MASTER

Multi-agent dynamic traffic assignment incorporating GPS diary and personalized supernetwork approach

Li, C.

Award date:
2016

Multi-agent dynamic traffic assignment incorporating GPS diary and personalized supernetwork approach

Author

Graduation candidate	Chaoyu Li
Student number:	0926551

Graduation committee

prof.dr. ir. B. (Bauke) de Vries	Chairman Master Program CME, TU/e
dr. Q. (Qi) Han	First supervisor, TU/e
dr. T.(Tao) Feng	Second supervisor, TU/e
ir. A. J. (Joran) Jessurun	Third supervisor, TU/e

Institutional Information

University	Eindhoven University of Technology
Faculty	Faculty of the Built Environment
Program	Construction Management & Engineering

Contents

List of tables & figures

English summary

Mobility issue is a serious concern in urban areas. Though a shift in transport mode choice can be seen in recent years from motorized vehicles to slow modes, i.e. cycling and walking, road congestion still frequently occur, indicating that a traffic planning system considering multi-modal transportation is demanding. Research efforts trying to replace the traditional four-step traffic planning models have been put into this context. The integrated activity-based modeling (ABM) and dynamic traffic assignment (DTA), looking at the issues from demand and supply sides respectively, offer an opportunity to improve the predictions in a high resolution

However, the cost functions adopted in the assignment are typically trip-based or tour-based, which fails in representing the multidimentional characteristics of individuals' activities and trips. In reality, however, individuals make a decision on a sequence of activities and trips at different time and places. For instance, a shopping location choice may involves the joint choice with parking places. Considering this issue from a perspective of full activity-travel programs, the degree of easiness (accessibility) to conduct the multiple activities in space-time needs to be taken into account in the assignment. This triggers the necessity to investigate: to what extent the integrated accessibility can reflect the true heterogeneity in choice preference and influences the results of dynamic assignment.

Integrated accessibility in space-time, extended from the spatial and general-cost based accessibility, has been increasingly discussed in recent in transportation and geography. Relative to the unimodal based accessibility indicators, the integrated accessibility provides insights to the evaluation of tours and/or full activity-travel programs in a comprehensive way. The latest breakthrough of such a concept is the supernetwork representation proposed by Arentze and Timmermans (2004) and extended by Liao et al (2011). They decomposed multi-modal transportation network into private vehicle networks (PVN) and public transport networks (PTN), based on the individual's activity program. The links interconnecting PTN or PVN represent unimodal travel, and those connecting PVN and PTN in the same activity state represent transport mode change; links between PTN from different activity states then represent conducted activities. This representation is capable of capturing the activity-travel pattern of heterogeneous individual at a high level of detail, without adding much computational burden.

The recent work by Liu et al. (2015) proposed an analytical model incorporating this approach. They formulated the discrete-time DUE condition as an equivalent path-based variational inequality (VI) problem, and adopted an iterative algorithm in MATLAB to realize the assignment. The model, however, is quite complicated in the sense that a list of unrealistic assumptions had to be made to ensure its convergence, e.g. the individual has complete knowledge of all utility and travel time. Agent-based simulation should be a better

choice instead, considering its high consistency with the supernetwork concept and capability of capturing behavioral realism. This research, therefore, aims at incorporating this supernetwork representation with simulation-based dynamic traffic assignment approach, further exploring the applicability of this model.

In order to incorporate the dynamic accessibility into the DTA, the utility function will be adjusted using the integrated scoring of individuals' activity-travel program (ATP). The parameters estimated using a mixed logit model based on a stated-choice experimental data, will be adopted to predict the space-time accessibility. The next step is to generate *plans* (ATP) for each *agent*, which was done using GPS data from Eindhoven, in combination with a data validation procedure to prepare the data for MATsim use. Based on these diary data, ATPs are then assigned to each *agent*, serving as input for simulation. The simulation procedures are carried out with a modified *Qsim*, that is, a queue-based time-step simulation module. Simulation of public transport with a given schedule is already enabled in *Qsim*, and the no-motorized activities are incorporated via adding a multi-modal contribution to each link object and using priority queues to order the agents based on their scheduled link leave time, in other words, teleportation.

The proposed approach is implemented and validated through the simulation using data from Eindhoven, The Netherlands, by comparing with the results using Charpar-Nagel cost functions. The incorporation results suggest that GPS data improves the behavioral realism MATsim captures in terms of initial plans with higher complexity and variety, which might add some extra computation burden. The trade-off is acceptable considering the fact that large-scale simulation would already take a long period. On the other hand, the comparison results indicate that the modified utility function is more suitable for multi-modal scenario, as the discrepancy between the initial performance of agent plans and the optimized ones become smaller, leading to a quicker convergence. Superiority of the proposed approach is thus confirmed from both the methodological and practical aspects. With the simulation results, this combined approach proves to be a promising solution for transportation planning.

However, some further studies are required to further validate its applicability. A primary improvement would be incorporating the GPS data with population synthesis, using iterative proportional fitting to generate a 20% synthetic population of the study area, which was not completed due to time and device limitation. Other improvements include the simulation of parking activities and joint decision simulation at household level, which would involve large-scale household level GPS data collection and extra coding effort within MATsim platform. Another attempt can be made to incorporate the integrated accessibility concept with *en route* adjustment, which is not applicable at this moment due to the limitation that such simulation is not enabled yet for iterative procedure.

Abstract

This thesis aims at incorporating the integrated space-time accessibility concept from supernetwork structure into the framework of simulation-based dynamic traffic assignment (DTA), trying to explore its effectiveness.

A methodology is proposed first, using multi-modal simulation to reproduce both motorized and non-motorized trips of real-life individual, together with a one-year GPS diary data as source for travel demand input, enhancing the resolution of detail this demand can capture. The simulation is then conducted using a multi-agent approach (MATsim), maximizing the benefits of representing the heterogeneous choice preferences of individuals in multidimentional activity and travel programs. Based on the estimates of a mixed logit model, the proposed approach is operationalized via modifying the utility functions for evaluation of individual's plan in response to the integrated accessibility concept. The supernetwork structure is also partially reproduced with a multi-layer network consisting of private vehicle network and public transit network.

Following the preparations above, the approach is eventually implemented by taking Eindhoven region as a study area. Results suggest that this integrated approach is a promising solution for transportation planning, and possess high applicability in other scenarios as no major adjustment is required. Future works include parking activity simulation and household joint-decision simulation, and incorporation with population synthesis for large agent population.

ACKNOWLEDGEMENT

Before presenting my report, I need to mention several persons that contributed greatly to the completion of this thesis in the past six months. First I would like to express my sincerest gratitude towards dr. Feng, my direct supervisor, who provided me with empirical guidance and precious advice since research proposal period. Without his help, there lies no possibility that I could go through all the obstacles by myself. I am also very grateful to dr. Han, my 1st supervisor, who gave me consistent encouragement during my graduation project, leading me to the final line with care and patience. I would also like to thank ir. Jessurun, as he helped me overcome several severe hindrances I encountered with expertise and offered a unique point of view. Special thanks to prof. Bauke de Vries for your generosity that allowed me to complete the thesis with enough timespan to keep everything explicit.

1. Introduction

1.1 Problem definition

Road traffic congestion, resulting from the imbalance between the supply of and the demand for transportation facilities, has been a global concern for urban areas since ancient times (Falcocchio and Levinson, 2015). In US only, the estimated economic loss of congestion reached 121 billion dollars in 2011, and witnessed an increasing trend since 1982 (Schrank et al, 2012). Reducing congestion occurrence thus is an urgent issue within the context of transportation planning, and the solution measures mostly fall into two categories: Either by building new traffic infrastructure to extend the capacity of road systems, or making efficient use of existing transport networks, the latter of which is more preferred today (Illenberger et al, 2008). A popular concept rose in recent years depicting such thought, is "Smart Mobility", which looks for possible solutions to traffic bottlenecks, from various perspectives such as traffic management, network design and mobility management (TNO).

In Netherlands, the urban traffic is also problematic, due to the increasing road traffic in major cities such as Amsterdam and Rotterdam. In recent years, a shift can be seen in transport mode choice from personal vehicle and public transport towards bicycle and walking. Such trend, however, end up causing bicycle congestion in some areas with high population density instead of relieving road pressure as expected (Rijkswaterstaat, 2015), which was not predictable with traditional transportation demand prediction methods, as they often focus merely on vehicular traffic and ignore the impact of slow-mode trips (Wong and Sezto, 2012). This indicates that a smarter traffic management system is needed to solve the mobility issue today, considering all possible aspects of individual's transportation mode choice. The solution can be two-folded: An advanced multi-modal transportation planning model with non-vehicular travel mode choice assignment to realistically evaluate regional travel performance, making efficient use of existing road network from a policy view; and a real-time traffic information provision system that collects sensor data, analyzes current network state and predict future flows with accuracy, offering travel guidance to individual users.

The development of such smart traffic management system has become an eager calling in current Intelligent Transportation Systems (ITS), requiring sophisticated models to represent real-life traffic networks, and complicated solution procedures that might easily lead to overwhelming computational effort (Peeta and Ziliaskopoulos, 2001). How to realize such deployment is a remaining issue, and has been widely discussed in various studies in transportation field. The motivational foundation of this research, therefore, is to help finding an answer to the planning part of the question, using a dynamic traffic simulation approach.

Based on these facts listed above, the problem to be solved here can be further defined as seeking a solution to realize a realistic multi-modal transportation planning method.

1.2 Research question

The main research question proposed here, therefore, can be concluded as:

How to improve the methodology of multi-modal transportation planning?

This question can be further broken down into several sub-questions:

1) What is the state-of-art dynamic traffic assignment approach within transportation planning context?

This question requires a thorough review of the current approach for transportation demand prediction, which can be found in Chapter 3.

2) Considering the characteristics of the multi-modal representation, what is the state-of-art practice platform?

To answer this question, the involved technique will be briefly introduced in Section 1.3.1 in order to give an initial definition of the methodology. This will be followed by a detailed reasoning backed up with rigorous logical binding in Chapter 4, after a thorough literature study.

3) What type of data will be required for the chosen method? How to process the data needed for the case study?

The data requirement as well as processing procedure will be explained in Chapter 4, and will be further illustrated in the case study described in Chapter 5.

4) What is the advantage of the proposed methodology?

This question will be answered throughout the following part of the thesis, as the majority of this paper is trying to demonstrate the chosen method with practical and technical detail. To be more specific, Chapter 3 and 4 will address the theoretical advantage of the model, which will be implemented in Chapter 5 together with an advanced data source applied in this thesis.

1.3 Research design

1.3.1 Methodological justification

The current trend in transportation planning has led to a combination of activity based modeling (ABM) for demand forecasting and dynamic traffic assignment (DTA) for traffic simulation (Chiu et al, 2011; Shiftan and Ben-Akiva, 2011) in replacing the traditional four-step model.

Unlike conventional trip-based models which only focus on vehicle movements, activity-based models believe that travel is derived from the necessity to participate in activities. Therefore, instead of trips focusing on destination and transport modes, ABM generates trip chains at an individual or household level, trying to predict the timing and sequence of activities and associated trips based on some behavioral principles (Rasouli and Timmermans, 2014).

No matter with approach is used, the Original-Desitnation (OD) matrix generated serves a basis for traffic simulation where a traffic assignment module is in general necessary to assign the traffic demand generated at zone levels into the road networks.

Traditionally, this was done by the so-called static assignment procedure where a user equilibrium is normally assumed. Basically, static traffic assignment, as is applied by default in the four step model, generates traffic volume on each link directly based on the O-D matrix, and calculates the route travel time as a sum of link travel times on the route (Chiu et al, 2011). On the other hand, dynamic traffic assignment is a generalization of such traditional assignment. The process consists of two main components, namely *travel choice* and *traffic flow*. The former component determines the traffic flow level on each road at each instant, given network performance represented as time-varying travel times, whereas the latter component describes how vehicular traffic spreads within a network, influencing network performance. The output of the *travel choice* therefore becomes the input of the *traffic flow*, vice versa. DTA then decides the flow pattern that satisfies the two components simultaneously, sometimes even targeting at reaching a dynamic user equilibrium (DUE) established for each departure time (typically ranging from a few seconds to several minutes) instead of the entire analysis period. Compared with static assignment, DTA proves to be superior in that traffic dynamics are better captured (Wong and Szeto, 2012).

Current DTA models, however, provide limited choices serving as solution to ABM demand forecast, as they usually adopt cost functions to find shortest path for traffic optimization, which are typically trip-based or tour-based, failing in representing the multidimentional characteristics of individuals’ activities and trips (Kachroo and Shlayan, 2013). Considering this issue from a perspective of full activity-travel programs (ATP), the degree of easiness (accessibility) to conduct the multiple activities in space-time needs to be taken into account in the assignment. This triggers the necessity to investigate: to what extent the integrated accessibility can reflect the true heterogeneity in choice preference and influences the results of dynamic assignment.

To this end, personalized supernetwork concept seems to be a profound representation serving such purpose. As the latest breakthrough within ABM context, it formulates the network as separate layers connected by transition links (access, egress and parking, etc.) and bounded by individual's limited activity-travel experience, captures the ATP of real-life person at a high level of detail. Though this representation is theoretically sound, reproducing the structure in a large-scale scenario would call for various simplifications, which will be addressed in this thesis.

Therefore, in this research, the methodology adopted will concentrate on operationalizing the theoretical and process model of the proposed transportation planning measure. The main will be integrating the personalized supernetwork approach and agent-based simulation. The major takes therefore are to identify the suitable models and investigate the way to implement them, addressing the contribution of a practical and easily-applicable solution that better captures the heterogeneity of various individuals. This largely relies on the correct choice of a tool package that can both carry out the required representations and prove the simulation to reach convergence in a recognizable way, restricting the options available here. Considering the software requirements and functionality, the software package, MATsim (Multi-agent transportation simulation), dedicated to agent-based modeling is chosen.

Another exploration in this thesis is the use of GPS diary for travel demand forecasting instead of traditional synthetic population derived from national travel surveys (NTS). The traditional source data are usually obtained with nationwide questionnaire surveys where participants are asked to recall their ATP on a specific day, which has been considered inaccurate and unable to capture day-to-day relations (Grengs et al, 2008). GPS diary, on the contrary, is advantageous for being able to capture detailed ATP of each participant during a continuous time period, which should be especially meaningful for multi-agent representation. However, similar attempts are not well applied yet, due to the shortcoming of the newly-developed data source, namely the limitation of sample size and validation accuracy, as well as complexity of conversion (Shen and Stopher, 2014). Therefore, in this thesis, the applicability of GPS data will be specifically investigated. The methodology will be further illustrated in Chapter 4, and implementd with a case study in Chapter 5.

In summary, the main contribution of this thesis lies in two aspects: Seeking a dynamic multi-modal traffic assignment solution incorporating supernetwork representation, and further explore its applicability within the smart mobility context, specifically for traffic planning part; integrating GPS diary with multi-agent DTA approach, testing the applicability of this new data source for large-scale multi-modal scenario.

1.3.2 Process design

The design of the research is a straightforward linear process, as is shown in Figure 1.1:

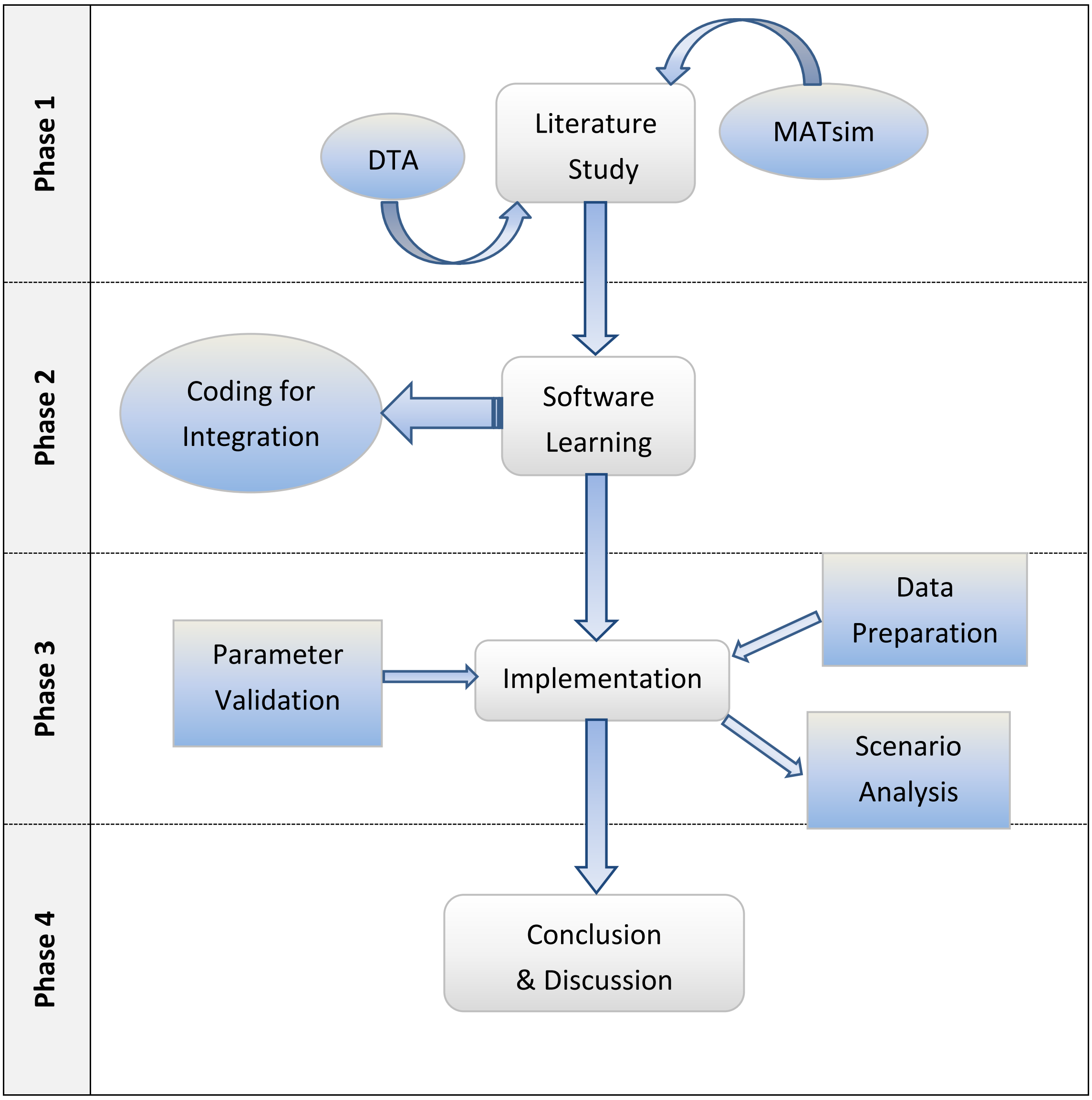


Figure 1. 1 Research design

The process is composed by four major phases:

Phase1: Conducting a literature study related to DTA and MATsim implementation, settling the theoretical foundation of this research. This has been mostly finished following the completion of research proposal, and gradually updated during the learning process. Chapter 3 will present the outcome of this procedure.

Phase2: Familiarizing with MATsim functions with the help of given examples and coding modules for the generation and integration of data from various sources, which would call for intensive programming effort. This is a vital precondition to Phase 3, and the original integration codes are attached in Appendix 2.

Phase3: Carrying out a case study located in Eindhoven region as the implementation of the proposed method. Parameter validation is done via adopting established result from previous studies and modifying it to the current scenario, which will be explained in Chapter 4. Data preparation step proves to be the toughest part of the research, since it involves processing raw real data that requires complicated conversion procedures before any actual use and will very easily cause error. This will be presented in details from Chapter 5.2 to 5.4. Following the completion of preparation, a scenario analysis is conducted comparing the simulation results between the scenarios with and without the supernetwork disutility integration, providing an evidence for the validation of results of the proposed method, as can be seen from Chapter 5.5 to 5.7.

Phase4: Concluding the research with a fully composed report, and giving some advices on future studies as well, as more contents will be found out during the research that requires extensions to the current methodology.

1.4 Research objectives and limitations

1.4.1 Objectives

Initiative: Improving current simulation-based DTA method with advanced concepts. The integrated space-time accessibility concept depicted from supernetwork representation complements the theoretical shortcoming of current DTA models capable of solving ABM formulations, since it captures high resolution behavioral reality without adding significant burden to computational effort, as has been proven in latest study with an analytical solution. This advanced concept, however, has not yet been incorporated with simulation-based approaches that are more promising due to higher compatibility with ITS deployment. The ultimate goal of this thesis, therefore, is to find/develop a simulation DTA approach capable of implementing the integrated accessibility concept within supernetwork frame.

Application: Developing an integrated method that is highly applicable. While the supernetwork representation involves multidimentional choice of trips and activities, a complete reproduction of the structure does not prove to be viable in simulation models, thus simplification is required regardless of the model used here. Agent-based simulation is the most suitable platform for such integration, especially the multi-agent representation from MATsim, which has most potential to reproduce the structure due to its individual agent resolution and capability of simulating bounded rationality. Thus a secondary goal of this study is to search a practical measure to implement the integration of methodologies.

Exploration: Testing the incorporation of advanced data source-GPS data. Being able to capture high resolution detail of participants' daily schedule, GPS diary seems to be a promising source for transportation planning studies as a replacement of travel survey.

Various studies have examined the validation measures to convert the raw GPS points into schedule records, but the incorporation of such outcomes with simulation-based approaches is not well addressed yet. Consequently, another goal of this thesis is to overcome the challenge of GPS data post-processing and incorporation with DTA, as an input for demand generation from a large geographical scale.

1.4.2 Limitations

The activity-travel program generated in this research is a typical one-day trip chain, while in fact individual's ATP is usually influenced by his/her weekly schedule (Schlich and Axhausen, 2003). Therefore in this project, it is also possible to consider multi-day scenarios, serving as an optional goal; however, this is disabled due to the limited sample size of GPS diary data.

Due to the fact that research effort hasn't been put into integration of ABM and agent-based measures (Dubernet and Axhausen, 2015), and several simplifications have to be made to reduce system complexity and computational burden, the following limitations are produced:

In real-life cases, individual's ATP is not only determined by himself/herself, but also influenced by the schedule of his/her family members; for example, the activity of shopping can be done by an individual that represents the whole family, thus should not be present in other member's schedule for a period of time. Attempts have been made to model such phenomenons (Gliebe and Koppelman, 2005). However, such effort will add extra complexity to the simulation process, also requiring more detailed data that is not easy to obtain from a large scale. This household level scenario therefore will not be taken into consideration, reducing the reality of the simulation to some extent.

The second limitation is related to the *en route* adjustment. That is, real-life individuals can sometimes adjust their plan halfway, resulting in an ATP different from assigned schedule, which was studied in the works of Mounce and Carey (2011). For complexity reason, this will be excluded as well.

Another phenomenon that frequently occurs in daily life is multi-person joint travel, explored by Liao et al (2013a). Similar to the reasons mentioned above, scenario of this level will not be considered.

1.5 Expected results

This simulation can generate various outcomes for further analysis:

1. If the time step of the MATsim model is set to one second, a dynamic assignment result can be visualized, offering a straightforward view.

2. A comparison can be conducted between the first iteration result and the final result, checking the effectiveness of this assignment procedure.

3. A scenario analysis can also be conducted, comparing the simulation result applied with supernetwork approach and one obtained via default settings, proving the superiority of the method.

1.6 Reader's guide

The content distribution of the following chapters of this thesis will be presented as follow:

To guarantee explicit illustrations, Chapter 2 will first give brief definitions of the concepts most relevant to problem formulation and research question. Chapter 3 will then conduct a thorough literature study regarding the proposed methodology, which will be followed by a detailed explanation of the research model in Chapter 4. The model will be implemented in Chapter 5 with a case study, and the results will be discussed in Chapter 6, concluding the whole thesis. Reference and appendixes will be attached at the end of the thesis.

2. Glossary

This chapter provides brief definitions of the concepts mentioned in the research question and problem formulation segments, assisting the reader to clearly comprehend the purpose of the thesis. The other concepts relevant to methodology will be explained in details in Chapter 3.

2.1 Traffic congestion

Road traffic congestion, resulting from the imbalance between the supply of and the demand for transportation facilities, is a global concern for urban areas. This phenomenon disrupts traveler/driver's daily schedule of business and family activities, reducing their job opportunities or productivity (Cambridge Systematics, 1994), and causes physical and mental stress as well (Wiesenthal and Hennessy, 1999). External costs that usually do not affect people directly should also be taken into consideration, such as air pollution emissions, noise, space consumption, fuel consumption, vehicle maintenance, etc (Bigazzi and Figliozzi, 2013). One initiative of this thesis is to use a more advanced transportation planning approach to help solving it instead of enhancing the capacity of existing road network, which normally would demand large amount of investment and a long period of time to finish, and bring inconvenience to the residents in the neighboring areas.

2.2 Smart mobility

This concept, proposed by TNO, emphasizes the efficient and sustainable transport of people and goods, featuring smart organization of traffic and smart use of advanced technology to assist it. It has raised interest of various business companies and research agencies, and supported by the Ministry of Infrastructure and the Environment in the Netherland, gradually becoming a core transportation research and application concept within Dutch context. It provides a universal solution direction for various transportation researches related to mobility issue.

2.3 Intelligent transportation systems (ITS)

According to the definition in EU Directive, Intelligent transportation systems (ITS) are advanced applications that incorporate information and communication technologies (such as sensors, cameras as data collection tools) with transport engineering, aiming at constructing safe, coordinated and "smart" road networks with high efficiency and environmental performance. Its incorporation with DTA mainly lies in the collection and process of real-time traffic data for flow information provision. In this research, GPS devices were mainly involved to collect diary data for transportation planning purpose.

2.4 Activity-based model (ABM)

Activity-based models are travel-demand forecast models founded on the argument that transportation routes do not exist for their own sake, but provide people with access to their desired activities. In contrast to trip-based models, ABM work at a disaggregate person-level rather than aggregate zone-level, and are advantageous for their explicit representation of space-time constraints and capability of capturing linkages among activities and travel, therefore would reproduce more realistic travel choice preference of real-life individual or household (Castiglione et al, 2015).

2.5 Dynamic traffic assignment (DTA)

Dynamic traffic assignment is a traffic assignment procedure that takes time-variant traffic flow into consideration, as a generalization of static traffic assignment. DTA is generally considered difficult due to huge number of routes involved and changing route set choices. Despite its difficulty, DTA is of practical importance, because it can be applied for offline transportation planning and policy evaluation, as well as real-time traffic management, which are believed to be the eager demand of current ITS (Szeto and Wong, 2012).

2.6 Data sources: National travel surveys (NTS) & GPS diary

Nation travel surveys are typical Dutch data source for transportation researches. They collect the activity and travel schedules of around 3% of the total population of a specific day with questionnaires/online surveys. The accuracy of this source is often doubtable because the data largely relies on the memory of participants.

GPS diary is an advanced traffic data collection method that makes use of mobile GPS devices. The longitude and latitude of the participants will be recorded automatically, therefore can capture periodic schedule of individual. However, the GPS records have to be validated into meaningful travel data before further usage, which might cause error. Despite this shortcoming, GPS diary is still believed to be promising for transportation applications.

2.7 Macroscopic, mesoscopic and microscopic models

Three types of simulation model differ in the resolution detail of simulated objects.
Macroscopic: simulates traffic flow, represented by vehicle speeds, traffic density and flow rates, being the simplest representation that still possesses popularity due to simplified flow propagation.
Mesoscopic: simulates both vehicle speeds and individual vehicles, efficiently evaluate route choice decisions, since it follows the traffic flow properties of macroscopic models without detailed examination of vehicle interactions.
Microscopic: car-following models for individual vehicles, allows simulation of lane switch and *en route* adjustment, better for evaluation of traffic with some geometric configurations (Kachroo and Shlayan, 2013).

2.8 Bounded rationality

The assumption that travelers tend to maximize their individual utility, but not necessarily to an absolute maximum level, due to their limited knowledge of the network states and alternative route information. This is one of the core concepts which supernetwork representation is based on, and also the main characteristic of MATsim agent.

3 Literature review

3.1 Introduction

Traditional transportation planning approach that is widely adopted around the world for investment decisions of large-scale scenarios, is based on a four-step model (Potts and Oliver, 1972):

Trip Generation: estimate the number of trips generated at origin zones and/or attracted to destination zones based on data such as household income, demographics, land-use plan, etc.

Trip Distribution: Use mathematical algorithms to generate an origin–destination (O–D) matrix, with each cell entry indicating the number of trips from one origin to one destination.

Modal Split: Attach each value of O–D matrix to various alternative travel modes. This can usually be accomplished using discrete choice analysis results from survey data (Ben-Akiva and Lerman, 1985).

Traffic Assignment: Assigns each O–D flow value onto various alternate paths from that O–D node, until the whole system reaches Wardrop's user-equilibrium (UE); that is, the travel times of all used routes between the same O–D pair are minimal (Wardrop, 1952).

This traditional four-step process cannot satisfy current ITS needs any longer, due to several fundamental limitations:

Lack of consistency with sub-models: For example, travel times from the assignment model are not necessarily consistent with ones used for the prediction of trip destinations;

Failure to capture the dependency among trips in the same chain: The model cannot predict secondary effects and complex behavioral adaptation patterns caused by external manipulation as a result;

Significant aggregation bias: As the model is based on the assumption that the proportion of trips in each period is constant, and all households in a zone are identical;

Lack of behavioral realism: The model is not based on the realistic assumption that individuals and households wish to realize their needs but subject to constraints such as time budget, car availability, etc (Rasouli and Timmermans, 2014).

As has been illustrated in Chapter 1.3.1, efforts have been put into developing new measures to overcome such shortcomings. Currently, on the demand side (travel demand forecast), the dominant approach to replace trip-based method is activity based modeling (ABM) (Shiftan and Ben-Akiva, 2011); on the supply side (traffic assignment), dynamic traffic assignment (DTA) proves to be a behaviorally sound alternative to the classic static assignment (Chiu et al, 2011).

This chapter will provide a brief exploration of the previous studies that has contributed to the solution of the replacement approach. Chapter 3.2 starts with a review of dynamic traffic assignment approaches, followed by a detailed evaluation of state-of-art simulation based approaches in Chapter 3.3. To better capture behavioral realism, microscopic multi-agent simulation is chosen in this study, the reasoning of which is explained in Chapter 3.4. Supernetwork concept was then introduced in Chapter 3.5, complementary to the theoretical soundness of the multi-agent simulation. This literature study is then concluded in Chapter 3.6, with a combined approach proposed and further explained in Chapter 4, also labeled as the main contribution of this study.

3.2 Dynamic traffic assignment

Since the initial work of Merchant and Nemhauser (1978) that considered discrete time-varying O-D flows instead of static matrix, DTA approaches have witnessed significant evolution, brewing two mainstreams: Analytical solutions and simulation-based measures, both of which usually share similar mathematical extraction for problem formulation, but try to solve it from different perspectives.

The former ones are usually based on extensions of the Lighthill–Whitham–Richards (LWR) model (Lighthill and Whitham, 1955; Richards, 1956), a macroscopic one-dimensional traffic model that uses traffic density and traffic speed for traffic flow propagation. They tend to seek a profound mathematical formulation of traffic network dynamics, giving numerical solutions for different scenarios. One latest example is a multibuffer model proposed by Garavello and Piccoli (2013), trying to describe the intersections via using buffer for each outgoing road, so that a junction will not be blocked if merely one exit is full. Though this model helps represent real-life driver's preference under such scenario and complement LWR road network, extraordinary mathematical complexity is added as a side effect.

Another research direction is to incorporate models that improve LWR flow propagation. Earlier efforts within this context, such as bottleneck models and exit flow models, fell short of capturing the spatial effect of queues, which was the major factor contributing to real-life congestion (Peeta and Ziliaskopoulos, 2001). A typical instance of analytical solution that has better performance is cell-based dynamic equilibrium approaches, which incorporate the cell transmission model (CTM) into DTA framework. The CTM, purposed by Daganzo (1994), divides highway into homogeneous sections (cells) whose lengths are determined by the distance traveled by free-flow traffic in discretized time intervals, and seeks approximate results under LWR conditions. These approaches, though being able to capture realistic traffic dynamics such as queue formulation, queue dissipation and queue spillback, possess an undesired property named as traffic holding-back, and usually lead to very high computational burden due to the complexity of the representation (Szeto, 2013).

As a conclusion, the analytical methods, similar to the examples given above, are often theoretically sound, capable of proving that the network has reached an UE state; however, due to the ill-behaved system properties caused by traffic realism and human behavior, a universal solution for general networks has not been found yet, hindering their deployment in real life (Szeto and Lo, 2005).

Compared to the analytical solutions mentioned above, simulation-based approach appears to be a more practical measure. They seek close-to-optimal solutions, trading off theoretical soundness with practicality to some extent. They usually conduct several iterations to obtain user-equilibrium, followed by a field data-based implementation process. Once successful, the simulation software will be prepared for various studies of the chosen case (Kachroo & Shlayan, 2013). Due to their utility, simulation approaches can be applied for all three levels of scenarios, including macroscopic, mesoscopic and microscopic.

3.3 Simulation-based DTA

There also exist two research directions for simulation-based models, namely real-time deployment and transportation planning improvement. Though the former branch focuses on real-time flow propagation, while the latter relies on iteration to optimize network mobility, the objective to capture traffic reality does not differ; therefore, a model with real-time data processing capability is usually bound with an offline planning version that share the same theoretical frame.

There exist several software packages dedicated to simulation based dynamic traffic assignment (DTA) implementation. Mesoscopic tools focus on traffic flow, including DYNASMART, DynaMIT and CONTRAM, etc. Microscopic packages reproduce each single vehicle, including SimTraffic, CorSim, VisSim, Paramics, AMSUNNG, MITsim, TRANSIMS, etc. A comparison among these software packages can be found in the work of Jeihani (2007), which is beyond the scope of this thesis, only several most vital models will be discussed here.

The first known simulation-based approach that follows realistic behavioral rules is DYNASMART (Dynamic Network Assignment Simulation Model for Advanced Road Telematics), a descriptive analysis tool developed by Jayakrishnan et al (1994). It generates vehicles from a direct translation of zone-to-zone O-D demand with fixed departure time during each time step, and allocates them randomly among selected links; all moving vehicles within a link then share the same speed resulting from the density at the end of the previous time step. Driver response simulation is performed at each node, where the path selection decision will be made. With such representations, this model is capable of studying the effectiveness of given traffic information/control system configurations, serving as the foundation of later simulation models. Its planning version, DYNASMART-P, recognizes four vehicle types (passenger cars, trucks, high occupancy vehicles and buses), allowing for mode and route choice assignment with fixed departure time. This model can still see usage in recent works, such as its deployment in Taiwan scenario (Hu and Liao, 2011).

The functionality and utility of simulation methods was further improved with DynaMIT (Dynamic Network Assignment for the Management of Information to Travelers), a DTA model intended for real-time travel guidance provision proposed by Ben-Akiva et al (1997). Given historical O-D records and real-time data from traffic sensors, this model is able to provide descriptive guidance about current network conditions in terms of O-D flows, link travel speed, traffic densities and queue characteristics, with the help of mesoscopic simulation from the beginning of the previous time step to the current period. Using the above estimation as input, an iterative simulation procedure can be carried out to predict future network states. Together with an *en route* driver behavior model, prescriptive guidance will be generated as a result, including suggestions of departure time, travel mode and route choice. Being the first simulation model with guidance provision capability, DynaMIT clarified solution direction for later studies. Its offline version, DynaMIT-P, is capable of modeling day-to-day traffic evolution of short-term planning projects, with route/mode/destination choice assignment corresponding to the information provision. The applicability of DynaMIT-P has been proven even in highly congested areas (Ben-Akiva et al, 2015).

For models dedicated to planning purpose, a tendency to incorporate complex network formulations or demand forecast that analytical measures struggle to solve has been witnessed, since simulation solutions trade accuracy for applicability in large-scale scenarios. RouteSIM, developed by Lee (1996), is an example of simulation solution towards the CTM. Two major enhancements were present in this model: Firstly adjustable cell lengths were adopted instead of homogeneous ones, with bigger cells representing section of long highway segments and small cells for intersections, so that computational requirement and flexibility of CTM was improved consequently; another contribution was to simulate traffic signal control on road intersections, capturing traffic reality to a higher level. The model was later embedded into a prototype internet-based GIS system developed by Ziliaskopoulos and Waller (2000) for various transportation applications.

The aforementioned approaches are typically trip-based models that simulate vehicular traffic on road network, which was reasonable based on the assumption that vehicles made up the majority of traffic flow in urban areas. Such assumption, however, might not be true at this moment, as was explained in Chapter 1.1, with the increasing preference of non-vehicular trips. For this reason, activity based models prove to be superior for more realistic travel demand prediction, despite the mathematical complexity of representations.

Among the simulation methods available for ABM, two open-source software packages have frequent appearance in recent studies, namely TRANSIMS (Transportation Analysis and

Simulation Systems) and MATsim. Though both adopting agent representation that simulates individuals instead of vehicles, these two differ in several aspects. TRANSIMS proves to be advantageous for being the only tool embedded with demand estimation module, and produces more realistic traffic flows obeying macroscopic flow characteristics (Jeihani, 2007). MATsim, on the other hand, has three traits that appear to be more satisfying for this research. For one thing, MATsim agents completely contain their individual attributes throughout modeling process, resulting in detailed information provision to personal or household level, while TRANSIMS splits them amongst modules and files. Secondly, MATSIM utilizes XML for data processing, and requires only one document type definition (DTD or schema) for all agents, therefore possesses higher data consistency. Apart from that, MATsim is also unique in its multi-agent design, allowing traveler to be resolved individually, capable of storing multiple plans which better reproduces bounded rationality of real-life people (Balmer et al, 2005). Considering the fact that behavioral realism is the major concern of this study instead of flow propagation, the multi-agent representation was chosen consequently.

3.4 MATsim: Multi-agent simulation for ABM

MATsim, developed by ETH Zurich, is an activity-based multi-agent simulation framework implemented with Java, and is especially designed for one-day simulation of large-scale scenarios (MATsim, 2015). The simulation is based on genetic algorithms, which keeps several instances of possible daily activity chains in the memory of each member of the population, and uses a fitness function to evaluate their performance in the form of a numerical score. The members are allowed to modify their activity chains via mutation and crossover, which simulates the behavior of real people. An iterative process will be conducted until the average population score stabilizes, which can be partially viewed as converging into an UE (Horni et al, 2015).

The fundamental version developed by Charypar and Nagel (2005) was capable of simulating individual’s daily activity and travel schedule, a trait that is inherited in all later versions and remain the core function of MATsim. The simulation, though being able to make full use of demand prediction from ABM methods, could not reproduce public transport system as TRANSIMS does, resulting in a significant loss in terms of traffic reality. This problem was eventually solved by Rieser (2010), who proposed a complete theoretical model for travel mode choice, including non-car plan generation, scoring, replanning and event handling, as well as strategy modules that allow trip mode switch for each agent. This model was further complemented with simulation of public transit vehicles, which was embedded in MATsim using a semi-automatic public transit route map-matching tool developed by Ordonez and Erath (2011). Another step forward was the simulation of non-vehicle trips such as cycling and walking, carried out via a separate module that calculates the movement speed of agent from its demographic characteristics (age, gender) and teleports the agent after the corresponding travel time is taken (Dobler et al, 2012). This module, however, did not integrate well with the existing bimodal simulation (car and public transit) due to some coding conflict. Extra effort will be called for to form a complete multi-modal simulation approach, which is one major contribution the author is trying to address.

Apart from the evolution of mode choice aspects, another major enhancement made to MATsim was a more specified within day replanning function developed by Illenberger et al (2008), where a one-shot simulation instead of iterative process was applied as a replacement of the iterative process. Storing historical travel times of the agents and analyzing reactive travel times within current network, this approach is capable of providing predictive travel times, therefore simulates *en route* adjustment of each agent under different types of information provision, similar to DynaMIT. This method extended the functionality of MATsim to a different level, igniting sparks for several later studies. This method, however, did not serve the purpose of transportation planning, as it is not practical to optimize traffic flow with a single time of simulation, thus is beyond discussion in this thesis.

When it comes to traffic flow simulation, MATsim also saw various expansions. Flow propagation in the early versions relied on *QueueSimluation*, an implementation of the time-step based queue model proposed by Balmer (2007), where the simulated time period was split up into equal pieces with a given duration—the time step size, and the state of the queues would be calculated within each time step. A remake of the original one-core simulation, together with several extension modules including traffic signal control and public transit simulation, made up the majority of *QSim*, the most used *mobsim* today. As an alternative, event-based approach for MATsim was developed by Charypari et al (2007) and later adapted to Java, namely *JDEQSim*, where calculations are done for each occurring event during simulation, so that its computational burden only scales with the number of agents and the complexity of their plans. In this thesis, however, *QSim* remain the choice, regardless of the potential computation time reduction from *JDEQSim*, considering the re-usability of the established scenario for further analysis and the extraordinary complexity caused by GPS diary incorporation, as is explained in Chapter 5.4.1.

One latest breakthrough within MATsim context that needs to be mentioned is the implementation of household choice models, considering the challenge brought by vehicle allocation. In their research, Dubernet and Axhausen (2015) made a successful attempt to simulate the joint decisions of household level with the help of a game theoretic model and social network integration. This effort pointed out a new future research direction, as household level data is also a commonly used data source for demand generation within transportation planning field. In the same article, they also revealed a list of choice models that have research potential for integration with multi-agent representation, among which the supernetwork frame proves to be most suitable, due to several common features they share, such as bounded rationality and activity-travel chain modification. The author, therefore, made an initial attempt in this thesis to merge these two models with simplified representations, trying to develop a DTA method with relatively sound theoretical underpinning and high applicability.

3.5 Supernetwork approach: integrated accessibility

Supernetwork is a concept originated from computer science, referring to a concatenation of networks interconnected by transfer links (Sheffi, 1985). In the sense of traffic engineering, its definition was further extended to a network connecting different networks for transport modes and the locations for activities (Arentze and Timmermans, 2004). In the paper of Liao et al (2013b), a supernetwork was decomposed into private vehicle networks (PVN, including car and bike) and public transport networks (PTN, including public transport and walking), based on the observation that an individual's activity program only involves a small proportion of destination nodes and transport system. The *links* interconnecting PTN and PVN represent uni-modal travel, and those connecting PVN and PTN represent transport mode change, such as parking or picking-up private vehicles; links between PTN then represent conducted activities. In this way, the whole-day activity-travel program was captured precisely. This representation can reduce the computational burden in large-scale scenarios without significant loss of theoretical soundness, thus can be a potential solution for the integrated approach (Liao et al, 2011).

In the research work of Huang et al (2015), an analytical measure named dynamic activity-travel assignment (DATA) has been proposed to incorporate the supernetwork representation mentioned above. They formulated the discrete-time DUE condition as an equivalent path-based variational inequality (VI) problem (Friesz et al, 1993), and proposed an iterative algorithm in MATLAB to solve it. For each traveler, the inflows of non-minimum disutility time-dependent ATP are swapped to the minimum disutility paths, until the stopping criterion is met.

This DATA method, however, has several drawbacks. First, the traffic assignment algorithms proposed was not efficient enough to handle large scale scenarios, given the quantity of feasible ATP generated. Second, the model falls short of a general treatment on activity duration choice. Also, this solution is based on the unrealistic assumption that travelers have complete information about activity-travel time and disutility, which is critical in case of behavioral realism.

Considering the characteristics of personalized supernetwork, the multi-agent simulation specifically from MATsim should be a better solution here. For one reason, it fits the core concept of ABM well, and is capable of describing microscopic scenarios, given detailed geographical, functional, and temporal data (Bazzan and Klügl, 2013). If we model each individual as an *agent*, and each part of his/her activity-travel program as an *event,* the theoretical underpinning of this solution method then becomes immediately obvious. For another, the main advantage of multi-agent simulation method lies in its capability to represent the *learning* process. Only part of the agents will realize that their current ATP are not of highest *utility*, and modify it in response in the next iteration, while this modification will still be limited to the *plans* they are assigned with. This process captures the reality that individual has bounded rationality, therefore better represents the heterogeneity of real human.

The agent representation in MATsim is resolved at individual level, therefore exactly fits the concept in supernetwork frame, and the ability to store multiple plans for each agent would reproduce the bounded rationality of individual to some extent. Also, thanks to the effort of its developers, the simulation of cycling and walking has been enabled recently, which accounts for another main reason to choose the package, despite its beginner-unfriendliness due to the nature of open-source software. The current simulation process, however, is not fully integrated as an entity for multi-modal simulation, but split among various modules. Developing a multi-modal DTA approach from these modules require extra effort considering data compatibility and coding conflict, which is one of the challenge the author managed to overcome.

3.6 Conclusion

In this literature study, the traditional four-step traffic planning model is first introduced, with its shortcomings listed, namely lack of consistency with sub-models, failure to capture the dependency among trips in the same chain, significant aggregation bias and worst of all, lack of behavioral realism. Due to these flaws, the model no longer satisfies current ITS requirements. Recent research effort focuses on seeking a replacement to this old-fashioned model, and one promising trend is to incorporate activity based modeling (ABM) and dynamic traffic assignment (DTA), the latter of which is the focus of this thesis.

A thorough review of DTA is provided here, from two solution directions. Analytical models aim at finding an accurate formulation of road network, and seek convergence of the mathematical model as the solution to traffic optimization. Typical instances include extensions of LWR model, and CTM incorporation models. However, a formulation of general network is not found yet, hindering these models from universal deployment in ITS, and the complexity of the formulation and solution process is also considered inefficient for real-time use, despite their theoretical soundness and solution accuracy. Simulation-based models, on the other hand, is more promising from application perspective. They usually use iterative process to obtain a partially released travel demand, trading solution accuracy for efficiency and applicability. Some trip-based approaches made remarkable contribution to the evolution of simulation models, including DYNASMART, DynaMIT and RouteSIM. For incorporation purpose, however, agent-based simulation is more suitable for ABM frames.

There exist several platforms for agent-based simulations, among them TRANSIMS and MATsim are most developed and frequently adopted in studies. TRANSIMS is more advanced in terms of embedded demand estimation functions and high quality flow propagation similar to macroscopic model output, while MATsim is more promising due to higher data consistency and unique multi-agent representation.

Since its appearance in 2005, MATsim has witnessed several major improvements, namely within day replanning enhancement for *en route* adjustment, public transport incorporation for bimodal simulation, and household joint decision simulation attempt. Unfortunately, the multi-modal simulation function, as an eager calling from latest transportation planning needs, remains immature in MATsim model, which is the challenge the author is trying to overcome. The integrated accessibility concept derived from personalized supernetwork frame is adopted here in order to complete the final piece for realistic multi-agent simulation, and these two methods in fact complement each other, as the individual resolution of agent from MATsim reproduces personalized activity-travel choice preference from supernetwork approach, and simulates bounded rationality as well in the form of multiple alternative plan for each agent.

This chapter gives a review of methodologies related to the effort of this thesis, meanwhile providing reasoning for the integrated approach from a historical perspective. The details of the research model of the given approach will be presented in Chapter 4, as the theoretical foundation of this study.

4 Research model

4.1 Introduction

As has been demonstrated in Chapter 3.4 and 3.5, current MATsim model is not fully capable of multi-modal simulation. In the latest published branch (0.7.0), public transits has been embedded into the core segment of Qsim, while the simulation of walking and cycling remains separate module and adopts within day replanning simulation instead of iterative process, causing consistency issues. The major problem, however, lies in the fact that the Charypar-Nagel fitness function is not capable of realistically evaluating the outcome of different travel mode choices and unable to simulate real-life individual's preference under multi-modal scenario as a result. A fitness function with validation results is called for, giving more realistic evaluation of given agent plans, and some extra effort is required for the generation and integration of data needed, both of which challenged the author during the research process.

This chapter will give a detailed description of the research model, and address the main contribution of this thesis. Chapter 4.2 introduces the very basic MATsim uni-modal simulation model and its process model considering vehicular flow propagation, as well as the Charypar-Nagel function for its genetic algorithms, which MATsim model relies on for plan improvement. Chapter 4.3 then briefly describes the incorporation of public transport system necessary for later development of multi-model simulation, where the author has to spend a large portion of effort for scenario establishment and data integration. Chapter 4.4 discusses the integrated accessibility and supernetwork concepts in detail, integrating them into multi-modal simulation process in the form of a validated and modified fitness function that takes different transport mode choice into consideration, which can be labeled as the main contribution of this thesis.

4.2 Unimodal simulation: Basic MATsim process

The very basic implementation of MATsim simulation only considers vehicular traffic without recognizing vehicle types, following the steps shown in Figure 4.1. The files marked in orange are either generated or modified by the author.

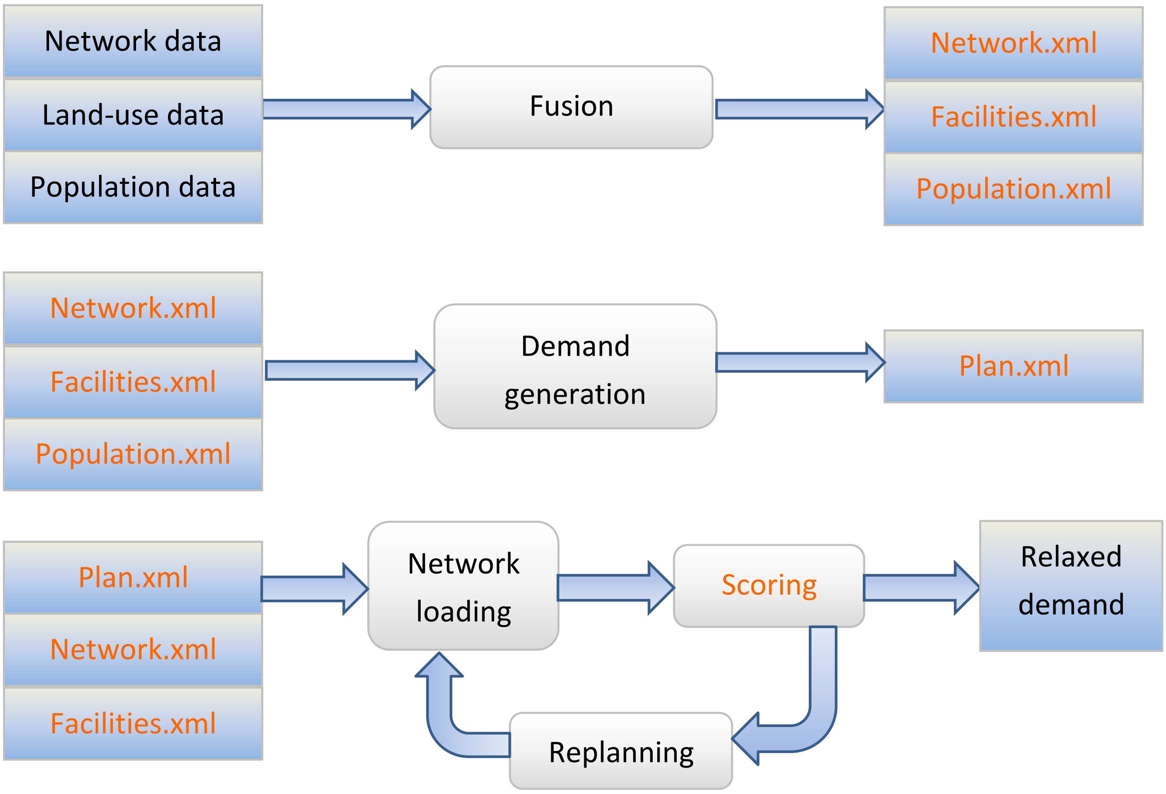


Figure 4. 1 MATsim process structure by Balmer et al (2008)

Fusion: A preparation process where the data files necessary for basic MATsim simulation will be created from various local data sources. The output of this step is an established scenario consisting of three categories of data:

1. Network file: A GIS layer of the study area, reproducing the geographic picture from nodes and links;
2. Population file: The initial plan of each agent, recording its activity location, start time, end time and main mode of travel (named “leg”);
3. Facility file: The location of facilities for various types of activities, with their opening hours recorded.

Demand generation: The process where an initial demand would be derived from the study area population’s daily activity chains. Plans (activity-travel program, in terms of population file) will then be generated, describing the activity chain of each agent. It is meant for representing the travel behavior of a study area, and can be derived from various data sources. A conventional one is national travel survey, where participants are asked to fill in paper or online questionnaires, recalling their activity-travel schedule of a specific day. These plans, usually with a large sample size, serve as the source of agents’ plan thereafter. One-to-one translation of these plans proves to be viable for simulating the whole nation, as the survey normally covers around 3 % of total population; however, for a specific area, the sample size might shrink to a large extent. A commonly used solution for such situation is population synthesis, which recreates the whole population using an iterative proportional fitting (IPF) procedure based on the demographics of the local area, such as gender and age distribution at a household level (Arentze and Timmermans, 2001).

Network loading: The simulation process of MATsim. Using the above input, the synthetic population will be generated amongst the given network, and conduct their daily schedules in pre-defined facilities. After first iteration, each plan from the initial demand will be associated with a utility score, which will be re-evaluated when iteration terminates, based on the plan's performance. The traffic flow propagation relies on a *mobsim* module named *QSim*, a queue-based time-step model with extensions of within day replanning, public-transit simulation and traffic signal control, as is mentioned in Chapter 3.4. The duration of its iteration increases proportional to the number of links, and is independent of the agent population, resulting in a stable simulation time for each studied network, suitable for large-scale scenarios (Dobler, 2010).

Replanning: The transition between two iterations, during which part of the agents will be allowed to generate new plans from the initial ones. For each agent, the plan with highest score will be carried out during iteration, and whenever an agent possesses excessive plans, the plan with lowest score will be removed from its memory, before the next iteration starts. The available replanning modules, based on whether they modify the plan randomly, can be categorized as Random Mutation module and Best Response module, with the former one used for time allocation modification of individual's schedule given mutation range (Balmer et al, 2009), while the latter optimizing the route choice of agents. Via a Router Module that seeks the route with the least negative utility as the best route, the Best Response module is a Landmarks-A* implementation of the Dijkstra algorithm (Lefebvre and Balmer, 2007).

Since MATsim uses genetic algorithms for traffic optimization, a fitness function is required for evaluation of the plan performance during network loading and replanning stages. The default function developed by Charypar and Nagel (2005) sums the utility of all activities and disutilites of traveling behavior:

$$F = \sum_{i=1}^{n} U_{act1,i} + \sum_{i=2}^{n} U_{trav}(loc_{i-1}, loc_i) \tag{1}$$

Where $U_{act1,i}$ refers to the utility of activity location *i*, and $U_{trav}(loc_{i-1}, loc_i)$ is the disutility of traveling from location *i-1* to *i*.

The utility of traveling is represented as:

$$U_{trav} = \beta_{trav} \times \text{time} \tag{2}$$

Where β_{trav} is the marginal utility of travelling, in the form of a monetary value, normally settled to -6 €/h.

The utility of each activity is a little more complicated:

$$U_{actl,i} = U_{dur,i} + U_{wait,i} + U_{latear,i} + U_{early.dp,i} + U_{shortdur,i} \quad (3)$$

Where U depicts the utility of the following variables of each activity: activity duration, waiting time, penalty for coming too late or leaving too early, and penalty for performing an activity for too short time. They are calculated as follow:

$$U_{dur} = \beta_{dur} \times \text{time} \quad (4)$$

$$U_{wait} = \beta_{wait} \times \text{time} \quad (5)$$

$$U_{latear} = \beta_{latear} \times (t_{start} - t_{latestar}) \text{ if } t_{start} > t_{latestar}\text{, else } =0; \quad (6)$$

$$U_{early.dp} = \beta_{early.dp} \times (t_{earliestdp} - t_{end}) \text{ if } t_{end} < t_{earliestdp}\text{, else } =0; \quad (7)$$

$$U_{shortdur} = \beta_{shortdur} \times (t_{shortdur} - (t_{end} - t_{start})) \text{ if } t_{end} < t_{shortdur}\text{, else } =0. \quad (8)$$

Where β is the marginal utility of each variable, t_{start} and t_{end} is the starting time and end time of the activity respectively, $t_{latestar}$ and $t_{earliestdp}$ is the latest starting time and earliest end time of such type of activity respectively, $t_{shortdur}$ is the shortest duration of this activity. Among them, the most frequently used parameters, namely β_{dur} and β_{latear}, are usually settled to 6 €/h and -18 €/h respectively.

After defined times of iteration, the demand data will be relaxed, with the average trip travel time of each simulated agent minimized and stabilizes.

4.3 Bimodal simulation: public transit incorporation

A major evolution of MATsim in recent years is marked by the contribution of Rieser (2010), Ordonez and Erath (2011), who added public transit to MATsim assignment, enabling bimodal simulation. Two extra input documents will be required as listed below:

4. Transit schedule file: The schedule of public transport system, describing the route and time table of each single bus or tram;
5. Transit vehicle file: The list of public transit vehicles in the study area and the parameters of each type of vehicle including length, width and capacity.

The network for MATsim simulation is usually converted from local extraction of OpenStreetMap (OSM) data. This extraction, however, is not sufficient for multi-modal simulation, as the original data do not contain the information related to transport modes available for each link; in other words, the extraction is merely a private vehicle network. To construct the public transport network, a complex procedure is needed. The route of each single public transit vehicle has to be marked on the existing network, so that the links where bus or tram travels can be picked out and modified into multi-modal links, which, together with the nodes they connect, become the component of the new PTN.

Unfortunately, there does not exist an open data source that hold storage of public transit route information that can be directly converted into xml format; that means, every path of each public transit vehicle has to be verified by the author himself. Currently, this has been proven practical via reproducing the path from transit data, and a widely used open data source for such purpose is General Transit Feed Specification (GTFS).

Following the generation of intial transit schedule, path verification is the vital step of this process. The solution is an semi-automatic tool developed by Ordonez and Erath (2011) that incorporates a shortest-path map-matching algorithm, which has become part of gtfs2matsimtransitschedule module. Each of the path is automatically calculated based on the instructions from Shape files and shortest-path principle, and then visualized in an user interface, where manual checking and editting is enabled. New nodes and links can be added if needed, and the path will then be re-calculated. Figure 4.2 briefly introduces the work flow. This procedure could gurantee very high accuracy if intensive effort would be made. The outcome of this step is the link sequence of each route at each departure time, as well as a multi-modal network that integrats PVN and PTN.

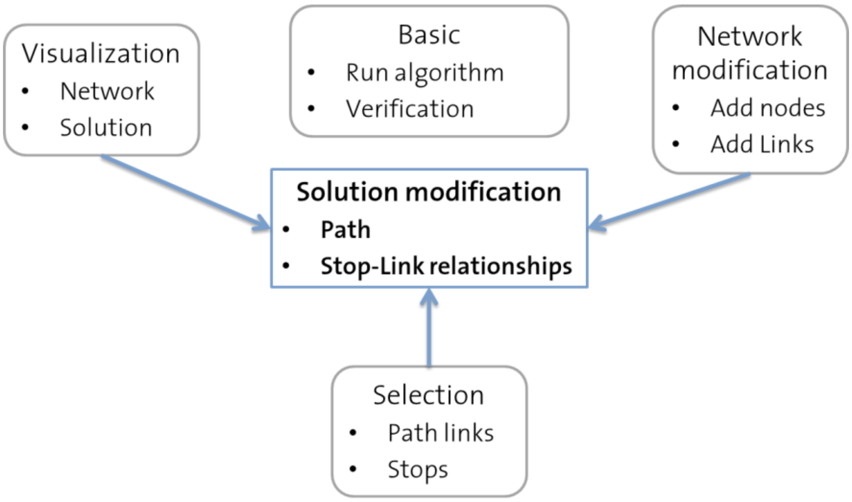


Figure 4. 2 Process of path verification from GTFS feed by Ordonez and Erath (2010)

4.4 Multi-modal representation with supernetwork frame

4.4.1 Supernetwork incorporation

To run a multi-modal simulation that involves bike and walk, the module has to be further adjusted. The approach used here is to incorporate the module written by Dobler et al (2012), who added a multi-modal contribution to each link object in mobsim, using a priority queue to order the agents based on their scheduled link leave time, so that the agents are “teleported” from the origin to destination after the travel time. This process described was, however, carried out by a separate simulation involving within day replanning, thus was not fully integrated with existing methods to form a compatible approach. An initial attempt of the author was to compose a module that convert the output plan from bimodal simulation into the input for cycling & walking simulation, which, however, did not prove to be necessary as the latest branch of MATsim already enabled multi-modal simulation shortly after this thesis started, with the teleport speed of non-vehicle modes defined in configuration.

For traditional transportation simulation that focuses on vehicular traffic flow in uni-modal network, the Charypar-Nagel function would be enough, which, however, is not the case for supernetwork approach. In a general supernetwork, each node denotes a location in real life, while each link represents a traveler's specific action, thus choosing a route is analogous to choosing an activity location, duration, time of participation and travel route (Ramadurai and Ukkusuri, 2010). To accurately simulate such structure, transport modes other than personal vehicle should also be taken into consideration, as well as the impact of transaction between different network layers such as waiting, access and egress, requiring more complicated activity-travel (dis) utility functions, which is the key of this research model.

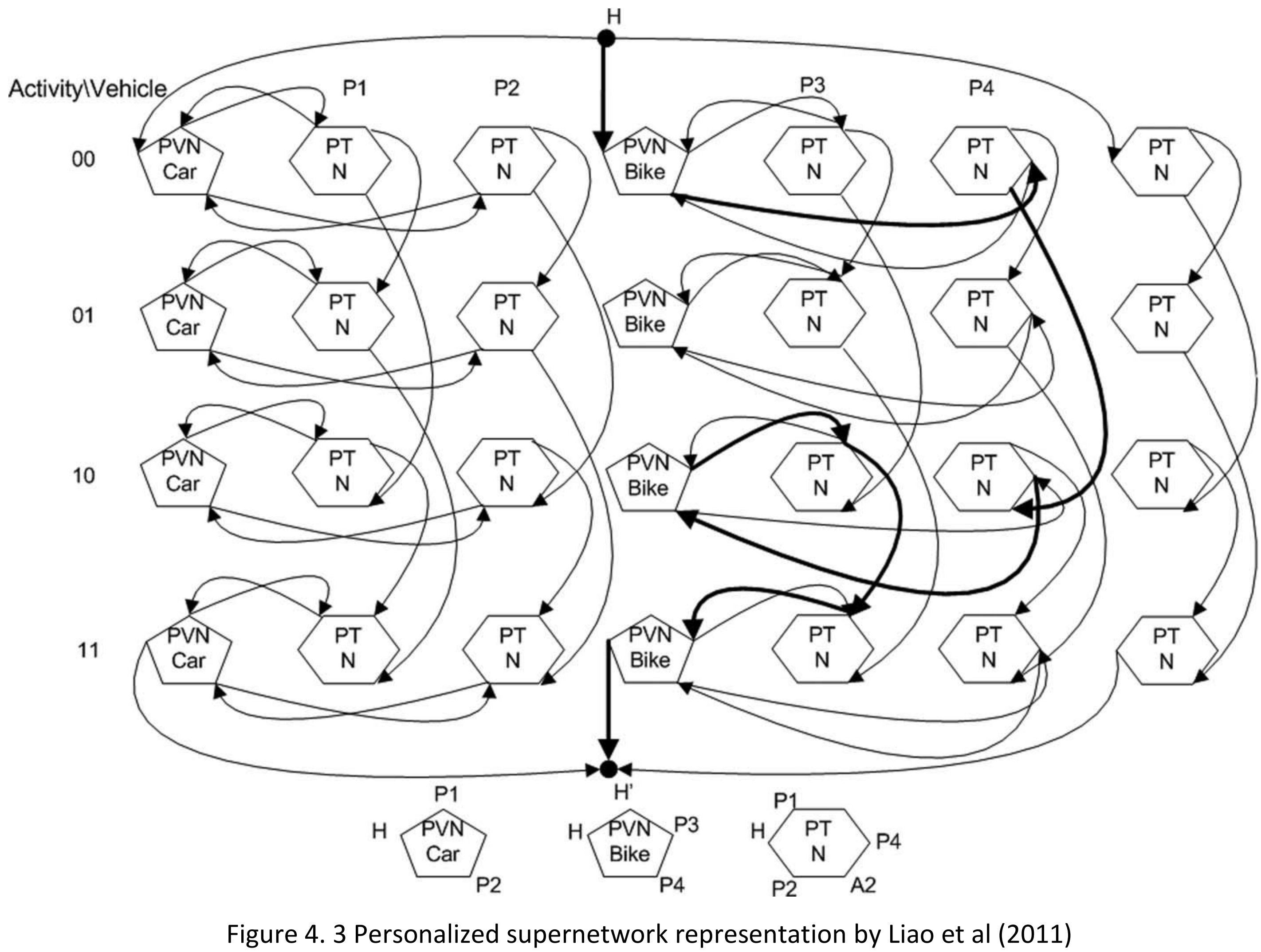


Figure 4. 3 Personalized supernetwork representation by Liao et al (2011)

As is mentioned in literature review, personalized supernetwork was composed of private vehicle networks (PVN, including car and bike) and public transport networks (PTN, including public transport and walking). Links interconnecting PTN or PVN are travel links; links between PVN and PTN in the same activity state are transition links, and links between PTNs from different activity states are transaction links. Figure 4.3 gives a sample representation. In this figure, H represents home location, where individual's daily schedule usually starts and ends; the whole-day plan conducted consists of a chosen set of polygons, involving three types of personalized network illustrated at the bottom of the figure, where P1, P2, P3 and P4 represents parking locations for car and bike respectively, serving as connection between PVN and PTN, with the other points of the polygons referring to activity locations interconnecting PVN or PTN, except for A2, which is the alighting location of PTN. This figure presents all possible route choices, with the bold lines describing a chosen route: This person starts from home on bicycle, then parks it near the bus stop and travels to the first activity location by bus; after the first activity is performed, he/she picks the bus back to where the bicycle is parked, and travels to the second activity location on bike; on finishing the second activity, he/she then picks the bike and go home.

Corresponding to the personalized supernetwork representation mentioned above, four transport modes are considered in this thesis: Car, bicycle for PVN and bus, walking for PTN. The intercity public transportation will not be taken into consideration, as it leaves the study area and cannot be followed in microscopic simulations. As for taxi, though can be modeled similar to bus with a higher cost parameter, it is not valuable enough to be included here, as the choice between bus and taxi, as well as the one between car and taxi, will add unnecessary complexity without significantly improving the model realism; also, the data availability of such mode is another concern, as conventional travel diary data usually don't distinguish between personal vehicle and taxi.

The (dis)utility functions proposed by Liao et al (2011) are also applied in this research, depicting the concept of integrated accessibility. The functions for each link type and activity type are listed below:

Travel link costs:

Walking: $$U_{isW1} = \beta_{isWt} \times \mathrm{time}_{W1} + \varepsilon_{isW1} \tag{9}$$

Bike: $$U_{isB1} = \beta_{isBt} \times \mathrm{time}_{B1} + \varepsilon_{isB1} \tag{10}$$

Car: $$U_{isC1} = \beta_{isCt} \times \mathrm{time}_{C1} + \beta_{isCc} \times \mathrm{cost}_{C1} + \varepsilon_{isC1} \tag{11}$$

Public transport: $$U_{isPT1} = \beta_{isPTt} \times \mathrm{time}_{PT1} + \beta_{isPTc} \times \mathrm{cost}_{PT1} + \varepsilon_{isPT1} \tag{12}$$

Transition link costs:

Parking: $U_{isPKv} = \beta_{isPKv} \times X_{isPKvp} + \varepsilon_{isPKvp}$ (13)

Picking up: $U_{isPUvp} = \beta_{isPUve} \times \mathrm{eTime}_{isPUvp}$ (14)

Boarding: $U_{isBDt} = \beta_{isBD} \times X_{isBDt} + \varepsilon_{isBDt}$ (15)

Alighting: $U_{isATt} = \beta_{isATe} \times \mathrm{eTime}_{isATt}$ (16)

Transaction link costs:

Activity: $U_{isCAjk} = \beta_{isCAj} \times X_{isCAjk} + \varepsilon_{isCAjk}$ (17)

Where U depicts disutility of the link, β represents various weight vectors, ε refers to an error item; eTime is short for egress time, and X denotes waiting time and location attractiveness.

In this research, the error items (ε) can be ignored, due to its extra complexity in case of simulation implementation, so the β values need to be validated according to the local scenario. Fortunately, a prior research had conducted a multinomial logit (MNL) process for this purpose, using the data from a nation-wide online questionnaire in Netherlands with high convincingness (Arentze and Molin, 2013); therefore, the author decided to advance from these established results, to incorporate them with multi-agent simulation.

In their paper, two multi-modal scenarios were studied, one focusing on trips within 5 kilometers, the other collects data for a longer distance (20 kilometers). Based on their research, as the trip distance lengthens, passengers tend to care less about access and egress time; therefore a scale parameter μ has to be multiplied to the utility function, that is:

$$U_i = \mu_i \beta_i \times \mathrm{time} + \varepsilon_i \tag{18}$$

Thus for trip distance longer than 5 kilometers, the scale parameter of 0.887 is also applied in this research. The marginal utilities can be further defined as:

$$\beta_i = \beta_{trav} \times r_i \tag{19}$$

Where r_i is the factor determined by transport mode chosen, β_{trav} is the marginal utility of travelling settled to -6 €/h.

4.4.2 Parameter estimation

The estimation results are shown in Table 4.1. In this table, t_ refers to time (distinguishing from the monetary cost functions), _access and _egress represents the boarding and alighting stage of each trip respectively, _main refers to the main mode of travel of a specific trip; t-Value is not directly used, but defines the possible range of estimated parameters.

To incorporate these parameters with MATsim simulation, a further transformation is required. Considering the fact that the default setting is meant for cars, the author decided to settle t_main_car value to 1.0 and re-scale the remaining values accordingly. The r_i estimation results can be found in Table 4.2.

The integration of the disutility functions can be achieved in various ways. Adjusting the core coding of MATsim replanning and compose a new module would be the most in-depth method which, absolutely, will take intensive effort and demand highly-skilled programming. A simplified solution is applied instead, by overriding the Charypar-Nagel function, as is shown in Appendix 3.

Parameter	Value(<5km)	t-Value	Value(<20km)	t-Value
t_access_walk	-0.11	-36.6	-0.11	-36.6
t_access_bike	-0.095	-13.5	-0.095	-13.5
t_access_pt	-0.084	-10.2	-0.084	-10.2
t_wait_access	-0.073	-8.6	-0.073	-8.6
t_main_bike	-0.076	-9.04	-0.076	-9.04
t_main_car_short	-0.043	-9.41	-0.079	-12.9
t_main_pt_short	-0.058	-16.77	-0.074	-18.5
t_main_transfer	-0.097	-22.7	-0.097	-22.7
t_park_car	-0.079	-11.7	-0.079	-11.7
t_wait_egress	-0.112	-16.2	-0.112	-16.2
t_egress_walk	-0.101	-37.4	-0.101	-37.4
t_egress_ptbike	-0.13	-24.9	-0.13	-24.9
t_egress_pt	-0.069	-10.2	-0.069	-10.2

Table 4. 1 β estimation results by Arentze and Molin (2013)

Parameter	Value(<5km)	r-Value	Value(<20km)	r-Value
t_access_walk	-0.11	2.558	-0.11	1.392
t_access_bike	-0.095	2.209	-0.095	1.203
t_access_pt	-0.084	1.953	-0.084	1.063
t_wait_access	-0.073	1.698	-0.073	0.924
t_main_bike	-0.076	1.767	-0.076	0.962
t_main_car_short	-0.043	1.000	-0.079	1.000
t_main_pt_short	-0.058	1.349	-0.074	0.937
t_main_transfer	-0.097	2.256	-0.097	1.228
t_park_car	-0.079	1.837	-0.079	1.000
t_wait_egress	-0.112	2.605	-0.112	1.418
t_egress_walk	-0.101	2.349	-0.101	1.278
t_egress_ptbike	-0.13	3.023	-0.13	1.646
t_egress_pt	-0.069	1.605	-0.069	0.873

Table 4. 2 β_i estimation results

4.5 Conclusion

In this chapter, the research model based on MATsim is presented, which has gone through a three-phase evolution. One thing that remains static is the multi-agent representation, which simulates one-day schedule of each individual of given population instead of every single vehicle within a road network as conventional trip-based microscopic model does. Such representation brings extra complexity to the traffic flow optimization goal, as the activity chain of each agent have to be taken into consideration in addition to its travel choices, resulting in a multidimensional simulation process consisting of three steps: demand generation, network loading and replanning. Demand generation is a mathematical procedure to produce agent plans, which is out of the scope of this chapter, and will be discussed in Chapter 5; the flow propagation (network loading) relies on a queue-based time-step model, with route assignment carried out by a shortest-path implementation of Dijkstra algorithms; meanwhile the activity choices are optimized using an iterative process based on genetic algorithms, with the overall performance of agent plans evaluated via a fitness function (replanning). This simulation process has witnessed major changes in terms of configurations and choice options, which in turn significantly enhanced the simulation capacity of MATsim.

The initial version is a unimodal simulation process that merely generates vehicular traffic flows regardless of vehicle types, and adopts a Charypar-Nagel fitness function to score agent plan, which sums up the utilities of conducting activities and disutilities of traveling, with penalty items regarding late arrival, early departure and short activity duration. An improved version enables bimodal simulation involving public transport vehicles, with the help of transit schedule deprived from GTFS feeds and transit route validated by a map-matching algorithm. The inclusion of a new choice dimension not only lifted the traffic reality MATsim captures to a higher level, but also established the fundamental structure for the implementation of PVN and PTN concepts from supernetwork frame. Backed up by these preconditions, the author tried to purpose a multi-modal simulation approach, integrating existing modules that separately reproduce vehicular traffic modes and slow modes, binding them together with a new criterion in place of Charypar-Nagel function, considering integrated space-time accessibility derived from personalized supernetwork frame. In this new function, the travel disutilities of different transport modes as well as transition actions (boarding, alighting and waiting) are taken into consideration, capturing the activity-travel program and choice preference of real-time individual at a high level of detail, with the corresponding parameters extracted and converted from previous experiment results.

This chapter introduces the default MATsim model and the integrated multi-modal simulation approach proposed by the author respectively, the latter of which is labeled as the major contribution of this thesis. The methodology will be implemented in Chapter 5 with a case study of Eindhoven region, where the second contribution (GPS diary incorporation as data source for demand generation) will be addressed.

5 Case study

5.1 Introduction

In this project, the Metropoolregio Eindhoven (MRE) was chosen as the study area. With a population of around 753,426, serving as the rail transport hub of the Noord-Brabant province, and possessing the second busiest airport in the Netherlands, this region satisfies the condition of high dense population and traffic (Rijkswaterstaat, 2015), being a good field lab for smart mobility study. Also, the extensive network of bicycle paths in this region suggests that cycling would be one of the major transport modes for the inhabitants, which would provide a good sample for multi-modal simulation. The data availability makes up another reason for the choice, as this area is where some famous research and development agencies such as TNO, TU/e, Philips, etc., are located, providing abundant localized data sources with expertise. The conditions altogether would guarantee the viability of this case study, as implementation of the proposed method.

This chapter would present the implementation process of the purposed method explained in Chapter 4 in terms of a case study. In Chapter 5.2, the geographic boundary of the study area and the network extracted within will be briefly introduced. This is followed by a detailed explanation of GTFS processing, the crucial step to construct a multi-modal network structure in Chapter 5.3. Chapter 5.4, as the majority of this chapter, will focus on demand generation, as it has to be localized considering the availability and characteristics of travel diary data, and involves complex coding for its implementation. Also, the measure applied for the incorporation of GPS diary data is presented in this segment, marked as the secondary major contribution of this research. Chapter 5.5 will briefly present the simulation configurations and introduce several simplifications applied in this study. The simulation results, together with analysis, will be presented in Chapter 5.6, coming to a conclusion in Chapter 5.7.

5.2 Initial extraction

The network used in this research is an OSM extraction downloaded from http://download.bbbike.org/osm/bbbike/Eindhoven/ which covers a large part of the MRE, including the following municipalities: Asten, Best, Deurne, Eersel, Eindhoven, Helmond, Nuenen, Oirschot, Someren, Son en Breugel, Valkenswaard, Veldhoven, Waalre, Gemert-Bakel, Heeze-Leende, Laarbeek and Geldrop-Mierlo, as can be seen in Figure 5.1. The boundary of the study area ranges from 51.33° N to 51.58° N and 5.26° E to 5.75° E, shaping a road network consisting of three motorways (A2, A50 and A67), four provincial roads (N279, N69, N397 and part of N266) and roads of lower levels.

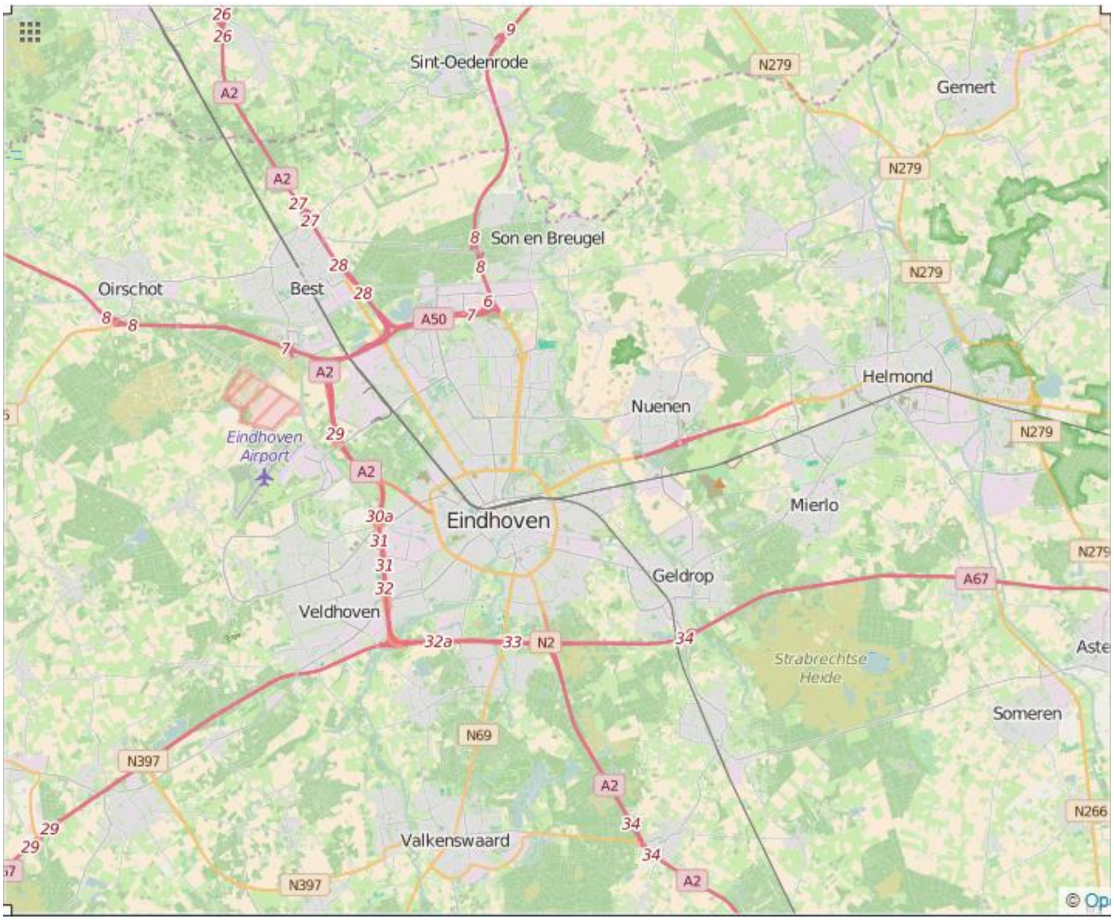


Figure 5. 1 Study area

The original extraction, stored in osm.pbf format, was converted with the help of Osmosis module, a java library specialized at processing OSM data. The following keywords were chosen to extract the road information, based on the definition from Osmosis wiki

```
--rb file="……"
--tf accept-ways
highway=motorway,motorway_link,trunk,trunk_link,primary,primary_link,secon
dary,secondary_link,tertiary,tertiary_link
--wx ……
```

(http://wiki.openstreetmap.org/wiki/Osmosis/Detailed_Usage_0.44):

In this case, the railway links were excluded, due to the fact that the shortest-path algorithm always failed to successfully recognized some paths around the Eindhoven centrum station when railway was involved; in other words, the path would always be settled to the rail road since it is the shortest ones even for buses, which is not true in real life. Also, the percentage of train travelers in both data sets that satisfy the boundary constraints were small enough to be ignorable, as the municipalities within the study area are mostly accessible with buses, and would cost much less, which made up for another reason to exclude the train modes for a better simulation reality.

The simulation of MATsim mainly uses Euclidean distance for calculation, which requires Cartesian coordinates for all the input data, therefore this network was also converted from WGS84 coordinate to the local coordinate system of EPSG:25831, with the help of geometry java library. A total of 35213 nodes and 83624 corresponding links were gathered as a result.

5.3 GTFS processing

A typical GTFS feed is mainly composed of following elements:
1. Agency: agencies that provide the data;
2. Stops: GPS record of the location of each stop;
3. Routes: the official name of each public transit lines;
4. Trips: the sequence of the stops of each route;
5. Stop_times: time table of each route;
6. Calendar: the availability of public transit service;
7. Shapes: rules for drawing lines that represent routes;

Based the information above, a basic transit schedule file can be obtained with the help of GTFS2MATSim module. This intial output consists of two parts:
1. transitStops: Describing the id, name and location of each stop, as well as the link assigned to it in OSM system;
2. transitRoutes: Describing the stop references and sequences of each route at each departure time;

The next step is to generate a transit vehicle file. Since no open data sources records the schedule of each vehicle, a rigorous solution is to estimate the vehicle id based on the official time table of each route, given the departure time at each origin/destination station and timespan for a whole trip. A sample estimation result is shown in Table 5.1, illustrating the vehicle id of the one departs at a specific time from the origin station of route_id 1762.

1762_0	1762_1	1762_2	1762_3	1762_4
5:38:00	6:08:00	6:41:00	7:01:00	7:21:00
6:00:00	6:30:00	7:07:00	7:27:00	7:47:00
6:26:00	6:51:00	7:31:00	7:51:00	8:11:00
6:49:00	7:17:00	7:57:00	8:17:00	8:37:00
7:11:00	7:41:00	8:21:00	8:41:00	9:01:00
7:37:00	8:07:00	8:47:00	9:07:00	9:27:00
8:01:00	8:31:00	9:11:00	9:31:00	9:51:00

Table 5. 1 Sample of vehicle estimation

However, a simplified procedure using random vehicle_id generation is applied in this thesis, as this vehicle file does not prove to affect the simulation reality significantly.

With all the information given, the finale comes with the completion of the transit schedule file. The link sequence obtained in previous step has to be written into the initial schedule file, so it is with the vehicle references. The output files can be found in Figure 5.2 and 5.3 respectively.

The GTFS feed used in this research was retrieved from an open source data site http://gtfs.ovapi.nl/, containing the transit information of whole Netherland. This source is effective from March 4th till August 7th 2016 which, unfortunately, does not fully match the time period of diary data collected for analysis. This discrepancy can be considered ignorable, because no major change occurred in public transit routes recent years in the study area.

```
<vehicleDefinitions>
    <vehicleType id=" ">
        <capacity>
            <seats persons=" "/>
            <standingRoom persons=" "/>
        </capacity>
        <length meter=" "/>
        <width meter=" "/>
        <accessTime secondsPerPerson=" "/>
        <egressTime secondsPerPerson=" "/>
        <doorOperation mode=" "/>
        <passengerCarEquivalents pce=" "/>
    </vehicleType>
</ vehicleDefinitions >
```

Figure 5. 2 transitVehicle file

Before any further processing, this GTFS feed has to be simplified to the local level, as test result revealed that reading in the whole data set would take more than an hour. The routes that depart from or arrive at Eindhoven were first picked out from route.txt file as the foundation of simplification; the records that match route_id were then extracted from trips.txt file, resulting in localized trip_id and shape_id , which in turn serve as extraction reference for stop_times.txt and shapes.txt respectively. Eventually stops.txt was checked against stop_id in stop_times.txt, finalizing the simplification procedure. All the files mentioned above were also adjusted to match the parsing sequence of gtfs2matsimtransitschedule module.

```
<transitSchedule>
    <transitStops>
        <stopFacility id=" " x=" " y=" " linkRefId=" " name=" " isBlocking="true "/>
    </transitStops>
    <transitLine id=" ">
        <transitRoute id=" ">
            <transportMode>bus</transportMode>
            <routeProfile>
                <stop refId=" " arrivalOffset=" " departureOffset=" " awaitDeparture=" "/>
            </routeProfile>
            <route>
                <link refId=" "/>
            </route>
            <departures>
                <departure id=" " departureTime=" " vehicleRefId=" "/>
            </departures>
        </transitRoute>
    </transitLine>
</transitSchedule>
```

Figure 5. 3 transitSchedule file

Following the parsing of GTFS feed is the matching procedure, which draws the public transit path on the given network. The shortest-path solution, as is mentioned in Chapter 4, was not as effective as expected. A typical failure can be found in Figure 5.4, as the algorithm failed to find the real shortest path from the stop marked in green toward the Eindhoven centrum station (up in north). This was because the link between the marked stop and its next one was defined as oneway in OSM data base; that means, the bus cannot travel back on the same path as the information from shape.txt suggests. Though the UI enables manually adding nodes and links, the coordinates of the added nodes were not correct, which did not help solving this issue.

A solution the author found that partially compensate this shortage was to manually re-editing the output network file, by merging the lines with wrong coordinates into a single link of this path. The manual editing process was repeated until no errors remain.

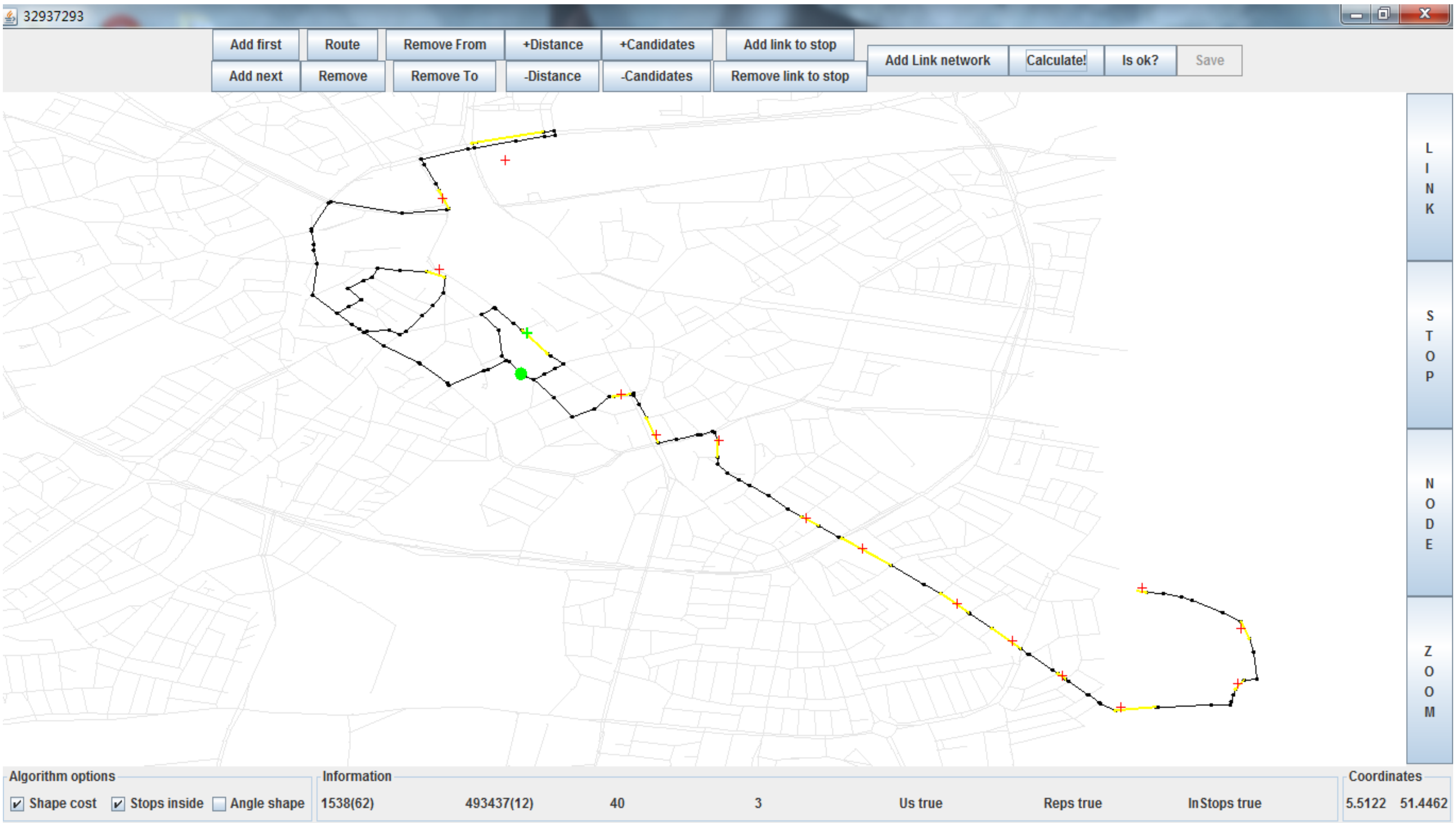


Figure 5. 4 Map-matching process with unrecognized link

In total, 47 public transit lines were validated after GTFS processing, excluding 4 routes that had more than 20% of their stops out of the study area: 103, 150, 149 and 121 (route name). As a result, 6701 multi-modal links were added including the stop links, forming the PTN layer together with corresponding nodes. A sample output file is displayed below in Figure 5.5

```
<network>
    <nodes>
        …………
    </nodes>
    <links capperiod=" " effectivecellsize=" " effectivelanewidth=" ">
        <link id=" " capacity=" " freespeed=" " permlanes=" " from=" " to=" " length=" "
modes="bus,car" oneway="1" origid=" "/>
    </links>
</network>
```

Figure 5. 5 multi-modal network sample

5.4 Demand generation with GPS diary

5.4.1 Data preparation

For this research GPS diary data are used instead of traditional travel survey. Compared with the survey data, GPS diary offer several important advantages: Storing travel information for a longer periods of time; not relying on the memory of respondent; providing linkages among complex trips, tours, and daily travel patterns; most importantly, addressing the dynamic properties of travel behavior, even for *en route* adjustment, that is, adjusting destination and route choice if unexpected situation occurs (Grengs et al, 2008).

The raw GPS data are merely massive points, thus a complex convert procedure is needed to identify the route choice of each individual:
1. Trip/segment identification (TI/SI): Identify the one-mode trip of each journey between different activity locations, usually with the help of rule-based algorithms;
2. Travel mode detection: Detect the travel mode of each trip identified, based on several criteria such as travel speed, acceleration/deceleration, and the information from the GIS database. The accuracy of this step is usually over 90%;
3. Trip purpose imputation: Detect the trip purpose of each journey. Unfortunately, only a few unconvincing approaches are available at this moment, with an accuracy rate from 60% to 75% (Shen and Stopher, 2014);

The GPS diary after processing is still not ready for use. A step prior to demand generation is to exclude erroneous part from the diary record, improving the accuracy of the output plans; this step varies from each data set, considering the variables of participants, sample size and method of preparation (translation or population synthesis). In this research, translation was adopted as a simplified procedure due to limited sample size that could not capture the travel pattern of the whole population.

The GPS diary used for this study was collected from 2012 to 2013 in the study area, where the participants were given a GPS-device that automatically recorded their locations. 84643 records were collected among 327 participants within 2 years, which was a good fit for the purpose of studying personalized supernetwork. As is mentioned in Chapter 4, the theoretical model was based on the fact that each person can choose among limited number of possible activity-travel schedules to execute his/her plan of the day, therefore a multi-day diary would better represent this characteristic than some travel patterns derived from national survey. A typical one-day record can be found in Table 5.2, including the most valuable variables for this research: user id, activity/travel location, start time and end time, duration, travel distance and travel mode, which recorded multiple trips within a journey, representing the route choice at a high level of detail.

user	start	end	type	distance(m)	activityType	duration(min)	lon	lat
5045	5:00:00	6:28:08	ACTIVITY		home	88	5.708544	51.53852
5045	6:28:08	7:05:45	CAR	48966.8		37		
5045	7:05:45	7:18:47	WALKING	1341.39		13		
5045	7:18:47	16:54:00	ACTIVITY	4587.67	paidwork	575	5.865834	51.84591
5045	16:54:00	16:58:31	WALKING	99.96		4		
5045	16:58:31	17:56:13	CAR	44813.7		57		
5045	17:56:13	5:00:00	ACTIVITY	544.81	home	663	5.708544	51.53852

Table 5. 2 GPS diary sample

This GPS diary, though validated in advance, still possesses the shortage of low overall accuracy (less than 70%). The main reason for such issue lies in the fact that the GPS device would lose signal in some area and gave incorrect location information, which could hardly be found out through algorithm-based validation, as the inconsistency between two activities/trips would not hinder its functionality. A further validation process is demanded for this data source:

1. Coordinate validation: The records where the coordinates were invalid could first be filtered out, as they either gave negative values or 0/90. They were usually caused by signal loss, and can be checked against the neighboring activities/trips and corrected.

2. Record filtering: Given the boundary of the study area, the records that were out of the range were excluded, considering the fact that MATsim simulation cannot settle a 9999 zone to include the agents that is out of the network. Though there might be a chance that some of the excluded records were usable if is not due to GPS device defect, the author did not find a good solution to distinguish them, and manual checking would take too much effort without significant improvement of simulation reality; thus exclusion would be a time-wise choice.

3. Mode correction: Even though mode detection is believed to have the highest accuracy among GPS validation procedure, the sample data still contains some erroneous records, for example, a long distance trip with mode defined as walking or cycling, and an in-house movement recognized as trip, et cetera. A set of rules corresponding to the experiments carried out by Arentze and Molin (2013) was adopted to correct the modes:
For walking trips, if the travel distance is longer than 2 kilometers, the mode would be reset to public transit (pt), which is also the case for cycling trips longer than 5 kilometers;
For car and pt trips, if the travel distance is less than 200 meters and this is not en route (at the middle of a journey), the modes would be reset to walking.

At the end of the process, a total of 9145 records gathered from 177 participants were selected for further study.

5.4.2 Demand generation

Theoretically, the most rigorous use of the diary data is to categorize the records according to day of the week, and carry out multiple simulations for each category, which, however, would be meaningless due to the very limited sample size. Apart from that, lack of demographic information of the participants stopped the author from generating a synthetic population out of the diary record. Fortunately, for the personalized supernetwork approach, representing the travel pattern of the whole study area is not the major concern; instead, reproducing the selection among multiple plans available for a single agent is the main target. Solutions for such scenario were not found in previous examples of MATsim yet, so the author decided to amplify the sample size via merging the data. Two assumptions were made for this purpose:

1. The plans from the data sources were assumed to be carried out at the same day by different members of the population;
2. The merged population satisfies the demographic characteristics of the study area;

With the above assumptions, though it is not possible to assign multiple initial plans to each agent as is supposed to theoretically, the possible alternatives are included in the simulated population that partially share the same home or work locations, which can be viewed as simulating the limited number of daily plan choices for each participant involved.

The demand generation of this project was inspired by the work of Horni and Balmer (Chapter 42 of Nagel and Axhausen, 2015), who made use of Switzerland census data and national travel survey for simulation, with the help of GPS records. The census data collected the demographic information and home/work location of each participant that took the travel survey, as well as the coordinate information of locations for different activities across the nation. From this data source, a facility file that collected the coordinates of activity and home locations was generated. On the other hand, the original plans derived from the travel survey were translated into an intermediate population, mainly recording the activity type and duration, which were then shuffled and assigned to the agents. For each simulated agent, the home and work location was settled as the same as that in the census file, while the plan it executed was a random choice from all the alternatives, with the activity location randomly picked from the facility file within a radius of current position. The radius would double if no facility was found, until the activity was successfully placed. Such simulated population can be assumed as a less rigorous synthetic one of the study area.

For this project, a similar method was adopted, but with personalized activity facilities instead of a collection of facility locations from points of interest (POI) data, based on the assumption that limited number of activity type and location will be chosen in a personalized supernetwork. Several files were prepared for this purpose:

Census.txt: Records the id of each participant, the zone id of his/her working locations and home coordinates. If the participant has no job, the zone id is set to -1;

Business_census.txt: Records the coordinate of each facilities and the type of activity it holds. This was derived directly from the GPS diary, with 8 categorized activity types: work, unpaid work, shopping, recreation, service (see the doctor, buy the ticket, deal with issues in city hall, etc.), bring/pickup people, education and social communication.

Municipalities.txt: The whole study area was separated into small square polygons with a length of 8000 meters. Each polygon was given a zone id, which was recorded in this file, together with the coordinates of the center of the zone.

Travelsurvey_persons.txt: Records the id of each participant and the day he/she received the survey in a year.

Travelsurvey_trips.txt: The most important file for demand generation. Its variable mainly includes person id, coordinates of origin and destination of the travel, travel mode and the type and duration of the next activity. Each event in the original diary was integrated into a single line for the sake of efficient input, which unfortunately disabled the possibility to reproduce multi-trip (multi-leg) journeys.

Given the data files mentioned above, an intermediate population of 2726 agents was generated. The activities in the intermediate plans were then renamed according to its duration, for instance, if an activity typed as “work” lasts for 2 hours, it will be renamed into “w2”. This step would further define each activity in a more precise pattern, so that the accuracy of the activity score will improve, as the related time variables (activity start time, end time, typical duration) would be customized to include exceptions that are normally not present in simple plans. For example, if an agent works from 19:00 to 21:00 after a dinner break, traditional agent simulation would either ignore it or give a negative score since the activity start time is later than normal daily work (usually starts from 9:00 to 17:00), which is not the case in real life, as such activity usually means extra wage; while in this scenario, this type of activity is also simulated with a positive score since it is part of the plan, so long as the activity type “w2” is correctly defined. Consequently, the behavioral reality of real life person will be better captured.

The re-defined intermediate population was then assigned to the population, generating the initial demand as the input for simulation. A sample of the plan can be found in Figure 5.6.

```
<population>
    <person id="10082_362" age="53" employed="yes">
        <plan selected="yes">
            <act type="h8" facility="3119" x="682806.5018" y="5702608.303" end_time="08:26:02" />
            <leg mode="bike" dep_time="08:26:02">
            </leg>
            <act type="w3" facility="3105" x="669845.8245" y="5695325.474" max_dur="03:28:00"
end_time="11:54:29" />
            <leg mode="bike" dep_time="11:54:29">
            </leg>
        </plan>
    </person>
</population>
```

Figure 5. 6 Sample population file

5.5 Simulation

The simulation procedure, also known as network loading, is carried out by MATsim core module that implements a time-step model with queue-based approach (Dobler, 2010), as has been explained in Chapter 4. Customized configuration is necessary for the module to be functional, mainly involves the input and output path of related files, the choice of *mobsim*, the start and end time of simulation, the replanning strategies and most importantly, the definition of activity and scoring.

For the scenario of Eindhoven region, multi-modal simulation has to be taken into consideration, which was enabled with strategy modules defined in Figure 5.7. 10% of agents were allowed to modify their plans after each iteration, either by mutate the time of activities, change the route of trips/journeys, or change the mode of travel. The multi-modal contribution of bike and walk travel modes, as is mentioned in Chapter 4, was realized via adding a description of teleported modes in a separate module.

Apart from the configurations that implement supernetwork structure, the disutility functions that serve as binding pieces for this approach were also integrated. A first step is to construct a class that overrides the Charypari-Nagel function and calls the controller to continue simulation process, which was also reflected in the configuration files, since the marginal utility values has to be adjusted, as is shown in Figure 5.8. A test scenario with default scoring settings was simulated as well, checking the impact of the supernetwork approach integration. For each scenario, results of 100 iterations and 500 iterations were obtained respectively, which would signal whether convergence was reached. Since large quantity of data would be obtained after each simulation attempt, mainly including the improved plans for each agent (output_plans), the events that took place at each time step duration the simulated time period (output_events), the count of transport mode adopted at each time step (legHistogram), and three figures that indicate the average score status, the computation time distribution per iteration and average travel distance per leg respectively. The results will be displayed and analyzed later in this Chapter, as they give a straightforward implication of the solution effectiveness.

```
<module name="strategy">
        <param name="maxAgentPlanMemorySize" value="5" />
        <param name="ModuleProbability_1" value="0.7" />
        <param name="Module_1" value="BestScore" />
        <param name="ModuleProbability_2" value="0.1" />
        <param name="Module_2" value="ReRoute" />
        <param name="ModuleProbability_3" value="0.1" />
        <param name="Module_3" value="TimeAllocationMutator" />
        <param name="ModuleProbability_4" value="0.1" />
        <param name="Module_4" value="ChangeTripMode" />
</module>
<module name="changeLegMode">
        <param name="modes" value="car,pt,bike,walk" />
</module>
<module name="planscalcroute">
        <param name="beelineDistanceFactor" value="1.3" />
        <param name="teleportedModeFreespeedFactor_pt" value="2.0" />
        <param name="teleportedModeSpeed_bike" value="6.01" />
        <param name="teleportedModeSpeed_" value="13.88888888888889" />
        <param name="teleportedModeSpeed_walk" value="1.34" />
</module>
```

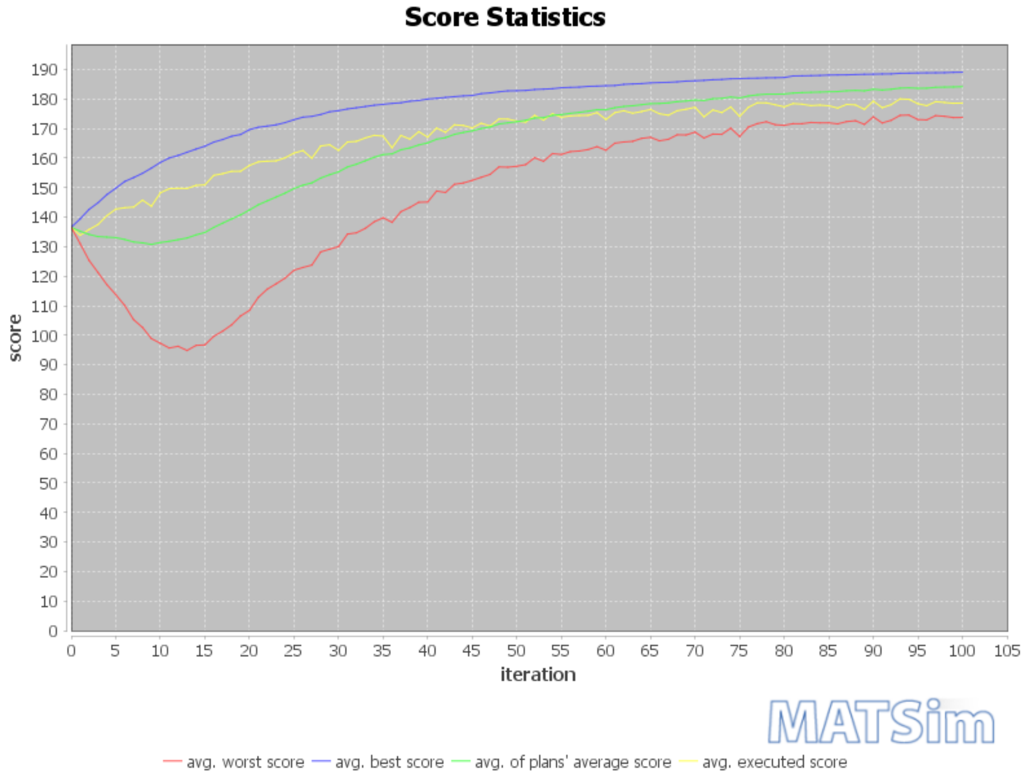
Figure 5. 7 Strategy modules for multi-modal simulation

```
<module name="planCalcScore">
        <param name="learningRate" value="1.0" />
        <param name="BrainExpBeta" value="2.0" />
       <param name="performing" value="+6"/>
        <param name="lateArrival" value="-18.0"/>
        <param name="traveling" value="-6" />
        <param name="waitingPt" value="-8" />
        <param name="utilityOfLineSwitch" value="-9.4"/>
</module>
```

Figure 5. 8 Modified scoring setting

5.6 Result & analysis

5.6.1 Simulation with default settings



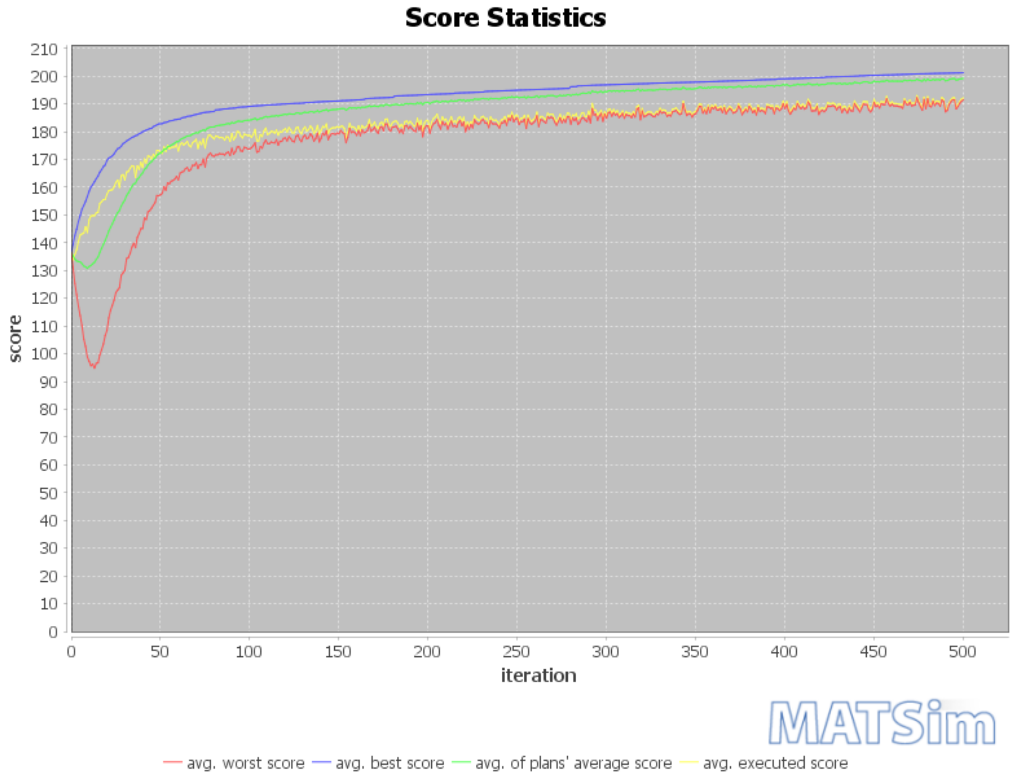


Figure 5. 9 Score status of simulation (100 iterations and 500 iterations)

Figure 5.9 offers a direct perspective for understanding the gradual uprising procedure of average score per plan, as the total count of iteration adds up. Though the scoring curve still does not go horizontal even after 500 iterations, the four curves have been stabilized to some extent, therefore we have enough evidence to assume that a 500-iteration result has reached convergence and UE is achieved, for the agents simulated here. On the contrary, 100-iteration results would prove to be less meaningful, and will not be mentioned further unless for test scenarios.

The computation time spent in this scenario is presented in Figure 5.10. As can be seen in the graph, a single iteration would cost an average of 50 to 55 seconds on a laptop with 8G RAM, except for the ones where an output plan is required, leading to 25 seconds extra. This proves to be the expense of adopting GPS diary data, since it captures the detailed ATP of real individual and possesses extreme complexity due to that. The corresponding plans generated also inherit the complexity and thus should consume more computational effort, which, however, is not very provable as no comparable cases exist. This possible trade-off between computational efficiency and behavioral realism is acceptable though, as in scenarios where tens of thousands of agents would be simulated, computational time is already measured in weeks (Balmer et al, 2009).

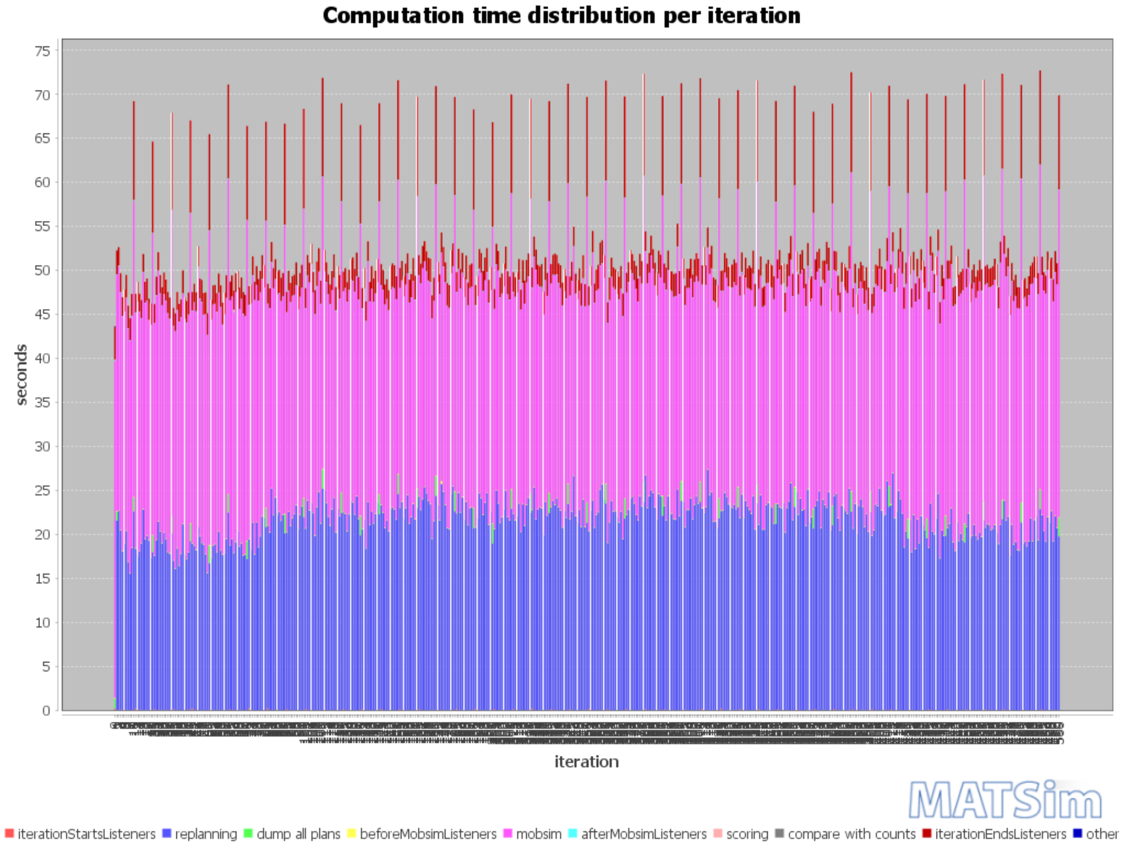


Figure 5. 10 Computation time distribution with default settings

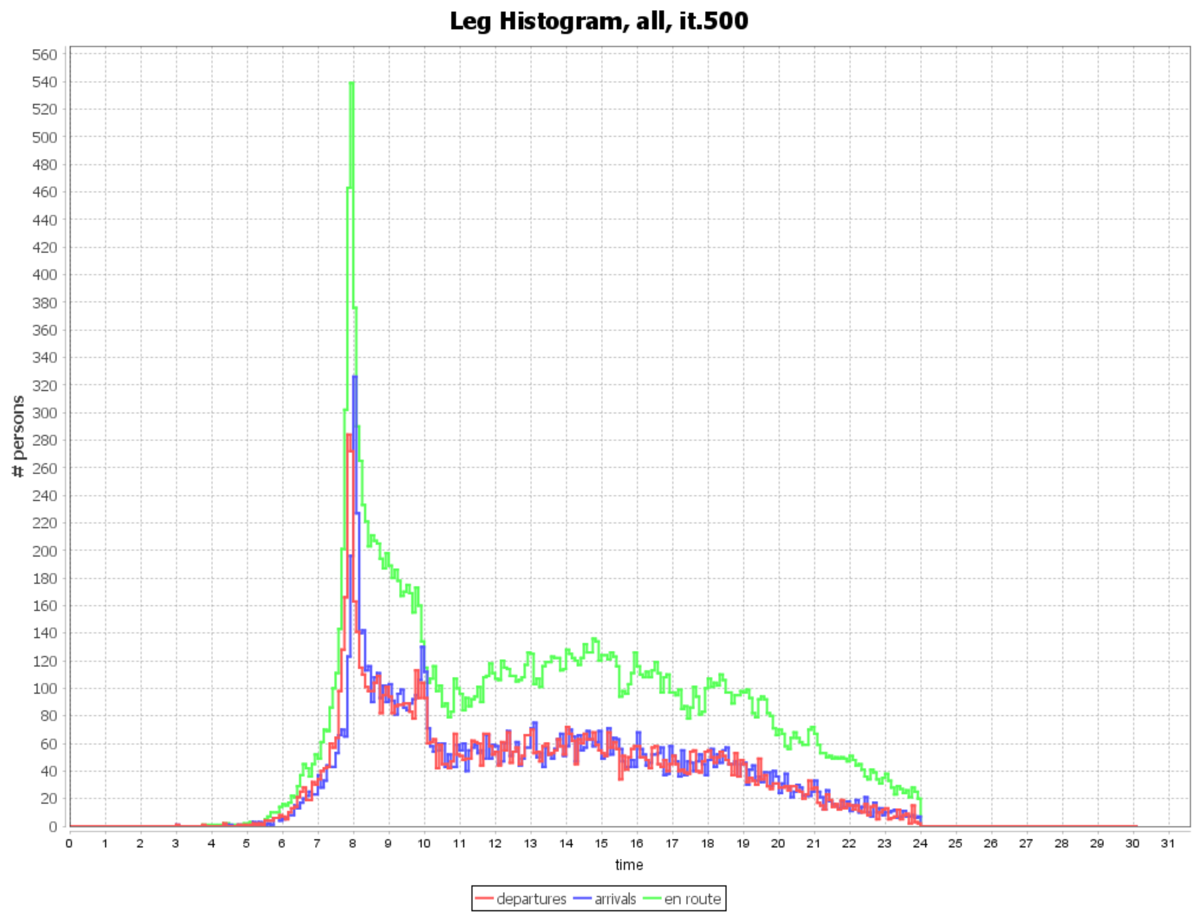


Figure 5. 11 Transport mode distributions

Figure 5.11 illustrates the travel pattern of the total population simulated. The outcome is not as promising as expected, since the most typical travel pattern, namely rush hours, are not apparent here. This would indicate that even though an amplification was carried out for demand generation, the limited sample size of the adopted GPS diary data still hinder it from reproducing the travel pattern of the whole study area. This might affect the accuracy of the simulation result, since the congestion at evening rush hours was not reproduced. However, considering the fact that the study emphasizes the representation of personalized supernetwork, such negative factors will not deny the value of the data used here, as well as the results generated.

Another meaningful result obtained here is the line chart that visualizes the continuous change of average travel distance per trip at each time step of simulation, as can be viewed in Figure 5.13. The corresponding average trip duration had been reduced from 00:21:23 to 00:09:39, meaning that the travel time was largely optimized after iteration. The final trip duration, however, should not be considered as a stable value, as the graph strictly follows a fluctuation pattern due to the random nature of mutation strategy applied here.

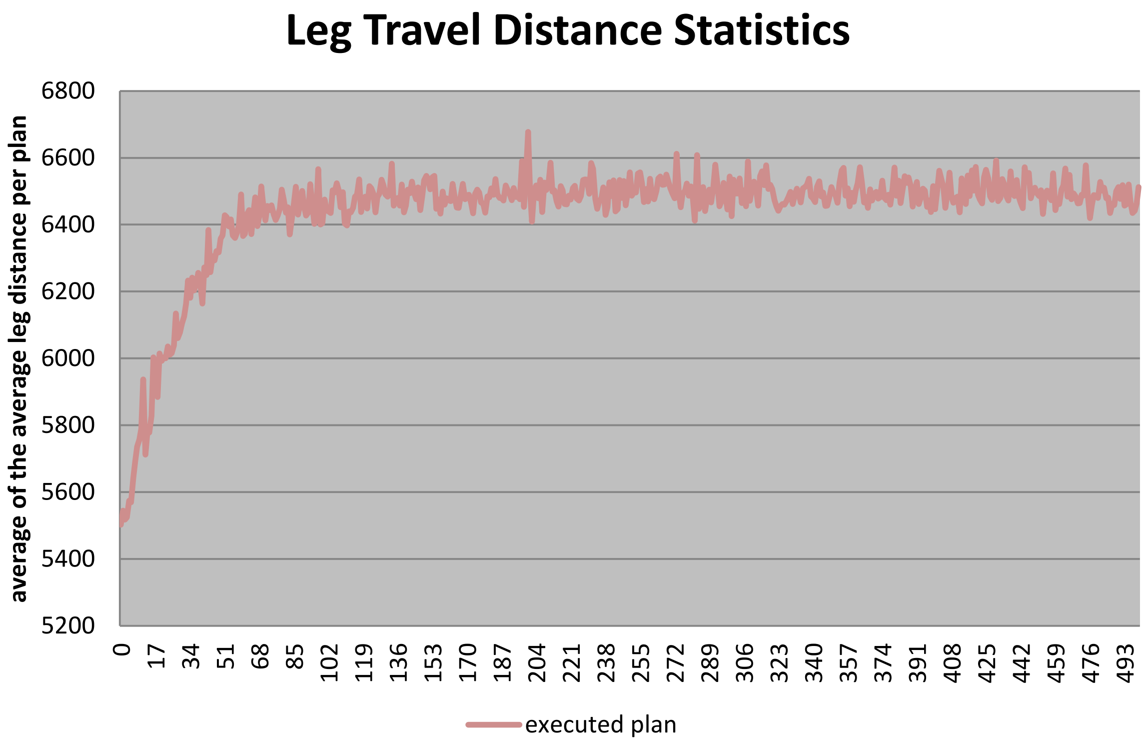


Figure 5. 12 Average leg distance with default settings

5.6.2 Simulation with supernetwork disutilites

Figure 5.13 Score status of simulation with supernetwork representation

Via adopting the new disutility functions, the scoring did appear to have some trouble giving correct values, which was the reason for no graduation displayed on vertical axis. Fortunately, this phenomenon has been proven not to influence the plan selection during replanning process, therefore would not affect the optimization results.

The main advantage of applying the new disutility functions was found in the results displayed in Figure 5.14. In comparison with the results in Figure 5.10, an average of 5 seconds' computation time was saved in each iteration, which would possibly become more significant in a more complicated scenario where huge quantities of agents would be involved.

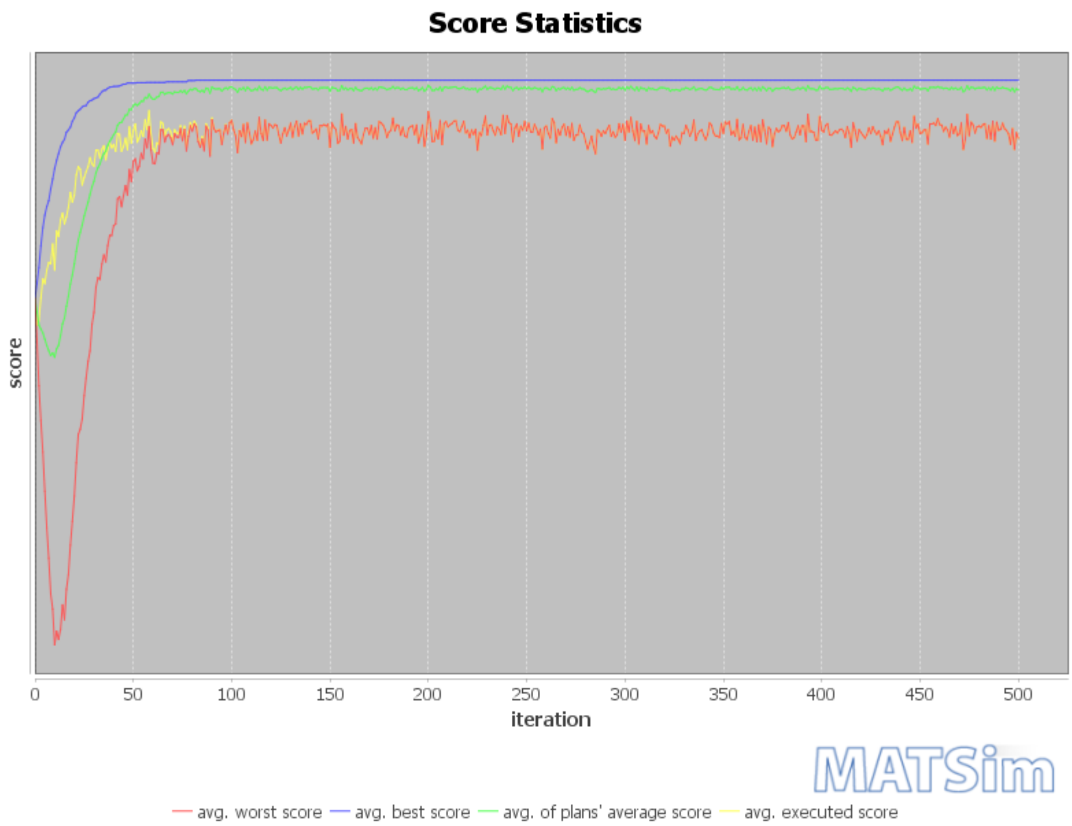


Figure 5. 13 Score status of simulation with integrated accessibility

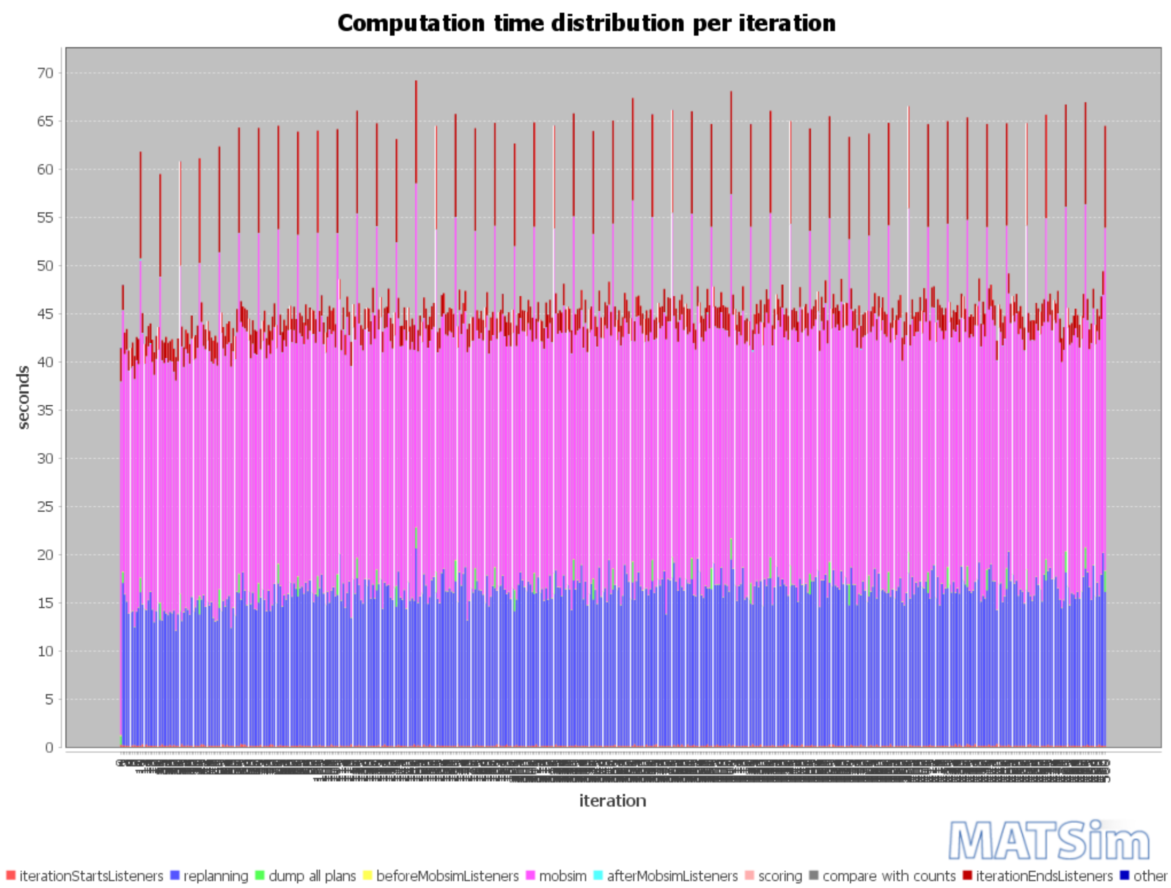


Figure 5. 14 Computation time distribution with integrated accessibility

Another major improvement was found out from two results. A travel time optimization from 00:24:19 to 00:09:57 was carried out corresponding to the line chart in Figure 5.15, indicating a steeper line toward a relatively stable average trip distance, compared with the results from Figure 5.12. This phenomenon, together with an evidence from Figure 5.13, if not considering the abnormal scores, implies that this integrated approach converges quicker compared than that with default settings. The rigor of this conclusion will be further tested in a larger scenario with synthetic population.

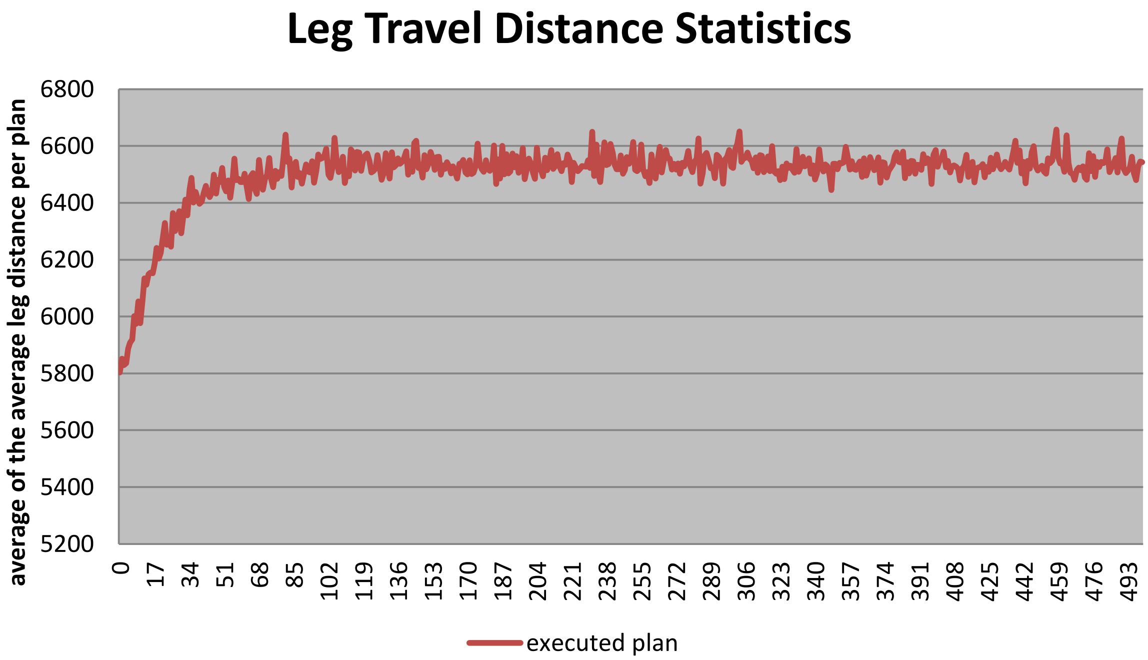


Figure 5. 15 Average leg distance with supernetwork representation

5.7 Conclusion

5.7.1 The integrated method

In this chapter, a case study is carried out in Eindhoven region to test the applicability of the proposed method, which is successful to a large extent. Following a complicated procedure of data tuning and software configuration, MATsim manages to conduct a simulation of around 2,300 agents within the study area, and significantly optimizes their average travel time. The success of the case study, together with some attempts in other researches using MATsim, indicates that same methodology can be applied to various locations around the world (European scenarios in the book Horni et al; Asian scenario in Zhuge et al, 2014 and Ordonez and Erath, 2011), as no major modification is done within the core modules, proving the applicability of the proposed approach.

5.7.2 GPS diary incorporation

For this case study, GPS diary instead of travel survey is used for demand generation. There exist several highlights of the data source, which, unfortunately, does not fully overcome the disadvantages it bring into this case study. Being able to store travel information for a longer periods of time does not help improving the accuracy of within-day simulation; it is true that the data does not rely on the memory of respondent, but the complex validation procedure affects its accuracy, further limiting the effective sample size; addressing the dynamic properties of travel behavior is a huge advantage, but is not made full use of in this scenario, because MATsim is not capable of simulating a travel leg consisting of multiple trips with different transport modes.

Though GPS diary as a more advanced data source does not prove to be superior in this case study for agent-based simulation, the major limitation, namely restricted sample size, can be easily overcome via having more participants involved. In that case, a multi-day simulation, a possible extension mentioned in section 1.4.2, can be realized for large-scale scenario as well; therefore, it is still correct to state that this advanced data source possesses promising potential for multi-modal simulation application.

This chapter presents a detailed implementation process of the proposed method regarding the two main contributions of the thesis: Multi-modal simulation approach using integrated accessibility concept from supernetwork frame, and GPS diary incorporation as data source for demand generation. This case study proves to be successful to some extent. Next Chapter will conclude the whole report and answer the research questions raised at the beginning of the thesis (Chapter 1.2).

6 Conclusion & Discussion

6.1 Conclusion

This chapter will answer the research question defined at the beginning of the thesis, namely "**How to improve the methodology of multi-modal transportation planning?**" from four perspectives corresponding to four sub-questions:

1) What is the state-of-art dynamic traffic assignment approach within transportation planning context?

A literature review regarding DTA is presented in Chapter 3, revealing two mainstreams of state-of-art DTA solutions. Analytical approaches are usually based on extensions of LWR model or incorporation with models from other fields, concentrating on accurate mathematical formulations of given road network, seeking convergence as the solution to traffic optimization. However, they lack a general formulation for universal scenarios, and possess complicated solution procedure inefficient for real-time use, therefore are not yet reliable for ITS application nowadays. In contrast, simulation-based models use iterative process to obtain partial user equilibrium for the network entity, trading solution accuracy for efficiency and applicability, therefore are more promising.

Among the various simulation-based approaches, macroscopic and mesoscopic models focus on traffic density and link travel speed, while microscopic models reproduce each single vehicle on the network, thus capture higher level of detail. This advantage is further enhanced with activity-based DTA that simulate the whole day schedule of individual instead of vehicle, and the side-effect of high computation burden is gradually compensated thanks to the fast development of IT devices. In conclusion, the activity-based agent simulation proves to be the most advanced DTA approach for transportation planning purpose.

2) Considering the characteristics of the multi-modal representation, what is the state-of-art practice platform?

As has been illustrated in Chapter3, TRANSIMS and MATsim are most developed and frequently studied platforms capable of multi-modal agent-based simulations. TRANSIMS is more advanced due to embedded demand estimation and flow propagation that obeys macroscopic rules, while MATsim appears to have more potential for higher data consistency and unique multi-agent representation, which enables further development in various scenarios that require individual resolution of real-life people. This thesis also makes use of this advantage, connecting the fragmented modules within MATsim with integrated space-time accessibility concept from supernetwork structure in terms of disutility functions, to propose a new multi-modal simulation approach for transportation planning purpose, as is demonstrated in Chapter 4.

3) What type of data will be required for the chosen method? How to process the data needed for the case study?

Chapter 5 presents a case study as implementation of the proposed method, and uses GPS diary data for travel demand forecast, which is validated from geographic point records into individual's activity travel program. Apart from the population input, a network extract from OSM database, as well as public transit schedule are required for the multi-modal simulation. The transit schedule are generated from GTFS feed data using a map-matching tool that identifies the travel route of each public transit vehicles as well as several data integration modules in Appendix 2, with the network extract at the same time updated to a multi-layer structure consisting of PVN and PTN. A three-phase process is then carried out for simulation.

The first phase, demand generation creates a synthetic population of the study area, and generates their initial plans. This is followed by network loading, which assigns the plans on the updated network and conduct flow propagation using a queue-based time-step model. The performance of each plan is evaluated according to the proposed disutility function, and modified via time mutation during replanning phase. Phase two and three will be repeated until a given time of iteration is reached (in this case 500 times), viewing as UE achieved. This simulation process, though following the MATsim basic route, has been modified specifically for multi-modal simulation considering vehicular modes and slow modes (cycling and walking) using advanced input source (GPS), can be labeled as the main contribution of this thesis.

4) What is the advantage of the proposed methodology?

In this thesis, a transportation planning method integrating multi-agent simulation and space-time accessibility as well as GPS source is proposed and tested. The integrated accessibility concept derived from supernetwork structure proves to be a suitable solution for multi-modal simulation, as it provides a realistic and profound criterion to evaluate individual's activity-travel program. Together with GPS diary data, the integrated approach is capable of capturing the behavioral realism of real life individual at a high level of detail, thus would theoretically reproduce better traffic reality. On the other hand, compared with analytical solutions, Multi-agent simulation from MATsim best carries out the supernetwork representations, since the bounded rationality of individual could be reproduced via limited memory of each agent, with their travel time optimized by rule of thumb. With the advantages listed above, the integrated approach proves to be theoretically intact and practically applicable, bearing the possibility to replace the traditional four step model that is still in use.

6.2 Discussion

There exist several possible extensions to the current research, which can be worked out in future studies:

Among the transition action defined in the theoretical model, only the boarding-alighting links are simulated in this thesis. The simulation of parking action as well as parking choice has been operationalized in MATsim contrib, which would take some extra effort to incorporate it with current multi-modal system. This integration can be realized in the close future.

The household level simulation is not taken into consideration in this thesis, due to limitations of data and time availability. An attempt has been made to implement it in MATsim (Dubernet and Axhausen, 2015). Combining this type of simulation with the proposed method would call for household level GPS data collection, which will for sure add extra difficulty to the validation procedure, especially for trip purpose imputation stage, as the purpose of multi-person activity within a household can vary. This might raise interest for future studies.

Another improvement lies in demand generation aspect, as large-scale GPS diary data can be collected in future studies for population synthesis of a study area. A combination of simulation performance and optimization results between demand forecast of the same area from two distinct data sources (GPS and NTS) will rigorously prove the pros and cons of each source, seeking potential application of the advanced data type within planning field.

7 Bibliography

1. Arentze, T. & Molin, E. (2013). Travelers' preferences in multimodal networks: Design and results of a comprehensive series of choice experiments. *Transportation Research Part A, 58,* 15–28.
2. Arentze, T. & Timmermans, H. (2001). Creating synthetic populations: approach and empirical results. *Association of European Transport.*
3. Arentze, T. & Timmermans, H. (2004). Multistate supernetwork approach to modelling multi-activity, multimodal trip chains. *International Journal of Geographical Information Science, 18:7,* 631-651.
4. Balmer, M., Axhausen, K. W. & Nagel, K. (2005). Agent-based demand-modeling framework for large-scale microsimulations. *Transportation Research Record, 1985 (1),* 125–134.
5. Balmer, M. (2007). Travel demand modeling for multi-agent traffic simulations: Algorithms and systems, Ph.D. Thesis, ETH Zurich, Zurich.
6. Balmer, M., Rieser, M., Meister, K., Charypar, D., Lefebvre, N., Nagel, K. & Axhausen, K. W. (2009). MATSim-T: Architecture and Simulation Times. Multi-Agent Systems for Traffic and Transportation Engineering, 57–78.
7. Bazzan, A., & Klügl, F. (2013). *A Review on Agent-Based Technology for Traffic and Transportation.* Cambridge: Cambridge University Press.
8. Ben-Akiva, M. E. & Lerman, S. R. (1985). Discrete choice analysis: theory and application to travel demand. *MIT Press series in transportation studies*. Cambridge, MA: MIT Press.
9. Ben-Akiva, M. E., Bierlaire, M., Bottom, J., Koutsopoulos, H. N. & Mishalani, R. (1997). Development of a route guidance generation system for real-time application. *Proceedings of the IFAC Transportation Systems 97 Conference, Chania, Greece,* 433-438.
10. Ben-Akiva, M. E., Gao, S., Lu, L. & Wen, Y. (2015). Combining disaggregate route choice estimation with aggregate calibration of a dynamic traffic assignment model. *Networks and Spatial Economics, Vol 15,* 559-581.
11. Bigazzi, A. Y. & Figliozzi, M. A. (2013). Marginal costs of freeway traffic congestion with on-road pollution exposure externality. *Transportation Research Part A, 57,* 12–24.
12. Cambridge Systematics, Inc. (1994). Impact of urban congestion on business, repaired for national cooperative highway research program project 2–17(5). *Transportation Research Board*, Washington, D. C.
13. Carey, M. (1992). Nonconvexity of the dynamic traffic assignment problem. *Transportation Research Part B, 26(2),* 127–33.
14. Castiglione, J., Bradley, M. & Gliebe, J. (2015). *Activity-Based Travel Demand Models: A Primer. Last accessed 08-06-2016 from* *http://onlinepubs.trb.org/onlinepubs/shrp2/SHRP2_S2-C46-RR-1.pdf*
15. CBS.nl. Population dynamics; birth, death and migration per region. *Last accessed 08-06-2016 from* http://statline.cbs.nl/Statweb/publication/?DM=SLEN&PA=37259eng&D1=0-1,3,8-9,14,16,21-22,24&D2=0&D3=154,211,296,347,356,393,396,492,505,611,792,805,1001,1003,1080,1087,1121&D4=0,10,20,30,40,(l-1)-l&LA=EN&VW=C

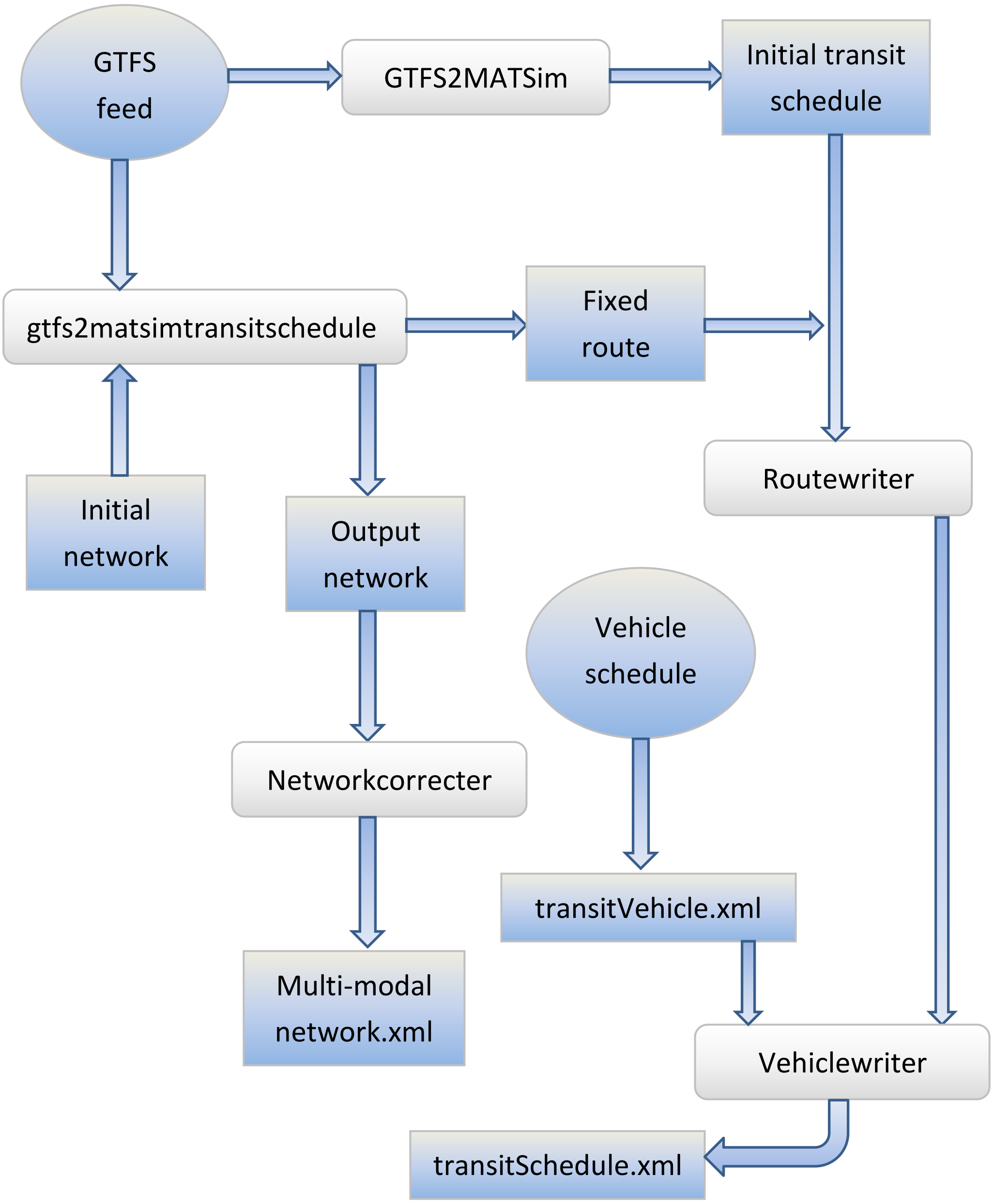
16. Charypari, D.& Nagel, K. (2005). Generating complete all-day activity plans with genetic algorithms. *Transportation, 32,* 369–397.
17. Charypari, D., Axhausen, K. W. & Nagel, K. (2007). An event-driven parallel queue-based microsimulation for large scale traffic scenarios. *Transportation Research Record, 2003,* 35–40.
18. Chiu, Y., Bottom, J., Mahut, M., Paz, A., Balakrishna, R., Waller, T. & Hicks, J. (2011). Dynamic traffic assignment: a primer. *Transportation research circular, E-C 153*.
19. Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research*, *28B,* 269–287.
20. Directive 2010/40/EU of the European Parliament and of the Council of 7 July 2010. *Last accessed 20-07-2016 from* http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2010:207:0001:0013:EN:PDF
21. Dobler, C. (2010). Implementation of a time step based parallel queue simulation in MATSim. in: *10th Swiss Transport Research Conference, Ascona, September 2010*.
22. Dobler, C. & Lämmel, G. (2012). Integration of a multi-modal simulation module into a framework for large-scale transport systems simulation. *Pedestrian and Evacuation Dynamics 2012,* 739-754
23. Dubernet, T. & Axhausen, K. W. (2015). Implementing a household joint activity-travel multi- agent simulation tool: first results. *Transportation, 42,* 753–769.
24. Friesz, T. L., Bernstein, D., Smith, T. E., Tobin, R. L. & Wie, B. W. (1993). A variational inequality formulation of the dynamic network user equilibrium problem. *Operations Research, 41 (1),* 179–191.
25. Garavello, M. & Piccoli, B. (2013). A multibuffer model for LWR road networks. *Advances in Dynamic Network Modeling in Complex Transportation Systems,* 143-161. Springer, New York.
26. Gliebe, J. P. & Koppelman, J. S. (2005). Modeling household activity-travel interactions as parallel constrained choices. *Transportation, 32(5),* 449–471
27. Grengs, J., Wang, X. & Kostyniuk, L. (2008). Using GPS data to understand driving behavior. *Journal of Urban Technology, 15(2),* 33-53.
28. Horni, A., Nagel, K. & Axhausen, K. W. (2015). *The Multi-Agent Transport Simulation MATSim. Last accessed 20-11-2015 from* *http://matsim.org/docs/tutorials/matsim-class-2015*
29. Hu, T. Y. & Liao, T. Y. (2011). An empirical study of simulation-based dynamic traffic assignment procedures. Transportation Planning and Technology, Vol 34, 467-485.
30. Huang, H. J., Liu, P., Liao, F. & Timmermans, H. (2015). Dynamic activity-travel assignment in multi-state Supernetworks. *Transportation Research Part B, Vol 7,* 24–43.
31. Illenberger, J., Flötteröd, G. & Nagel, K. (2008). Enhancing MATSim with capabilities of within-day re-planning. paper presented at: *the IEEE Intelligent Transportation Systems Conference, Seattle, WA*.

32. Jayakrishnan, R., Mahmassani, H. S. & Hu, T. Y. (1994). An evaluation tool for advanced traffic information and management systems in urban networks. *Transportation Research Part C, Vol 2, 129-147.*
33. Jeihani, M. (2007). A review of dynamic traffic assignment computer packages. *Journal of the Transportation Research Forum, 46,* 35–46.
34. Kachroo, P. & Shlayan, N. (2013). Dynamic Traffic Assignment: A Survey of Mathematical Models and Techniques. *Advances in Dynamic Network Modeling in Complex Transportation Systems, vol.2,* 1-25.
35. Lam, W. H. K. & Yin, Y. (2001). An activity-based time-dependent traffic assignment model. *Transport Research Part B, 35,* 549-574.
36. Lee, B. S. (1996). *A cell transmission based assignment-simulation model for integrated freeway/surface street systems. Last accessed 22-07-2016 from* https://etd.ohiolink.edu/!etd.send_file?accession=osu1103731343&disposition=inline
37. Lefebvre, N. & Balmer, M. (2007). Fast shortest path computation in time-dependent traffic networks. In: *The 6th Swiss Transport Research Conference, Ascona, Switzerland, September 2007.*
38. Liao, F., Arentze, T. A. & Timmermans, H. J. (2011). Constructing personalized transportation networks in multi-state supernetworks: a heuristic approach. *International Journal of Geographical Information Science, 25:11,* 1885-1903.
39. Liao, F., Arentze, T. A. & Timmermans, H. J. (2013a). Multi-state supernetwork framework for the two-person joint travel problem. *Transportation, 40,* 813-826.
40. Liao, F., Arentze, T. A., Timmermans, H. J. P. (2013b). Incorporating space-time constraints and activity-travel time profiles in a multi-state supernetwork approach to individual activity-travel scheduling. *Transportation Research Part B, 55,* 41–58.
41. Lighthill, M. J. & Whitham, G. B. (1955). On kinematic waves. i:flow movement in long rivers. ii:a theory of traffic on long crowded roods. *Proceedings of the Royal Society of London, A229,* 281–345.
42. Lin, D., Eluru, N., Waller, S. T. & Bhat, C. R. (2008). Integration of activity-based modeling and dynamic traffic assignment. *Transportation Research Record: Journal of the Transportation Research Board,* 52-61.
43. MATSim. (2015). *Multi-Agent Transportation Simulation. Last accessed 16-11-2015 from http://www.matsim.org.*
44. Merchant, D. K.& Nemhauser, G. L. (1978). A model and an algorithm for the dynamic traffic assignment problems. Transport Science, 12(3), 183–199.
45. Mounce, M. & Carey, M. (2011). Route swapping in dynamic traffic networks. *Transportation Research Part B, 45,* 102–111.
46. Ordonez, S. & Erath, A. (2011). Semi-automatic tool for map-matching bus routes on high-resolution navigation networks. Eidgenössische Technische Hochschule Zürich, IVT - Institut für Verkehrsplanung und Transportsysteme.
47. Peeta, S. & Ziliaskopoulos, A. K. (2001). Foundations of dynamic traffic assignment: the past, the present and the future. Network Spatial Econ, 1(3/4), 233–265.

48. Potts, R. B. & Oliver, R. M. (1972). Flows in transportation networks. Mathematics in science and engineering. New York: Academic Press.
49. Ramadurai, G. & Ukkusuri, S. (2010). Dynamic user equilibrium model for combined activity-travel choices using activity-travel supernetwork representation. Netw Spat Econ, 10, 273–292.
50. Rasouli, S. & Timmermans, H. (2014). Activity-Based Models of Travel Demand: Promises, Progress and Prospects. The International Journal of Urban Sciences 2014, vol 18 (1), 31-60.
51. Richards, P. I. (1956). Shockwaves on the highway. *Operation Research, 4,* 42–51.
52. Rieser, M. (2010). Adding transit to an agent-based transportation simulation concepts and implementation. Ph.D. Thesis, Technische Universität Berlin, Berlin.
53. Rijkswaterstaat. (2015). *Traffic in the Netherlands 2015. Last accessed 04-01-2016 from* *http://www.traffic-quest.nl/en/reports*
54. Schlich, R. & Axhausen, K. W. (2003). Habitual travel behavior: Evidence from a six-week travel diary. *Transportation, 30,* 13-36.
55. Schrank, D., Eisele, B. & Lomax, T. (2012). 2012 Urban Mobility Report. Texas Transportation Institute, College Station, TX. *Last accessed 13-01-2016 from* *http://media2.kjrh.com/html/pdfs/2012urbanmobilityreport.pdf*
56. Szeto, W. Y. & Lo, H. K. (2005). Dynamic traffic assignment: review and future research directions. Journal of Transportation Systems Engineering and Information Technology, 5(5), 85–100.
57. Szeto, W. Y. (2013). Cell-based dynamic equilibrium models. *Advances in Dynamic Network Modeling in Complex Transportation Systems,* 163-192. Springer, New York.
58. Sheffi, Y. (1985). Urban transportation networks: Equilibrium analysis with mathematical programming methods. Prentice Hall
59. Shen, L. & Stopher, P. R. (2014). Review of GPS travel survey and GPS data-processing methods. *Transport Reviews: A Transnational Transdisciplinary Journal*.
60. Shiftan, Y. & Ben-Akiva, M. (2011). A practical policy-sensitive activity-based travel demand model. The Annals of Regional Science, 47 (3), 517–541.
61. TNO. *Smart Mobility. Last accessed 04-01-2016 from* *https://www.tno.nl/en/collaboration/expertise/earth-life-and-social-sciences/smart-mobility/*
62. Wardrop, J. (1952). Some theoretical aspects of road traffic research. *ICE Proceedings: Part II, Engineering Divisions, 1,* 325–362.
63. Wiesenthal, D. A. & Hennessy, D. L. (1999). Traffic congestion, driver stress, and driver aggression. *Aggressive Behavior 25*.
64. Wong, S. C. & Szeto, W. Y. (2012). Dynamic traffic assignment: model classifications and recent advances in travel choice principles. *Central European Journal of Engineering, 2(1),* 1-18.

65. Ziliaskopoulos, A. K. & Waller, S. T. (2000). An Internet-based geographic information system that integrates data, models and users for transportation applications. *Transportation Research Part C, Vol 8,* 427-444.
66. Zhuge, C., Shao, C., Gao, J., Meng, M. & Xu, W. (2014). An Initial Implementation of Multiagent Simulation of Travel Behavior for a Medium-Sized City in China. *Mathematical Problems in Engineering, Vol 2014.*

Appendix 1 Work flow of multi-modal integration process in terms of modules



Appendix 2 Sample coding for GTFS assistive modules

Networkcorrecter

```
public class networkcorrecter {

public static void main(String[] args) {
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
         factory.setValidating(true);
         factory.setIgnoringElementContentWhitespace(true);
         try {
              DocumentBuilder builder = factory.newDocumentBuilder();
              File file = new File("Myproject/eindhoven/input/network/eindhoven-all.xml");
              Document doc = builder.parse(file);

              Node network = doc.getElementsByTagName("network").item(0);
              NodeList networkChildNodes = network.getChildNodes();
              for (int i = 0; i < networkChildNodes.getLength(); i++) {
                  Node networkChildNode = networkChildNodes.item(i);
                  if (networkChildNode.getNodeName().equals("links")) {
                       NodeList links = networkChildNode.getChildNodes();
                       for (int j = 0; j < links.getLength(); j++) {
                            Element link =(Element) links.item(j);
                            NamedNodeMap linkAttributes = link.getAttributes();
                            String length = linkAttributes.getNamedItem("length").getNodeValue();
                            String freespeed = linkAttributes.getNamedItem("freespeed").getNodeValue();
                            Double l = Double.parseDouble(length)/1000;
                            Double v = Double.parseDouble(freespeed)*4;
                            link.setAttribute("length", l.toString());
                            link.setAttribute("freespeed", v.toString());
                            }
                       }
                  }
            // Use a Transformer for output
            TransformerFactory tFactory =
                        TransformerFactory.newInstance();
                        Transformer transformer = null;
```

```
                try {
                    transformer = tFactory.newTransformer();
                } catch (TransformerConfigurationException e) {
                    e.printStackTrace();
                }
                transformer.setOutputProperty(OutputKeys.INDENT, "yes");
                transformer.setOutputProperty(OutputKeys.METHOD, "xml");
                transformer.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
                transformer.setOutputProperty(OutputKeys.DOCTYPE_SYSTEM,
"http://www.matsim.org/files/dtd/network_v1.dtd");
                transformer.setOutputProperty("{http://xml.apache.org/xslt}indent-amount", "2");

                FileOutputStream outputStream = new
FileOutputStream("D:/matsim/Myproject/eindhoven/input/network/eindhoven2.xml");

                DOMSource source = new DOMSource(doc);
                StreamResult result = new StreamResult(outputStream);
                try {
                    transformer.transform(source, result);
                } catch (TransformerException e) {
                    e.printStackTrace();
                }

            } catch (ParserConfigurationException e) {
            } catch (SAXException e) {
            } catch (IOException e) {
            }
        }
}
```

Routewriter

```
public class routewriter {

    public static void main(String[] args) {
        Map<String, List<String>> routesByDeparture = new HashMap<>();
        try {
            BufferedReader reader = new BufferedReader(new InputStreamReader(new
FileInputStream("eindhoven/input/gtfs-nl/temp/finishedTrips.txt")));
            String departureID = null;
            while (true) {
                String line = reader.readLine();
                if (line == null) {
                    break;
                }
                if (departureID == null) {
                    departureID = line;
                } else {
                    String[] split = line.split(";");
                    routesByDeparture.put(departureID, Arrays.asList(split));
                    departureID = null;
                }
            }
        } catch (IOException e1) {
            e1.printStackTrace();
        }

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
         factory.setValidating(true);
         factory.setIgnoringElementContentWhitespace(true);
         try {
             DocumentBuilder builder = factory.newDocumentBuilder();
             File file = new File("eindhoven/input/transitSchedule.xml");
             Document doc = builder.parse(file);

             Node transitSchedule = doc.getElementsByTagName("transitSchedule").item(0);
             NodeList transitScheduleChildNodes = transitSchedule.getChildNodes();
```

```
                    for (int i = 0; i < transitScheduleChildNodes.getLength(); i++) {
                        Node transitScheduleChildNode = transitScheduleChildNodes.item(i);
                        if (transitScheduleChildNode.getNodeName().equals("transitLine")) {
                            NodeList transitRoutes = transitScheduleChildNode.getChildNodes();
                            for (int j = 0; j < transitRoutes.getLength(); j++) {
                                Node transitRoute = transitRoutes.item(j);
                                NodeList transitRouteChildNodes = transitRoute.getChildNodes();
                                for (int k = 0; k < transitRouteChildNodes.getLength(); k++) {
                                    Node transitRouteChildNode = transitRouteChildNodes.item(k);
                                    if (transitRouteChildNode.getNodeName().equals("departures")) {
                                        Node departure = transitRouteChildNode.getChildNodes().item(0);
                                        if (departure != null) {
                                            NamedNodeMap departureAttributes = departure.getAttributes();
                                            String departureID =
departureAttributes.getNamedItem("id").getNodeValue();
                                            List<String> routeIDs = routesByDeparture.get(departureID);
                                            Node route = doc.createElement("route");
                                            for (String routeID : routeIDs) {
                                                Node link = doc.createElement("link");
                                                Node refID = doc.createAttribute("refID");
                                                refID.setNodeValue(routeID);
                                                link.getAttributes().setNamedItem(refID);
                                                route.appendChild(link);
                                            }
                                            transitRoute.insertBefore(route, transitRouteChildNode);
                                        }
                                        break;
                                    }
                                }
                            }
                        }
                    }
                    …………// Use a Transformer for output similar to previous one
                } catch (ParserConfigurationException e) {
            } catch (SAXException e) {
            } catch (IOException e) {
            }
        }
}
```

Vehiclewriter

```
public class vehiclewriter {

    public static void main(String[] args) {
    try{
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
         factory.setValidating(true);
         factory.setIgnoringElementContentWhitespace(true);
            DocumentBuilder builder = factory.newDocumentBuilder();
            Document doc1 = builder.parse(new File("eindhoven/input/transitSchedule3.xml"));
            Document doc2 = builder.parse(new File("eindhoven/input/transitVehicles.xml"));

            Node transitSchedule = doc1.getElementsByTagName("transitSchedule").item(0);
            NodeList transitScheduleChildNodes = transitSchedule.getChildNodes();
            ArrayList<String> vehicleIds = new ArrayList<String>();
            for (int i = 0; i < transitScheduleChildNodes.getLength(); i++) {
              Node transitScheduleChildNode = transitScheduleChildNodes.item(i);
              if (transitScheduleChildNode.getNodeName().equals("transitLine")) {
                  NodeList transitRoutes = transitScheduleChildNode.getChildNodes();
                  for (int j = 0; j < transitRoutes.getLength(); j++) {
                      Node transitRoute = transitRoutes.item(j);
                      NodeList transitRouteChildNodes = transitRoute.getChildNodes();
                      for (int k = 0; k < transitRouteChildNodes.getLength(); k++) {
                          Node transitRouteChildNode = transitRouteChildNodes.item(k);
                          if (transitRouteChildNode.getNodeName().equals("departures")) {
                              Node    departure = transitRouteChildNode.getChildNodes().item(0);
                              if (departure != null) {
                                  NamedNodeMap departureAttributes = departure.getAttributes();
                                  String vehicleId =
departureAttributes.getNamedItem("vehicleRefId").getNodeValue();
                                  vehicleIds.add(vehicleId);
                                  }
                              }
                          }
                      }
                  }
              }
```

```
                Node transitVehicle = doc2.getElementsByTagName("vehicleDefinitions").item(0);

                Set<String> vehicles = new HashSet<String>(vehicleIds);
                for (String vehicleid : vehicles) {
                        Node vehicle = doc2.createElement("vehicle");
                        Node id = doc2.createAttribute("id");
                        id.setNodeValue(vehicleid);
                        vehicle.getAttributes().setNamedItem(id);
                        transitVehicle.appendChild(vehicle);
                }

               …………// Use a Transformer for output

           } catch (ParserConfigurationException e) {
            } catch (SAXException e) {
            } catch (IOException e) {
            }
      }

}
```

Appendix 3 Coding of utility function modification

```
public class scoring {
    public static void main(String[] args) {
        String configFile = "eindhoven/config-disutility.xml" ;
        final Scenario scenario = ScenarioUtils.loadScenario(ConfigUtils.loadConfig(configFile));

        Controler controler = new Controler(scenario);
        controler.setScoringFunctionFactory(new ScoringFunctionFactory() {
            @Override
            public ScoringFunction createNewScoringFunction(Person person) {
                SumScoringFunction sumScoringFunction = new SumScoringFunction();

                final CharyparNagelScoringParameters params =
                    new CharyparNagelScoringParameters.Builder(scenario, person.getId()).build();
                sumScoringFunction.addScoringFunction(new CharyparNagelActivityScoring(params));
                sumScoringFunction.addScoringFunction(new CharyparNagelLegScoring(params,
scenario.getNetwork()));
                sumScoringFunction.addScoringFunction(new CharyparNagelMoneyScoring(params));
                sumScoringFunction.addScoringFunction(new CharyparNagelAgentStuckScoring(params));

                sumScoringFunction.addScoringFunction(new SumScoringFunction.LegScoring() {
                    private double score;
                    @Override public void finish() {}
                    @Override
                    public double getScore() {
                        return score;
                    }

                private int ccc=0 ;
                private double calcLegScore(final double departureTime, final double arrivalTime, final Leg
leg) {
                    double tmpScore = 0.0;
                    double travelTime = arrivalTime - departureTime;
                    ModeUtilityParameters modeParams = params.modeParams.get(leg.getMode());
```

```
if (modeParams == null) {
    if (leg.getRoute().getDistance() < 5000){
        if (leg.getMode().equals(TransportMode.transit_walk)
||leg.getMode().equals(TransportMode.walk)) {
            modeParams = params.modeParams.get(TransportMode.walk);
            tmpScore += travelTime * modeParams.marginalUtilityOfTraveling_s*2.581;
        }
        else if (leg.getMode().equals(TransportMode.pt)){
            modeParams = params.modeParams.get(TransportMode.pt);
            tmpScore += travelTime * modeParams.marginalUtilityOfTraveling_s*1.349;
        }
        else if (leg.getMode().equals(TransportMode.bike)){
            modeParams = params.modeParams.get(TransportMode.bike);
            tmpScore += travelTime * modeParams.marginalUtilityOfTraveling_s*1.767;
        }
        else {
            modeParams = params.modeParams.get(TransportMode.other);
        }
}
else {
    if (leg.getMode().equals(TransportMode.transit_walk)) {
        modeParams = params.modeParams.get(TransportMode.walk);
        tmpScore += 0.887*travelTime * modeParams.marginalUtilityOfTraveling_s*1.335;
    }
    else if (leg.getMode().equals(TransportMode.pt)){
        modeParams = params.modeParams.get(TransportMode.pt);
        tmpScore += 0.887*travelTime * modeParams.marginalUtilityOfTraveling_s*0.937;
    }
    else if (leg.getMode().equals(TransportMode.car)){
        modeParams = params.modeParams.get(TransportMode.car);
        tmpScore += 0.887*travelTime * modeParams.marginalUtilityOfTraveling_s;
    }
    else if (leg.getMode().equals(TransportMode.bike)){
        modeParams = params.modeParams.get(TransportMode.bike);
        tmpScore += 0.887*travelTime * modeParams.marginalUtilityOfTraveling_s*0.962;
    }
    else {
        modeParams = params.modeParams.get(TransportMode.other);
```

```
                                        }
                                    }
                                }
                            }
                            @Override
                            public void handleLeg(Leg leg) {
                                double legScore = calcLegScore(leg.getDepartureTime(), leg.getDepartureTime() + 
leg.getTravelTime(), leg);
                                this.score += legScore;
                            }
                        });
                        return sumScoringFunction;
                        }
                    });
            controler.run();
            }
}
```