relative time offset to the departure time of the first stop of the route. Following the route profile, the network route, which is a series of network links describing the complete route the transit vehicles follows, is defined (`route`). Finally, a transit route also requires a list of `departures`. These denote the exact time the transit vehicle starts the corresponding transit route. In the departure section, explicit vehicles may also be assigned. Transit vehicles are defined and initiated in a `transitVehicles.xml` file which will be discussed later on.

The following figure provides an overview of the relationships and structure of the transit schedule file via an entity-relationship diagram.



**Figure 7 Entity-Relationship (ER) model of transit schedule data structure (Poletti 2016)**

Typically, the data for the transit schedule file is not created from scratch but rather extracted from existing transit data exchange formats, such as the ones described in section 4.2. This can be realised through tools which convert the data provided in the data exchange formats to MATSim transit schedules elements such as the GTFS converters developed by Zilske and Kühnel (2019) or Ordóñez and Erath (2011), the HAFAS converter developed by Bösch and Ciari (2015) or the converters provided in the PT2MATsim package (Poletti 2019). Through these, an unmapped transit schedule according to the MATSim input file structure, can be generated. Unmapped refers to a schedule where the stop sequence and departure times are defined but not the network route or reference links (Poletti 2016).

The network routes used by the public transit vehicles are generally not included in the common data exchange formats. They must be reconstructed based on the stop sequences. Given the particular nature of public transit routing, this task is not so straightforward. MATSim possesses the possibility to simulate concurrent use of the road network by different modes. However, this feature has remained largely unused in the past and subsequently, mostly simulations ignore the interactions of public transit and private vehicles on the network links (Ben-Dor, Dmitrieva et al. 2017). Typically, public transit vehicles are modelled to use a separate network from private cars as this was most readily achievable given the available data. One way to generate such a network route is using the “CreatePseudoNetwork” API (Rieser 2018). This builds a network with direct links between transit stops according to the route profiles of the unmapped schedule file. The outputs are modified transit schedule and network files. The previously unmapped `transitSchedule.xml` code will have the network routes and reference links for each transit route added. Likewise, the new network file will have the generated public transit route network merged to it. The network routes are generated as artificial, separate links from the car network links. That is, private cars and public transit are not interacting with each other on the same links. Moreover, the connecting links generated are the straight-line distances between stops. Only one link is created to connect two consecutive stops, based on the Euclidean distance. These distances do not exactly correspond to the actual travel routes and distances the public transit vehicles travel between stops. This may be problematic in areas where transit stops are more dispersed.

In order to make use of the multimodal features in MATSim and observe the concurrent use of the network by public transit and private cars, the route travelled by public transit vehicles must be known. Map matching methods which utilize GPS data to identify this route is fairly well explored and an active field of research (Quddus, Ochieng et al. 2007). GPS information is in many cases not available though, and gathering this data for larger regions is costly. Up until recently, documentations on algorithms producing this route without GPS data had been sparse. In 2016 the PT2MATSim (Poletti 2016) package was developed, a comprehensive tool package which facilitates the creation of multimodal networks and transit schedules for MATSim from openly available data formats.

The PT2MATSim package offers tools to convert public transit data from HAFAS, GTFS and OSM to an entirely mapped MATSim transit schedule. The following diagram presents the data sources and workflow of the PT2MATSim package.



Figure 8 PT2MATSim data sources and workflow (Poletti 2016)

The `Gtfs2TransitSchedule.java` and `Hafas2TransitSchedule.java` classes convert the information contained in the public transit data formats to unmapped MATSim schedules. As mentioned in 4.2.3, OSM files are not suitable for schedule data yet, however a further converter, the `Osm2MultimodalNetwork.java` class creates a multimodal network for MATSim. The core of the PT2MATSim package is the `PublicTransitMapper`, which implements a pseudo routing algorithm that reconstructs the routes for public transit vehicles based the stop locations, sequences and the street network. The algorithm computes the least-cost path from the first to the last stop of a transit route, under the constraint that the path must contain a link candidate for each transit stop on the route. The mapper is initiated via a config file which declares the necessary input parameters. Finally, the package also provides a plausibility check to assure the transit routes are feasible (Poletti 2019).

The `transitVehicles.xml` file describes the specific vehicles serving the public transit lines. In the first part of the file the vehicle types are described (bus, train, tram etc.), including their passenger capacities and also vehicle dimensions. These can include the width, length and passenger car equivalent unit (PCE) of the transit vehicles. In the second part of the file actual vehicles must be listed and described with an identifier and one of the previously defined vehicle types (Rieser 2016). This can represent the actual transit vehicle fleet of the public transit provider. Alternatively, should this data not be available or necessary for the simulation, a transit vehicle can also be

assigned for every departure of the schedule. In the following, a simple example of such a file is given.

```
<?xml version="1.0" encoding="UTF-8"?>

<vehicleDefinitions xmlns="http://www.matsim.org/files/dtd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.matsim.org/files/dtd
    http://www.matsim.org/files/dtd/vehicleDefinitions_v1.0.xsd">

  <vehicleType id="bus">
    <description>Hybrid bus</description>
    <capacity>
      <seats persons="28"/>
      <standingRoom persons="49"/>
    </capacity>
    <length meter="12"/>
    <width meter="2.5"/>
    <accessTime secondsPerPerson="1.0"/>
    <egressTime secondsPerPerson="1.0"/>
    <doorOperation mode="serial"/>
    <passengerCarEquivalents pce="3.0"/>
  </vehicleType>

  <vehicle id="hy_0" type="bus"/>
  <vehicle id="hy_1" type="bus"/>
  <vehicle id="hy_2" type="bus"/>
  <vehicle id="hy_3" type="bus"/>
</vehicleDefinitions>
```

Figure 9 Example `transitVehicles.xml` file (Rieser 2016)

## 5. Methodology

With the aim of analysing the public transit modelling capabilities and sensitivities presented in the MATSim simulation environment, this thesis will examine and compare different methods to implement public transit services in MATSim. This analysis is carried out through creating a MATSim model for the district of Wunsiedel in Germany. Public transit will be simulated for the region utilizing three different methods. The first method will simulate public transit with a fully-multimodal schedule and network, generated using the PT2MATSim package. The second method simulates public transit using a separate network from the private car users. The third method of implementation will not be explicitly simulating public transit and public transit users will be teleported. All scenarios will use the same basic data and configuration parameters, where appropriate. A further analysis is presented with artificially induced capacity constraints on the network, in order to observe the effects of congestion in each implemented method.

## 5.1. Wunsiedel im Fichtelgebirge

### 5.1.1. Region

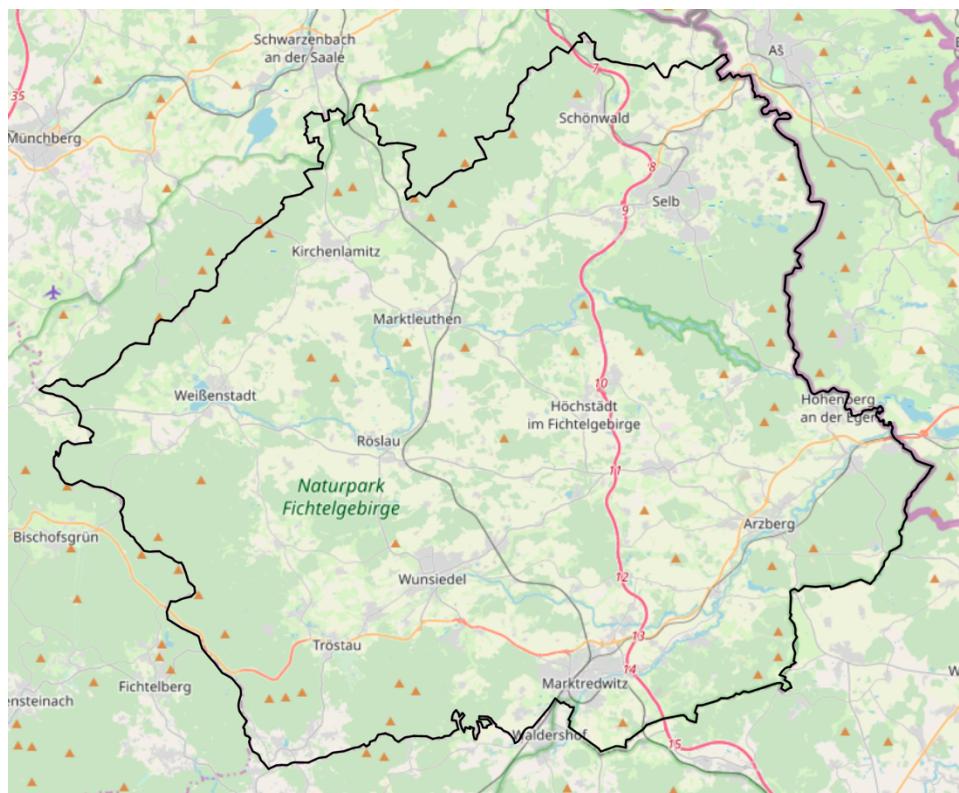
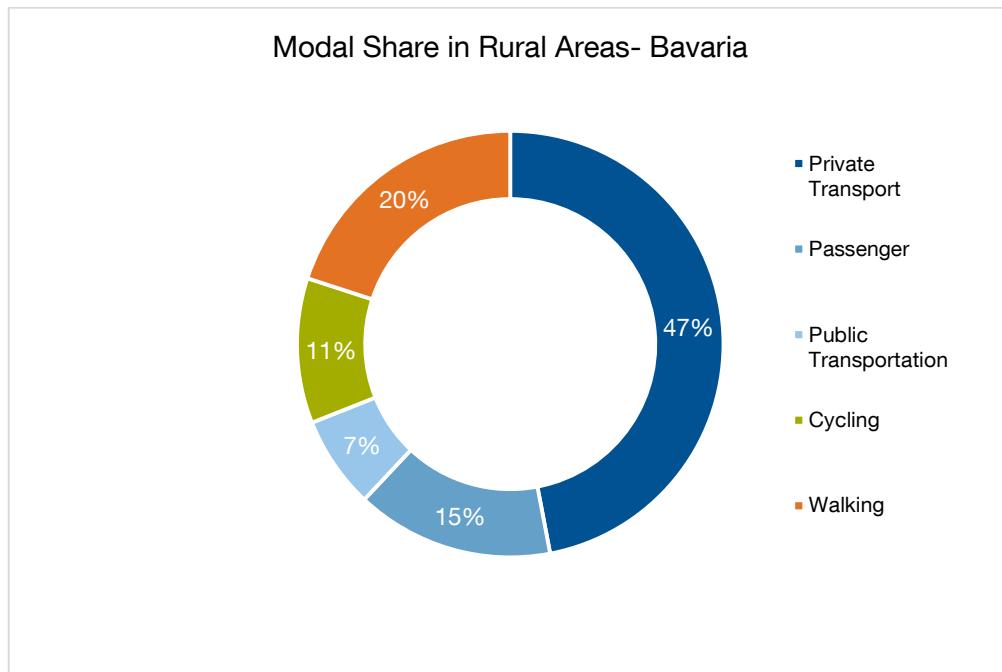


Figure 10 District of Wunsiedel (OpenStreetMap contributors 2019)

The Upper Franconian district Wunsiedel in the Fichtelgebirge is situated in the north-eastern low mountain range of Bavaria. It comprises 9 towns and 8 municipalities with roughly 73 000 inhabitants on 606,40 km<sup>2</sup> of landmass. The population is largely focused in the southeast of the district in two regional centers Wunsiedel Stadt and Marktredwitz and a further mid-sized regional center in Selb (Landkreis Wunsiedel im Fichtelgebirge 2016). With a population density of 125 persons per km<sup>2</sup> (Bayerisches Landesamt für Statistik 2018), the district of Wunsiedel can be described as predominantly rural as defined in the regional typology by the Organization for Economic Co-operation and Development (OECD) (OECD 2011).

### 5.1.2. Transportation

According to the national travel survey *Mobility in Germany* (MiD, for its abbreviation in German) from the year 2008, the total share of private car users (private transport plus passengers) in rural Bavaria amounts to 62% (Bundesministeriums für Verkehr Bau und Stadtentwicklung 2008).



**Figure 11** Modal share in rural areas in the state of Bavaria, adapted from (Bundesministeriums für Verkehr Bau und Stadtentwicklung 2008).

The following graph depicts the number of trips for the district of Wunsiedel by trip purpose according to a household travel survey conducted by the bus operator Regionalbus Ostbayern (RBO) in 2017. Work trips make up the highest number of trips for the district, followed by education trips.



**Figure 12 Total traffic in the district of Wunsiedel by trip purpose (Jürgens and Fritsch 2017).**

The local public transit agency for the district of Wunsiedel is the Verkehrsgemeinschaft Fichtelgebirge (VGF). The VGF services 25 bus lines throughout the district (Verkehrsgemeinschaft Fichtelgerbirge 2018). A comprehensive map of the entire network plan and stops serviced by the VGF can be found in Appendix A. Typical for rural regions, the public transit offerings in the district of Wunsiedel are strongly characterized by the settlement structure and disperse population distribution of the area (Jürgens and Fritsch 2017).

## 5.2. Data Preparation

The minimal core data required for a MATSim scenario consists of a network, population and configuration file. The following describes the development of these files for the Wunsiedel scenario. Likewise, the preparation of the public transit schedule and vehicle files are discussed.

### 5.2.1. Network

Networks consist of nodes and links with individual identifier ids. Nodes are described by their x-and y-coordinates. Links contain further information such as the connecting *to* and *from* nodes. Moreover, they have attributes describing traffic-related characteristics such as the length of the link, the flow capacity, the maximum travel speed permitted on the link, the number of lanes available as well as the permitted modes on the links (Horni, Nagel et al. 2016).

Network files are typically converted from OpenStreetMap (OSM) data. As previously mentioned, several tools exist to transform OSM data into a MATSim road network. For the scenarios developed in this thesis, OSM data containing the relevant road

infrastructure of the district of Wunsiedel were download using the Overpass API tool (OpenStreetMap 2018) and converted using the *Osm2MultimodalNetwork.java* class provided in the PT2MATSim package (Poletti 2019).

Special attention must be given to the coordinate system of the network. MATSim requires the use of a Cartesian coordinate system. This is due to that distances between coordinate points are calculated using the Pythagoras theorem in the MATSim code. Using a spherical coordinate system for this is very complex and computationally expensive and would therefore slow down such calculations (Horni, Nagel et al. 2016). The network in the Wunsiedel scenario uses Gauß-Krüger 4 coordinate projections.

### 5.2.2. Population

Travel demand in MATSim is generated through agents fulfilling their day plans. One plan per agent is selected and executed by the mobsim. As mentioned earlier, the full set of agents in a scenario form a population. Travel demand is thus described in either the `population.xml` file which contains a list of all agents and their plans, or alternatively the `plans.xml` files, which lists all of the day plans of a MATSim simulation.

The demand generation for the Wunsiedel scenario is based on the code previously developed within the realms of the MobiDig project. Likewise, much of the information on commuter data and geographical files was made available through this. The following activities have been defined for the Wunsiedel scenario.

**Table 3 Activity types**

Activity type	Typical duration [hours]	End Time
home	16	7:00
work	9	17:00
school	5	13:00

Given the available data, a simplified demand was generated based on commuter data. As seen in Figure 12, work trips make up the highest number of trips for the district. Through this, commuters are able to give a feasible representation of typical peak hour traffic demand structures. In order to represent this in the MATSim simulation, in a first step, an O-D matrix based on the commuter statistics for the district is integrated in

the demand generation code for the scenario. The commuter data has been collected in 2017 by the Federal Labour Office of Germany and provides the statistics on commuters in the High Franconia region based on municipalities. Commuters in terms of employment statistics are all employees, subject to social insurance contributions, whose place of work differs from their place of residence. The following tables list the relevant commuter flows. The commuters leaving the district of Wunsiedel are aggregated by the municipality of their place of residence. The commuters entering the district of Wunsiedel have been aggregated by the municipality of their place of work. They are listed according to the official municipality key codes or “Amtlicher Gemeindeschlüssel” (AGS) in German.

**Table 4 Number of Commuters leaving district Wunsiedel, aggregated by the municipalities of their place of residence**

<b>AGS</b>	<b>Place of Residence</b>	<b>Number of Commuters</b>
9479111	Bad Alexandersbad	183
9479112	Arzberg, Stadt	1.333
9479126	Höchstädt i.Fichtelgebirge	283
9479127	Hohenberg a.d.Eger, Stadt	305
9479129	Kirchenlamitz, Stadt	848
9479135	Marktleuthen, Stadt	751
9479136	Marktredwitz, Stadt	2.461
9479138	Nagel	456
9479145	Röslau	438
9479147	Schirnding, Markt	228
9479150	Schönwald, Stadt	703
9479152	Selb, Stadt	2.173
9479158	Thiersheim, Markt	414
9479159	Thierstein, Markt	282
9479161	Tröstau	616
9479166	Weissenstadt, Stadt	569
9479169	Wunsiedel, Stadt	1.896
<b>Total</b>		<b>13.657</b>

**Table 5 Number of commuters entering district Wunsiedel, aggregated by the municipalities of their place of work.**

<b>AGS</b>	<b>Place of Work</b>	<b>Number of Commuters</b>
09479111	Bad Alexandersbad	142
09479112	Arzberg, Stadt	473
09479126	Höchstädt i.Fichtelgebirge	69
09479127	Hohenberg a.d.Eger, Stadt	141
09479129	Kirchenlamitz, Stadt	120
09479135	Marktleuthen, Stadt	201
09479136	Marktredwitz, Stadt	5.726
09479138	Nagel	24
09479145	Röslau	222
09479147	Schirnding, Markt	94
09479150	Schönwald, Stadt	468
09479152	Selb, Stadt	3.461
09479158	Thiersheim, Markt	50
09479159	Thierstein, Markt	109
09479161	Tröstau	579
09479166	Weissenstadt, Stadt	569
09479169	Wunsiedel, Stadt	2.279
<b>Total</b>		<b>14.379</b>

The tables present insight on the commuter flows within the district. The population in the district of Wunsiedel is not focused in one dominating rural center, but rather in three mid-sized agglomerations: Marktredwitz Stadt, Wunsiedel Stadt and Selb Stadt. This is also reflected in the commuter flows, as these three towns have the largest number of commuters entering and leaving.

In order to assign the activities locations, the O-D relations have to be combined with shapefiles containing administrative boundaries in order to give the municipalities realistic georeferenced positions within the network. This is done via the AGS municipality keys. The shapefiles were retrieved from the service center of the Federal Agency for Cartography and Geodesy (Bundesamt für Kartographie und Geodäsie 2018). It is important to note the spatial reference system of the shapefiles used, as transformations

need to be implemented in the code to match up the network coordinates with the shapefile coordinates.

This now enables activities to be paired with a location in the network within the vicinity of municipality boundaries. The exact locations are drawn randomly via the code within the corresponding boundaries.

In a next step, the O-D relation methods are defined a share of motorized private transport, that is, from the overall modal split the share of private car users. For this model, this number was taken from the most recent available traffic survey, MiD, from the year 2008, 62% (Bundesministeriums für Verkehr Bau und Stadtentwicklung 2008).

Finally, the amount of intra-town employee traffic must also be produced. That is, the number of employees living and working in the same town must be generated. This is done by subtracting the number of commuters entering each town from the number of employees at the workplace subject to social insurance registered for each town, which is taken from Bavarian State Bureau of Statistics for the year 2016 (Bayerisches Landesamt für Statistik 2017).

In order to increase the precision of the simulation, the locations of the *home* activities within the district of Wunsiedel are georeferenced in a more detailed manner into the code. For this, an official population file for the High Franconia region was used. The file is a csv file listing the land uses in the whole region, with corresponding x- and y-coordinates and also specifies the number of inhabitants at each, where appropriate. The csv file was filtered for residential buildings. This list of residential buildings with their x- and y-coordinates are integrated into the code in such a way that when a *home* location in an O-D pair lies within the district of Wunsiedel, which is determined over the AGS key, the agents are placed at one of the residential buildings in the list. If the agent's home location doesn't fall in the district of Wunsiedel, a random point is drawn in the corresponding municipality within the shapefile.

Likewise, according to a file containing the top 100 companies with their headquarters in the district of Wunsiedel, further details were incorporated into the code. The file on the companies lists exact locations and number of employees at each company. Once again, the x-and y-coordinates of the companies are integrated into the code in such a way that when a *work* location in an O-D pair lies within the district of Wunsiedel, which

is determined over the AGS key, the agents are placed at one of positions on the list. This is accessed in the code based on a normalized count function in order to produce the number of agents with work locations in Wunsiedel weighted according to the size of the company. This is deduced through generating the ratio of employees at each individual company over all registered employees within the list. Again, if the agent's work location doesn't fall in the district of Wunsiedel, a random point is drawn in the corresponding municipality within the shapefile.

The second largest number of trips in the district of Wunsiedel are made for education. In order to integrate this into the simulation, an activity representing pupils attending elementary and secondary school will be defined. The information for this is taken from a file providing the locations (x-and y-coordinates) of all schools in the district of Wunsiedel. It also provides the number of pupils attending each school.

The assumption made, is that pupils will attend a school in the same municipality in which they live in. Thus, if the origin municipality is the same as the destination municipality, a *school* activity will be created in one of the school locations listed in the file according to the corresponding municipality, once again linked via the AGS key. This is only applied to 11.5% of intra-trips made in the district. The following table shows the age distribution of the district of Wunsiedel according to the population census of the district for the year 2016.

**Table 6 Age distribution of the population in the district of Wunsiedel (Bayerisches Landesamt für Statistik 2018).**

Age [years]	Percentage of Population [%]
under 6	4,8
6 to 15	8,4
15 to 18	3,1
18 to 25	6,5
25 o 30	5,2
30 to 40	11,6
40 to 50	11,4
50 to 65	24,5
65 or over	24,5

As can be taken from the table, 11.5% of the total population is between the ages of 6 and 18, the ages at which you attend primary and secondary school. A further

assumption and significant limitation of this is that this share is tied to the commuters. That is, it is calculated as being part of the commuter trips and so assumes the pupils are sharing rides or carpooling with the commuters. This would not be accurate as a significant share of the pupils may travel independently to and from school, especially when using public transit.

The final *home* activity does not require an end-time. This is due to that a simulation in MATSim is a single day and the plans “wrap around” the whole day so that the final activity merges with the first activity of the plan (Rieser 2016).

This concludes the activities and legs used to generate the population for the Wunsiedel scenario. Whilst the population generated in this project provides some valuable insight on the travel behaviour of the district, it is a very simplified approach based on the data available.

### **5.2.3. Configuration**

Settings and parameters for the configuration of a MATSim simulation are stored in a so-called `config.xml` file. This file builds the connection between the user and the simulation. It is essentially a list of settings and parameters which dictate the way the simulation behaves. Each parameter and its value are listed and categorized into logical groups or modules (Rieser 2016).

Basic information such as the file paths of the various input data files, output directories, but also parameters defining coordinate system, capacity flow factor, storage flow factor and number of iterations, amongst many others, spanning over every step of the MATSim cycle, can be specified in this file.

### **5.2.4. Public Transit**

For the explicit simulation of public transit (pt) in the MATSim framework, two additional input files are required, as described in section 4.3.2, a `transitSchedule.xml` and `transitVehicle.xml` file. Unfortunately, for the region of Wunsiedel, the existing transit schedule data available (GTFS, HAFAS) was largely inconsistent or incomplete and so this approach was not implementable for the model in this thesis. The next section will discuss the alternative workaround method applied.

The relevant information for the bus lines serving the Wunsiedel district, such as departure times and schedules as well as stop positions and sequences was taken from the local public transit provider of Wunsiedel, VGF (Verkehrsgemeinschaft Fichtelgebirge 2018), as well as the electronic journey planner ‘Fahrplan Bayern’ provided by the Bavarian Railway Company, BEG for its abbreviation in German (Bayerische Eisenbahngesellschaft 2018).

An external plugin for the Java OSM editor, JOSM (JOSM 2014), has been developed in order to extend the core functionalities for the use of data in MATSim. This plugin simplifies the conversion of OSM data to MATSim networks. The plugin also facilitates the production of transit schedules. This process is however very labour-intensive and generally only suitable for smaller scenarios. Every stop area of the entire transit network, including the stop position and platform, along with the connecting links, must be manually located and drawn in the editor. Additionally, the JOSM-plugin is also limited in the actual handling of larger data sets. Typically, the plugin will stop working due to memory and processor constraints. Nevertheless, due to a lack of viable alternatives, the plugin was used for the at least partial generation of the transit network of Wunsiedel. That is, the unmapped schedule was generated in JOSM.

The following is an overview of the theoretical workflow necessary to create schedule based public transport for MATSim in JOSM (Kühnel 2018).

1. Import car network.
2. Create and define a transit line.  
[“Route Master Relation” in JOSM]
3. Create and define transit route(s) and add them to the route master relation.  
(One transit route represents only one way of a transit line)  
[“Public Transit Route Relation” in JOSM]
4. Draw the network links which the transit vehicles travel on to serve the transit routes.  
[MATSim links]
5. Add the network links to the corresponding transit routes.

6. Create and define stop facilities. A stop facility must be defined as a “Stop Area” in OSM. Furthermore, a “Stop Area” consists of a “Stop Position” and “Platform”.  
[“Stop Area” relation, “Stop Position” and “Platform” in JOSM]
7. Add stop facilities to the corresponding transit routes.
8. Save updated network.
9. Export transit schedule.

For the generation of the unmapped transit schedule, the working method used for this thesis corresponds to the above listed steps but omitting the tasks in numbers 4.) and 5.). This therefore excludes the drawing of the transit route network paths. This had to be done in order to preserve memory and processing effort in JOSM. 440 transit stops and 43 transit routes were generated in this way for the Wunsiedel scenario. The generation of the transit vehicle network routes will be described for each scenario in the following section, accordingly.

Regarding the `transitVehicle.xml` file, a transit vehicle type “bus” was defined. This vehicle type was given a passenger car equivalent (PCE) unit of 3. An unlimited vehicle fleet was assumed, and thus a separate vehicle was defined for every departure in the schedule.

## 6. Analysis Scenarios

### 6.1. Multimodal

The fully multimodal network, which models the concurrent use of all modes on the same network links, was generated using the PT2MATSim packages (Poletti 2019), as described in 4.3.2. The *PuplicTransitMapper* tool generates the missing routes of the unmapped transit schedule and creates a completely mapped transit schedule as well as a fully-multimodal network file as output. This following Figure 13 shows the fully-multimodal network of Wunsiedel for the multimodal implementation of public transit. The links in blue represent the bus routes and the nodes in green the specific stop locations of the VGF bus lines.

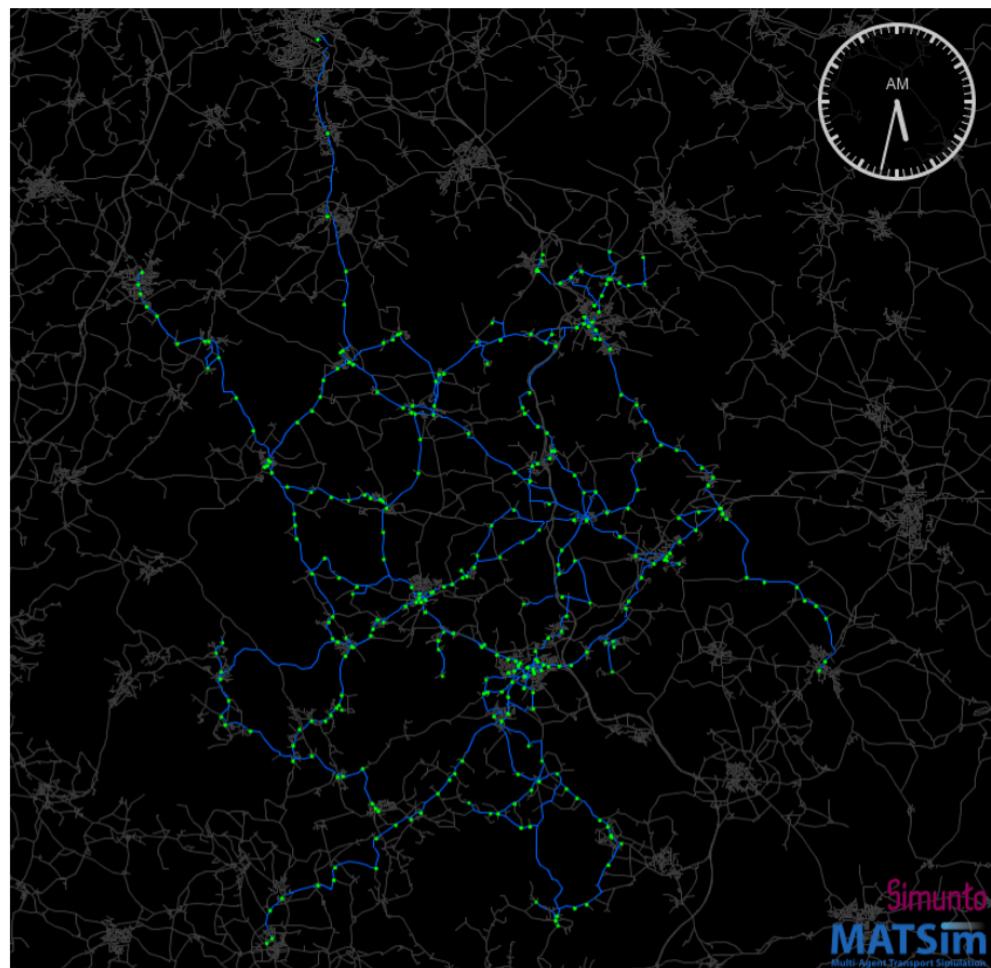


Figure 13 Fully multimodal network of Wunsiedel

## 6.2. Separate Network

The network for the scenario in which public transit and private transport use separate networks in the model has also been created using the PT2MATSim package. For the generation of this network, the parameter which defines the distance for possible link candidates of the algorithm generating the routes of the unmapped transit schedule has be set to 0. Through this, all transit routes are generated as artificial links. In the following Figure 14, the network for the separate network public transit solution is shown.

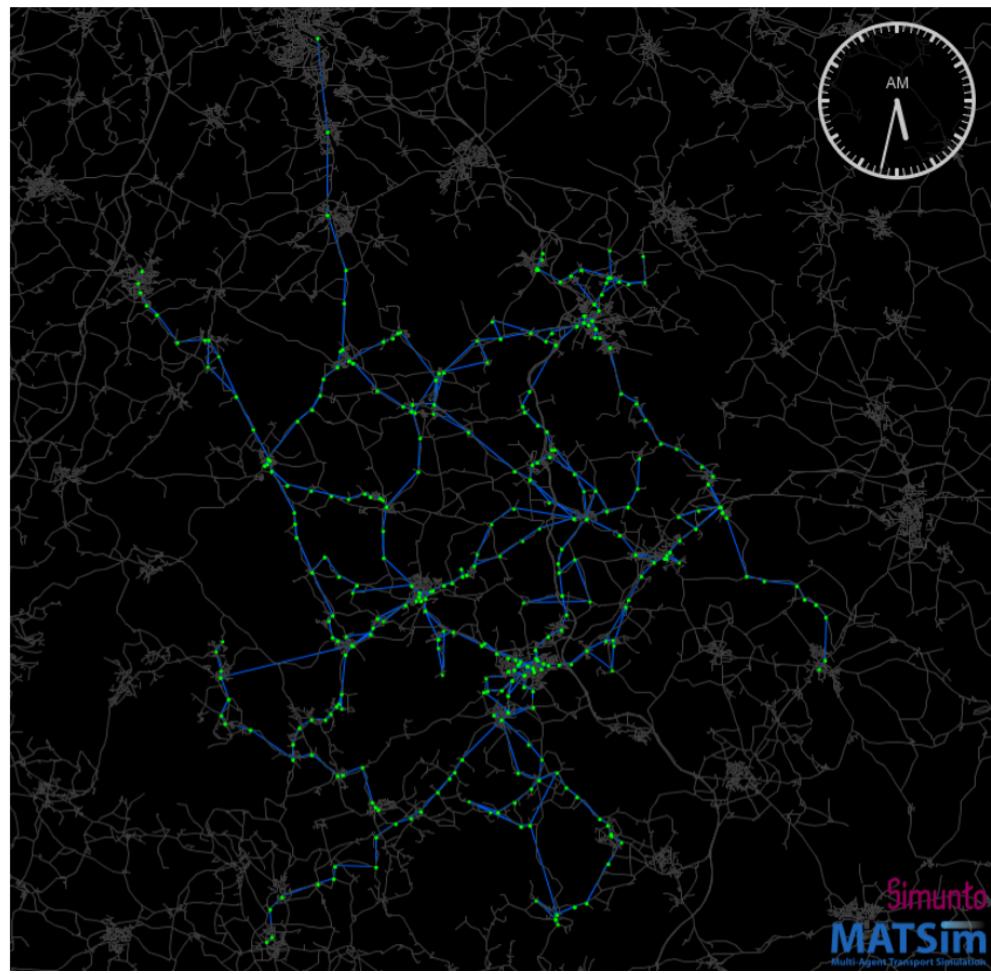


Figure 14 Wunsiedel network with public transit routes modelled on a separate network

### 6.3. Teleportation

Teleportation is executed by default in MATSim for every non-registered main mode. When teleportation is used, the QSim generates a departure event after the ending of an activity and then an arrival event according to an estimated travel time. This travel time can be computed in two ways.

- Teleported mode free speed factor: this method computes the teleported route for the mode in question, using the exact same travel distance as the car route and a travel time based on cars travelling that route at free speed, multiplied by the factor applied.
- Teleported mode speed: this generates a teleported route using a so-called *beeline distance factor* as well as a *teleported mode speed factor*. A beeline distance refers to the shortest, straight line route between two point. Thus, this distance is multiplied by the given factor. The travel time is then computed as the calculated distance divided by the given mode speed factor. This method is useful for modes whose travel times are unrelated to car travel times.

In the Wunsiedel model, as buses travel on the same road network as cars, the first teleportation method was applied. The travel time was set to take 2 times as long as the car travel time when travelling at free speed. This factor is traditionally applied for public transit, based on a statement by the Berlin public transit company claiming a goal of achieving door-to-door travel times not exceeding 2 times those of cars (Rieser 2010). Teleportation is activated through simply disabling the use of the public transit feature in the config file. For the network, the multimodal network file was used. Either one of the network files from the above-mentioned implementation methods could have been used as the definition of the public transit mode in the network links are not considered when public transit is deactivated.

### 6.4. Analysis

In order to gain insight on the capabilities presented by MATSim to model public transit with regards to varying modelling aspects, the three implementation methods discussed above will be evaluated on the following aspects. All scenarios have been

executed for 100 iterations, and other configurations parameters and basic data was kept the same, where applicable.

Population Sample	PT Solution	Model Runtime	Mode Statistics	Travel Time	Route-Time Diagram for PT Vehicles	Spatial Representation	Score Statistics
1%	Multimodal	✓	✓	✓	✓		
	Separate	✓	✓	✓	✓		
	Teleportation	✓	✓	✓			
10%	Multimodal	✓	✓	✓	✓		
	Separate	✓	✓	✓	✓		
	Teleportation	✓	✓	✓			
100%	Multimodal	✓	✓	✓	✓	✓	
	Separate	✓	✓	✓	✓	✓	
	Teleportation	✓	✓	✓		✓	
100% Congestion Scenarios	Multimodal	✓	✓	✓	✓		✓
	Separate	✓	✓	✓	✓		✓
	Teleportation	✓	✓	✓			✓

#### 6.4.1. Scalability

It is recommended that a typical MATSim simulation uses a 10% population sample due to resulting computational restrictions and performance reasons of the simulation. Additionally, this downscaling must also be reflected in the links of the network over the storage and flow capacity factors. These are defined as follows.

**Equation 1 Flow capacity factor**

$$\text{flow capacity} = \frac{\text{capacity value of link}}{\text{capacity period of network}} \times \text{flow capacity factor}$$

**Equation 2 Storage capacity factor**

$$\text{storage capacity} = \frac{\text{length of link} \times \text{number of lanes}}{\text{effective cell size}} \times \text{storage capacity factor}$$

In theory, these two factors should be scaled by the same parameter, however it is recommended that the storage capacity factor is rather scaled according to the following (Nagel 2016):

- $f_{CapacityFlow} = f_{Scale}$
- $f_{StorageCapacity} = f_{Scale}^{(0.75)}$

As recommended by the MATSim calibration for public transit vehicles by Ben-Dor, Dmitrieva et al. (2017) bus PCE units were implemented and scaled by the same factors. The population samples, with the flow and storage capacity factors as well as bus PCE parameters which have been implemented for the Wunsiedel model are summarized in the following table. Each simulation was executed for 100 iterations.

**Table 7 Analysis Scenarios including population sample sizes, bus PCE, flow capacity factor and storage capacity factors**

Scenario	Population Sample	Bus PCE	Flow capacity factor	Storage capacity factor
1	1%	0.03	0.01	0.06
2	10%	0.3	0.1	0.18
3	100%	3	1	1

#### 6.4.2. Performance

Simulations typically have to balance a trade-off between level-of-detail and computational efficiency. That is, increasing levels of detail require increasing computational times. The effects of the varying accuracies the public transit implementations mean with respect to computational efforts will be examined through presenting the typical mobism, replanning and total simulation times of one iteration.

#### 6.4.3. Spatial Representation

As accessibility plays a significant role on mode choice, the three public transit implementations will be examined on whether spatial properties of the public transit infrastructure are considered by each solution. Simulated agents using public transit as a mode will be depicted together with the infrastructure of the public transit on the network. This is done through processing the event files of the simulations in the visualization software Simunto Via (Simunto 2018).

#### 6.4.4. Capacity

In order to examine the sensitivities of each public transit implementation with respect to capacity constraints on the network, two further scenarios of analysis will run the full 100% population sample with reduced flow and storage capacity factors on the network links. This will induce congestion on the network. The public transit implementations will be examined with regards to the following aspect:

- Average travel times by mode. It is the average of the travel time of all trips made by every agent for every purpose. It is computed via an EventHandler.
- Modal statistics over a simulation
- Agent score statistics over a simulation
- Route-time diagrams depicting scheduled and simulated performances of a public transit route. Always the same route and direction is examined.

## 7. Results

The following chapter presents the results of running the MATSim scenarios described in section 6.4

### 7.1. Performance

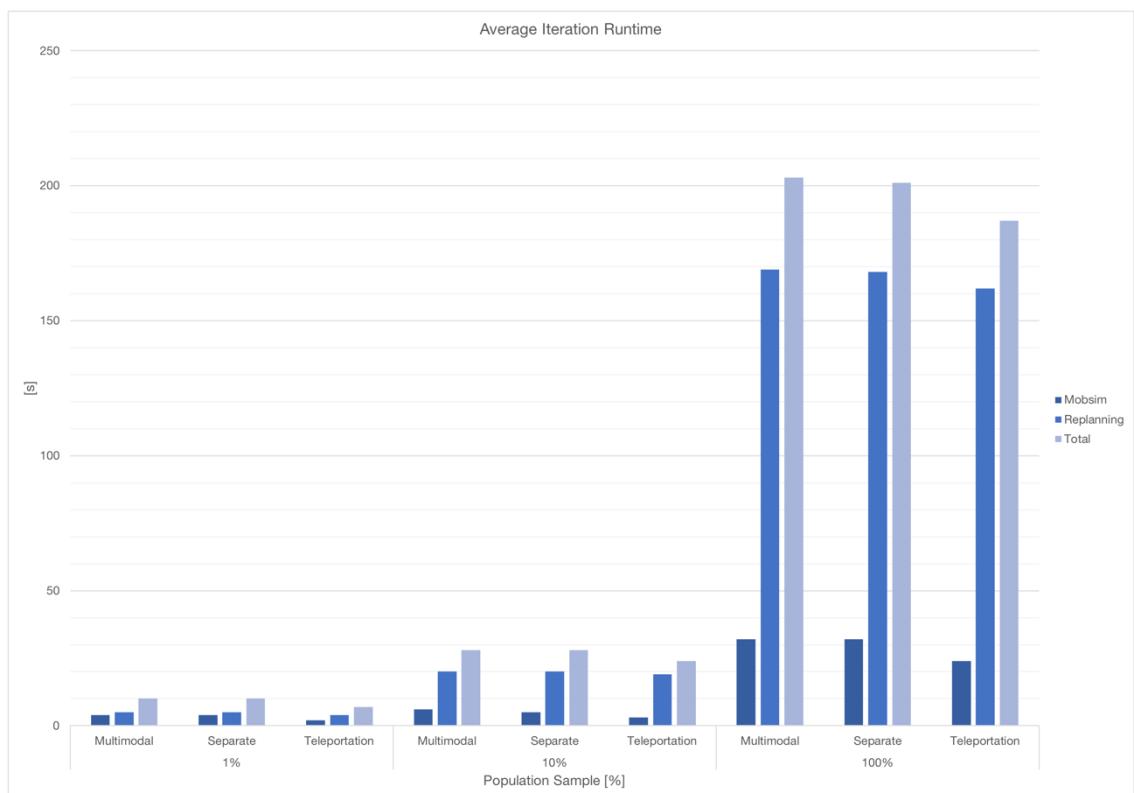
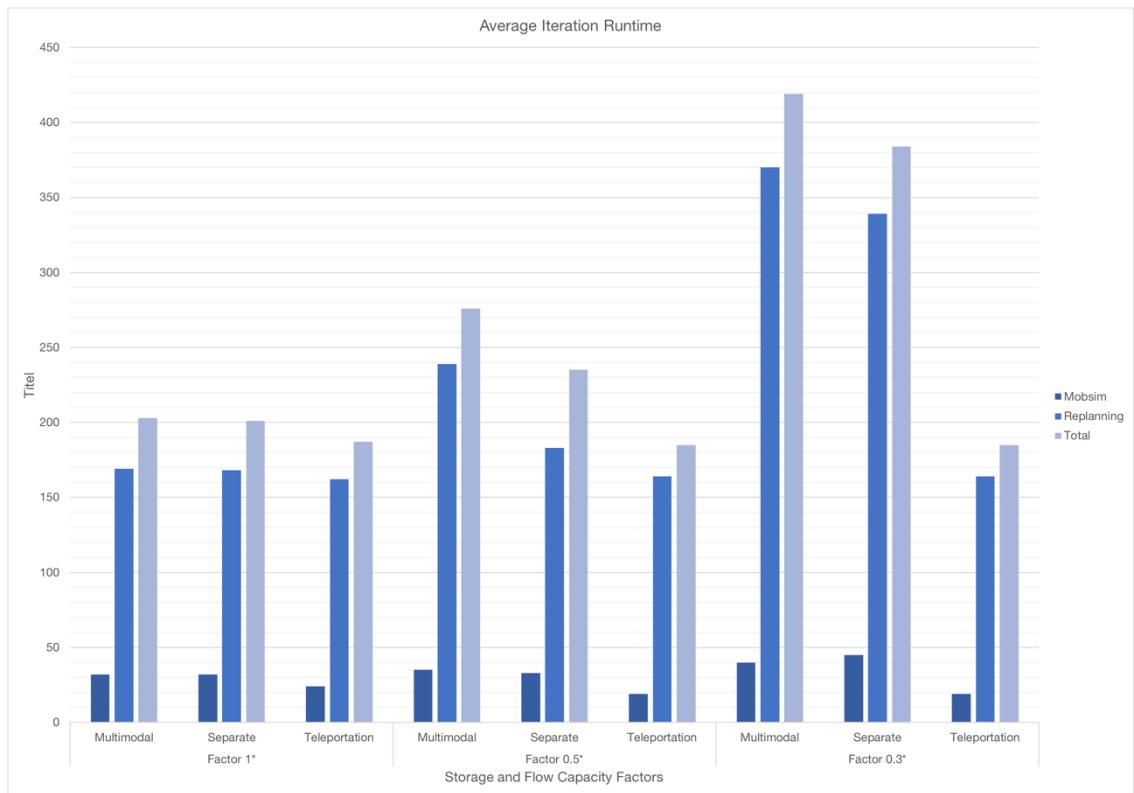


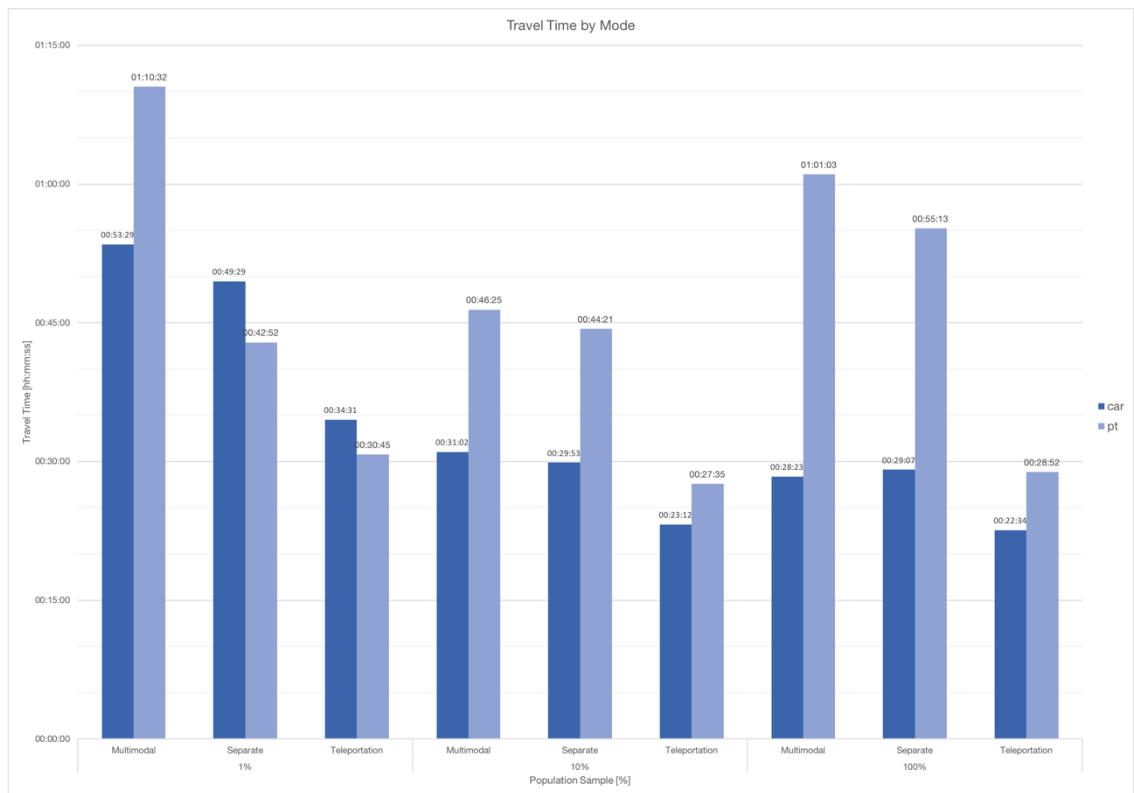
Figure 15 Average iteration runtimes by population sample size



**Figure 16 Average iteration runtimes by storage and flow capacity factors of the network.**  
 \* Factors according to 6.4.1

## 7.2. Scalability

In order to observe effects of scalability in each implementation method, the travel times for each mode in the scenarios are presented.



**Figure 17 Average travel time by mode, for different population size samples.**

Moreover, for the multimodal and separate network scenarios route-diagrams of a transit route, route 4 from Marktredwitz to Hof are given. These depict the scheduled and simulated results. As the teleportation scenario does not explicitly model public transit and its vehicles, these cannot be generated.

### 7.2.1. 1% Population Sample

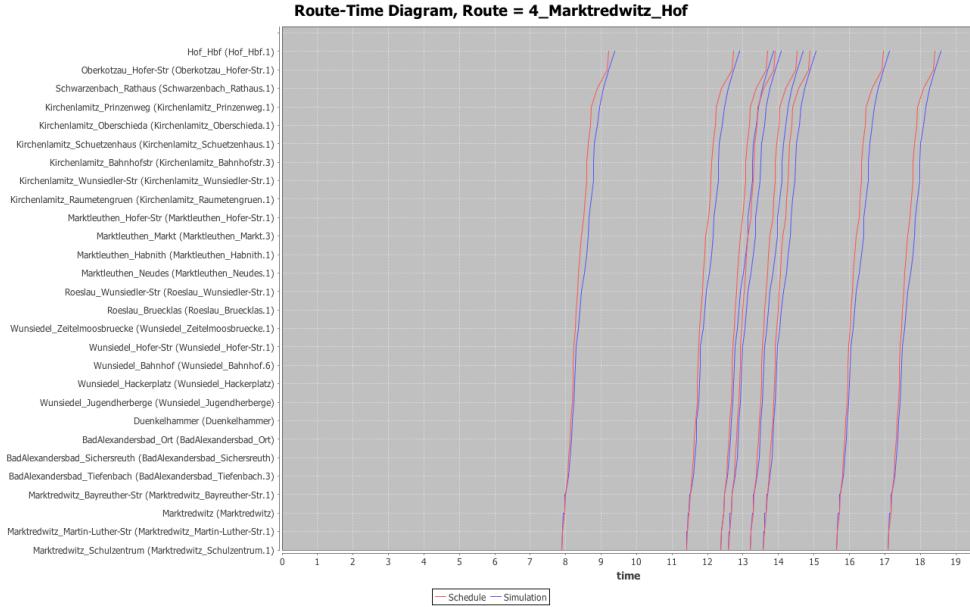


Figure 18 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof

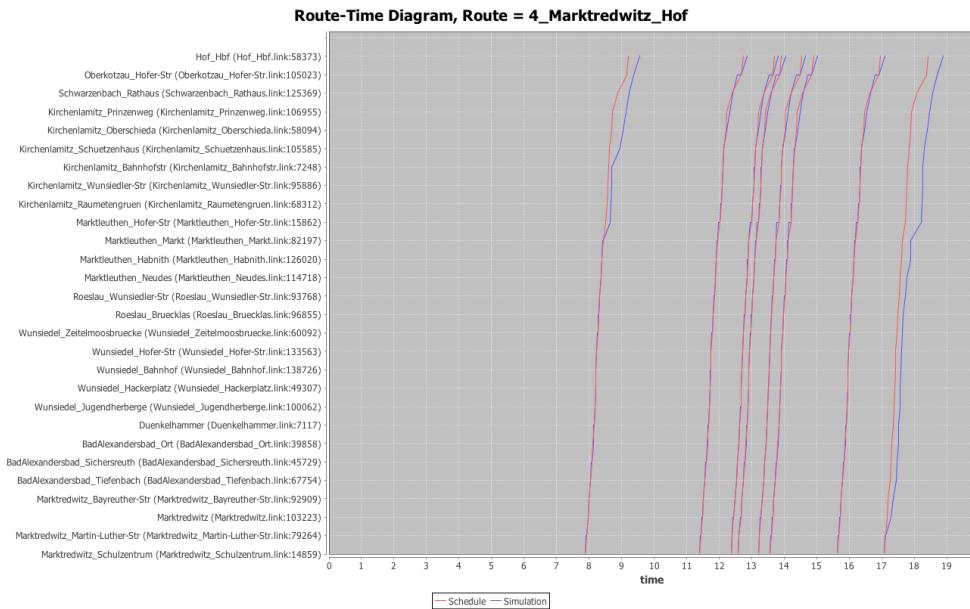


Figure 19 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof

### 7.2.2. 10% Population Sample

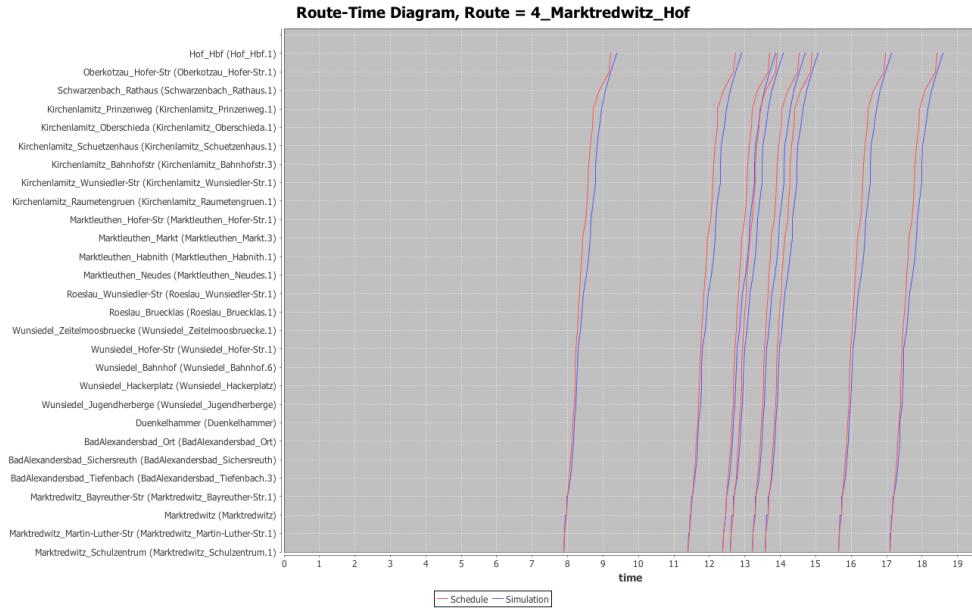


Figure 20 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof

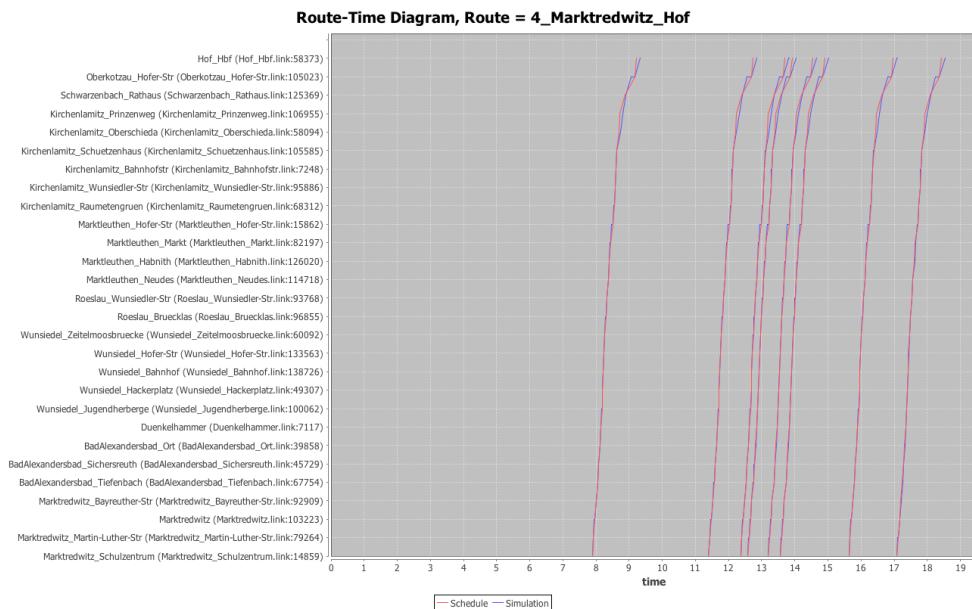


Figure 21 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof

### 7.2.3. 100% Population Sample

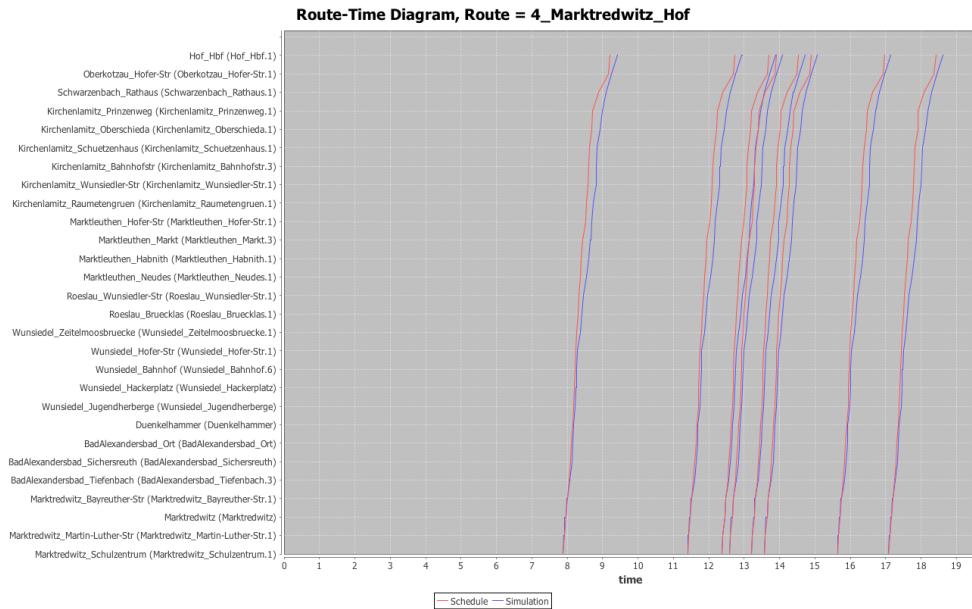


Figure 22 Separate scenario: route-time diagram for bus route 4: Marktredwitz to Hof

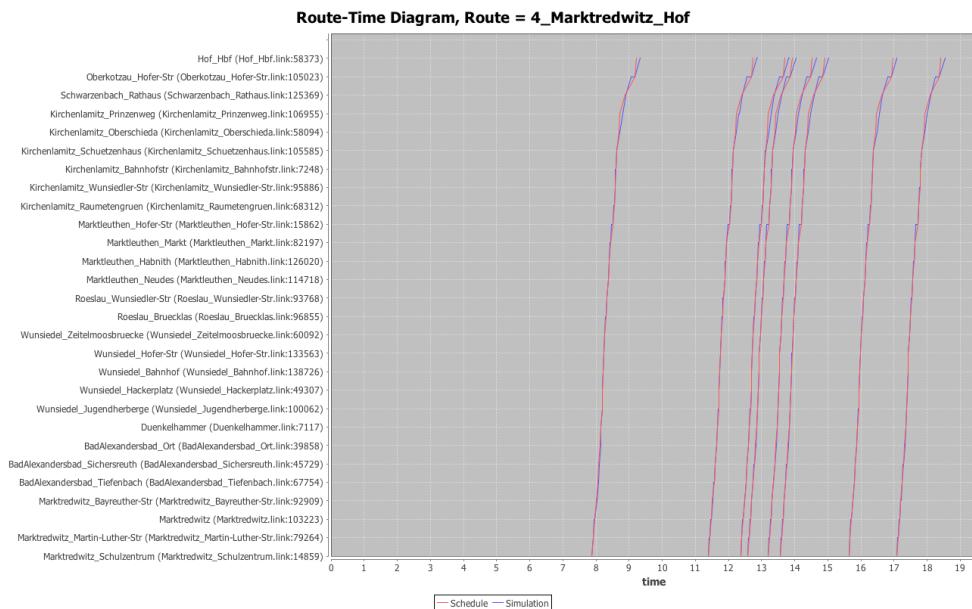


Figure 23 Multimodal scenario: route-time diagram for bus route 4: Marktredwitz to Hof

### 7.3. Spatial Representation

The following images depict the distribution of simulated agents in the scenarios who chose public transit as their mode for trips.

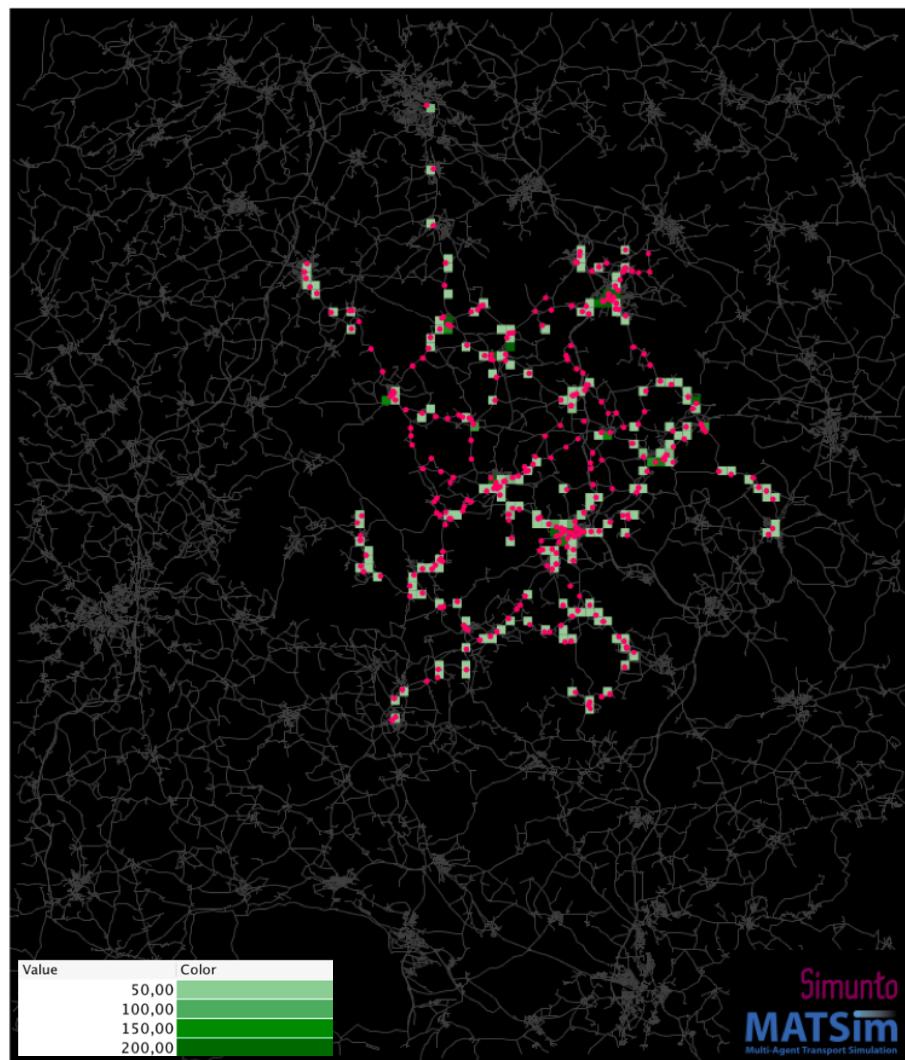


Figure 24 Multimodal network scenario

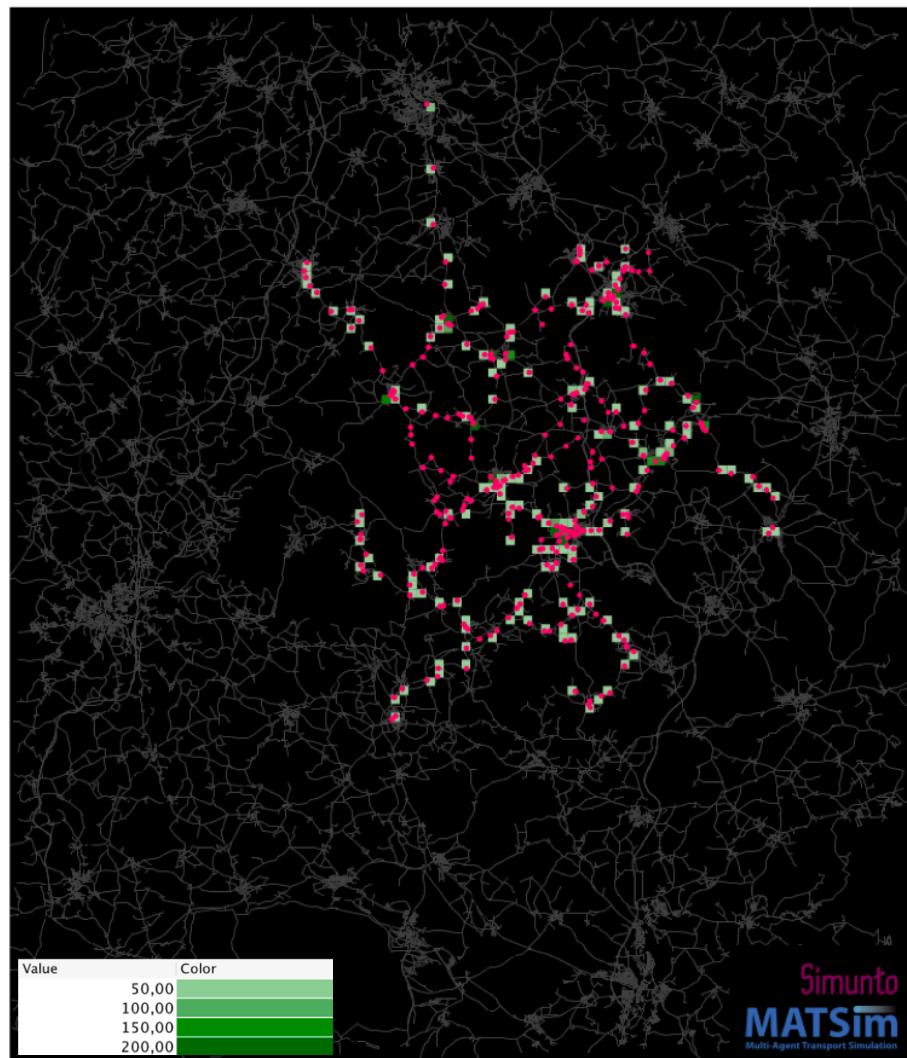


Figure 25 Separate network scenario

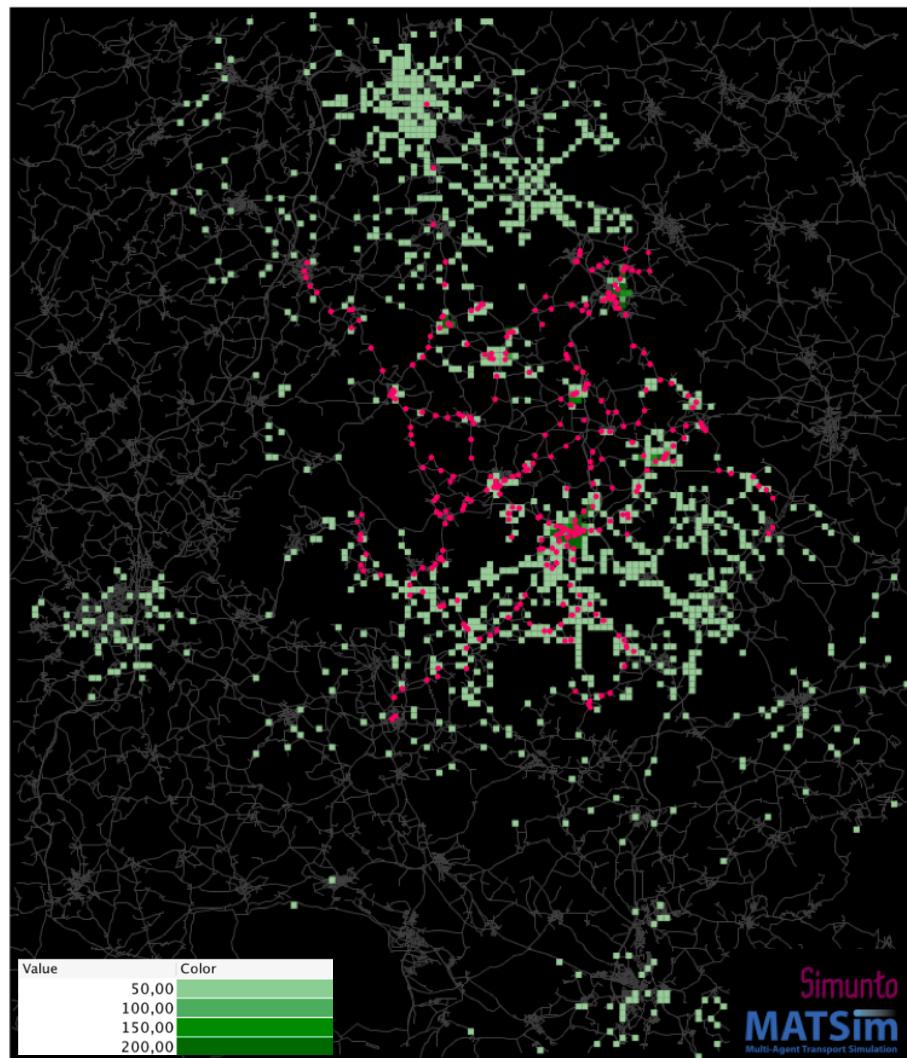


Figure 26 Teleportation scenario

The following figure presents a closer view of the multimodal and separate network scenarios side by side.

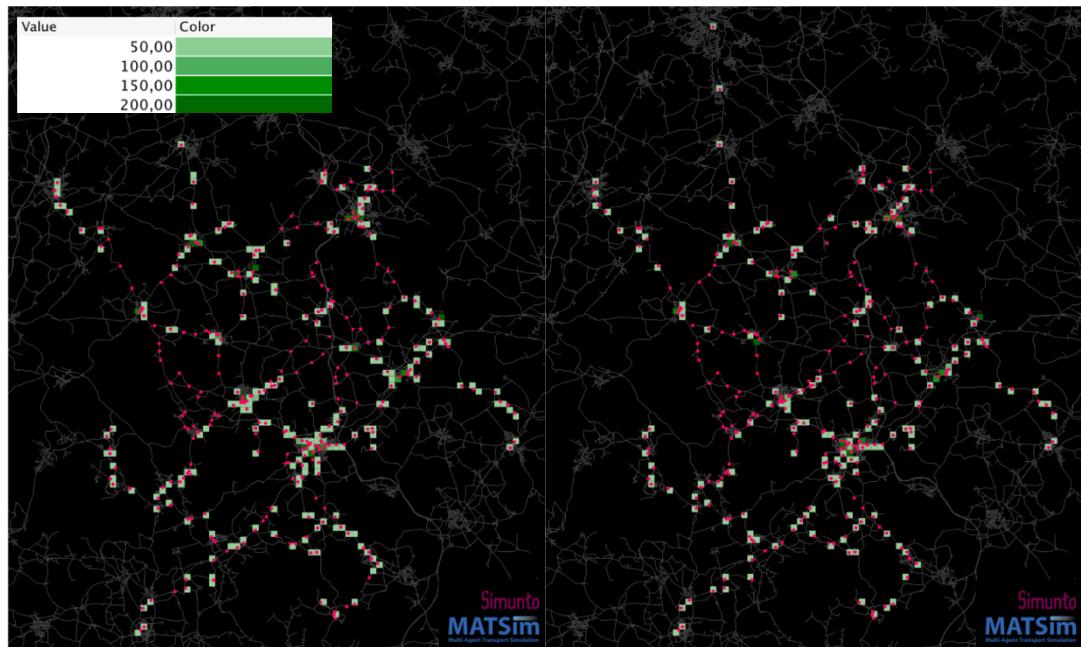


Figure 27 Close ups of multimodal and separate network scenarios, respectively.

## 7.4. Capacity

### 7.4.1. Mode Statistics Normal Traffic Conditions

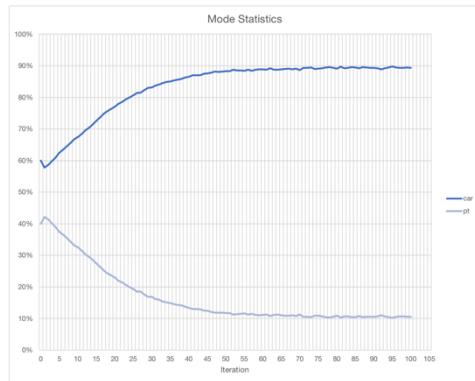


Figure 28 Multimodal network scenario with normal traffic conditions

### Flow Capacity Factor = 0.5, Storage Capacity Factor = 0.6



Figure 31 Multimodal network scenario with flow capacity factor = 0.5, storage capacity factor=0.6

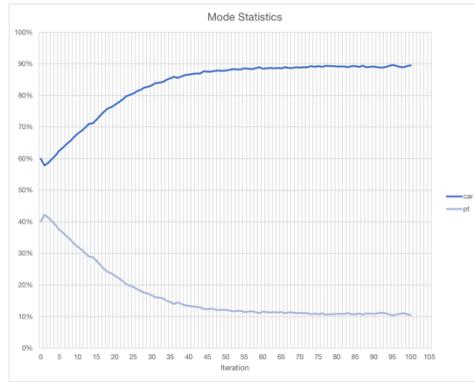


Figure 29 Separate network scenario with normal traffic conditions

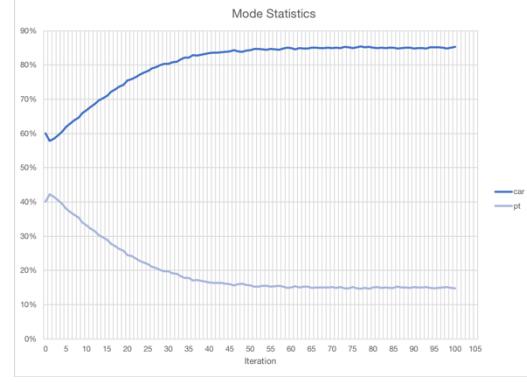


Figure 32 Separate network scenario with flow capacity factor = 0.5, storage capacity factor=0.6

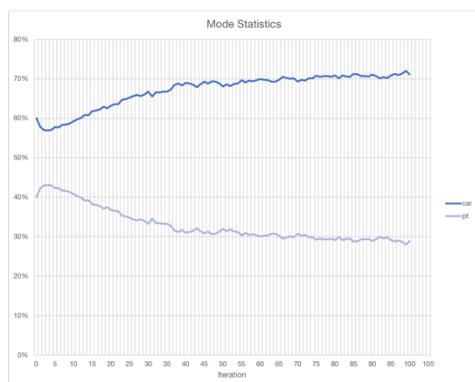


Figure 30 Teleportation scenario with normal traffic conditions

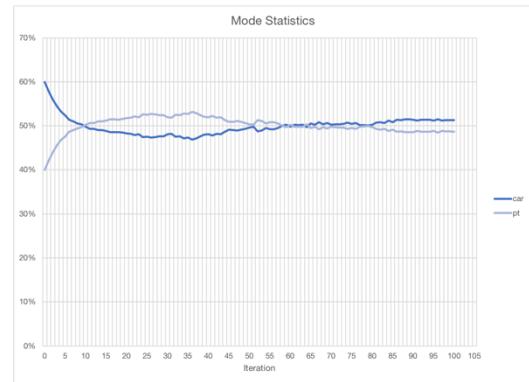
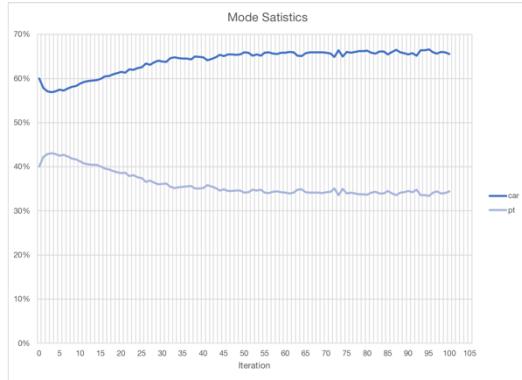
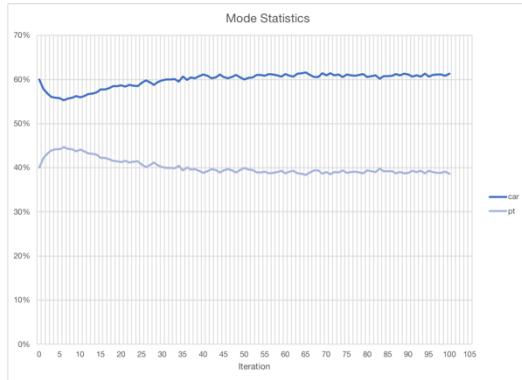
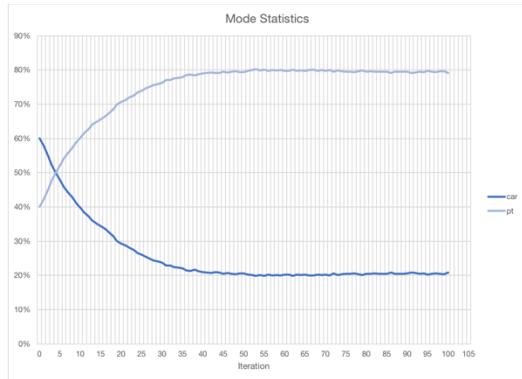
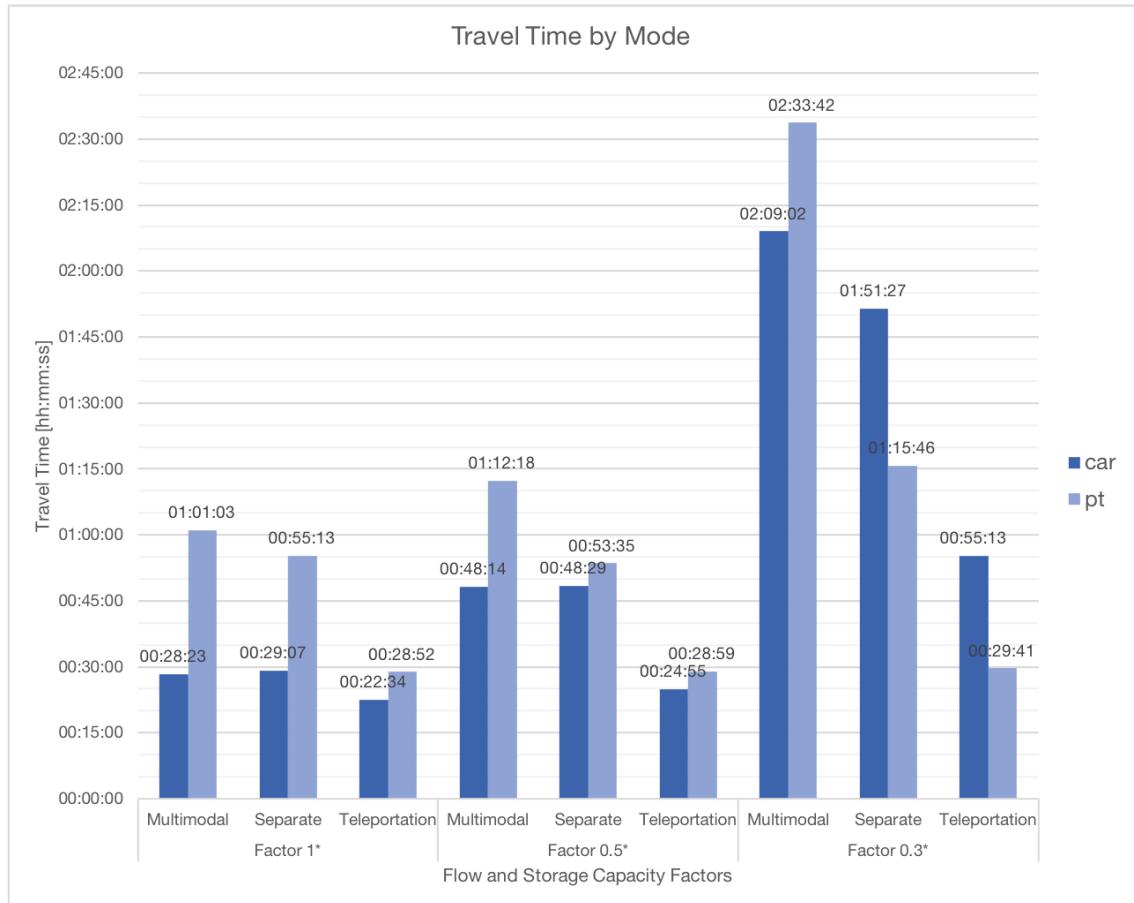


Figure 33 Teleportation scenario with flow capacity factor = 0.5, storage capacity factor=0.6

**Flow Capacity Factor = 0.3, Storage Capacity Factor = 0.4****Figure 34 Multimodal network scenario with flow capacity factor = 0.3, storage capacity factor=0.4****Figure 35 Separate network scenario with flow capacity factor = 0.3, storage capacity factor=0.4****Figure 36 Teleportation scenario with flow capacity factor = 0.3, storage capacity factor=0.4**

### 7.4.2. Travel Time

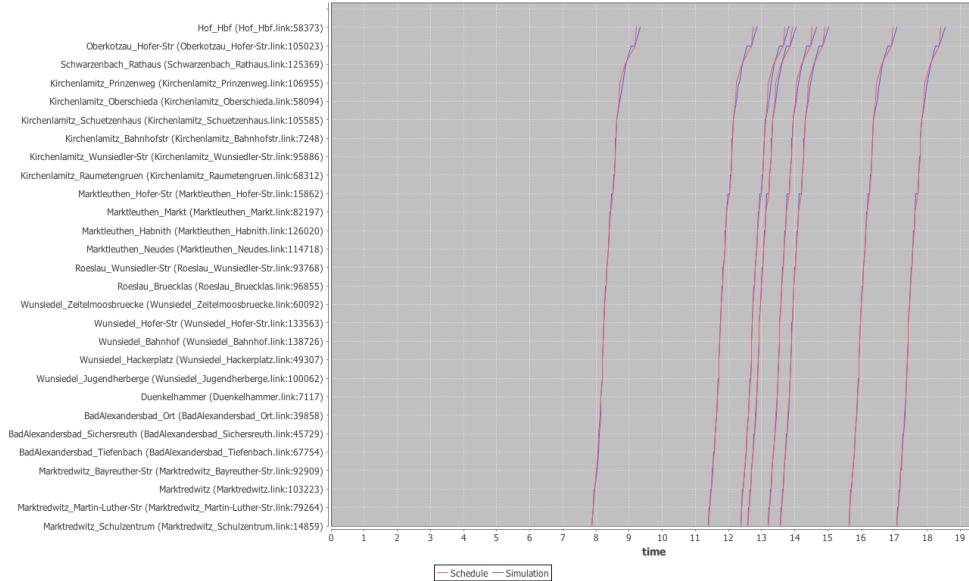


**Figure 37 Average travel times by storage and flow capacity factors of the network.**  
 \* Factors according to 6.4.1

### 7.4.3. Route-Time Diagrams

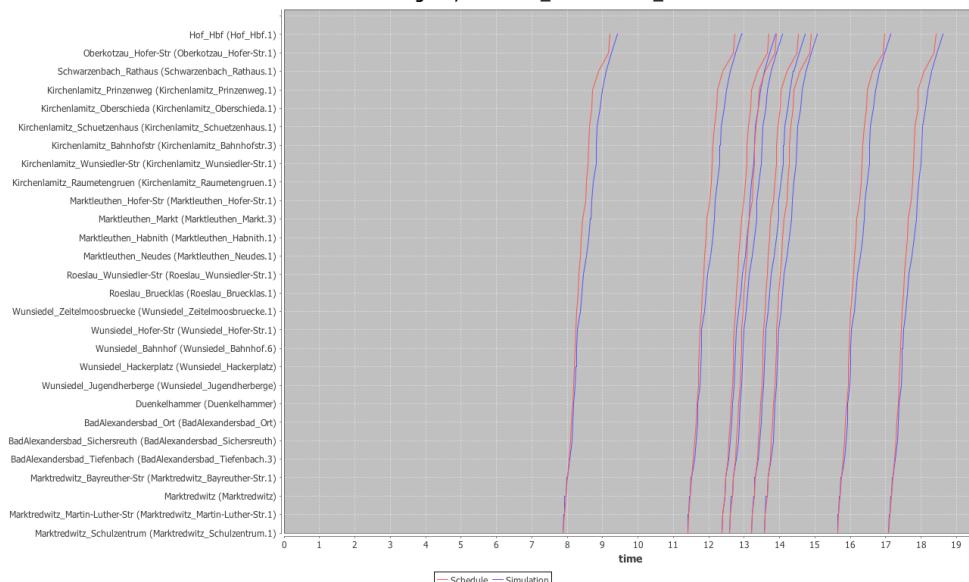
#### Normal traffic conditions

**Route-Time Diagram, Route = 4\_Marktredwitz\_Hof**



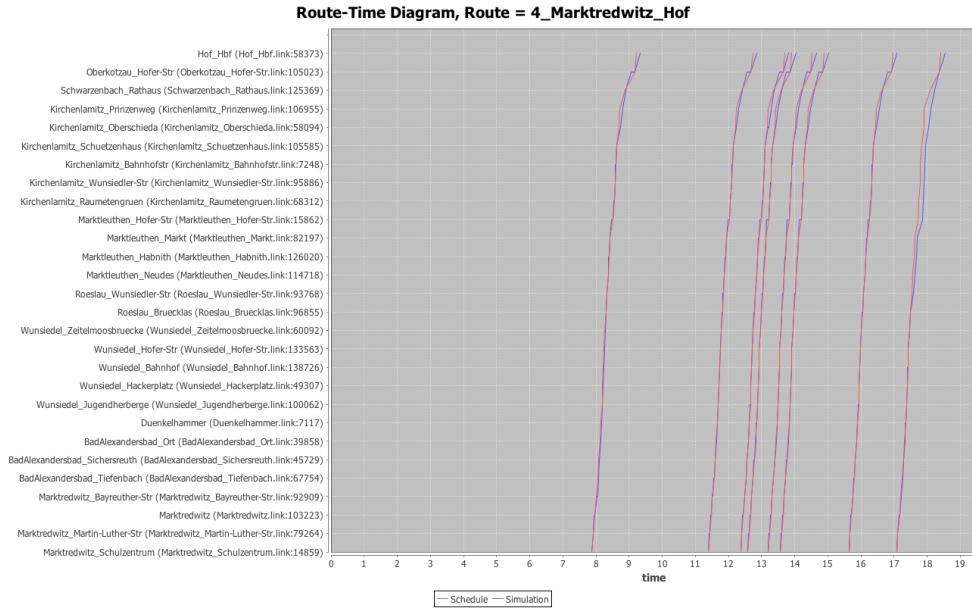
**Figure 38 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof at normal traffic conditions.**

**Route-Time Diagram, Route = 4\_Marktredwitz\_Hof**

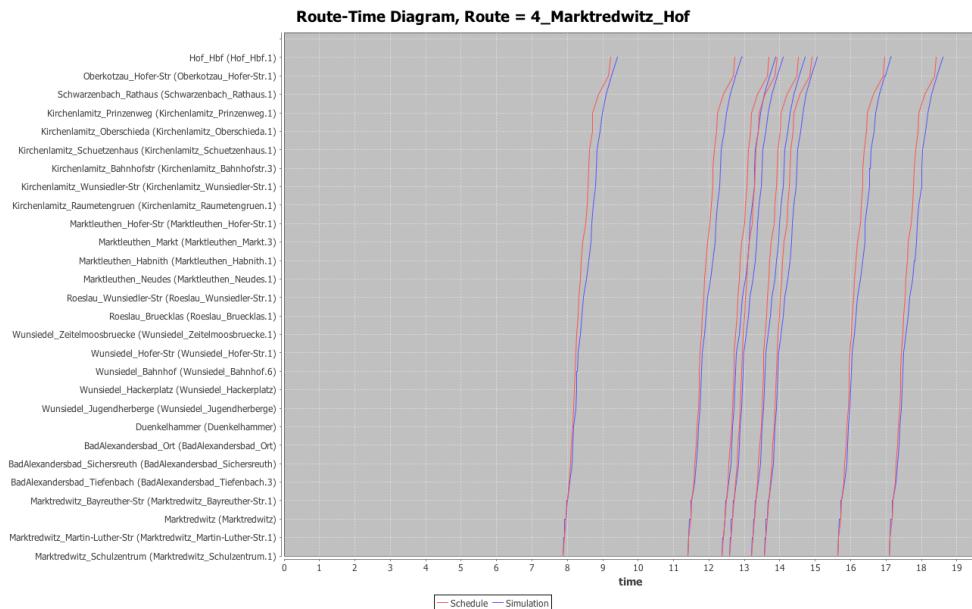


**Figure 39 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof at normal traffic conditions.**

### Flow capacity factor = 0.5, storage capacity = 0.6

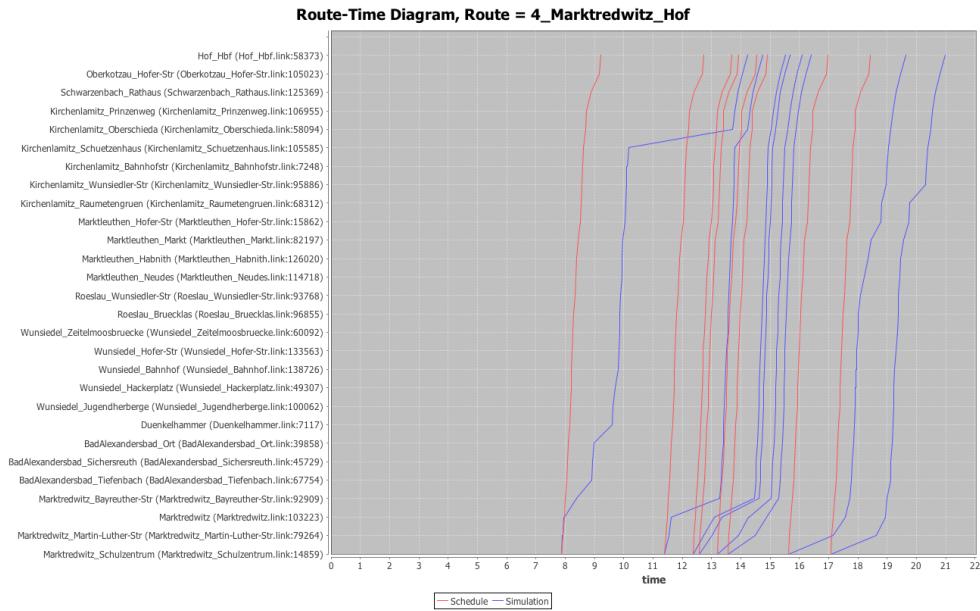


**Figure 40 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.5, storage capacity = 0.6.**

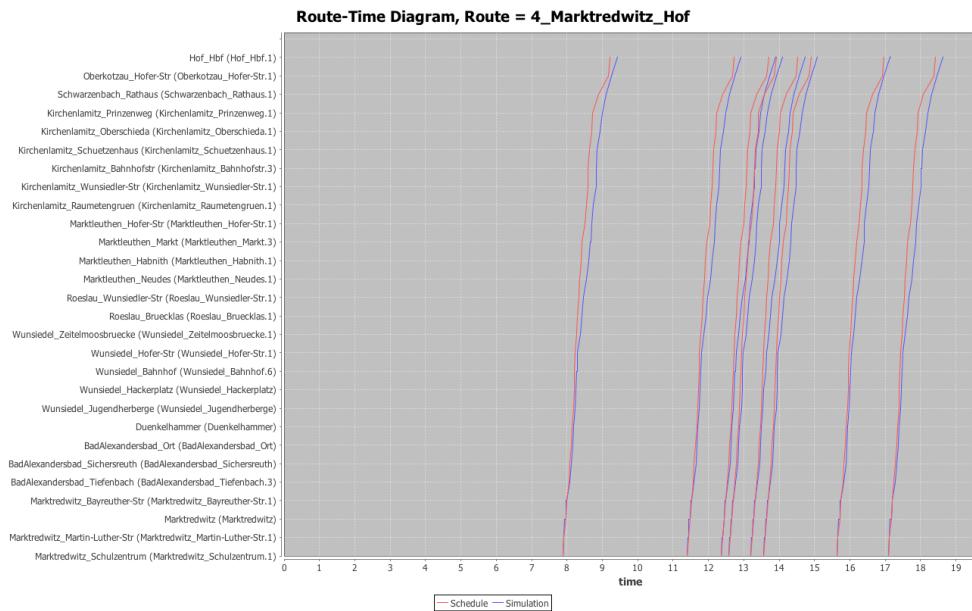


**Figure 41 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.5, storage capacity = 0.6.**

### Flow Capacity Factor = 0.3, Storage Capacity Factor = 0.4



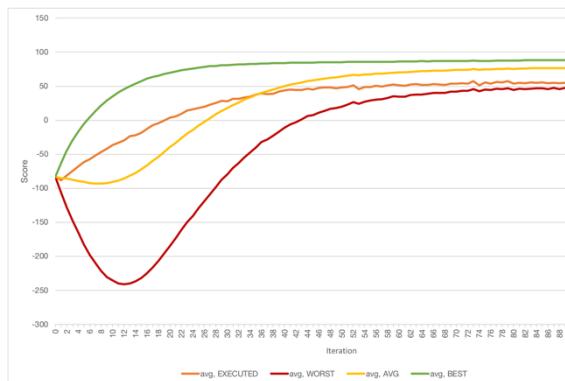
**Figure 42 Multimodal network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.3, storage capacity = 0.4.**



**Figure 43 Separate network scenario: route-time diagram for bus route 4: Marktredwitz to Hof with flow capacity factor = 0.3, storage capacity = 0.4.**

#### 7.4.4. Score Statistics

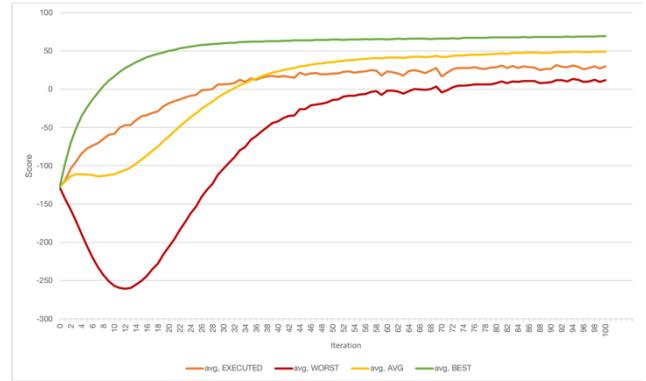
##### Normal Traffic Conditions



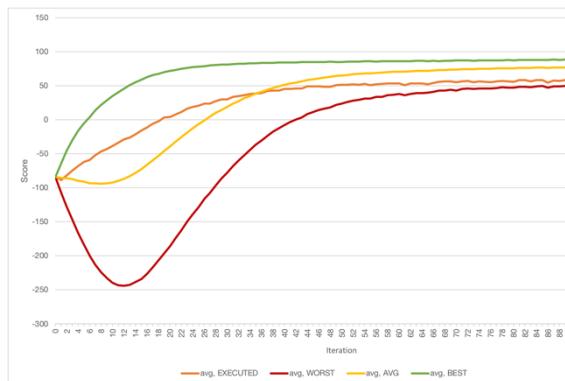
**Figure 44** Score statistics for multimodal network scenario.

##### Flow Capacity Factor = 0.5,

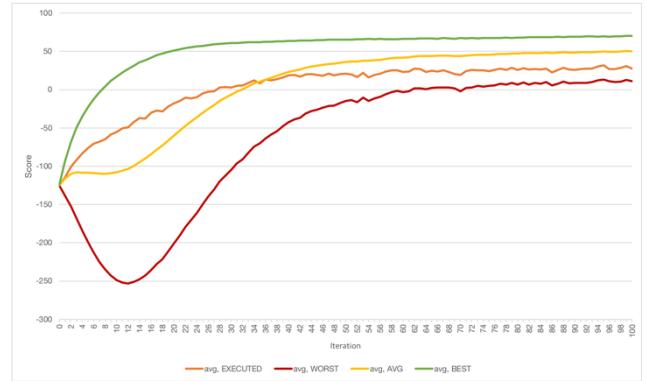
##### Storage Capacity Factor = 0.6



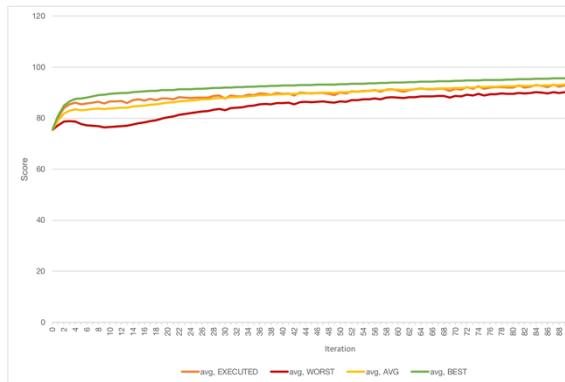
**Figure 47** Score statistics for multimodal network scenario.



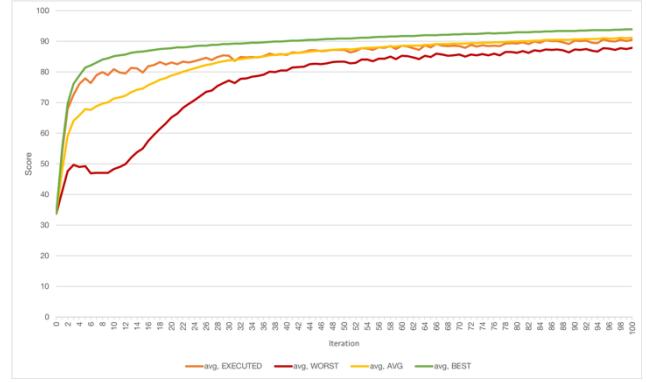
**Figure 45** Score statistics for separate network scenario.



**Figure 48** Score statistics for separate network scenario.

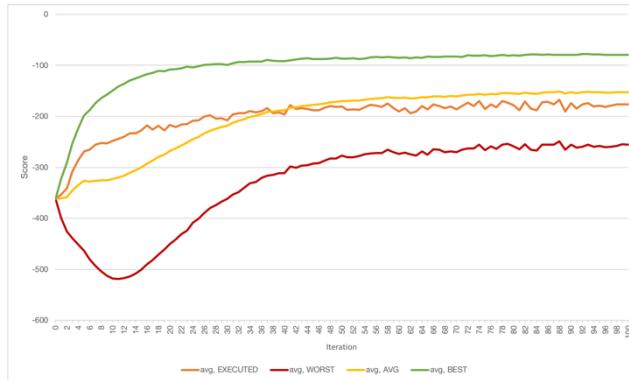


**Figure 46** Score statistics for teleportation scenario.

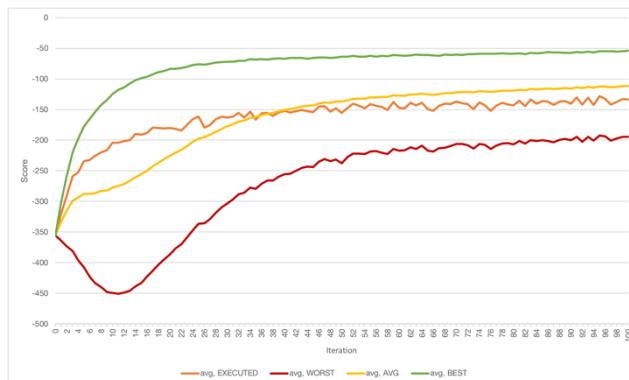


**Figure 49** Score statistics for teleportation scenario.

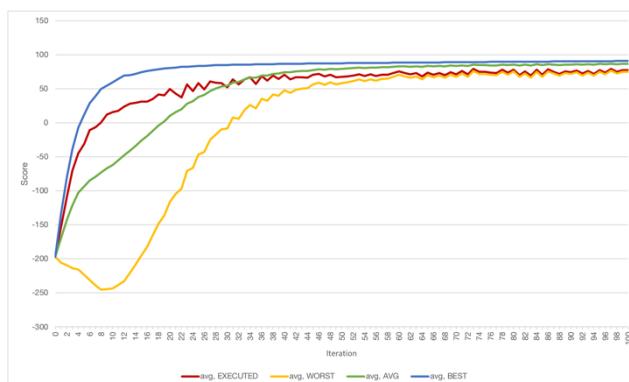
**Flow Capacity Factor = 0.3, Storage Capacity Factor = 0.4**



**Figure 50 Score statistics for multimodal network scenario.**



**Figure 51 Score statistics for separate network scenario.**



**Figure 52 Score statistics for teleportation scenario.**

## 8. Discussion of Results

### 8.1. Performance

As to be expected, the more complex or detailed representations of public transit required longer simulation times per iteration. All teleportation scenarios resulted in shorter simulation times when compared to the other two public transit implementation scenarios for every setting. The simulation times of the multimodal and separate scenarios were mostly similar in normal traffic conditions as seen in the bar chart of Figure 15.

The default public transit router employed in MATSim uses a modified version of Dijkstra's (1959) shortest path algorithm. A graph or network is created from the transit schedule with vertices and edges representing connections between stops. The query times are reflected in the replanning stage of the simulation.

Figure 16 displays the simulation times for 100% population samples with varying storage and flow capacity factors in a bar chart. The bars of factor 1 represent normal traffic conditions. The bars of factor 0.5 and 0.3 represent congestion conditions on the network. The increased complexities of agents optimising path and mode choice in these conditions is reflected in the increased replanning times of the two scenarios. In the extreme case where the storage and flow capacities factors were reduced to only 30%, the multimodal scenario brought up the highest replanning times.

### 8.2. Scalability

The travel times presented in Figure 17 show that for all three scenarios generally equivalent results were generated in both the 10% and 100% population samples. This is especially true for the teleportation scenario. The travel times of agents using pt were longer than those using car for all scenarios. The multimodal and separate network scenarios both brought up longer travel times for pt mode in the 100% sample relative to the 10% sample.

The 1% population sample brought up very different results for all three implementation methods. Notably, the separate network and teleportation scenarios displayed longer travel times for agents using cars. The multimodal implementation still generated longer travel times for pt users and generally longer travel times for all modes.

The route-time diagrams of the separate network scenario display identical patterns for all three population sample sizes. These contain delays between the scheduled and simulated performances.

The multimodal implementation displays delays of up to 30 mins in the simulated performance of the 1% sample. The route-time diagrams of the 10% and 100% population samples bring up equivalent results. The overall delays are far more minimal compared to the separate network implementation in those sample sizes.

### 8.3. Spatial Representation

The distribution of agents who chose pt as their mode of transport for a simulated trip in the multimodal and separate network scenarios present a correlation to the public transit stop locations on the network. The agents of the teleportation scenario as presented in Figure 26, are distributed throughout the entire modelling region with no connection to the locations of the transit stops.

### 8.4. Capacity

To observe the behaviour of each public transit implementation under capacity constraints in the network, two further scenarios were tested in which the storage and flow capacity factors were reduced. These limit the number of vehicles which fit on a link as well as the number of vehicles which are able to leave a link, respectively.

#### **8.4.1. Flow Capacity Factor = 0.5, Storage Capacity Factor = 0.6**

In the scenarios where the flow capacity is reduced to 50% and storage capacity to 60% at a population size of 100%, the mode statistics presented by the multimodal and separate implementations show only very little transformations. In both cases a slight increase in the pt mode share can be observed once the shares stabilize, yet in both cases this is less than 5%. Also compared to each other, the outcomes appear

similar. These observations are also reflected in the score statistics of the simulations. Both the multimodal and separate network scenarios produce very comparable statistics throughout the iterations. Overall, the shape of the curves remain similar to those in the normal traffic states, yet the stabilisation of the scores happens at lower values. Moreover, the range between average worst score and average best scores achieved by the agents becomes larger in both.

Regarding the average travel times the multimodal network scenario shows an increase for both car and pt users. The separate network scenario shows an increase in the car travel time and slight decrease in the pt travel times.

The route-time diagrams show only minor variations between the normal traffic conditions and congestion condition with regards to the schedules and simulated performance of the transit vehicles on the network. The separate network scenario remains seemingly unchanged. The multimodal scenario presents only minor delays.

The teleportation scenario's mode share statistics exhibit major changes under the 50% flow capacity and 60% storage capacity scenario, as seen in Figure 33. The mode shares between car and pt barely stabilize throughout the 100 iterations, however fluctuate around the 50% mark.

The travel times achieved by agents in the teleportation scenarios remain very close to those in normal traffic conditions. The average car travel times increase by less than 3 minutes and the pt travel times only by 7 seconds.

#### **8.4.2. Flow Capacity Factor = 0.3, Storage Capacity Factor = 0.4**

In the scenarios where the flow capacity of the links is reduced to 30% and the storage capacity to 40%, the effects of the capacity constraints are far more observable in all scenarios. For the multimodal network scenario, the share of pt users stabilizes at around 34% and for the separate network scenario at around 39%. The share of pt users in the normal traffic condition scenario is 11%.

The travel times experienced by the agents in the multimodal network scenario increase for both car and pt users by almost 1.5 hours. The pt travel times remain longer than the car travel times.

The effects of congestion are also seen in the route-time diagrams of the multimodal network scenario. All trips in the simulation show significant delays for the route.

This is also reflected in the score statistics on the multimodal network scenario. The average of best plateau close to the -100 mark.

The travel times experienced by the agents in the separate network scenario also increase. The agents travelling by car take up to 1.5 hours longer. The pt users, on the other hand, only experience an increase in the average travelling time of under 30 minutes. The pt travel times are overall less than those of the car users.

The average best scores in the separate network scenario become steady around a score of - 50.

The route-time diagram of the separate network remains seemingly unchanged to those witnessed in the normal traffic condition scenario.

In the teleportation scenario the mode share clearly stabilizes at 80% for pt and 20% for car. The average travel times of pt users increase only by 49 seconds, compared to the normal traffic condition scenario. The average car travel time increases by over 30 mins. This makes the average car travel times longer than those of pt users. In all three observed teleportation scenarios the public transit travel time stayed almost constant. This is due to the way it is defined. The teleported mode free speed factor in this model dictates that pt users will always have a travel time twice as long as cars travelling the same route at free speed.

When looking at the score statistics of the teleportation scenario, the fluctuations at the beginning of the simulation are far greater than in any of the other teleportation scenarios, yet they stabilize at almost 100 by the end of the simulation.

## 9. Conclusion

This thesis presents the varying implementation methods for public transportation in the MATSim simulation framework. This includes the explicit modelling of pt on a fully-multimodal network, on which the concurrent use of private and pt vehicles on the network links are simulated, modelling pt on a separate network to the base car network and teleportation, in which pt is not explicitly simulated and agents are simply removed from one location and placed at another location according to an estimated travel time. These three implementation methods were analysed according to several parameters in order to give insights on the sensitivities of each public transit solution to various modelling aspects. Based on the analysis presented in this thesis, an overview can be gained on the features each public transit solution is capable of providing for given modelling aspects.

The three implementation methods can be assorted based on the levels of realism provided in their representations of public transit in the modelling framework. The least amount of realism is achieved by the teleportation method. Results exhibited that due to the fact that public transit is not explicitly simulated, spatial and temporal characteristics of the public transit system are neglected. Neither the elements of a transit schedule, nor the public transit infrastructure, including the physical geometries and locations of the stops, are incorporated in the modelling process. This was also seen in the distribution of agents using pt as their mode of transport, as they displayed no correlation to the public transit infrastructure. Moreover, operation hours of the public transit services are also disregarded. Closing times or frequencies have no influence on the availability of pt services in the simulation. Overall, teleportation disregards most aspects of accessibility or quality of service indicators of the public transit system. Moreover, any notions of capacities or traffic flow indicators are also not incorporated by this method. The travel times of teleported pt users remained generally unchanged in all traffic conditions. Given the low level of detail, teleportation offers the lowest computation times of all three implementation methods.

Considerably more realism is provided when explicitly simulating public transit on a separate network. This method incorporates both the temporal and spatial proportions not addressed by the teleportation method. The spatial distribution of agents using pt

as their mode of transit in the Wunsiedel scenario demonstrated the correspondence to the public transit infrastructure on the network. As the simulation is based on a transit schedule, temporal elements provided in the transit schedule are explicitly modelled. Moreover, transit vehicles have capacities. The separate network method is not able to represent the influence of the traffic state of the overall transportation system. As the transit vehicles travel on an independent network, interactions with other modes cannot be represented. Therefore, transit vehicles will not be influenced by private car traffic and will also not affect private car traffic through for example, serving stops. This effect was observable in the results presented in this thesis. Despite a capacity overload on the network, pt travel times did not exceed those of cars in the scenario with only 30% flow capacity and 40% storage capacity on the network. Likewise, the route-time diagrams displaying the scheduled and simulated performance of transit vehicles remained unchanged throughout all scenarios, regardless of the traffic state on the network. This is unrealistic as transit vehicles would also be stuck in traffic, which would have effects on travel times and meeting the scheduled departure times of the transit schedule. Given this, the separate network transit implementation method is suitable for all traffic-state independent modelling scenarios or regions with no congestions. This includes for example subway vehicles which travel on separate underground networks, or buses using dedicated bus lanes.

Finally, the most level of realism is offered by the fully-multimodal network implementation method. It encompasses the same spatial and temporal accuracies as the explicit simulation of transit on the separate network and is also capable of incorporating effects of the overall traffic state of the transportation system. Major congestion conditions were reflected in all analysis points discussed in this thesis such as modal statistics, the score statistics of simulated agents, average travel times as well as the ability of the transit vehicles to stick to their schedule times. Also, the fully-multimodal implementation permits the use of more detailed public transit vehicle fleets. Due to that the entire network is connected and available to all modes, transit vehicles can be reused, facilitating the representation of accurate vehicle compositions of public transit agencies. This public transit solution is therefore recommendable for comprehensive studies of transportation systems as it provides the most accurate representation, from the implementation methods observed in this thesis, of not only public transit but also the influence competition for capacity by different modes has on a system-wide level.

Furthermore, the use of a fully-multimodal network improves visualization and simulation credibility.

## Bibliography

Axhausen, K. W. (2000). "Activity-based modelling: Research directions and possibilities." Arbeitsberichte Verkehrs- und Raumplanung **48**.

Balmer, M. (2007). Travel demand modeling for multi-agent transport simulations Algorithms and systems, Eidgenössische Technische Hochschule Zürich.

Balmer, M., et al. (2006). "Agent-Based Demand-Modeling Framework for Large-Scale Microsimulations." Transportation Research Record: Journal of the Transportation Research Board: 125-134.

Bayerische Eisenbahngesellschaft (2018). "Bayern Fahrplan für Bus und Bahn." from <https://www.bayern-fahrplan.de/de/auskunft>.

Bayerisches Landesamt für Statistik (2017). Sozialversicherungspflichtig beschäftigte Arbeitnehmer in den Gemeinden Bayern am 30. Juni 2016. Fürth.

Bayerisches Landesamt für Statistik (2018). Statistik kommunal 2017 Landkreis Wunsiedel i.Fichtelgebirge 09 479. Fürth.

Ben-Dor, G., et al. (2017). MATSim simulations in the Tel Aviv Metropolitan Area: Direct competition between public transport and cars on the same roadway. hEART 2017: 6th Symposium of the European Association for Research in Transportation. Haifa, Israel.

Bhat, C. R., et al. (2013). SimAGENT Activity-Based Travel Demand Analysis, Southern California Association of Governments.

Bösch, P. M. and F. Ciari (2015). A Multi-Modal Network for MATSim. 15th Swiss Transport Research Conference. Ascona.

Bundesamt für Kartographie und Geodäsie (2018). "Verwaltungsgebiete 1:250 000 - Stand 01.01.2018." Dienstleistungszentrum des Bundes für Geoinformation und Geodäsie. from [http://www.geodatenzentrum.de/geodaten/gdz\\_rahmen.gdz\\_div?gdz\\_spr=deu&gdz\\_akt\\_zeile=5&gdz\\_anz\\_zeile=1&gdz\\_unt\\_zeile=14&gdz\\_user\\_id=0](http://www.geodatenzentrum.de/geodaten/gdz_rahmen.gdz_div?gdz_spr=deu&gdz_akt_zeile=5&gdz_anz_zeile=1&gdz_unt_zeile=14&gdz_user_id=0).

Bundesministerium für Verkehr und digitale Infrastruktur (2017). "Mobilität digital Hochfranken - MobiDig." Retrieved November 18, 2018, from <https://www.bmvi.de/SharedDocs/DE/Artikel/DG/mfund-projekte/mobilitaet-digital-hochfranken-mobidig.html>.

Bundesministerium für Verkehr Bau und Stadtentwicklung (2008). Mobilität in Deutschland 2008 Ergenbisbericht. Bonn und Berlin.

Caliper (2019). "TransCAD Transportation Planning Software." Retrieved 05/2019, from <https://www.caliper.com/tcovu.htm>.

Castiglione, J., et al. (2015). Activity-Based Travel Demand Models: A Primer. Washington, D.C., Transportation Research Board

Cervero, R., et al. (2004). Transit-Oriented Development in the United States: Experiences, Challenges, and Prospects. Washington, D.C., Transit Cooperative Research Program (TCRP).

DAT.Mobility (2019). "Transport modelling: OmniTRANS." Retrieved 05/2019, from <https://www.dat.nl/en/solutions/transport-modelling-omnitrans/>.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. Numerische Mathematik **1**: 269-271.

Geroliminis, N., et al. (2014). "A three-dimensional macroscopic fundamental diagram for mixed bi-modal urban networks." Transportation Research Part C **42**: 168-181.

Google (2016). "GTFS Static Overview." Retrieved 05/2019, from <https://developers.google.com/transit/gtfs/>.

HaCon (2019). "HAFAS: Trip Planner und Travel Companion." Retrieved 05/2019, from <https://www.hacon.de/loesungen/trip-planner-und-travel-companion/>.

Hoogendoorn, S. P. and P. H. L. Bovy (2001). "State-of-the-art of Vehicular Traffic Flow Modelling." Proceedings of the Institute of Mechanical Engineers. Part I: Journal of Systems and Control Engineering **215**(4): 283-303.

Horni, A., et al. (2016). The Multi-Agent Transport Simulation MATSim. London, Ubiquity Press Ltd.

INRO (2019). "Emme: The World's Most Trusted Transportation Planning Software." Retrieved 05/2019, from <https://www.inrosoftware.com/en/products/emme/>.

JOSM (2014). "Java OpenStreetMap Editor." from <https://josm.openstreetmap.de/>.

Jürgens, L. and F. Fritsch (2017). "Mobilitätskonzept für den Landkreis Wunsiedel." Bürgerforum Wunsiedel.

Kühnel, N. (2018). Applied Transport Modelling with MATSim, Professorship for Modelling Spatial Mobility, Technische Universität München.

Landkreis Wunsiedel im Fichtelgebirge (2016). Radverkehrskonzept Landkreis Wunsiedel im Fichtelgebirge.

- Macal, C. M. and M. J. North (2010). "Tutorial on Agent-based Modeling and Simulation." Journal of Simulation **4**: 151-162.
- Menendez, M., et al. (2016). NetCap: Intermodale Strecken-/Linien- und Netzleistungsfähigkeit: Schlussbericht 2004/032. 1563, Schweizerische Vereinigung der Verkehrsingenieure und Verkehrsexperten (SVI).
- Nagel, K. (2016). "Can I run MATSim with sample populations, e.g. with 10% of the true population size?". Retrieved 06/2019, from <https://github.com/matsim-org/matsim-code-examples/wiki/faq-48955430>.
- Nuzzolo, A. and U. Crisalli (2009). The Schedule-Based Modelling of Transportation Systems: Recent Developments. Schedule-Based Modelling of Transportation Networks: Theory and Applications. N. H. M. Wilson and A. Nuzzolo, Springer. **46**: 1-26.
- OECD (2011). Regional Typology, Organization for Economic Cooperation and Development.
- OpenStreetMap (2018). "OpenStreetMap Overpass API." Retrieved August 8, 2018, from [https://wiki.openstreetmap.org/wiki/Overpass\\_API](https://wiki.openstreetmap.org/wiki/Overpass_API).
- OpenStreetMap contributors (2019). Planet dump retrieved from <https://planet.osm.org>.
- OpenStreetMap Wiki (2017). "About OpenStreetMap." Retrieved 05/2019.
- OpenStreetMap Wiki (2017). "Elements." Retrieved 05/2019, from <https://wiki.openstreetmap.org/wiki/Elements>.
- Ordóñez, S. and A. Erath (2011). Semi-Automatic Tool for Map-Matching Bus Routes on High-Resolution Navigation Networks. 16th International Conference of Hong Kong Society for Transportation Studies. Hong Kong, Hong Kong Society for Transportation Studies. **727**.
- Ortúzar, J. d. D. and L. G. Willumsen (2011). Modelling Transport. Chichester, West Sussex, John Wiley & Sons Ltd.
- Peeta, S. and A. K. Ziliaskopoulos (2001). "Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future" Networks and Spatial Economics **1**(3-4): 233-265.
- Poletti, F. (2016). Public Transit Mapping on Multi-Modal Networks in MATSim. Institut für Verkehrsplanung und Transportsysteme. Zürich, Eidgenössische Technische Hochschule Zürich. **Master**.
- Poletti, F. (2019). "PT2MATSim." Retrieved 06/2019, from <https://github.com/matsim-org/pt2matsim>.

PTV Group (2019). "PTV Visum." Retrieved 05/2019, from <https://www.ptvgroup.com/de/loesungen/produkte/ptv-visum/>.

Quddus, M. A., et al. (2007). "Current map-matching algorithms for transport applications: State-of-the art and future research directions." Transportation Research Part C **15**: 312-328.

Rasouli, S. and H. Timmermans (2014). "Activity-based models of travel demand: promises, progress and prospects." International Journal of Urban Sciences **18**: 31-60.

Rieser, M. (2010). Adding Transit to an Agent-Based Transportation Simulation: Concepts and Implementation Faculty V Mechanical Engineering and Transport Systems. Berlin, Technical University of Berlin.

Rieser, M. (2016). Modeling Public Transport. The Multi-Agent Transport Simulation MATSim. A. Horni, K. Nagel and K. W. Axhausen. London, Ubiquity Press: 105-110.

Rieser, M. (2018). "Class CreatePseudoNetwork." MATSim API Specification. Retrieved June 3, 2018, from <https://www.matsim.org/apidocs/core/0.4.0/org/matsim/pt/utils/CreatePseudoNetwork.html>.

Rieser, M., et al. (2018). Adding Realism and Efficiency to Public Transportation in MATSim. 18th Swiss Transport Research Conference. Monte Verità, Ascona.

Rieser, M., et al. (2016). Generation of the Initial MATSim Input. The Multi-Agent Transport Simulation MATSim. Andreas Horni, K. Nagel and K. W. Axhausen. London, Ubiquity Press: 61-64.

Simunto (2018). "Simunto Via." Retrieved July 12, 2018, from <https://www.simunto.com/via/>.

Smith, L., et al. (1995). TRANSIMS: TRansportation ANalysis and SIMulation System. 5th TRB National Transportation Planning Methods Applications Conference. Seattle.

SVI (2017). "Schweizerische Vereinigung der Verkehrsingenieure und Verkehrsexperten ". Retrieved 07/2019, from <http://www.svi.ch/web/index.php?id=4&L=36>.

TransitLand (2019). "Feed Registry." Retrieved 05/2019, from <https://transit.land/feed-registry/>.

Verkehrsgemeinschaft Fichtelgebirge (2018). "VGF." from <http://www.vgf-web.de/startseite/>.

Wahba, M. and A. Shalaby (2008). MILATRAS: a new modeling framework for the transit assignment problem. Schedule-Based Modelling of Transportation Networks: Theory and Applications, Springer. **46**: 171-194.

Zilske, M. and N. Kühnel (2019). "GTFS2MATSim." Retrieved 06/2019, from <https://github.com/matsim-org/GTFS2MATSim>.

Zilske, M., et al. (2011). OpenStreetMap for traffic simulation. 1st European state of the map : OpenStreetMap conference. Wien, OpenStreetMap Austria u.a.: 126-134.

## **Statement of Independent Work**

I hereby confirm that this thesis was written independently by myself without the use of any sources beyond those cited, and all passages and ideas taken from other sources are cited accordingly.

Munich, July 23, 2019

---

**Katharina Mehlstäubler**

## Appendix A

### VGF Network Plan

