OULUN YLIOPISTO
UNIVERSITY of OULU

FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Thang Luu

OTSS: OULU TRAFFIC SIMULATION SYSTEM

Master's Thesis
Degree Programme in Computer Science and Engineering
September 2017

Thang L. (2017) OTSS: Oulu Traffic Simulation System. University of Oulu, Degree Programme in Computer Science and Engineering. Master's Thesis, 56 p.

ABSTRACT

This thesis presents the design and the implementation of Oulu Traffic Simulation System (OTSS), a traffic simulation system for the City of Oulu, Finland. Following agent-based approach, the simulation generates artificial agents that represent the population synthesis of the City of Oulu. Data from several sources, including official statistics, government-organized open data and crowdsourced information were collected and used as input for the simulation. Two traffic demand models are presented in this thesis: (1) the random model which generates traffic trips as random, discrete events; and (2) the activity-based model which defines traffic trips as sequential events in the agents' day plan. The software development of the system follows the spiral model of software development and enhancement. During the implementation, several development cycles were conducted before the UML software design. The system was executed on two computation systems to test its real-time performance. To evaluate the traffic models, data extracted from the simulation was compared with aggregated survey data from Finnish Transport Agency and traffic count stations around the city. The results showed that a typical server is capable of running the simulation, and even though there were differences in the duration and distance of individual trips, the simulation reflects real-life traffic count significantly well.

Keywords: smart city, smart transportation, traffic simulation

TABLE OF CONTENTS

FOREWORD

This thesis was done at the Center for Ubiquitous Computing at the University of Oulu. I would like to thank my supervisors Dr. Timo Koskela, Mr. Toni Alatalo and Prof. Timo Ojala for their continuous support during my studies and thesis.

I am very grateful to Finland and especially University of Oulu for giving me the opportunity to pursue my master's degree.

Finally, I am thankful to my classmates in the CSE/UBICOMP International Master Programme in 2014. You made my European journey enjoyable and meaningful!

Oulu, 06.10.2017

Thang Luu

ABBREVIATIONS

3D	Three-dimensional space
GUI	Graphic User Interface
IoT	Internet of Things
ITS	Intelligent Transportation System
MATSim	Multi-Agent Transport Simulation
MovSim	Multi-model open-source vehicular-traffic Simulator
ODbL	Open Data Commons Open Database License
OS	Operating System
OSM	OpenStreetMap
OSMF	OpenStreetMap Foundation
POI	Point of Interest
SUMO	Simulation of Urban Mobility
TraCI	Traffic Control Interface
UI	User Interface
UML	Unified Modeling Language
XML	eXtensible Markup Language

1. INTRODUCTION

Since early 1990s, "smart city" has covered many aspects of everyday lives of urban citizens. One of its application is smart transportation, which take advantage of sensor networks, the Internet of Things (IoT) and other technical means to change the traditional transport system, and to establish the smart traffic management system to achieve the integration of urban planning, construction, management and operations.

During the urban planning and transportation network designing phase, traffic management methods and algorithms need to be calibrated and tested to ensure their efficiency and effectiveness. Because such a step is too big to be done in real-life, it is simulated in computer systems instead.

Traffic simulation packages have been developed and researched intensively throughout the years, resulting in a wide range of software packages published with both open source and proprietary licenses. Many of the software packages are quite mature, as can be seen in the analysis of four such open source packages presented in this thesis. Using the software packages, many simulations have been conducted in communities around the world.

The objective of this thesis is to develop and evaluate a traffic simulation under an urban environment. The simulation should reflect correctly the real-world traffic situation of a medium-sized city. It needs to support a large number of vehicles and to be able to visualize the state of the simulation in a real-time manner. The user should also be capable of interacting with the simulation.

This thesis presents the design and implementation of a real-time traffic simulation in the context of the City of Oulu. Position in the region of Northern Ostrobothnia, Finland, Oulu is a city and municipality of 196.828 inhabitants (30 June 2015). It is the most populous city in the Northern Finland and the fifth most populous city in the country. It is estimated that by 2020, the population of the city will be 250.000, which require more urban infrastructure and transportation systems.

The structure of the thesis is organized as follows: Chapter 2 describes the literature review about smart cities and their applications, then go deeper into the literature concerning traffic simulation. Chapter 3 reviews four traffic simulation packages and provides the reasoning to choose one of them for implementing the simulation. Chapter 4 and 5 document the design of the simulation in terms of simulation model design and software design. Chapter 6 explains the evaluation of the developed simulation. Chapter 7 focuses on the discussion about how well the defined objectives were met by the simulation what are its limitations. Finally, Chapter 8 presents the conclusions.

2. SMART CITY

2.1. Smart City Definition

“Smart city” is a relatively new concept. In the early 1990s, that catchword was used to signify how urban development was turning towards technology [1]. To be specific, Collins [2] defined smart community as “a geographical area ranging in size from a neighborhood to a multi-county region within which citizens, organizations and governing institutions deploy embrace and NICT (New Information and Communication Technology) to transform their region in significant and fundamental ways”. Later, Komnikos [3] further delineates the term of an “intelligent city” with three more additional aspects. The first aspect concerns the application of a wide range of electronic and digital applications to communities and cities, combined with ideas about the cyber, digital, wired, informational or knowledge-based city. The second aspect is the embedded information and communication technologies in the city. And the third one is the spatial territories that bring ICTs and people together to enhance innovation, learning, knowledge, and problem-solving. In general, Komnikos sees intelligent city as “... territory with high capacity for learning and innovation, which is built-in the creativity of their population, their institutions of knowledge creation, and their digital infrastructure for communication and knowledge management”.

In 2007, in an attempt to rank medium-sized European cities based on their “smartness”, Rudolf *et al.* [4] introduced six characteristics of a smart city that can be graded or quantified:

1. **Smart economy** covers the economic aspects of a city, including innovation, entrepreneurship, trademarks, productivity and flexibility of the labour market and the integration in the (inter)national market.
2. **Smart people** is defined by the quality of local citizens in the community in term of education or working skills, and the social interactions and openness toward "outer" world of the community.
3. **Smart governance** involves aspects of the political environment of the community, including public services for citizens and everyday-operations of the administration.
4. **Smart mobility**: local and international accessibility as well as the availability and modern and sustainable transport systems with additional support from information and communication technologies.
5. **Smart environment** is characterized as the attractive natural conditions (climate, green space etc.), pollution, resource management and environmental protection.
6. Finally, **smart living** covers various aspects of life quality such as cultural, health, safety, housing, tourism. [4, 5]

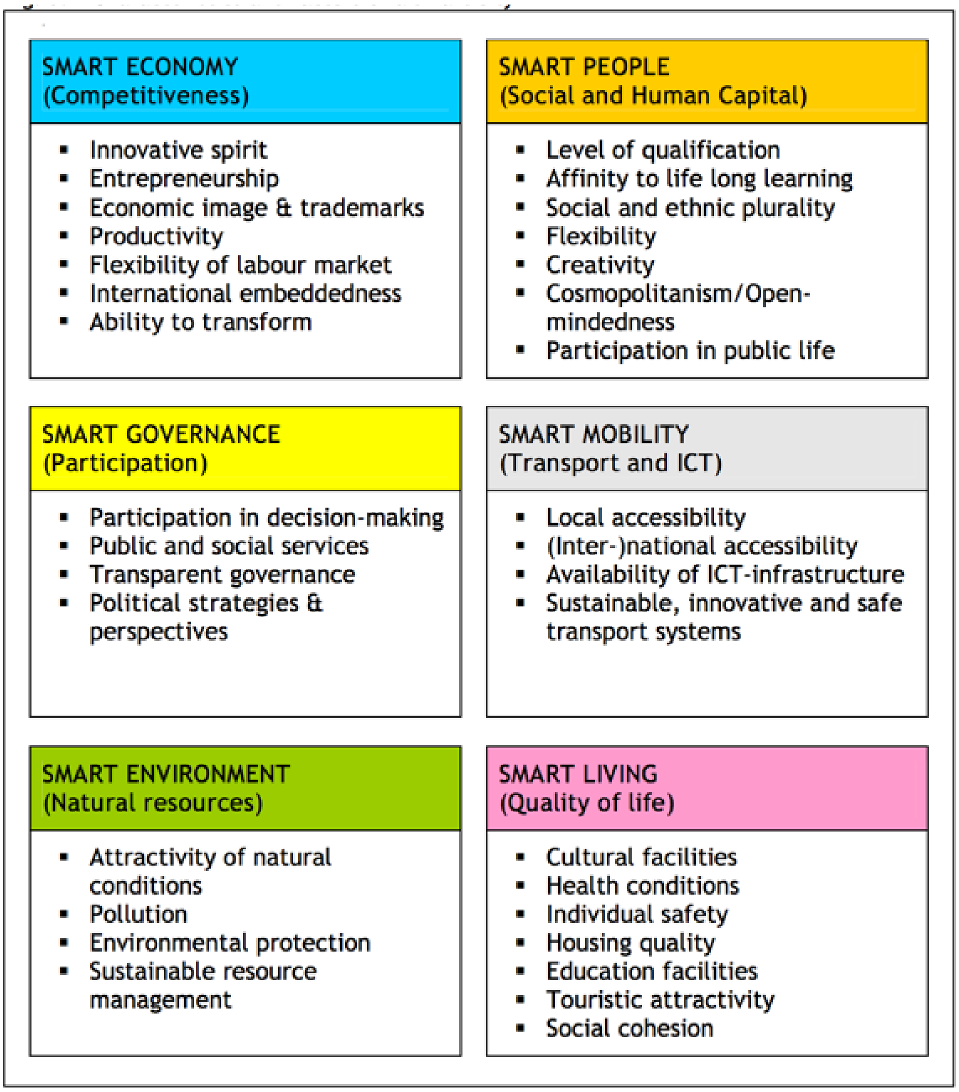

Figure 1. Characteristics of a smart city (Giffinger, 2007).

To sum up, a smart city is a well-defined geographical area which takes advantages of high technologies in the everyday living of its citizens as well as the administration and development of the city government [6].

2.2. Applications of Smart City

People's lives become richer and wealthier in smart cities [7]. To ensure that their communities and regions become "smart" and stay "smart", there are application areas of smart city local and regional policymakers can deploy [8]:

1. **Construction of wireless city** on the basis of powerful fiber-optic network and the technology of Wi-Fi, Mesh and WiMAX to cover the whole city with wireless broadband network.
2. **Construction of smart home** to take all items in citizen's everyday life as a terminal to be brought into the network, achieving the centralized and remote control of electrical and mechanical equipment through the interaction of various networks and terminals, which can also be used for user identification and management [9].

3. **Construction of smart transportation** [10] to take good advantage of sensor networks, IoT and other technical means to change the traditional transport system, and establish the smart traffic management system, including adaptive traffic signal (automatic control of traffic lights according to flow time) control system, urban traffic control management system to achieve the integration of urban planning, construction, management and operations, and provide comprehensive support for other subsystems of smart city system.
4. **Smart public service and construction of social management** [11] to provide basic platform services for comprehensive urban planning, emergency response, community management, and turn the government into a one-stop service system.
5. **Construction of smart urban management** to bring effective management and service of urban infrastructure, population and events through intelligent collection and analysis of data.
6. **Construction of smart medical treatment** to help hospitals to achieve the smart medical care and intelligent management of medical materials, and support the digital collection, processing, storage, transmission and sharing of internal medical information, equipment information, drugs information, personnel information and management intelligent.
7. **Construction of green city** [10] to achieve the networking and interoperability of various systems posed by different devices and make comprehensive use of various resources of monitoring and alarm to establish a new urban model and system of green city.
8. **Construction of smart tourism** [10] to achieve the establishment of a set of solutions to fulfill the management and tourism-related tasks such as tourism online services, management of customer relation, management of operational area, development of domestic and overseas tourism market, intelligent management system of monitor, collection of tourism information and forecast of tourism development.

2.3. Traffic Simulation

2.3.1. Definition

During the planning phase, traffic management methods and algorithms need to be calibrated and tested to ensure their efficiency and effectiveness. Unfortunately, due to the high cost, the lack of public acceptance, and especially the inability to control the experiment environments, normally only limited field tests are performed.

Traffic simulation packages are implemented to fulfill such needs. Designed to support detailed analysis of the dynamic traffic phenomena, they are important tools for analyzing transportation systems, especially in the presence of ITS technologies [12].

There are three kind of traffic simulation models: *macroscopic*, *mesoscopic* and *microscopic* [13], depending on their level of detail. From top down, we have *macroscopic* models, which represent traffic as a continuous flow similar to gas-kinetic or hydrodynamic theory. At a more detailed level, *mesoscopic* model represents individual vehicles on an aggregate manner, for example, the overview data of a trip. The most detailed models are *microscopic*, which record the behavior of vehicles and drivers in every aspects possible at the operational level. When comparing the three models, macroscopic and mesoscopic models lack the capability of modeling complicated roadway geometry changes as well as detailed functions and features of traffic control and management types such as actuated control and freeway ramp metering [14]. As a result, microscopic models stand out and are often used when evaluating complicated issues, especially in cooperation with advanced traffic information and control systems.

Traffic simulations for transportation planning typically consist of the following five steps [15]:

1. **Population generation**: disaggregate demographic data to obtain individual households and individual household members, with certain characteristics such as street address, car ownership or household income [16].
2. **Activities generation**: generate a set of activities (home, going shopping, going to work etc.) and activity location for each individual for a day [17].
3. **Modal and route choice**: generate modes and routes that connect activities at different locations for each individual [18].
4. **Traffic micro-simulation**: execute plans for each individual simultaneously and obtains the result of interactions between plans.
5. **Feedback**: iterates these steps until goals are obtained.

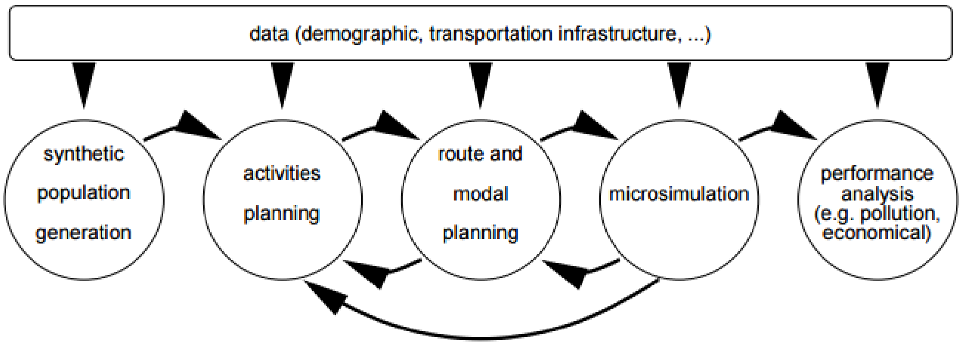

Figure 2. Traffic simulation implementation steps (Raney, 2002).

2.3.2. *Examples of Traffic Simulation Systems*

Real world traffic scenarios from the city of Bologna is a part of iTETRIS project ("An Integrated Wireless and Traffic Platform for Real-Time Road Traffic Management Solutions"). The project was co-funded by the European Commission and was concerned in developing a simulation system for evaluations of large-scale traffic management solutions that work via vehicular communications. Within the project, three traffic simulation scenarios were implemented according to three regions of Bologna city:

1. **The Andrea Costa scenario** includes the area around the football stadium and was setup to simulate the mobility of big events such as football matches or concerts.
2. **The Pasubio scenario** includes the area around the hospital and common routes to the football stadium.

3. **Andrea Costa and Pasubio joined scenario** includes the areas of Andrea Costa and Pasubio scenarios, including traffic demand for a football match.

Data for the project was provided by municipality of the city, including roads, positions of traffic lights, traffic light plans, inductive loop positions and measures. After the implementation, the scenarios were evaluated by comparing simulation data with real world traffic flow measured by 636 detectors around the city.

In 2012, a large-scale agent-based transport simulation model was implemented for Singapore. A total of 4.3 million agents were added to the system in contrast to 4.3 million real world people with access to private vehicles of the city. The agent population was generated using iterative proportional updating and weighted random sampling of household survey records. Facility information, including information about residential facilities, work facilities, education facilities and secondary activity facilities were gathered from their respected management authority. Activities plans of each agent were generated using weighted random sampling, with each activity was chosen from a sample of 1000 qualifying facilities based on that agent's socio-infographic attributes (education, income, occupation). The system utilized MATSim as transport simulator software. To validate, the authors compared a series of information including distribution of trip distance and duration, road counts, mode choice and travel speed measured in the simulation with real world measurements.

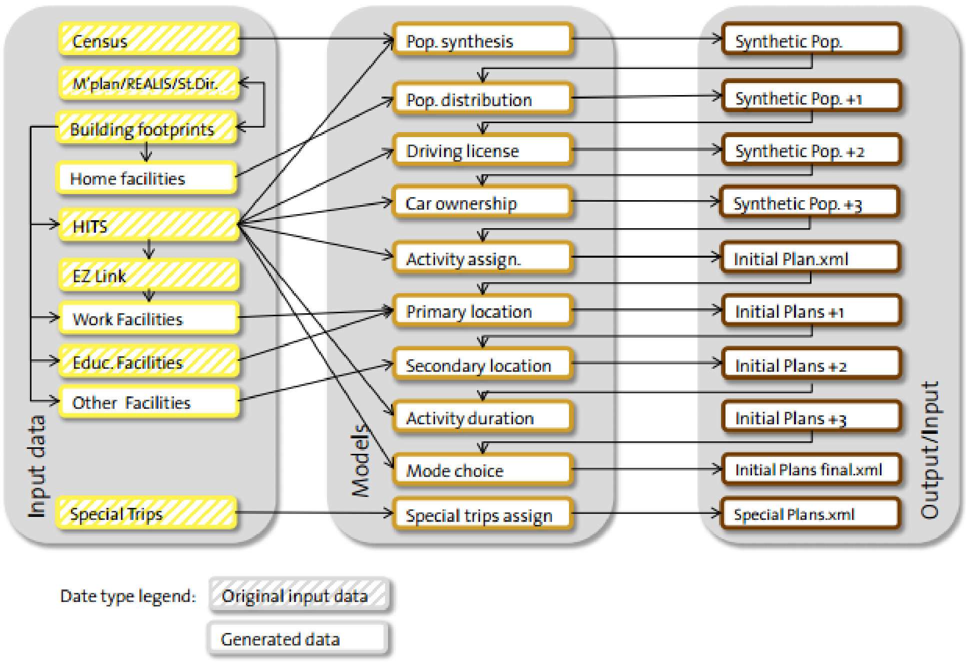

Figure 3. Structure of Singapore simulation system (Erath, 2012).

In 2014, Technical University of Munich and Department of Environmental Science, Aarhus University implemented a transport simulation at Landshuter Allee, an 8-lane major urban arterial road in the city of Munich. Traffic demand of the project was derived from very detailed survey data provided by the municipality of Munich. The system used MATSim as multi-agent transport simulation and resulting simulated traffic volumes are calibrated with traffic counts that are available for a large number of street sections. The aim of the project was to research the relation between linked traffic and air pollution model.

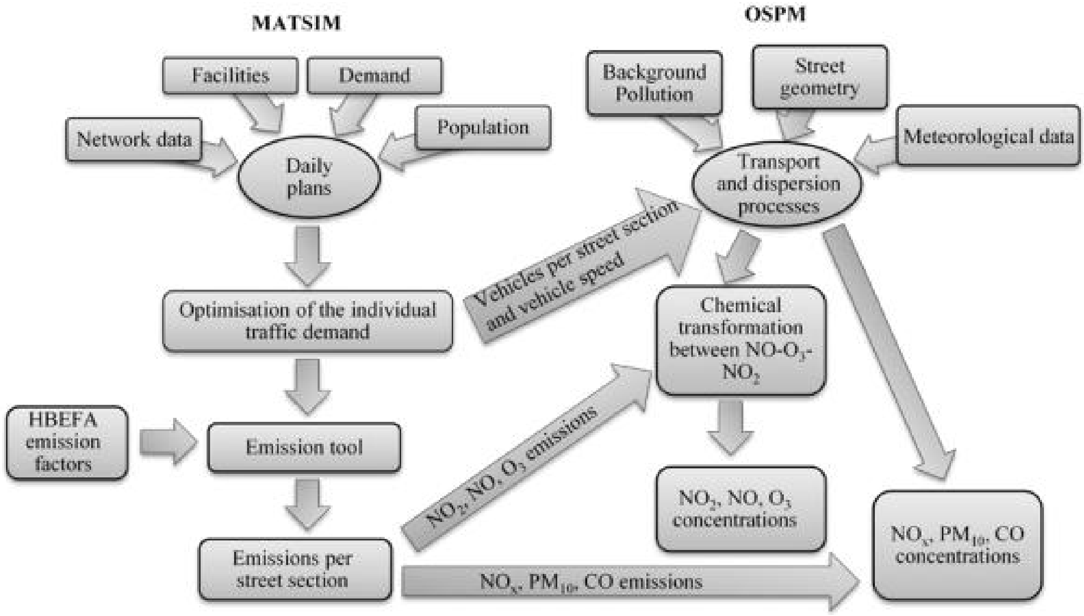

Figure 4. An integrated approach of a traffic and air pollution model used in Munich, Germany (Hülsmann, 2014).

In 2013, a transport simulation study was conducted to deal with evacuation of Aliaga, Izmir, where chemical facilities densely take place and present the evacuation process via simulation models. In this study, two pieces of simulation software were used, MATSim a microscopic simulation model and Cube Avenue a mesoscopic simulation model. Network data used in the project was taken from OpenStreetMap and was converted to a shape file using NetworkGenerator.class provided by MATSim. Three scenarios were implemented with the aim to provide the minimum time for arrival of evacuees to health care centers and gathering areas.

3. TRAFFIC SIMULATION SOFTWARE PACKAGES

In order to investigate different traffic simulation strategies, various microscopic traffic simulator software packages have been reviewed and compared. In this chapter, we give a general review of the investigated software packages, their features and differences. In the scope of this thesis, four open source traffic simulation packages were reviewed:

1. "**S**imulation of **U**rban **MO**bility" (SUMO) is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks" [21].
2. "**M**ulti-**A**gent **T**ransport **Sim**ulation" (MATSim) is an open-source framework to implement mesoscopic large-scale agent-based transport simulations. [22]
3. MITSim is a simulation-based laboratory developed for evaluating the impacts of alternative traffic management system designs at the operational level. [23]
4. "**M**ulti-model **o**pen-source **v**ehicular-traffic **Sim**ulator" (MovSim) is a microscopic traffic simulator with xml-based configuration and csv text output. [24]

3.1. Operating System Portability

Operating system (OS) portability is a feature of a software package that is becoming popular with the development of new and improved OSs. It was found that only one software package (Mitsim) is restricted by OS and works in Linux only. The other three packages can be used in Windows, Linux and MacOS. Two of the packages (Movsim and Matsim) are OS-neutral as they have been implemented in Java, and thus they exploit the portability of Java Virtual Machine, in which Movsim can even be programmed as an Android application [25].

3.2. Package Documentation and User Interface

It is important for complex software systems to have an extensive user manual and/or easy to follow user interface (UI).

SUMO, Mitsim and MATSim provide an extensive user manual in many forms, including on-site documentation, discussion forum, mailing lists and books. Unfortunately, we were not able to locate Movsim's user manual. Also, while SUMO and MATSim provide execution packages, Mitsim and Movsim only give source code and require user to compile the code themselves.

Concerning graphical user interface (GUI) of the simulation packages, it was found that SUMO, MITSim and Movsim provide GUI that is easy to use. Matsim does not have a GUI, and user run the package by executing a Java class.

3.3. Creation of Traffic Networks

Road network is important aspect in a traffic simulation package. At the time this thesis was written, there was not any standard for road network definition so each package provided their own file format.

Concerning approaches to create road networks, not all packages were found to provide ways to develop complex networks. SUMO, MATSim and Movsim use human-readable XML file format to represent road network so it is possible to manually write up the road network file. SUMO even defines several file formats such as nodes, edges, traffic lights and conjunctions and then use an application to convert them to the road network file. Mitsim use a text file format but it provides a graphical editor to create/edit networks.

Also, to simulate very large traffic networks, it is essential to be able to convert road networks from map data sources and other simulation software packages such as OpenStreetMap, OpenDRIVE and shapefiles. Within the reviewed simulation packages, only Movsim lacks this feature.

3.4. Simulation Output

One of the best way for non-technical users to see the result of a traffic simulation is to actually view the simulation running in real-time. Looking at graphical representation is also a good method to find important events and the exact time when they happened. This method also benefits real-time simulation in terms of tracking positive or negative effects on traffic flow when something has changed. All the reviewed microscopic simulation package have some kind of graphical representation of the simulation. Movsim even has a 3D view in which users can view the simulation inside a vehicle.

Figure 5. 3D view of a Movsim simulation.

In addition to real-time graphical representation, statistical output will give additional information the viewers might miss. Similar to the input, there are no output standards that the simulation systems use. Despite the differences in an output file format, all systems produce a similar set of outputs. All the reviewed simulation packages produce text files up to event level such as agents' plans, trip information and aggregated data such as segments and network link statistic. In addition to that, SUMO also produces a dump file with the position of all vehicles at a certain point in time.

3.5. Additional Capabilities

Additional features of simulation packages give new ways for researchers to exploit the simulation. The following additional capabilities were found that were worth mentioning in the reviewed simulation packages:

1. Simulation of **different vehicle types**: all the reviewed packages have this feature with different level of complexity. For example, in SUMO vehicle type can be defined with its own speed, size, acceleration and many characteristics.
2. Means to **restrict road access depending on vehicle type/class**: prevent certain types of vehicles to run on certain roads, for example trucks can not run in the city center. SUMO, MATSim and MITSim provide this feature.
3. Means to **configure traffic lights**: While all our reviewed packages provide traffic light functionality, traffic light pattern for Movsim could not been configured and the default pattern was therefore used.
4. Means of **adding public transport and public transport stops**: For a complex simulation system with simulation agents planning their own daily trips and means of commuting, this feature is of great importance. Movsim was the only package missing this feature.
5. Means of **blocking a street and vehicle re-routing**: this feature only affect microscopic packages when the viewer wants to see the effect of blocking a street on the traffic flow on-the-fly. All the reviewed simulation packages have this feature.
6. Ability to **simulate very large traffic networks**: Within our reviewed packages, SUMO, MATSim and MITSim are able to simulate at city-wide level. Because the lack of conversion function, this feature in Movsim could not be tried.

3.6. Summary

In order to choose a traffic simulation package to be used in this thesis, all four packages were closely compared in Table 1. Out of four reviewed packages, SUMO, MATSim and MITSim were very extensive and provided rather similar features. During the simulation implementation phase, SUMO was chosen as the simulation package because of its good usability and the massive support to be gained from its documentation and mailing list.

Table 1. Comparison of reviewed microscopic simulation packages.

	SUMO	MATSim	MITSim	MovSim
Operating System Portability	Windows, Linux, Mac	Linux only	OS - independent	OS - independent
Documentation	Extensive user manual	Extensive user manual	Extensive user manual	No documentation
User Interface	GUI and command-line interface	Command-line interface	GUI and command-line interface	GUI and command-line interface
Ability to create traffic networks	- Human-readable XML file and GUI to create and edit traffic networks - Can convert traffic networks from popular sources	- Human-readable XML file - Can convert traffic networks from popular sources	- GUI to create and edit traffic networks - Can convert traffic networks from popular sources	Human-readable XML file
Simulation output	Text output and visual representation of the simulation	Text output and visual representation of the simulation	Text output and visual representation of the simulation	Text output and visual representation of the simulation
Additional capabilities	Support the simulation of different vehicle types, block certain vehicle types on certain road, traffic lights, public transport, on-the-fly blocking and rerouting vehicles, simulate city-wide network.	Support the simulation of different vehicle types, block certain vehicle types on certain road, traffic lights, public transport, on-the-fly blocking and rerouting vehicles, simulate city-wide network.	Support the simulation of different vehicle types, block certain vehicle types on certain road, traffic lights, public transport, on-the-fly blocking and rerouting vehicles, simulate city-wide network.	Support the simulation of different vehicle types, on-the-fly blocking and rerouting vehicles.

4. DESIGN OF TRAFFIC SIMULATION MODEL

This Chapter describes the design of the traffic simulation model of Oulu Traffic Simulation System (OTSS) that mimics traffic environment in the City of Oulu, Finland. Traffic demands for the simulation are generated through socio-characteristic properties of people living in Oulu.

4.1. Overview

Traditionally, demand generation systems were created based on aggregate traffic quantities represented by isolated trips. That kind of system, while only requiring small level of input, could not reflect the temporal interdependence of activities and trips. For example, a delay of scheduled activities caused by increased travel time due to congestions would be propagated through the agents and could result in shortened duration of activities or even postpone them.

To solve that problem, OTSS creates an activity-based demand generation system using a dynamic traffic assignment package system. The package creates a sequential list of activities and trips connecting those activities for every person in the area. In that case, demand generation is derived as the concept of daily activity for each agent in the system.

There are two types of methods that have been developed to generate daily activity plan: *random generation* and *rule-based generation*. In the first type, random utility theory is used to generate daily activity plans. The system combines real-time data from a surveillance system (composed, for instance, of loop detectors, probe vehicles, incident detection systems, etc.) [26] and use it to create traffic demands for the simulation. In the second type, demand generation is based on psychological decision rules observed in travel diary or stated-adaptation surveys. The system creates a set of agents representing citizens of the location in question then generate a day plan for each individual agent.

The method presented in this thesis follows the rule-based generation method. For each agent, reported activity chains and preferred activity duration is assigned using the Day Activity Schedule Model System introduced by Bowman and Lawrence [17]. The model explicitly translates daily plans into a *primary tour* (involved work or school), and a set of *primary sub-tours* and *secondary tours*. Each tour has a starting time (can be fixed time or in relation to the previous activity), location, and duration of the activity. For each agent whose activity plan features work or education, fixed location to perform such operation is assigned. Secondary activity locations like shopping or leisure are assigned using a random sampling approach based on the prior activity location and finishing time.

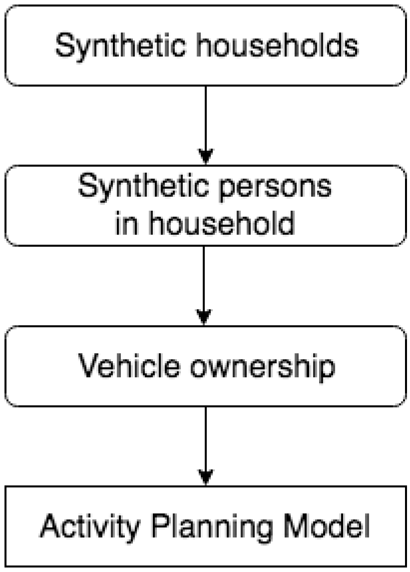

Figure 6. Demand generation in OTSS.

The process to generate traffic demand in the system is illustrated in Figure 6. First, the system generates synthetic households, fills each household with its respective persons (agents) and vehicles. Then for each agent, a day plan is generated based on that agent's characteristics and household properties.

4.2. Population Synthesis

In absence of a detailed full population census, the agent population was generated using weighted random sampling of household survey records. Without a detailed survey sample, data from Statistics Finland's PX-Web databases [52] was used, which provided a publicly available breakdowns of the most recent census in the City of Oulu. In case the information did not exist in the scope of the City of Oulu, the same indicator for the whole Finland was used instead.

On a household level, we used the following indicators:

1. Household income
2. Household size
3. Household type
4. Car ownership

On a person level, the indicators were as followed:

1. Age
2. Gender
3. Income (share with Household income)
4. Driving license holdership

The census used in this thesis is shown in Appendix 1. For the generation of households, each characteristic of a household was randomly assigned a level with respect to its proportion. The proportion of each characteristic of a household are shown in the following sections.

4.2.1. Household Size Proportion

This characteristic dictates the number of persons living in a household. The data was extracted from Statistics Finland's PX-Web.

Table 2: Household size proportion.

Household size	**Proportion (%)**
1 person	42.2
2 persons	33.2
3 persons	10.8
4 persons	8.9
5 persons	3.3
6 persons	1.5

4.2.2. Household Income Proportion

This characteristic of a household describes the total income of a household in comparison to other households in the same area. The data was extracted from Statistics Finland's PX-Web.

Table 3: Household income proportion.

Household income	**Proportion (%)**
Level I (Lowest - income 10 %)	12.5
Level II	10.6
Level III	10.2
Level IV	9.9
Level V	9.9
Level VI	9.7
Level VII	9.7
Level VIII	9.6
Level IX	9.3
Level X (Highest - income 10 %)	8.7

4.2.3. Household Type Proportion

This characteristic shows the type of the household based on data from Statistics Finland's PX-Web. “Registered male couple” and “Registered female couple” were removed because of their small size and because they did not state if the couples have children. “Cohabiting couple” was merged with “Married couple”, because these characteristics have the same meaning in the simulation.

Table 4: Household type proportion.

Household type	**Proportion (%)**
Couple without children	50.53
Couple with children	37.13
Single parent with children	12.34

Other rules were also applied to this characteristic as followed:

1. Households with one person would be assigned “Single adult” as a household type.
2. Households with more than two persons could only be “Couple with children” or “Single parent with children”.
3. In the scope of this thesis, cohabitants in the same apartment without family relation are considered “Single adult”.

4.2.4. Vehicle Generation

74% of households had at least one car in 2016, of which 17% of households had two cars and around 3% had three or more cars [27]. Because the number of households owning more than three cars only occupied a small percentage and because the number of households owning more than four cars was not available, generated households in the simulation were assumed to have at maximum three cars.

Without a full statistical report of vehicle types in use, the proportion number of vehicles registered in 2016 [28] was used instead. The report stated that the most common models of passenger cars in the register of Mainland Finland were Toyota Corolla (around 41%), Volkswagen Golf (around 35%) and Toyota Avensis (around 24%).

4.2.5. Household Address

The address of each household was randomly assigned from the list of roads typed as “residential”.

4.2.6. Agent Generation

In the system, the number of persons in each household was affected by its household type. “Couple” households have two adults, while “Single” households only have one adult. There are three socio-characteristics of a person what were taken into consideration:

1. **Gender**: *Male* and *Female*

2. **Age group**: *Child* (1 to 14 years old), *Young adult* (15 to 17 years old), *Adult* (18 to 64 years old) and *Senior* (65 years old and above)
3. **Working status**: *Pupil*, *Student*, *Employed* and *Unemployed*

Based on gender, age and job status of population in Finland, 16 segments of population were calculated. Proportion of each segment is shown in Table 5.

Table 5. Population proportion.

Group name	**Gender**	**Age group**	**Working status**	**Proportion**
Group 1	Female	Adult	Employed	21.25
Group 2	Female	Adult	Student	3.7
Group 3	Female	Adult	Unemployed	7.68
Group 4	Female	Child	Pupil	9.9
Group 5	Female	Senior	Employed	0.14
Group 6	Female	Senior	Student	0.01
Group 7	Female	Senior	Unemployed	8.6
Group 8	Female	Young adult	Student	1.7
Group 9	Male	Adult	Employed	21.08
Group 10	Male	Adult	Student	3.99
Group 11	Male	Adult	Unemployed	8.71
Group 12	Male	Child	Pupil	10.39
Group 13	Male	Senior	Employed	0.16
Group 14	Male	Senior	Student	0.01
Group 15	Male	Senior	Unemployed	0.64
Group 16	Male	Young adult	Student	1.81

4.2.7. Driving License

By the end of 2014, the driving license register in Finland hold details for about 3.7 million license holders [29]. With the total population of 5,363,637, (1,061,140 people were younger than 18) it means that 86% of the eligible people have a driving license.

4.3. Point of Interest Information

In the simulation, points of interest (POI) are geographic locations where activities can be performed by an agent at certain moments in time. For each POI, required information included type of activity (education, work, shopping, restaurant, leisure etc.) that could be performed in a POI, address of the POI, its geographic coordinate, as well as its opening and closing times.

For the lack of a complete list of facilities in English, unofficial online resources were used to generate POIs in our simulation. For the list of businesses and companies in the City of Oulu, data from BusinessOulu [30] was extracted. Additional data including addresses, opening and closing hours, type of the POI was gathered from Google Map [31] and OpenStreetMap [32].

- ***Work facilities***: BusinessOulu provided a database of businesses and companies in the Oulu region with detailed information about their name, address and size of the company. POIs from GoogleMap were also extracted as small businesses.
- ***Education facilities***: Location of elementary schools, junior high schools, high schools, vocational schools and universities.
- ***Secondary activity facilities***: Secondary facilities are a broad set of POIs found on Google Map POI service
- ***Housing and residential facilities***: Roads with the type attribute “residential” were used as home addresses for the agents.

4.4. Transport Network

To get the smallest granularity of the system, a high-resolution transport network model is required. To create this model, data from OpenStreetMap (OSM) was used.

The OSM project is a knowledge collective that provides user-generated street map [33]. The project, born at the University College London in July 2004, aimed to create a set of map data that was free to use and editable. As of January 2017, OSM had more than three million registered users, with more than 40.000 users actively uploading or editing maps per month. Its database contained more than 3.50 billion accumulated objects (nodes, ways, relations, etc.) with nearly three million objects added per day. OSM’s contributors are not only private individuals but also national mapping agencies and openly-license mapping data source such as National Land Survey of Finland's Topographic Database under National Land Survey open data licence. OSM data is licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF). Users are free to copy, distribute, transmit and adapt OSM data as long as the users credit OpenStreetMap and its contributors.

Being the leading global example of the effectiveness of crowdsourcing of geodata [34], OSM project has been the foundation of a number of scientific publications across a wide spread of research fields. In 2008, the first analysis was conducted that investigated the data quality of roads in OSM for England [35]. Followed by further publications about its accuracy in Ireland [36], Germany [37], France [38] and more quantitative detailed investigation showed that OSM had a highly positional accuracy, especially in highly populated areas.

In OTSS, road networks were downloaded from OSM inside the coordinate of 64.864240° to 65.118303° North, 25.196334° to 25.844493° East, as the City of Oulu has coordinates of 65°01' N, 25°28' E [39].

The network file extracted from OSM is quite extensive, featuring 31.873 nodes, 77.167 network edges and 162 traffic lights with their thorough information about capacity, allowed speed and number of lanes. Given the data source's development model and gradual development, the network file is used as input information for the simulation.

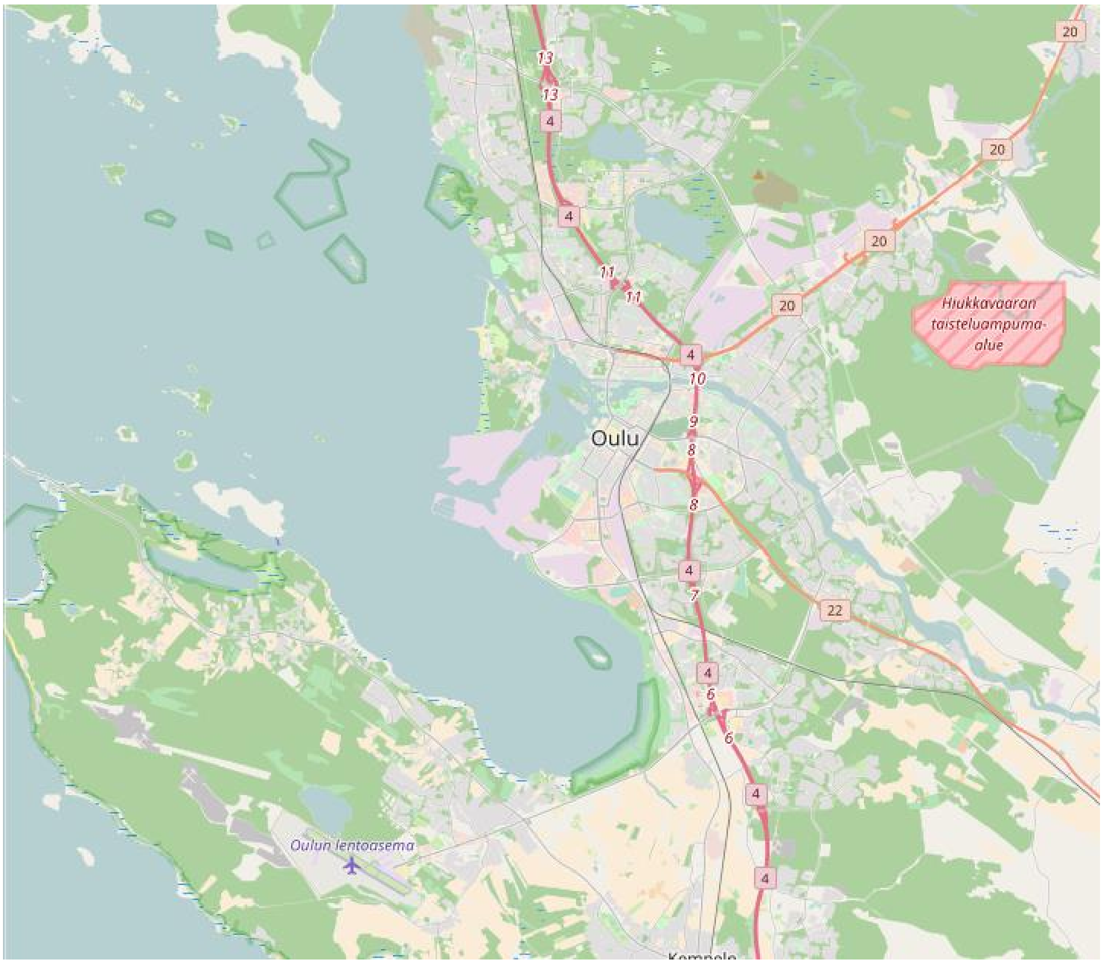

Figure 7. Road network area in OTSS.

4.5. Generating Traffic Demand

OTSS has two models for generating activity plans: (1) random demand model; and (2) activity-based demand model.

4.5.1. Random Traffic Demand Model

According to Finland Traffic Survey in 2012, each able-person in Finland commute three trips every day. The random traffic demand model was created based on that

statistics. For each agent that has a driving license and a car, three tours are generated from a random place of the city to another random place, at a random time.
The route of the tours is calculated using Dijkstra's algorithm.

4.5.2. Activity-based Traffic Demand Model

The traffic generation presented in this thesis falls into the area of activity-based demand generation. The vehicles do not run by themselves, but are rather controlled by a driver. In turn, the drivers' intention is not to produce traffic but to manage his/her day plan in a rational way. Everyday each person has to do a set of activities, most probably not at the same place, thus the traffic production. To plan a day, many decisions are made by a person:

- Activity type decision (Should I go to my friend's or go to a pub?)
- Location choice decision (Should I go to this pub or that pub?)
- Activity start time decision (Should I go now or half an hour later?)
- Activity duration decision (How long should I stay there?)
- Traffic mode decision (Should I drive my car or walk there?)
- Route choice decision (Should I go this way or that way to get there?)

In this model, each agent is assigned a set of activities that need to be done during that specific day, sequentially. Each action is assigned a fixed location, starting time and duration. Starting from the morning, an agent will wait until the starting time of his/her first activity to begin, then move on to the next until the end of the day. Between the activities, the agents need to move between places, thus creating traffic demand for the simulation.

In OTSS, activity chains are created based on three main activities: work (duration 7-9 hours), shopping (duration 0.5 to 1.5 hour), leisure (duration 2 to 3 hours). The process of generating activity chain for an agent is as shown in Figure 8.

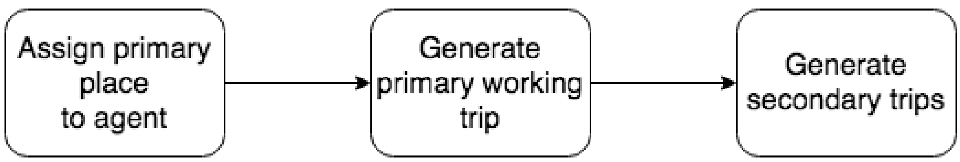

Figure 8. Process to generate agent's activity chain.

- Each POI extracted from BusinessOulu and GoogleMaps is assigned a weighted value based on its size. Each agent is then assigned a POI as the primary working place taking into account the weight of that point.
- The primary working trip start at a random time between 7AM and 9AM. The agent commutes from its household address to the primary place, and work there for 7 to 9 hours.
- Secondary trips can start from right after work or at a later time from home. Up to two trips can be generated for each agent, one trip for shopping, one trip for socializing.
    - For a shopping trip, a shop is chosen randomly as the destination of the trip. The agent goes there, stays for a duration ranging from 30 minutes to 1.5 hours, then goes home.

- For leisure trip, a shop is chosen randomly as the destination. The agent goes there, stays for a duration ranging from 2 to 3 hours, then goes home.

The route of the tours is calculated using Dijkstra's algorithm.

5. SOFTWARE DESIGN AND IMPLEMENTATION

The software process of OTSS follows the spiral model of software development and enhancement, as proposed by Boehm in 1988 [40]. Figure 9 provides an overview of the spiral model with several development cycles, each cycle begins with the identification and evaluation of the objectives of the project, alternative means of implementing the system, and the constraints imposed on the application of the alternatives. After that, a prototype is built to test the feasibility, operational usefulness and robustness of the project. At the end, the evaluation of the prototype is conducted. If the prototype resolves all the risks and constraints, the next step of the waterfall model follows. Otherwise, another development cycle is conducted.

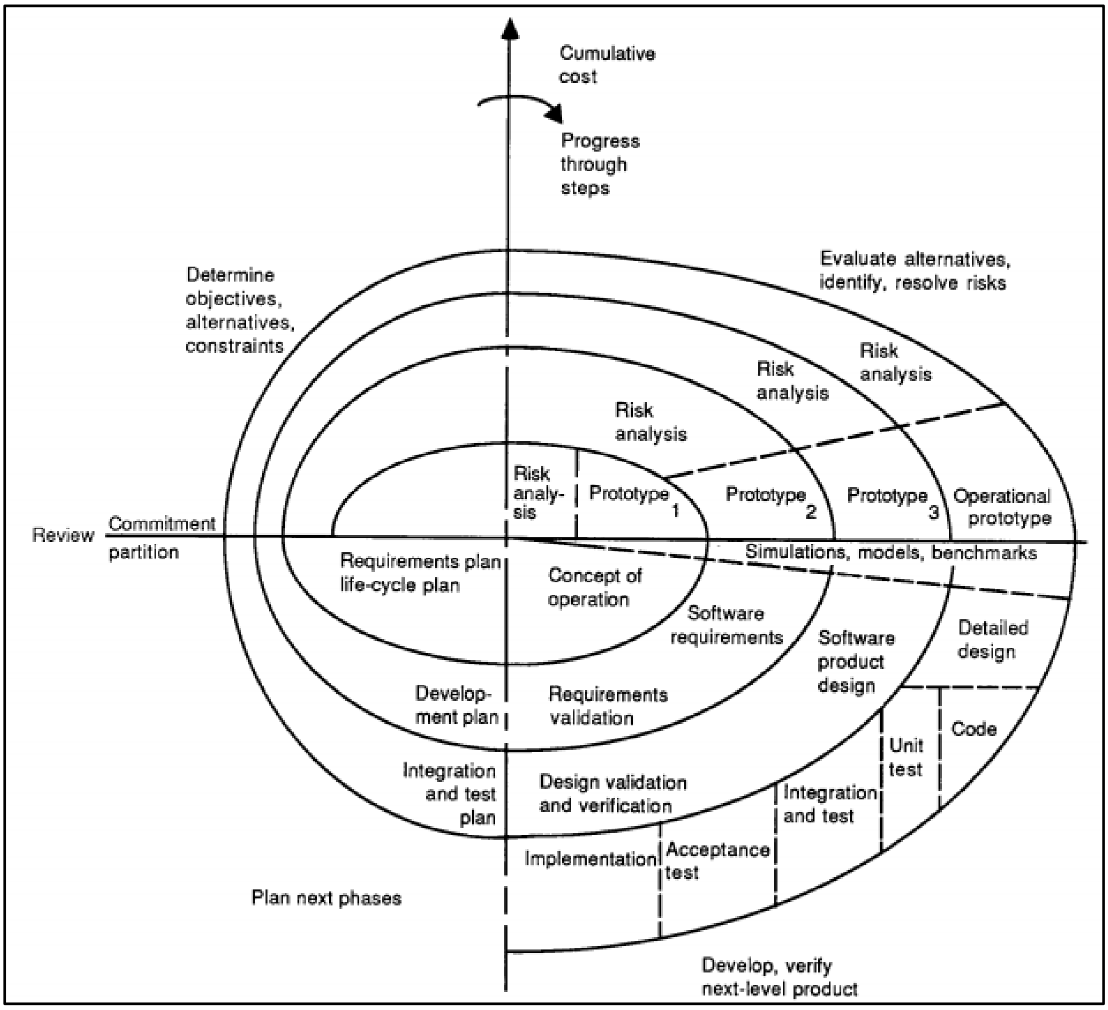

Figure 9. Spiral model of software process (Boehm, 1988).

5.1. System Description and Requirements

This section describes the requirement gathering, spiral cycles, system design and implementation of the simulation system.

5.1.1. System Description

OTSS simulates traffic situation in the City of Oulu. Upon startup, it generates virtual households, people (agents) with their socio-characteristics and daily plans according to Section 4. Then, the system simulates the agents' commuting and the interaction between them when they use vehicles on the streets in a real-time, continuous manner. The system can be configured to start at the beginning of the day, or at the system time. Speed rate of the simulation system can also be configured to match with the real-time speed rate (a second in the system equals to a second in the real world), or to be as fast as the host machine allows.

For researching purpose, the user can get the statuses of any vehicles running in the simulation. The user can set "beacons" to get statistical data of a certain road.

5.1.2. System Requirement Specification

In this section, we will briefly discuss the software requirement specifications of OTSS. The purpose of this requirement specification is to define the functionality and the interface of the simulation in question.

The general functional requirements of the simulation system are listed and described in Table 6.

Table 6. General functional requirement of the simulation.

ID	Requirement
1	The system can generate households and agents with their socio-characteristics and daily plans using the rules defined in Chapter 4.
2	The system can execute and monitor agents' actions in a real-time and continuous manner.
3	The system provides a visual representation of the simulation world.
4	The system can measure statistical data at certain points in the city.
5	The system allows user to block certain positions in the network (to simulate some events or traffic accidents).

5.2. Development Cycles

5.2.1. Cycle 0: Feasibility Study

Cycle 0, feasibility study, was conducted over a period of one month. As indicated in Table 6, the objective was to assess the feasibility of the project within the constraints of resources and timetable, as well as finding suitable technology for the project. One objective of this cycle was to find a solution that can be developed by a single person in the timespan of one year.

Table 7. Traffic Simulation System, Cycle 0.

Objectives	Feasibility assessment. Identifying technologies that are the most suitable for the simulation.
Constraints	Programming languages and external applications are runnable on Windows and MacOS. Feasible for one developer.
Alternatives	4 simulation packages.
Risks	No available technology or software package suitable for the simulation. The system is too extensive for one person to develop.
Risk resolution	Develop the whole system in-house. Limit the boundary of the system.

After the literature review, one solution to consider was to use existing traffic simulation packages to simulate the commuting of vehicles in the model. In this cycle, a total of four packages were reviewed. SUMO was chosen as the simulating package to be used because of its features and ease of use.

5.2.2. Cycle 1: Prototype Development

Cycle 1, prototype development, was conducted over a two months period, with the main content shown in Table 8. The main objective of this cycle was to develop a running prototype to test the technical feasibility of the project. Alternatives to choose from were Java, .NET and Python, with the only constraint was to be able to work with SUMO simulation package.

Table 8. Traffic Simulation System, Cycle 1.

Objectives	Develop a running prototype.
Constraints	Programming language used must be compatible with SUMO simulation package.
Alternatives	Can use Java, .NET or Python.
Risks	The author of this thesis didn’t have hand-on experience with .NET and Java.
Risk resolution	The examples of the application were researched thoroughly

More thorough review about SUMO showed that the binary package provided a Python library with APIs to interact with the simulation through an interface called TraCI (**Tra**ffic **C**ontrol **I**nterface) [41]. With this option, Python was chosen as the main programing language for the implementation and Flask, a Python's web microframework [42], was used as web container for its user interface.

5.2.3. Cycle 2: Stability Improvement

After cycle 1, a workable prototype of OTSS was implemented. However, because of the limitations of SUMO and Flask, the simulation was not stable and suffered from regular crashes. The instability of the simulation prototype led to the initiation of this phase.

This phase was conducted over a period of five months, with the main content shown in Table 9. The main objective of this phase was to create a stable version of the simulation so that its termination could only be triggered by the user or hardware failure.

Table 9. Traffic Simulation System, Cycle 2.

Objectives	Create a stable version of the simulation.
Constraints	Use Python, Flask in the new prototype.
Alternatives	Develop the simulation from scratch.
Risks	Cannot find a workable solution.
Risk resolution	Run simulation in a daily basis.

After a thorough trial-and-error debugging, it was found that the simulation had two problems:

- The simulation application kept track of its history in RAM and did not release the information after a vehicle exit the simulation. When the server ran out of memory, it crashed.
- Flask frequently crashed because the default runtime (which is not stable and only suitable for development environment) was used.

A new architecture for the simulation was drafted to deal with these issues. The simulation and the user interface were separated to two runtime packages, which communicated with each other using remote object communication technology called Pyro [43]. The built-in web container for Flask was also replaced by uWSGI [44], a stable web and application container.

5.3. System Design

5.3.1. System Architecture

The architecture of OTSS is illustrated in Figure 10. Aside from SUMO, the system consists of three modules: *Data Extraction*, *Simulation Controller* and *Web Interface*.

- **Data extraction** package extracts data from Google Map, OpenStreetMap and BusinessOulu and converts it to suitable format for the simulation.
- **Simulation Controller** generates agent population, control their daily plans and connect with the SUMO simulation package through TraCI connection. It adds vehicles to the simulation, controls vehicles to stop and start through SUMO.
- **Web Interface** provides a graphical user interface for users to control and view the simulation.

The communication between the three modules was implemented via Remote Method Invocation.

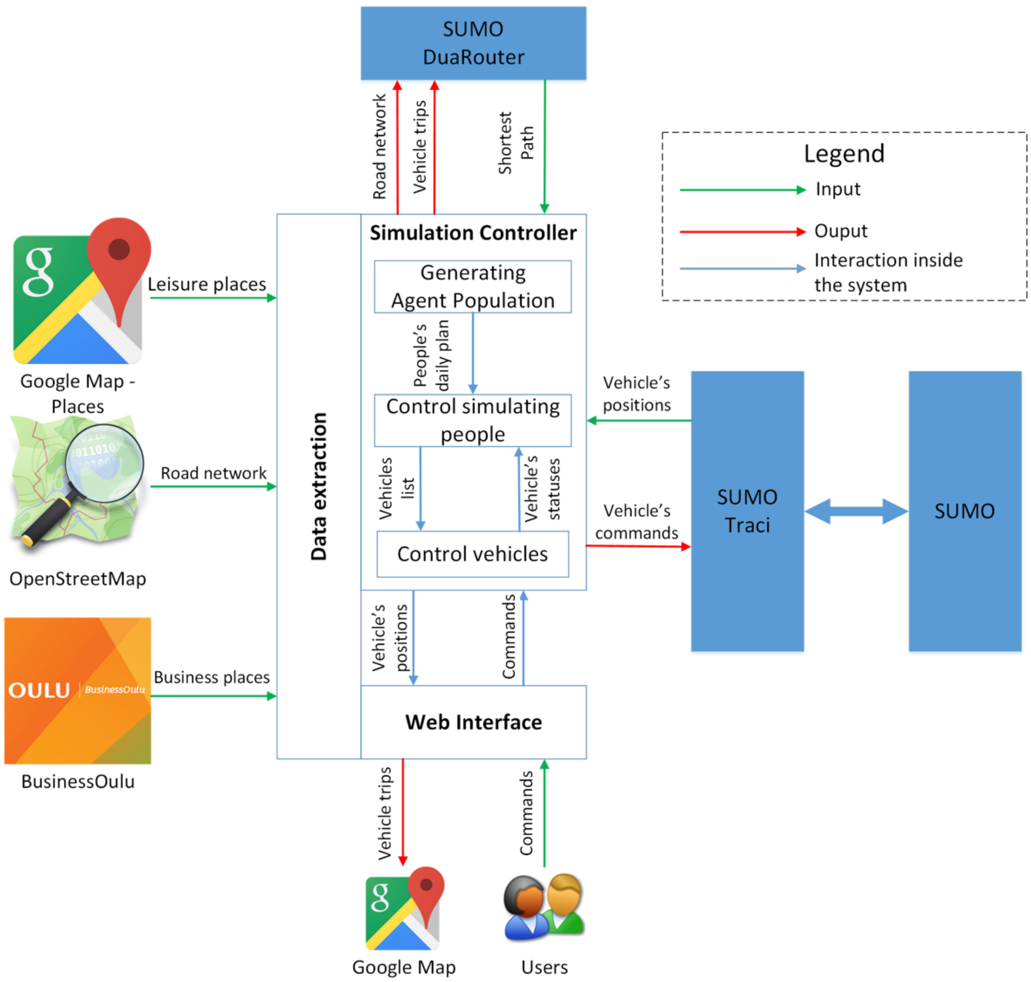

Figure 10. System architecture of OTSS.

5.3.2. Class Diagram

A class diagram shows a set of classes, interfaces, collaborations and their relationships. Class diagrams involves global system description, such as the system architecture, and detailed aspects such as the attributes and operations within a class.

From the result of the spiral development cycles, a class diagram was drawn as shown in Figure 11. It contains the following classes:

- **Vehicle**: store information about each car in used in the simulation.
- **Person**: store information and generate agents who operate in the simulation.
- **Household**: store information and generate households in the scope of the simulation.
- **Action**: an operation that an agent can execute in the scope of a day.
- **TrafficManager**: monitor the generation of households, agents and their day plans; control the simulation.

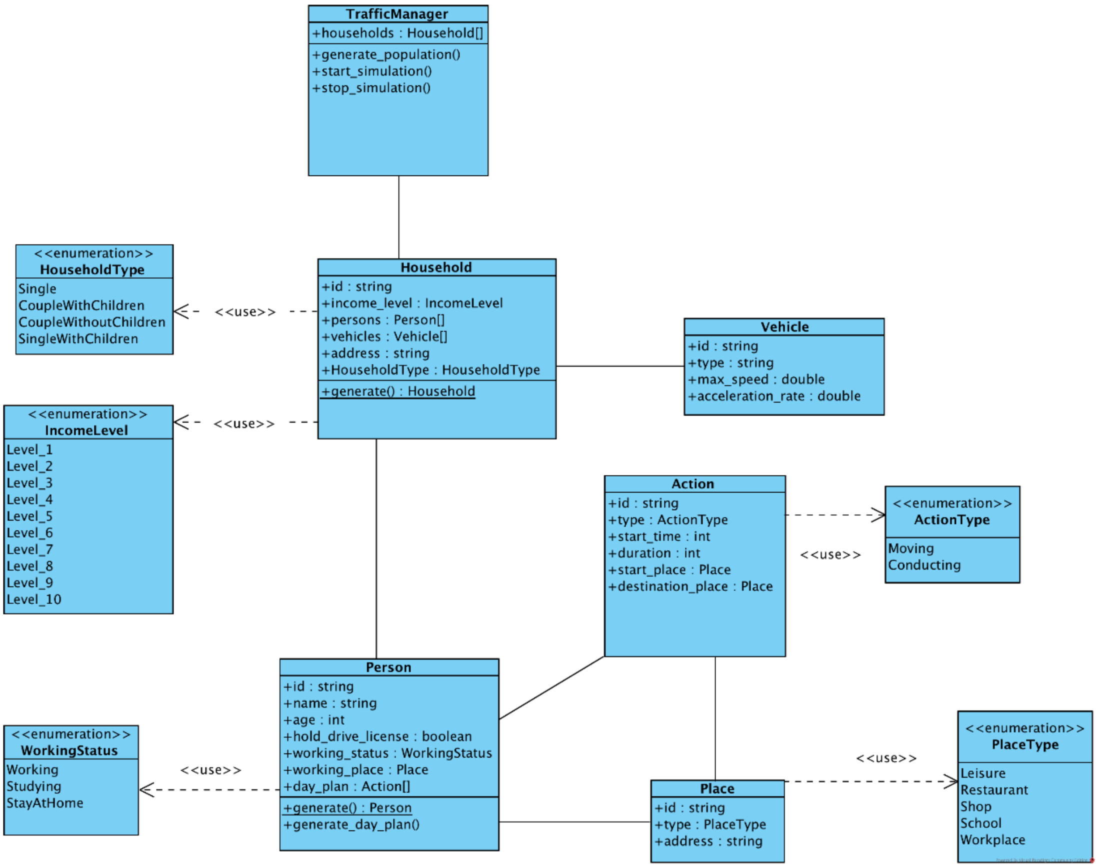

Figure 11. Class diagram of OTSS.

5.3.3. Sequence Diagram

Sequence diagram is one type of interaction diagram, which shows an interaction among a set of objects and their relationships (another kind of interaction diagram is a collaboration diagram). The purpose of the sequence diagram is to document the sequence of messages among objects in a time-based view.

The vertical "lifelines" represents objects of interest. Messages are shown flowing between object lifelines. UML supports the notation of response time in sequence diagrams, which makes it feasible to specify the performance requirements for a real-time system. Time flows from top to bottom.

Sequence diagrams in this section are created based on class diagram shown in Figure 12 and the description of the simulation.

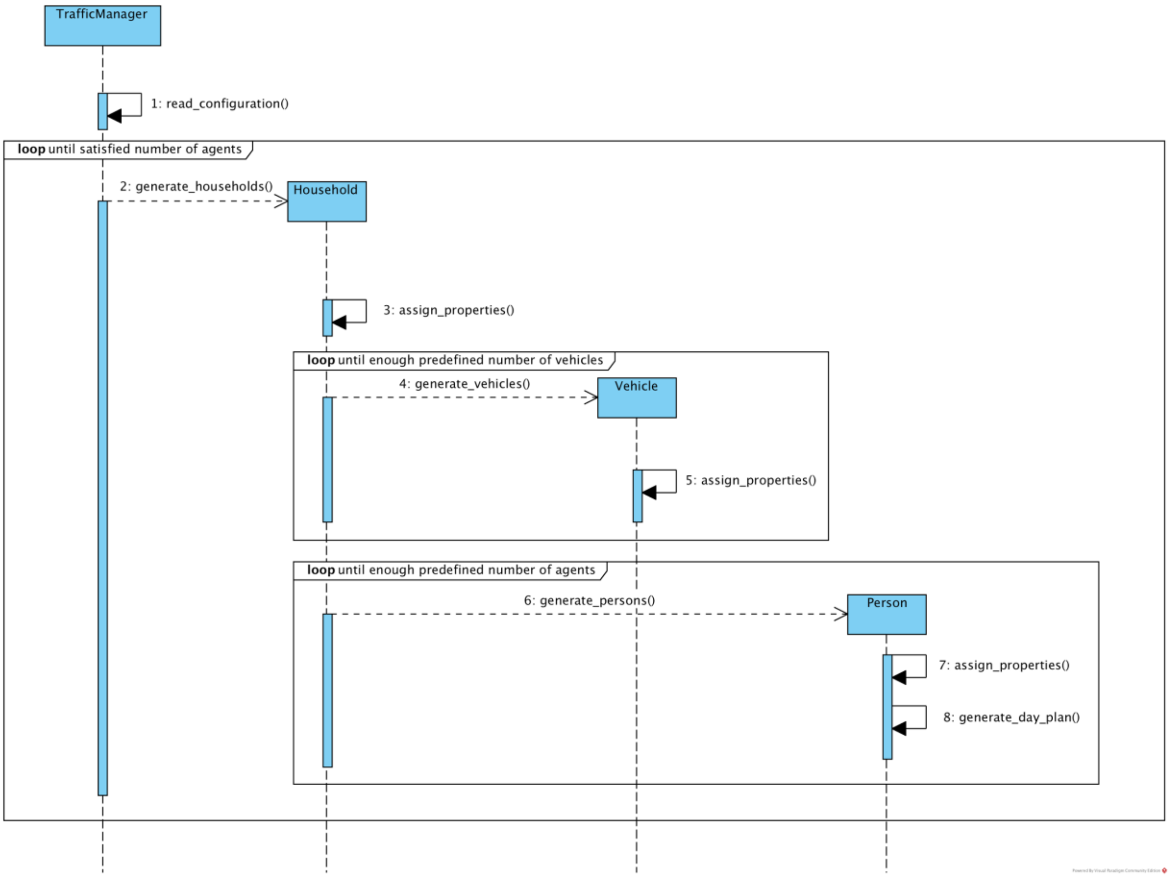

Figure 12. Sequence diagram to generate agent population.

To create an agent population, the traffic manager instance first reads configuration files to get the expected population count, the traffic road network and the information about building facilities that were gathered in the modeling phase. Then, the traffic manager instance creates households, household persons and vehicles by randomly assigning them a property value according to the defined proportions (see Section X).

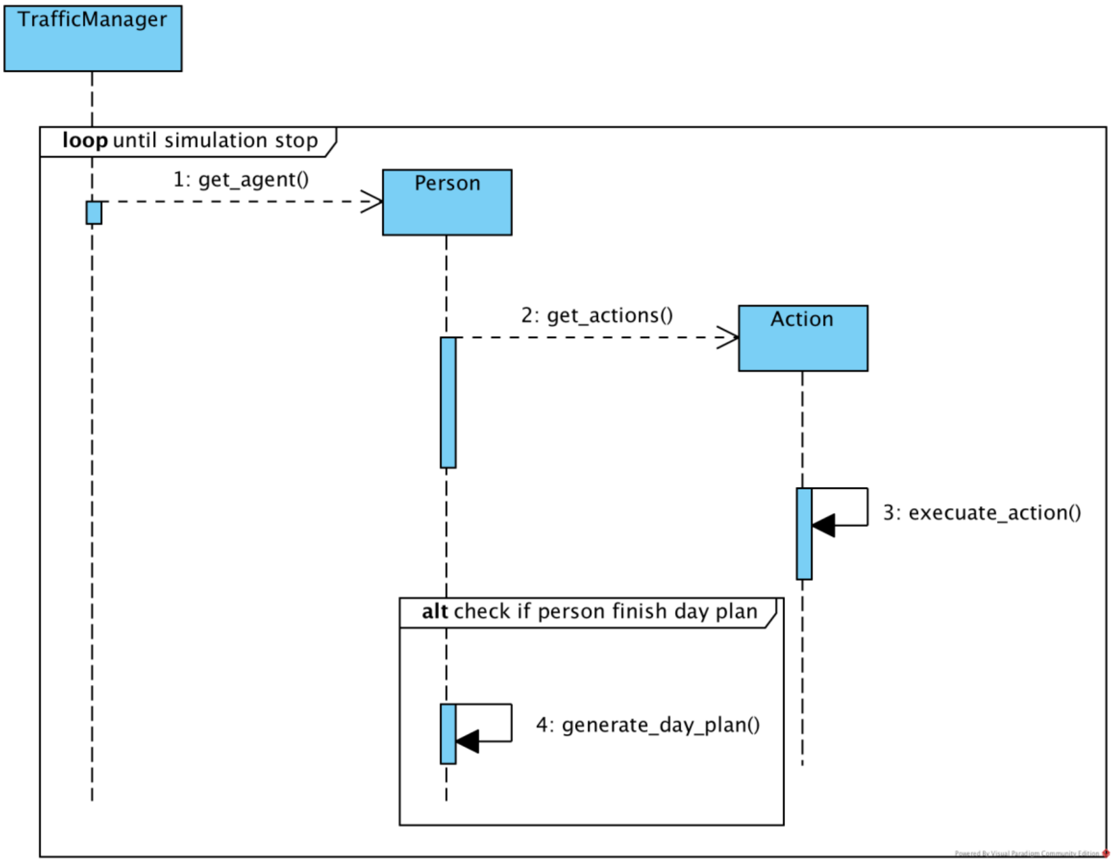

Figure 13. Sequence diagram to simulate the traffic model.

Upon startup, OTSS fetches agents' data from database, then goes into a loop, in which each cycle represents a real-life second. In each loop cycle, the system iterates through all the agents and executes the first action. If an action is a traffic task, the system will register it and wait for the finishing signal from the SUMO simulation package. If the action is not a traffic task, the system update starting time of the next action and then remove the current action from the action queue. If an agent has no action left, the system will generate its day plan for the next day.

5.4. Interface to SUMO

The connection between Simulation Controller module and SUMO has been implemented through an interface called Traffic Control Interface (TraCI). TraCI provides access to a running road traffic simulation and allows to retrieve values of simulated objects and to manipulate their behaviour on-the-fly. In the most basic form, TraCI uses a TCP based client/server architecture to provide access to SUMO. Over the years, developers have developed TraCI interfaces for different programming languages, including Python.

TraCI commands are split into 13 domains, which are gui, lane, poi, simulation, trafficlights, vehicletype, edge, inductionloop, junction, multientryexit, polygon, route, person and vehicle. The following commands are used in this thesis:

- Add vehicles to simulation
- Get information of vehicles running in the simulation

- Command a vehicle to stop
- Command a vehicle to start
- Advance the simulation to the next simulation second

5.5. User Interface

The simulation provides a web interface to visualize its current status. The web GUI displays the virtual time of the simulation, the number of vehicles running on the simulation and their positions.

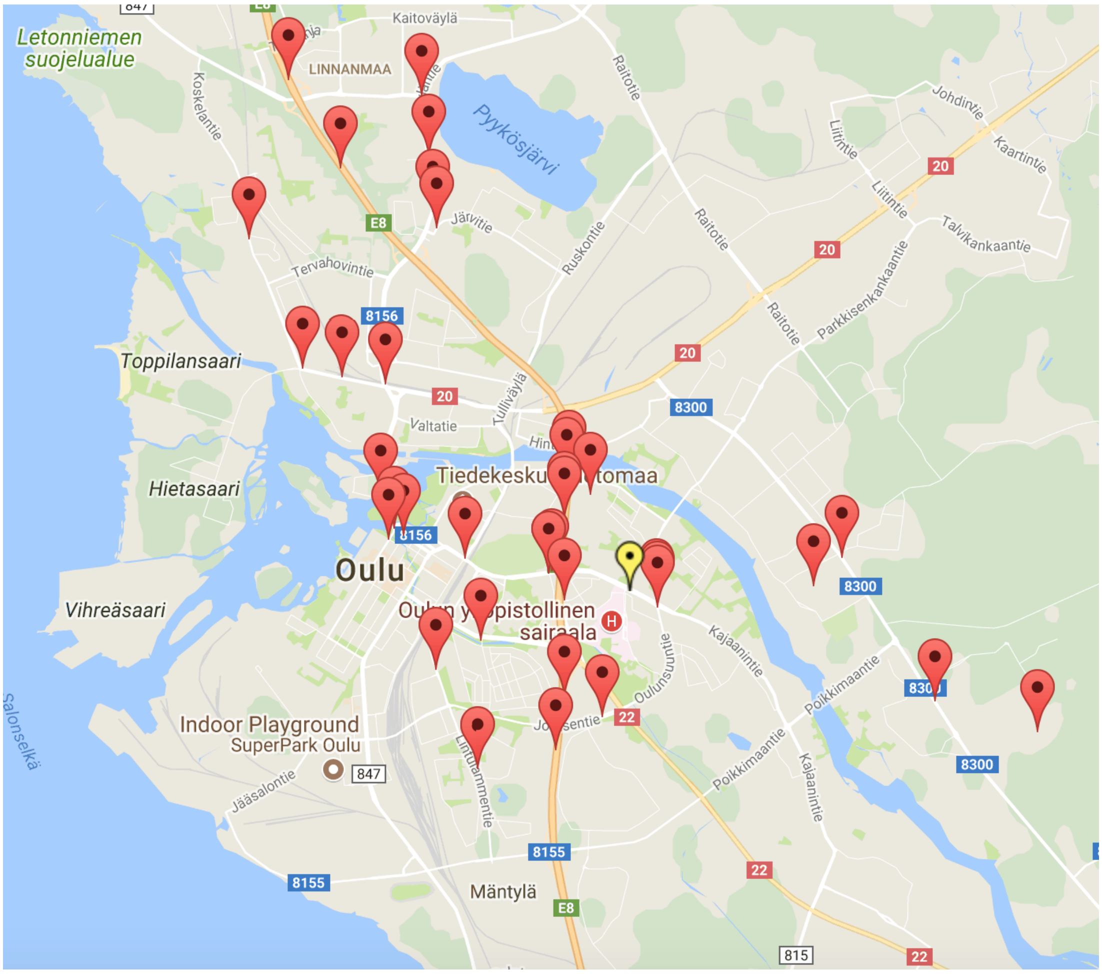

Figure 14. User interface of the simulation.

Users can interact with the simulation by adding new congestions, or moving and deleting existing congestions in the web GUI. Upon facing the congestion, the vehicle will stop and wait until the congestion is removed. Because the lack of direction information from Google Map interface, vehicles running on both side of the road network are blocked by congestion.

On the map, vehicles are represented by red markers, while congestions are represented by yellow markers. To add a congestion, user right-clicks on the desired position on the map then selects “Add congestion” in the pop-up menu. To remove a congestion, user right-clicks on it and then selects “Remove congestion” in the pop-up menu.

6. EVALUATION

This chapter presents the evaluation of the developed simulation system. The simulation starts with the data extraction of POIs in the City of Oulu to be used as input data for the simulation. The simulation is then run for both random traffic demand model and activity-based traffic demand model three times to evaluate their performance and accuracy.

6.1. Data Extraction

OTSS gets input data from three different sources: (1) OSM for road network and traffic lights; (2) Google Maps for POIs; and (3) BusinessOulu for workplace related information. Data from OSM can be downloaded and then converted to SUMO-friendly format with ease. However, data from Google Maps and BusinessOulu require more work as they provide only addresses. The addresses need first to be converted to latitude and longitude map system, and second, to the coordinate system used in SUMO.

6.1.1. Data Extracted from Google Maps

Using parameter types of *leisure*, *meal*, *school*, and *store*, a total of 590 places were extracted from Google Maps as shown in Figure 15.

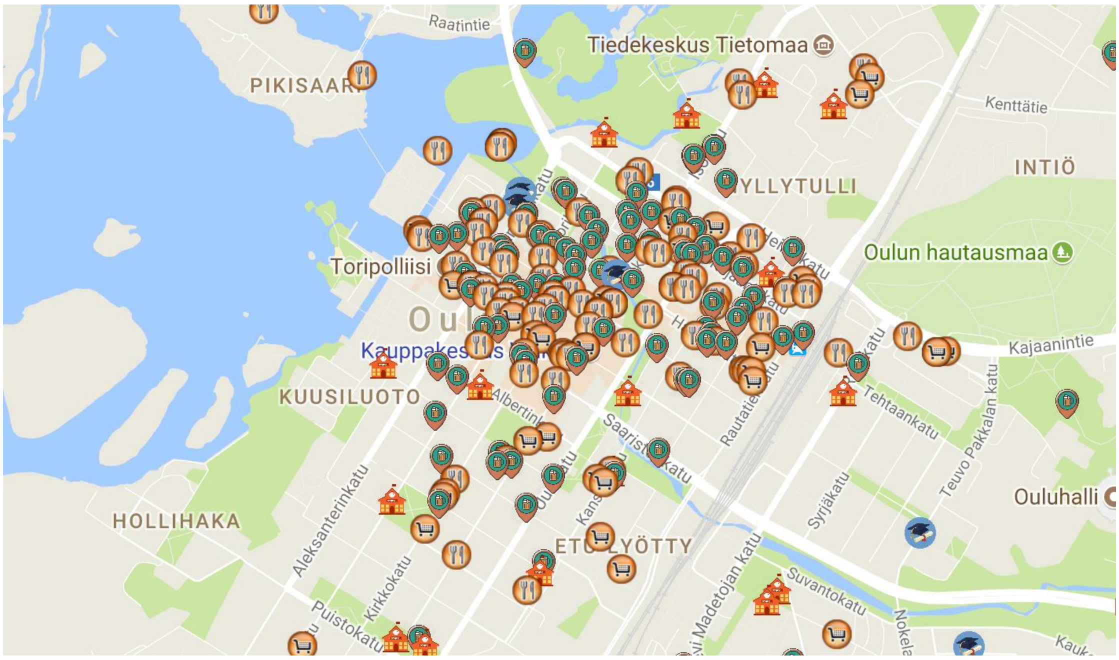

Figure 15. Distribution of the places in city center of Oulu.

Table 10. Aggregate information of POI data.

Type	Number of places
Leisure places	171
Restaurants	186
Schools	66
Universities	15
Stores	152

6.1.2. Data Extracted from BusinessOulu

A total of 597 companies and offices were extracted from the BusinessOulu website [30]. Their distribution is illustrated in Figure 16.

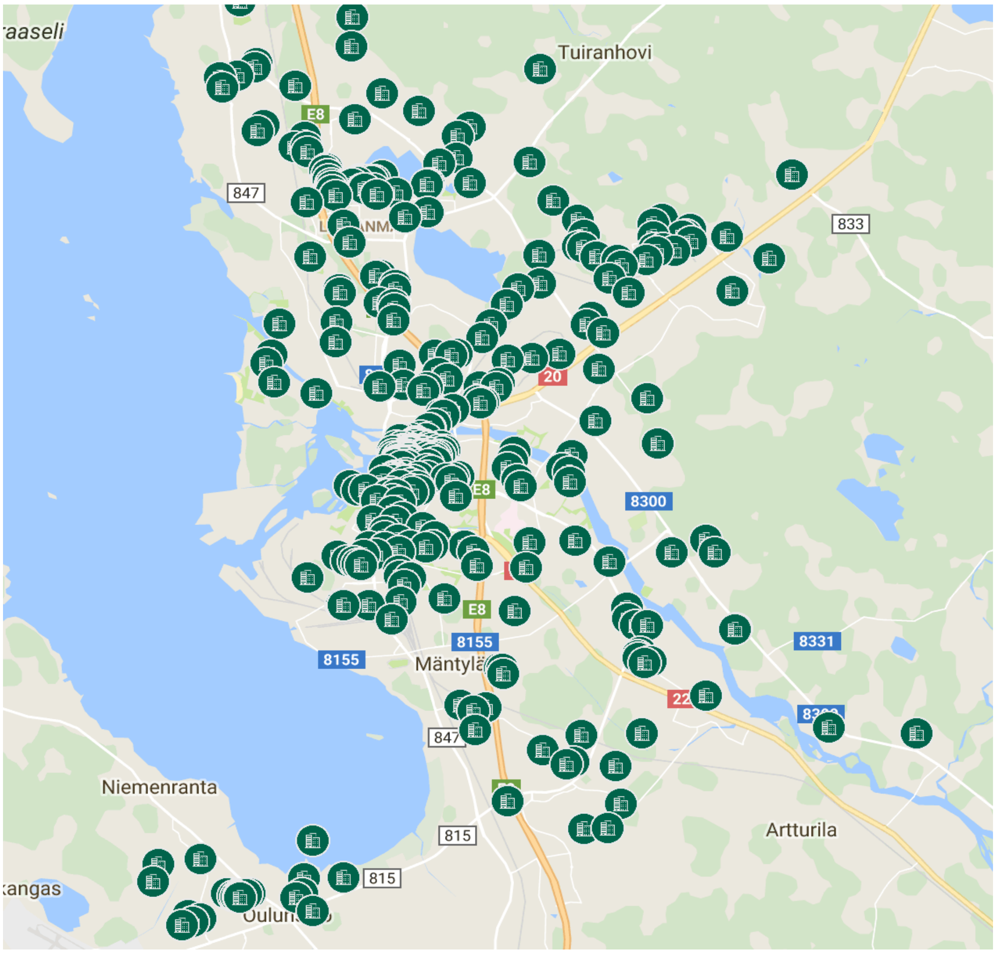

Figure 16. Companies and offices extracted from BusinessOulu.

6.2. Simulation Execution

The simulation is executed three times for (1) *random traffic demand generation model* and another three times for (2) *activity-based traffic demand generation model*. The results are then evaluated in comparison with the population synthesis aggregated data from Statistics Finland as well as with the road counts, distribution of trips and distance in middle-sized cities data from Finnish Transport Agency.

6.2.1. Evaluation of Agent Generation

To reduce both CPU and RAM loads of the scenario, a population of 60.000 agents were generated, which constituted ⅓ of the population of the City of Oulu. The resulting number of generated households and agents are compared with the survey reports fetched from Statistics Finland website on an aggregated level.

6.2.1.1. Evaluation of household generation

Households of the simulation were generated one by one until the population condition was fulfilled. Because the number of persons in each household is randomly selected, the number of households is different for each simulation run. On average, the total number of households in six (n=6) agent generation runs was 29,593.17 units (s = 182.13 units). Results of household generation process are presented in Table 11.

Table 11. Household sizes and vehicles in OTSS.

	Simulated	**Actual data**
Household size		
1 person	42.27 ± 0.11	42.20
2 persons	33.18 ± 0.18	33.20
3 persons	10.79 ± 0.04	10.80
4 persons	8.94 ± 0.13	8.90
5 persons	3.31 ± 0.05	3.30
6 persons	1.51 ± 0.03	1.50
Number of vehicles per household		
0 vehicle	25.95 ± 0.13	26.00
1 vehicle	53.98 ± 0.12	54.00
2 vehicles	17.05 ± 0.16	17.00
3 vehicles	3.01 ± 0.02	3.00

The distribution of households is illustrated in Figure 17.

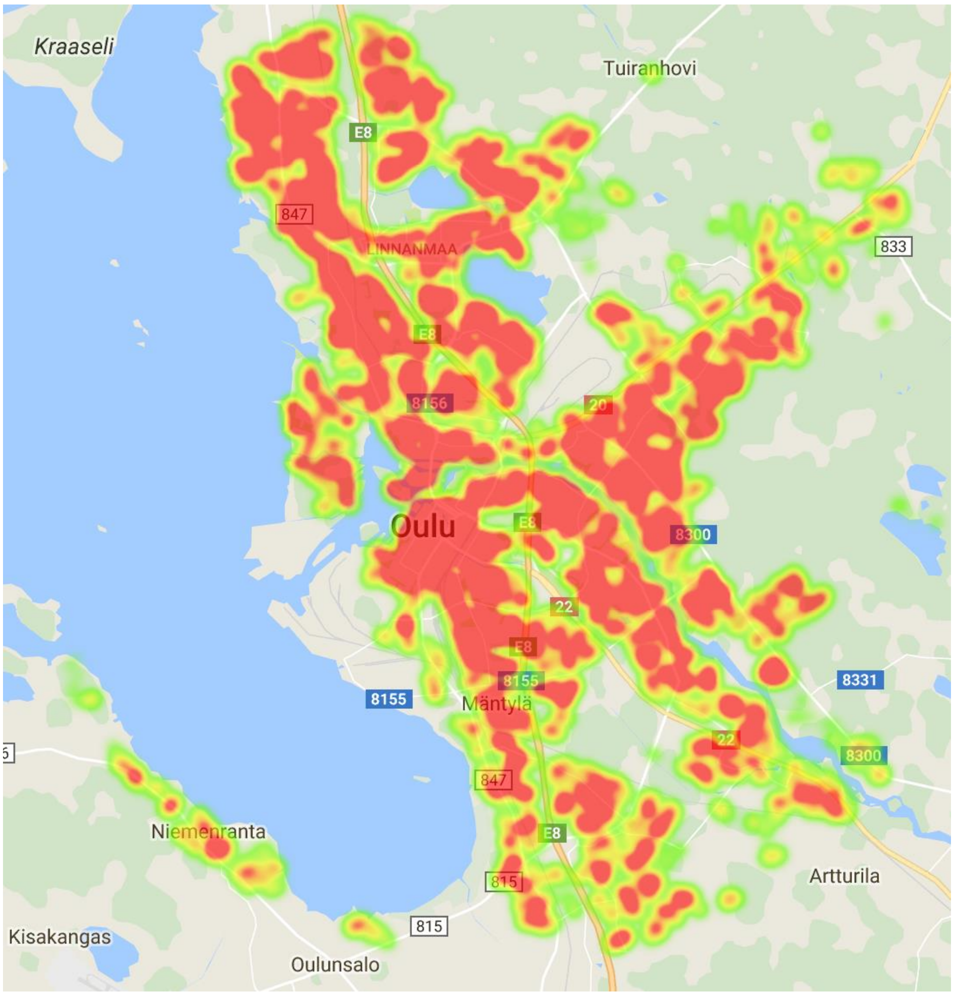

Figure 17. Distribution of households in OTSS.

6.2.1.2. Validation of Population Group Generation

The results of population group generation are presented in Table 12.

Table 12: Household driving licenses and population groups in OTSS.

	Simulated	**Actual proportion**
Driving license holding		
Hold driving license	65.43 ± 0.07	68.98
Not hold driving license	34.57 ± 0.07	31.02
Population groups		
Group 1	21.34 ± 0.06	21.25
Group 2	3.71 ± 0.02	3.7

Group 3	7.64 ± 0.07	7.68
Group 4	9.90 ± 0.05	9.9
Group 5	0.15 ± 0.00	0.14
Group 6	0.01 ± 0.00	0.01
Group 7	8.60 ± 0.04	8.6
Group 8	1.72 ± 0.03	1.7
Group 9	21.06 ± 0.02	21.08
Group 10	4.00 ± 0.06	3.99
Group 11	8.75 ± 0.04	8.71
Group 12	10.48 ± 0.03	10.39
Group 13	0.16 ± 0.01	0.16
Group 14	0.01 ± 0.00	0.01
Group 15	0.64 ± 0.00	0.64
Group 16	1.83 ± 0.00	1.81

6.2.2. Traffic Demand Evaluation

The traffic demand evaluation is conducted based on two criteria: (1) distributions of trip distance and duration; and (2) road counts.

6.2.2.1. Distribution of Trip Distance and Duration

On average, the duration of trips in OTSS are shorter than the average trip time reported in the Finnish national travel survey [47]. In the survey, the average trip time was 18.64 minutes, which is longer compared to both the random traffic demand model (M = 15.48 minutes; t(83962) = -118.75; p = 2.2e-16) and the activity-based traffic demand model (M = 17.80 minutes; t(98489) = -13.663; p = 2.2e-16) as shown by the one sample t-test results.

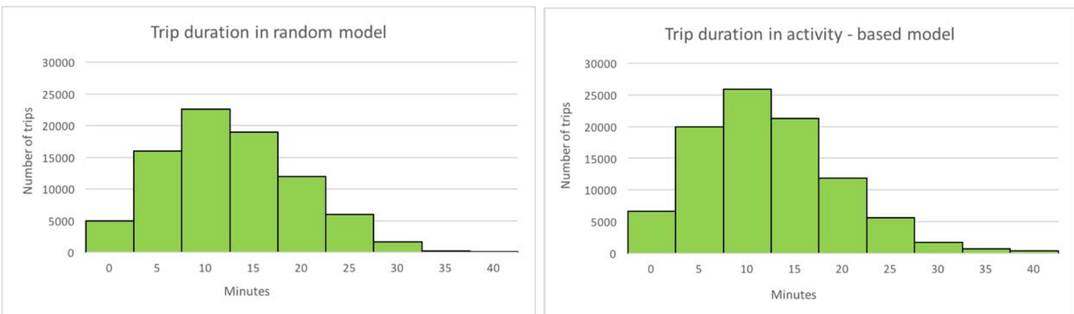

Figure 18. Histograms of trip duration in the random and activity-based models.

There are also differences in the distance of each trip observed in both traffic demand generation models. One sample t-test results show that the average trip distance is significantly less than the average trip distance reported in Finnish national travel survey [47]. In the survey, the average trip distance was 15.9km, which is significantly longer compared with both the random model (M = 8.76km; t(83962) = -511.85; p = 2.2e-16) and the activity-based model (M = 8.11km; t(98489) = -634.68; p = 2.2e-16).

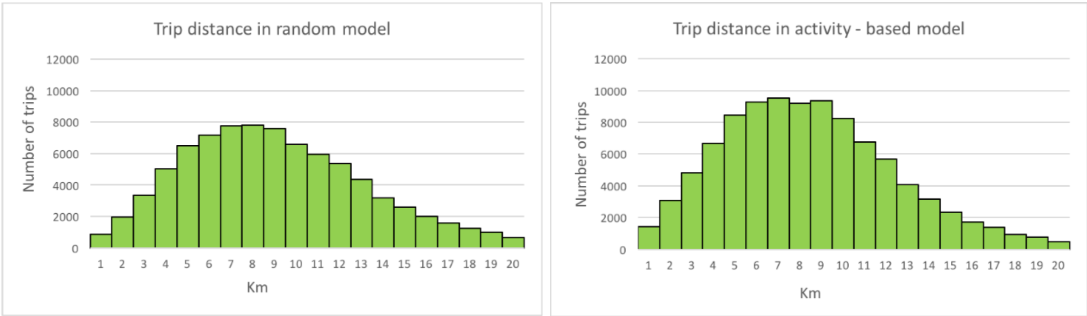

Figure 19. Histograms of trip distance in random and activity-based model.

Figure 20 displays the number of vehicles per hour in both random and activity-based models. While the random model gave a relatively flat number of vehicles, there were two peaks with the number of vehicles in the activity-based model; from 7AM to 9AM and from 4PM to 6PM.

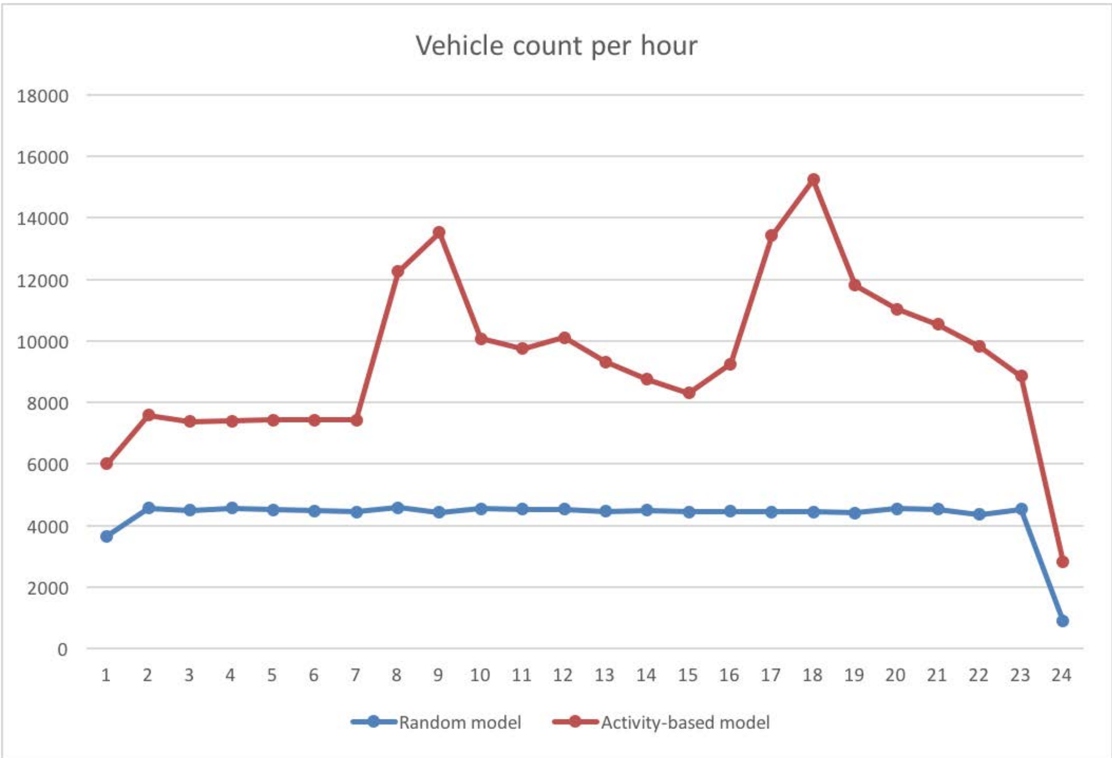

Figure 20. Number of vehicles running in the simulation per hour.

6.2.2.2. Road Counts

Since 2016, the City of Oulu has provided open data sets [45] free of charge for public use. Along with data about buildings, government organizations, finance, the City of Oulu also provides a number of road count stations of 55 points in the Northern Ostrobothnia in which 10 road count stations are positioned inside the City of Oulu.

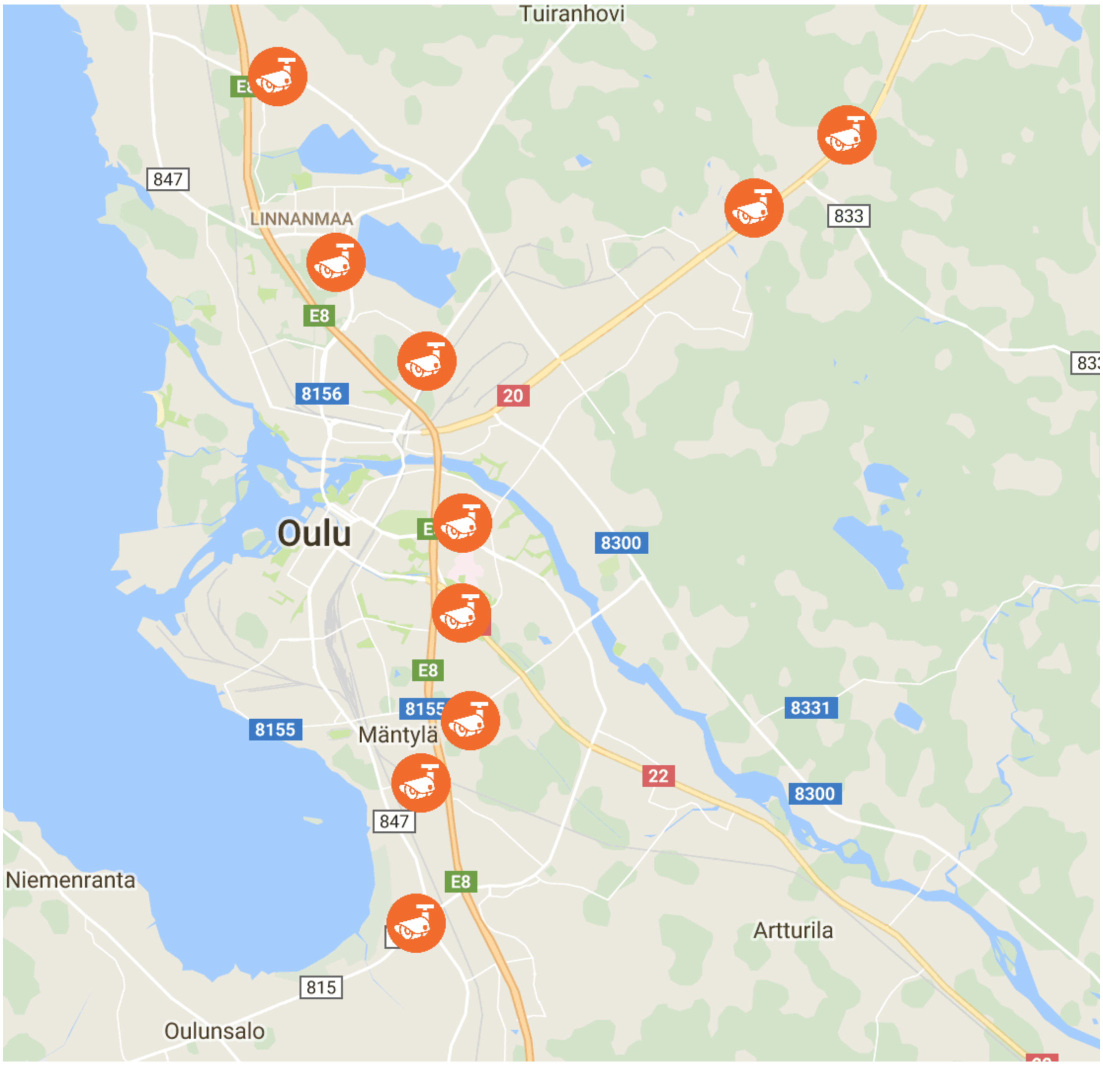

Figure 21. Position of road count stations inside the City of Oulu.

The road count data was collected in three days, from 21st of August 2017 to 23rd of August 2017 for each hour. The average value for each hour was used to compare with the same road count station in the simulations. Because the difference in the population between the City of Oulu and the population in the simulations, the data was transformed by a log10 algorithm. The log10 transformation helps to review the differences in proportion instead of the absolute value. Scatterplots of the data can be seen in Figure 22.

Spearman's rank-order correlation tests were run to determine the relationship between the number of vehicles passing by road count stations reported by the City of Oulu open data set and the data calculated in the simulations. There was strong, positive and statistically significant correlation between vehicle count in real-life compared with vehicle count in the random traffic generation model ($r(240) = .162$; $p = .012$) as well as vehicles count in the activity-based traffic generation model ($r(240) = .297$; $p = .000$)

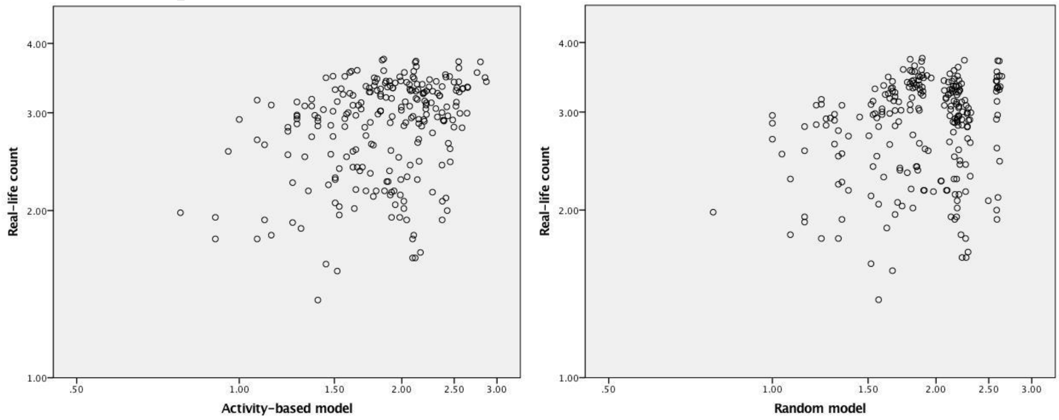

Figure 22. Scatterplot comparing the number of vehicles in real-life and in the simulation.

6.2.3. Computational Intensity

The simulation was executed on two different computation systems:

- System 1: Laptop computer with one Intel Core i7 featuring quad cores (double thread) at 2.6 GHz and 16 GB of 1600 MHz DDR2 RAM.
- System 2: Laptop computer with one Intel Core i5 featuring dual cores (double thread) at 2.6 GHz and 8 GB of 1600 MHz DDR2 RAM.

In both systems, the simulation was run three times for each traffic generation model, each time for the duration of a single simulation day. For this purpose, the simulations were run at a maximum speed.

Two paired-samples t-test were conducted to compare population generation and simulation duration in the two systems. For population generation, there was a significant difference in the duration in System 1 (M=213 seconds, SD=22) and System 2 (M=513 seconds, SD=53); $t(6)=-18.86$, $p = 0.001$. For simulation runtime, there was a significant difference in the duration in System 1 (M=410 minutes, SD=44) and System 2 (M=694, SD=51; $t(6)=-8.35$, $p = 0.001$. These results suggest that the available computer resources greatly affect the performance of the simulation. In particular, the amount of available RAM greatly affected the performance of the simulation. Additionally, those results show that the simulation can perform reasonably well on a regular server. Figures 23 and 24 show the computation time for both computation systems.

Figure 23. Computational time for generating population synthesis.

Figure 24. Computational time for OTSS in two systems.

7. DISCUSSION

The purpose of this study was to develop and evaluate a traffic simulation under an urban environment. The result of the study was the design, implementation and evaluation of OTSS, a microscopic traffic simulation for the City of Oulu. This chapter discusses in detail the results of the evaluation of OTSS. In addition, the limitations of this thesis are presented.

7.1. Achievement of Objectives

After the simulation model design and the system design and implementation, a total of six simulation runs were conducted to analyze the performance of the random and activity-based simulation models. To evaluate the reliability of the simulation models, they were both compared with real world data collected from surveys and traffic counts. In addition, the simulation models were benchmarked with different computational systems.

The results of the evaluation showed that the proportion of each population segment in the simulation is statistically equal with the data reported in the Statistics Finland database [46]. This means that the generated agent population fully represented the population synthesis of the City of Oulu.

In term of traffic demand generation, the result also showed that an average distance and duration of individual trips in the simulation were statistically shorter than the corresponding values reported in the National Travel Survey [47]. While the differences in trip duration were not very large (less than a minute in case of activity-based traffic demand model), the differences in the distance of trips was significant. One reason for these differences could be the issue of the relatively small area and the distribution of POIs that were used in the simulation (10km radius with the majority of POIs in the city center). Also, the survey data used covered the whole Finland, and therefore, the data may be affected by people living in less populated areas.

One of the differences between the random and the activity-based models is the number of vehicles existing in the simulation per hour. In the random model, the numbers of vehicles were relatively constant through the day. In comparison, there were two peaks in the number of vehicles in the activity-based model. This quality of the simulation conforms to the simulation of Bologna [49], a city with a similar size in Italy.

When comparing the number of vehicles go pass count stations, there was a correlation between the number of vehicles passing by in the simulation and in real-life, which showed the reliability of the implemented traffic demand generation models. Similar to the simulations of Bologna city [49], Singapore [19] and Munich [20], this is the most important indicator of a successful city-wide traffic simulation.

In term of system benchmarking, the simulation could perform well on a typical laptop computer. With the simulation which constituted only ⅓ of the total population of the city, it took only around eight hours to simulate a day using the maximum performance available. The benchmarking results also revealed that that it would take a strong workstation to execute the simulation with the complete population, but a typical laptop computer could be used for a smaller scale simulation.

The simulation provides great utility and many possibilities for research and planning of transportation systems in urban environments. Specifically, City of Oulu officers and researchers can use the simulation to plan and evaluate new transportation designs for the city. The simulation can also be integrated with the existing VirtualOulu [50] and Oulu3D [51] projects to provide functionality for both systems: VirtualOulu provide 3D visualization for the simulation; while the simulation provides a lively atmosphere for the 3D city model.

7.2. Limitations and Future Work

The limitations of this thesis are fully acknowledged. First, input data for the agent generation and traffic demand was collected as aggregated data for the whole country. In many cases, the smallest scale that could be collected was at the province level. Because of the relatively large land mass of Finland and the small population, people in the rural areas would have a different travel pattern compared to people in the City of Oulu.

Second, the simulation only deployed three activity-based scenarios based on the agents' working status. In reality, household structure, role in a household, personal and financial capabilities, and activity commitments are important factors on activity schedule choice primarily through the selection of activities (purpose and priorities) and through travel preferences (timing, mode and destination) [17]. Those socio-characteristics were not accounted in this simulation.

Third, the simulation only allows simple user interaction. When blocked by the user, vehicles stop and wait until released for an infinite time. Also, because the low resolution from Google Map interface and the lack of computer-usable direction information, vehicles on both sides of the road stop in front of a congestion. In practice, drivers react differently when being blocked by congestion, including changing line, cutting in front of another vehicle or changing route [49].

A good improvement for the simulation can be the ability to update road networks on-the-fly to simulate the effects of traffic accidents, road constructions and city events. When blocked, vehicles running in the simulation will change their routes according to the new design.

Another good improvement can be the ability to add vehicles to the simulation. When added, the vehicle can go to a random place in the road network. This feature can prove to be useful to plan large scale events in the city.

8. SUMMARY

This thesis documents the design and implementation of a microscopic traffic simulation under urban environment, specifically the City of Oulu. Input for the simulation was collected from open data provided by Finnish Government and reliable crowdsourced map network. The simulation follows agent-based approach, in which traffic demand was generated by a set of intelligent agents with socio-characteristics similar to the synthesis of Oulu citizens. Two traffic demand generation models were designed for the simulation: (1) random model; and (2) activity-based model. In addition to the GUI visualization, the simulation also allows simple user interaction, in which users can block vehicles through a web interface. After three rounds of spiral software development cycles, the simulation system was implemented and executed.

Raw data was extracted from the simulation and compared with aggregated traffic survey results and values from vehicle count stations. The results showed significant correlation in traffic count numbers, even though there were differences in the duration and distance of individual trips. Overall, the simulation reflected well the real-world traffic simulation in the reviewed urban area. In addition, performance tests show that the simulation can be executed in a real-time manner.

The simulation provides great utility and many possibilities for research and planning of transportation systems in urban environments. After further improvement, the simulation can be used as a traffic planning tool for investigating the effects of traffic accidents, road constructions and city events. The simulation can also be integrated with the existing smart city projects of Oulu to improve functionality and aesthetics of both systems.

9. REFERENCES

[1] Gibson, D. V., Kozmetsky, G., & Smilor, R. W. (Eds.). (1992). The technopolis phenomenon: Smart cities, fast systems, global networks. Rowman & Littlefield.

[2] Collins, B., Paquet, G., Roy, J., & Wilson, C. (2002). E-Governance and Smart Communities: A Social Learning Challenge.

[3] Komninos, N. (2002). Intelligent cities: innovation, knowledge systems, and digital spaces. Taylor & Francis.

[4] Giffinger, R., Fertner, C., Kramar, H., Kalasek, R., Pichler-Milanovic, N., & Meijers, E. (2007). Smart cities, Ranking of European medium-sized cities (2007). Vienna University of Technology.

[5] Caragliu, A., Del Bo, C., & Nijkamp, P. (2011). Smart cities in Europe. Journal of urban technology, 18(2), 65-82.

[6] Dameri, R. P. (2013). Searching for smart city definition: a comprehensive proposal. International Journal of Computers & Technology, 11(5), 2544-2551.

[7] Glaeser, E. L., & Berry, C. R. (2006). Why are smart places getting smarter. Taubman Center Policy Brief, 2.

[8] Su, K., Li, J., & Fu, H. (2011, September). Smart city and the applications. In Electronics, Communications and Control (ICECC), 2011 International Conference on (pp. 1028-1031). IEEE.

[9] Han, D. M., & Lim, J. H. (2010). Design and implementation of smart home energy management systems based on zigbee. IEEE Transactions on Consumer Electronics, 56(3).

[10] Sanchez, L., Muñoz, L., Galache, J. A., Sotres, P., Santana, J. R., Gutierrez, V., ... & Pfisterer, D. (2014). SmartSantander: IoT experimentation over a smart city testbed. Computer Networks, 61, 217-238.

[11] Szabó, R., Farkas, K., Ispány, M., Benczur, A. A., Bátfai, N., Jeszenszky, P., ... & Besenczi, R. (2013, December). Framework for smart city applications based on participatory sensing. In Cognitive Infocommunications (CogInfoCom), 2013 IEEE 4th International Conference on (pp. 295-300). IEEE.

[12] Toledo, T., & Koutsopoulos, H. N. (2004). (2004) Statistical validation of traffic simulation models. Transportation Research Board Annual Meeting Proceedings.

[13] Burghout, W., Koutsopoulos, H., & Andreasson, I. (2005). Hybrid mesoscopic-microscopic traffic simulation. Transportation Research Record: Journal of the Transportation Research Board, (1934), 218-255.

[14] Owen, L. E., Zhang, Y., Rao, L., & McHale, G. (2000, December). Street and traffic simulation: traffic flow simulation using CORSIM. In Proceedings of the 32nd conference on Winter simulation (pp. 1143-1147). Society for Computer Simulation International.

[15] Raney, B., Voellmy, A., Cetin, N., Vrtic, M., & Nagel, K. (2002). Towards a microscopic traffic simulation of all of Switzerland. Computational Science—ICCS 2002, 371-380.

[16] Beckman, R. J., Baggerly, K. A., & McKay, M. D. (1996). Creating synthetic baseline populations. Transportation Research Part A: Policy and Practice, 30(6), 415-429.

[17] Bowman, J. L. (1998). The day activity schedule approach to travel demand analysis (Doctoral dissertation, Massachusetts Institute of Technology).

[18] Jacob, R., Marathe, M., & Nagel, K. (1999). A computational study of routing algorithms for realistic transportation networks. Journal of Experimental Algorithmics (JEA), 4, 6.

[19] Erath, A., Fourie, P., Van Eggermond, M., Ordóñez, S., Chakirov, A., & Axhausen, K. (2012, July). Large-scale agent-based transport demand model for Singapore. In 13th International Conference on Travel Behaviour Research (IATBR). Toronto: International Association for Travel Behaviour Research.

[20] Hülsmann, F., Gerike, R., & Ketzel, M. (2014). Modelling traffic and air pollution in an integrated approach–the case of Munich. Urban Climate, 10, 732-744.

[21] Behrisch, M., Bieker, L., Erdmann, J., & Krajzewicz, D. (2011). SUMO–simulation of urban mobility: an overview. In Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation. ThinkMind.

[22] MATSim | Multi-Agent Transport Simulation (2017). URL: http://www.matsim.org/. Accessed 23.05.2017.

[23] INTELLIGENT TRANSPORTATION SYSTEMS LAB (2017). URL: https://its.mit.edu/software/mitsimlab/. Accessed 23.05.2017.

[24] Movsim (2017). URL: http://www.movsim.org/. Accessed 23.05.2017.

[25] MovSim - Android Apps on Google Play (2017). URL: https://play.google.com/store/apps/details?id=org.movsim.movdroid. Accessed 23.05.2017.

[26] Ben-Akiva, M., Bierlaire, M., Koutsopoulos, H. N., & Mishalani, R. (2002). Real time simulation of traffic demand-supply interactions within DynaMIT. In Transportation and network analysis: current trends (pp. 19-36). Springer US.

[27] Households' consumption (Statistics Finland) (2016). URL: http://www.stat.fi/til/ktutk/2016/ktutk_2016_2016-11-03_tie_001_en.html. Accessed 23.05.2017.

[28] Vehicle stock in Finland (2016). URL: http://www.stat.fi/til/mkan/2016/mkan_2016_2017-03-23_tie_001_en.html. Accessed 23.05.2017.

[29] Finnish Traffic Safety Agencty (2017). URL: https://www.trafi.fi/en/road/i_have_a_driving_licence/driving_licence_models_used_in_finland. Accessed 23.05.2017.

[30] BusinessOulu (2017). URL: http://www.businessoulu.com/. Accessed 23.05.2017.

[31] Google Map (2017). URL: https://map.google.com/. Accessed 23.01.2017.

[32] OpenStreetMap (2017). URL: https://www.openstreetmap.org/. Accessed 23.05.2017.

[33] Haklay, M., & Weber, P. (2008). Openstreetmap: User-generated street maps. IEEE Pervasive Computing, 7(4), 12-18.

[34] Chilton, S. (2009, November). Crowdsourcing is radically changing the geodata landscape: case study of OpenStreetMap. In Proceedings of the UK 24th International Cartography Conference.

[35] Haklay, M. (2010). How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. Environment and planning B: Planning and design, 37(4), 682-703.

[36] Ciepłuch, B., Jacob, R., Mooney, P., & Winstanley, A. C. (2010, July). Comparison of the accuracy of OpenStreetMap for Ireland with Google Maps and Bing Maps. In Proceedings of the Ninth International Symposium on Spatial Accuracy Assessment in Natural Resuorces and Enviromental Sciences 20-23rd July 2010 (p. 337). University of Leicester.

[37] Zielstra, D., & Zipf, A. (2010, May). A comparative study of proprietary geodata and volunteered geographic information for Germany. In 13th AGILE international conference on geographic information science (Vol. 2010).

[38] Girres, J. F., & Touya, G. (2010). Quality assessment of the French OpenStreetMap dataset. Transactions in GIS, 14(4), 435-459.

[39] Information about Oulu (2017). URL: https://www.ouka.fi/oulu/english/information-about-oulu. Accessed 23.05.2017.

[40] Boehm, B. W. (1988). A spiral model of software development and enhancement. Computer, 21(5), 61-72.

[41] Introduction to TraCI. URL: http://www.sumo.dlr.de/wiki/TraCI/. Accessed 23.05.2017.

[42] Flask, a Python Microframework (2017). URL: http://flask.pocoo.org/. Accessed 23.05.2017.

[43] PYRO: Python Remote Objects (2017). URL: http://www.pythonhosted.org/Pyro/. Accessed 23.05.2017.

[44] The uWSGI Project (2017). URL: https://uwsgi-docs.readthedocs.io/en/latest/. Accessed 23.05.2017.

[45] Oulunliikenne.fi avoin data (2017). URL: http://avoindata.ouka.fi:5000. Accessed 23.05.2017.

[46] Tilastokeskus (2017). URL: http://stat.fi/. Accessed 23.05.2017.

[47] Nation Travel Survey | Finnish Transport Agency (2017). URL: http://www.liikennevirasto.fi/web/en/statistics/national-travel-survey/. Accessed 23.05.2017.

[48] Bieker, L., Krajzewicz, D., Morra, A., Michelacci, C., & Cartolano, F. (2015). Traffic simulation for all: a real world traffic scenario from the city of Bologna. In Modeling Mobility with Open Data (pp. 47-60). Springer, Cham.

[49] Shinar, D., & Compton, R. (2004). Aggressive driving: an observational study of driver, vehicle, and situational variables. Accident Analysis & Prevention, 36(3), 429-437.

[50] Virtual Oulu | UbiOulu.fi (2017). URL: http://www.ubioulu.fi/en/VirtualOulu/. Accessed 23.05.2017.

[51] Oulu3D | Reflecting Reality (2017). URL: http://www.oulu3d.fi/en/. Accessed 23.05.2017.

[52] Balmer, M., Cetin, N., Nagel, K., & Raney, B. (2004, July). Towards truly agent-based traffic and mobility simulations. In Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1 (pp. 60-67). IEEE Computer Society.

10. APPENDICES

Appendix 1. Population census

Appendix 1. Population census

Dwelling population by income decile.

	Household-dwelling unit population in income decile, persons	**Share of population belonging to the income decile, %**
Total	195,511	100
I (Lowest - income 10 %)	24,350	12.5
II	20,641	10.6
III	19,939	10.2
IV	19,331	9.9
V	19,336	9.9
VI	19,047	9.7
VII	19,003	9.7
VIII	18,741	9.6
IX	18,131	9.3
X (Highest - income 10 %)	16,992	8.7

Household population by household size (in Finland).

	Population	**%**
1 person	1,112,342	42.2
2 persons	874,880	33.2
3 persons	285,433	10.8
4 persons	234,939	8.9
5 persons	87,506	3.3
6+ persons	39,239	1.5

Household type (in Finland).

Family type	
Married couple without children	528,539
Married couple with children	424,185
Cohabiting couple without children	215,620
Cohabiting couple with children	122,657

Mother with children	150,274
Father with children	31,452
Registered male couple	1,023

Occupation by sex and age.

Both sexes	**Age groups total**	**0-17**	**18-64**	**65-**
Whole population	198,525	44,876	125,046	28,603
Labour force	96,803	15	96,223	565
Employed	8,009	-	79,525	565
Unemployed	16,713	15	16,698	-
Persons outside the labour force	101,722	44,861	28,823	28,038
0-14 years old	38,116	38,116	-	-
Students, pupils	21,092	6,604	14,448	40
Conscripts, persons in non-military service	295	-	295	-
Pensioners	37,298	14	9,335	27,949
Other persons outside the labour force	4,921	127	4,745	49
Males	**Age groups total**	**0-17**	**18-64**	**65-**
Whole population	98,909	23,013	63,735	12,161
Labour force	49,496	4	49,187	305
Employed	39,908	-	39,603	305
Unemployed	9,588	4	9,584	-
Persons outside the labour force	49,413	23,009	14,548	11,856
0-14 years old	19,518	19,518	-	-
Students, pupils	10,916	3,401	7,492	23

Conscripts, persons in non-military service	285	-	285	-
Pensioners	16,682	10	4,862	1,181
Other persons outside the labour force	2,012	80	1,909	23
Females	**Age groups total**	**0-17**	**18-64**	**65-**
Whole population	99,616	21,863	61,311	16,442
Labour force	47,307	11	47,036	260
Employed	40,182	-	39,922	260
Unemployed	7,125	11	7,114	-
Persons outside the labour force	52,309	21,852	14,275	16,182
0-14 years old	18,598	18,598	-	-
Students, pupils	10,176	3,203	6,956	17
Conscripts, persons in non-military service	10	-	10	-
Pensioners	20,616	4	4,473	16,139