

UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Siim-Toomas Marran

Simulating Cellular Network Behaviour in the Context of Urban Environment

Master's Thesis (30 ECTS)

Supervisor(s): Amnir Hadachi, PhD
Artjom Lind, MSc

Tartu 2017

Simulating Road Traffic for Generating Cellular Network Logs in Urban Context

Abstract:

In the last years, the use of mobile phone data logs start to attract a lot of researchers' attentions from various disciplines. Those logs help the scientist to understand and predict human behaviour. The mobility logs, like Call Detail Records and GPS data, show where to people commute, how often do they commute and, usually, those logs also say why. These logs hold knowledge about our society, from that data the knowledge could be extracted and used for multiple purposes. The scientists could analyse through the movement how to plan the road infrastructure, generate target advertisement based on forecasting peoples displacement, new positioning technology, population control software, etc.

But there are limits on the people's mobility data. Those information logs are heavily protected by the government privacy data laws to protect the personal rights. Additionally, the mobile operators are interested in their own commercial solutions and therefore their interest to share vital information is low. Here, in this thesis, we show that this cumbersome problem can be over-stepped by prototyping a cellular network behaviour simulator to generate the logs for us through different scientific commuting models inherited from the traffic simulation program.

The result of this thesis reveals that this approach is feasible and shows multiple expansion possibilities how to produce even more real-life like mobility logs. The development of the cellular network behaviour simulation has shown huge potential and even bigger possibilities than predicted in the beginning. Since, our cellular network behaviour simulation is integrated with already existing open-source, highly configurable, road traffic simulator basing on the scientific human behaviour models produce with considerable value data.

Keywords: The traffic simulation; Cellular network behaviour simulation; The mobility logs

CERCS: P170

Liikluse simuleerimine mobiilside võrkude logide genereerimiseks linnakeskkonnas

Lühikokkuvõte:

Viimastel aastatel on hakanud mobiilside andmestik paeluma aina rohkem teadlasi erinevatelt teadusdistsipliinidelt. Need andmed aitavad mõista inimeste käitumis- kui ka liikumismustreid. Mitmed mobiilsusandmestikud (nagu näiteks *Call Detail Records* mobiilside andmed) ning GPS andmed näitavad inimeste liikumissagedust ja -põhjust. Need andmestikud sisaldavad endas väärtuslikku informatsiooni ühiskonna kohta. Töödeldud informatsiooni saab kasutada mitmel otstarbel. Teadlased saaksid andmestiku põhjal planeerida teedevõrgustikke, paremini suunata inimestele reklaame arvestades nende paiknemist, luua uusi positsioneerimistehnoloogiaid, arendada rahvastikukontrolli tarkvara jne.

Vaatamata tehnoloogilistele võimalustele on inimeste mobiilsusandmestikud väga raskesti kättesaadavad, sest need on kaitstud riiklike regulatsioonide poolt, kuna riivavad inimeste privaatsust. Teine tegur on mobiilioperaatorite enda huvi luua inimeste mobiilsusandmetel põhinevaid kommertslahendusi. Selline situatsioon ei innusta operaatoreid jagama äriiselt vajalikku informatsiooni kolmandate osapooltega. Antud magistritöö käigus näidatakse, kuidas sellest raskest probleemist üle saada arendades mobiilsidevõrgu käitumissimulatsiooni prototüüpi. Genereerides andmeid läbi erinevate teaduslike liikumismudelite, mida võimaldab meile liiklussimulatsiooni tarkvara.

Uurimistöö tulemusena selgus, et selline lähenemine on resultatiivne ja omab mitmeid laienemisvõimalusi. Täheledatai mitmeid võimalusi koostööks teiste uurimisvaldkondadega, et muuta genereeritavaid mobiilsusandmeid reaalelule sarnanevateks. Mobiilsidevõrgu käituvussimulatsioon on näidanud suurt potentsiaali ning arendamise käigus avaldusid võimalused, mida algselt ei osatud oodata. Mainitud mobiilsidevõrgu käituvussimulatsioon on integreeritud eksisteeriva liiklussimulatsiooni tarkvaraga, mis on vabavara ning mida on võimalik laialdaselt konfigureerida. Liiklussimulatsiooni tarkvaras kasutatavad inimkäitumise mudelid põhinevad erinevate teadustööde tulemustel ning seetõttu mobiilsidevõrgu käituvus- ning liiklussimulatsiooni sümbioosi tulemusel genereeritud andmed on märkimisväärse väärtusega.

Võtmesõnad: Liiklussimulatsioon, Mobiilsidevõrgu simulatsioon kasutaja perspektiiv-

vis, mobiilsed logid

CERCS: P170

Contents

Acknowledgements	7
Abbreviations and Acronyms	8
1 Introduction	11
1.1 Background	11
1.2 Motivation	12
1.3 Objectives	12
1.4 Limitations and challenges	13
1.5 Road map	13
2 State of the Art	15
2.1 Introduction	15
2.2 SUMO	16
2.2.1 Vehicular communication	18
2.2.2 The SUMO Suite	18
2.2.3 SUMO technologies	22
2.3 Cellular Communication	22
2.3.1 Mobility data	22
2.3.2 Cellular networks	23
2.3.3 Basic cellular characteristics	24
2.3.4 GSM architecture	26
2.3.5 Generations	29
2.3.6 Mobility management	31
2.3.7 Description of main mobility events	34
2.4 Conclusion	35
3 Cellular communication network simulation	37
3.1 Introduction	37
3.2 Used technology	37
3.3 System architecture and design	38
3.3.1 Scope	38

3.3.2	New models	38
3.3.3	Visualization	40
3.3.4	Core mechanics	40
3.3.5	Data import, output and feature configurations	42
3.3.6	Extent of the development	43
3.4	MESGEN and HexagonGen tools	43
3.4.1	MESGEN	43
3.4.2	HexagonGen	46
3.5	Mobility behaviour and management	46
3.5.1	Microscopic simulation devices	46
3.5.2	Mobility management	48
3.5.3	Wave propagation from the transmitter	52
3.5.4	Location update, Cell selection and reselection, Handover	53
3.5.5	Mobility events and the simulation cycle	55
3.5.6	Mobility CDR-like logs	55
3.6	Conclusion	56
4	Results and Analysis	57
4.1	The mobility logs	57
4.1.1	GPS logs	57
4.1.2	Cellular network simulation logs	58
4.1.3	MESGEN and HexagonGen	60
4.2	Graphical upgrade	62
4.3	The benchmarking	62
5	Conclusion and Future Perspectives	63
5.1	Conclusion	63
5.2	Future work	64
	References	74
	Appendix	75
	I. Licence	75

Acknowledgements

First, I want to thank my supervisor Dr. Amnir Hadachi for instructing the thesis and guiding me through the Dark Times. My eternal gratitude.

The thesis would not have been so interesting and educating without soon-to-be-a-doctor Artjom Lind's passionate talks about the cellular networking, its related technologies and algorithms. Thank you, Artjom!

Lastly, I would like to share my sincere gratitude to my friend Adam A. Waggoner for his expertise in English language and for being supportive during the difficult times.

Abbreviations and Acronyms

This section clarifies main technological and philosophical terms used in the paper.

3GPP - 3rd Generation Partnership Project

AT - Attention; this two-character abbreviation is always used to start a command line to be sent from TE to TA

BS - Base Station

BSC - Base Station Controller

BSS - Base Station Subsystem

BTS - Base Transceiver Station

CC - Call Control

CDR - Call Detail Record

CM - Connection Management

CS - Circuit Switched

DBPSCH - Dedicated Basic Physical Subchannel

EDGE - Enhanced Data rates for GSM Evolution, 2.5G

EIR - Equipment Identity Register

ETSI - the European Telecommunications Standards Institute

GERAN - GSM EDGE Radio Access Network

GIS - Geographic Information System

GMM - GPRS Mobility Management

GPRS - General Packet Radio Service, 2.5G

GPS - Global Positioning System

GRA - GERAN Registration Area

GSM - Global System for Mobile Communications, originally Groupe SpécialMobile, 2G

GUI - Graphical User Interface

HLR - Home Location Register

IE - Information Element

IMSI - International Mobile Subscriber Identity

ITS - Intelligent Transport Systems

IVC - Inter Vehicle Communication

LAI - Location Area Identity
LAPDm - Link Access Procedure on the Dm
LMSI - Local Mobile Station Identity
LTE - Long-Term Evolution, 4G
MAC - Medium Access Control
ME - Mobile Equipment
MESGEN - Mobility Event Simulation Generator
MM - Mobility Management
MS - Mobile Station
MSC - Mobile-service Switching Center
NSS - Network Switching Subsystem
OpenGL - Open Graphics Library
OSM - OpenStreetMap
PD - Public Data
PDCP - Packet Data Convergence Protocol
PDN - Public Data Network
PDP - Packet Data Protocol
PLMN - Public Land Mobile Network
POI - Point Of Interest
PRAT - Rated Output Power
PSTN - Public Switched Telephone Network
RFID - Radio-Frequency IDentification
RLC - Radio Link Control
RRC - Radio Resource Control
RSCP - Received Signal Code Power
RSSI - Received Signal Strength Indicator
SACCH - Slow Associated Control Channel
SIM - Subscriber Identity Module
SRBCC - Single Radio Voice Call Continuity
SUMO - Simulation of Urban MObility
TI - Transaction Identifier
TMSI - Temporary Mobile Subscriber Identity
TraCI - Traffic Control Interface

TRAU - Transcoder and Rate Adaption Unit
UE - User Equipment
UMTS - Universal Mobile Telecommunications System, 3G
USIM - Universal Subscriber Identity Module
VANET - Vehicular Ad Hoc Network
V2D - Vehicle-to-Device
V2I - Vehicle-to-Infrastructure
V2V - Vehicle-to-Vehicle
V2X - Vehicle-to-Everything
VLR - Visitor Location Register
XML - Extensible Markup Language
XSD - XML Schema Definition

1 Introduction

1.1 Background

Modern lives are full of so much commuting and travelling that it has generated a huge need for a large variety of technologies and solutions to manage, help and/or even entertain the population. This preposterous quest has been adapted over time by the development and spread of mobile technology, which has slowly alienated people from the Public Switched Telephone Network (PSTN) and normalized cellular communication in masses.

In 1982, The Confederation of European Posts and Telecommunications formed Global System for Mobile Communications (GSM), which should render previous mobile technologies (so-called 1G technologies) obsolete and lead Europe into the future under one standard, what we nowadays call 2G. Less than 10 years later, Finnish Radiolinja made the first GSM call in Finland, thereby announcing new possibilities to researchers and enterprises, whom had been previously divided between different wireless telephone technologies. [70]

Standardization and the wide access of the mobile technology has boosted range of research fields, most notably in this thesis' point of view, like intelligent transportation systems, geospatial technologies with access to Global Positioning System (GPS) data, reproductively producing and enabling new technologies and solutions in the mobile sphere itself. In 1996, the United States Federal Communication Commission issued a set of rules to govern the availability of basic 911 services and the implementation of wireless services, especially interesting for us, the locating the caller, who is in need of help [7]. The positioning of people has been one of the key challenges in the field of mobile technology. All phones do not have GPS or wireless network functionalities and therefore, this sets multiple limitations in terms of possibilities to provide location-based services. Before the smartphone revolution, government institutions and mobile operators relied on GSM mobility data, stressed the necessity to use cellular information for developing localization applications [33].

Cellular operators and mobile users have vital information to produce the next-generation of mobile positioning algorithms, but unfortunately, there are major limitations of accessing such data, because of various private data related laws and the

operators' self-interest to develop money earning solutions. A further limitation is the extent of effort it takes to gather data, even with the help of volunteers situated in various research scenarios. The solution is to produce the mobility data through simulating the human behavioural mobility in the urban environment, while attributing the simulated model with a mobile device, which has its own characteristics.

1.2 Motivation

The main motivation of this thesis lies in the reality that different legal institutions strongly protect cellular network related mobility data mainly because of users' privacy concerns, and limited due to competitive strategies between companies. Therefore, in order to investigate wireless communication solutions and applications, we need an easy way to produce training and testing datasets.

Through simulating people's real life commutes and social behaviours we could generate a source of information which could be used for understanding human mobility patterns. These patterns could assist in various goals: designing and engineering cellular networks, urban infrastructure and transport planning, advertising, event and population management, epidemics monitoring and control, and analyzing and predicting issues related to human migration.

By adding cellular network on top of the traffic simulation, we get a possibility to simulate Vehicle-to-Everything (V2X) communication through the mobile technology, more accurately Vehicle-to-Infrastructure (V2I), while a cellular device is attached to the vehicle and communicates with a mobile operator.

1.3 Objectives

This thesis has two main divisions: open-source SUMO traffic simulator [4] and cellular network behaviour. The primary objective is to research the cellular network components' and its devices' functioning to prototype their operation principles in the traffic simulator and produce synthetic cellular network logs.

This requires us to work through the vast codebase to analyze, design and implement new tools, integrations, and features to finally have, from 3GPP [1] specifications derived, mobility data producer.

The objectives:

- Research SUMO codebase for upcoming developments;
- Design XML importers and exporters;
- Design of cellular network;
- Representation of cellular towers and cells graphically;
- Mobility event simulator for depicting human's mobile behaviour;
- Simulation of Mobile behaviour inside of user defined cellular network;
- Investigate and prototype of basic vehicle-to-infrastructure communication and data generation.

1.4 Limitations and challenges

The challenges comprised of SUMO and cellular network related documentation. SUMO has a very long development history, and therefore, some of the technologies have become legacy and even deprecated. Over the years, there have been many engineers with different styles and preferences, but general guidelines for the development have kept somewhat order in the project.

Another issue lies in the development pipeline, the lack of feature branching and accepting only tested code into the master could lead into better project health. Currently, regression tests in the project note if it is a known bug or not. In software engineering, the master branch is typically reserved for review and tests passing code.

3GPP specifications about GSM and newer standards had information divided into many documents and therefore, for example, internal device communication between different layers and cellular network components were not always easily understandable, resulting in ambiguity or certain corner cases being unexplained or explained in another specification document.

1.5 Road map

The thesis dissertation has been structured in the following manner.

Chapter 2: Presents the State of the Art about the SUMO traffic simulator and cellular communication technologies.

Chapter 3: Outlook of the work which has been done during the implementation of the cellular network simulation.

Chapter 4: Describes the results which have been achieved during the development and testing.

Chapter 5: Concludes the work, supplying results and giving an insight into future perspectives.

2 State of the Art

The chapter is describing applications and technologies behind the traffic simulation package and gives basic knowledge about how mobile technology works on the viewpoint of this thesis. Providing an overview of relevant literature and existing alternating works, definitions and their properties used in the paper for designing and developing the cellular network behavioural simulation prototype in cooperation with existing traffic simulation software.

2.1 Introduction

The need for mobility data for science research to determine user location, predict or classify human behaviour in various scenarios can be exhibited by a few examples which describe, with some variety, the dependence on mobile specific data. Authors in search of a localization algorithm, are stressing a need for of minimal network location technology as an alternative to current triangulation or trilateration based technologies, because those technologies are incapable of localizing the targets when there is only one distinct base station. In their dataset, approximately 50% of Call Detail Records (CDR), were served by only one base station. People who are developing mobile applications and depend on the localization feature do not have available full and easy access to CDRs, due to privacy concerns and anonymization challenges. Therefore, the authors emphasize the need to develop technologies which are qualified to work with minimal network information. One article proposes a solution by using Bayesian inference for the localization problem and managed to achieve a 20% growth in accuracy when they switched from blind approach to their newly implemented solution. [59] Needless to say, access to generated data with one base station antenna coverage is exactly a scenario which can be resolvable with cellular network behaviour simulation software.

Positioning and human behaviour research have been quite intertwined over the recent decades when it comes to extracting knowledge how people act and move in society. One paper proposes a methodology to detect social events by using the cellular information with their Bayesian location inference framework. They differentiated mobility behavior among social event attendees and people's everyday routine by retracing attendees steps. This approach has been deemed useful in managing crowds and detect-

ing potential emergency situations. [75] The simulation of human behaviour could be defined through the traffic simulation, which has vehicle, public transport and pedestrian support with a possibility to define every entities' mobility pattern from various researches, which has been done in the respective field.

One example of cellular network's designing and altering capabilities to suit research scenario needs, is the changing of 2G cellular network macro cells to 5G micro cells to produce training and test data for model development. This is a very realistic scenario, as several researchers have raised the need for optimal network planning for future heterogeneous networks with high number of small cells since growth in the number of users will increase the density inside the cell. Underlining the importance of geometric modeling for network design and analysis, because currently heterogeneous networks are moving towards to a highly irregular network infrastructure and therefore causing interference, optimal resources allocation may be negotiated and that all results in difficult load balancing between small cells and macro cell due to random network topology and non-deterministic users' behavior. [72] The thesis' prototype serves as a possibility to model and test the new infrastructure and how it responds in compare of previous versions.

The need has been verified by slightly different application examples from the mobile technology sphere. To solve the problem there has been taken SUMO traffic simulation software to overcome human movement in the urban context and implemented cellular networking behaviour to generate mobility logs.

2.2 SUMO

SUMO (Simulation of Urban MObility) is a traffic simulation package which have been developed by the employees of the Institute of Transportations Systems at the German Aerospace [2] and SUMO community. It is an open source project, licensed under GNU General Public License (GPL) [30], with considerable support from external authors who appreciate the software's high portability, extendibility, performance, microscopic design to handle serious large road network simulations without inbuilt limitations towards limiting the network size and number of simulated entities.

Academics from German Aerospace Center started developing SUMO in 2001 and released the first open source version in 2002. The idea behind the open source is simple

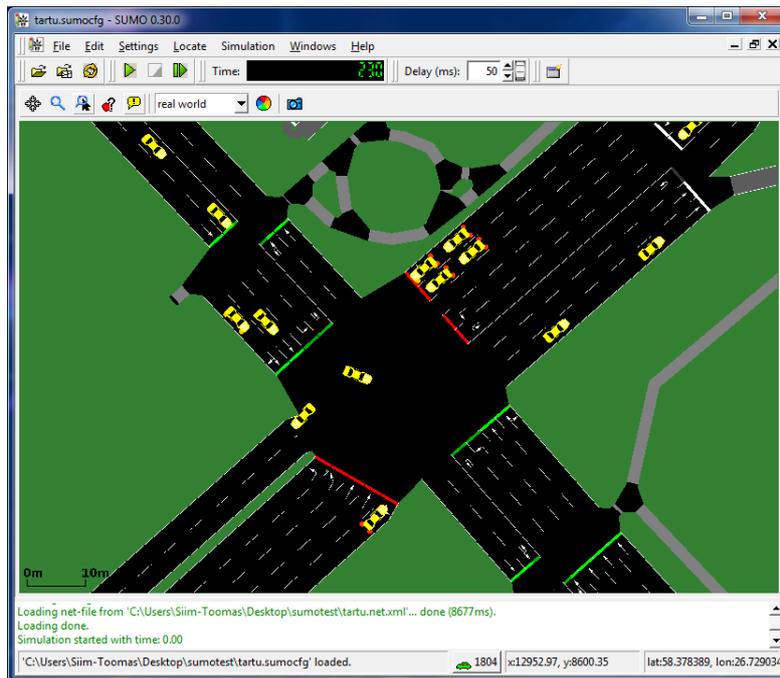


Figure 1. SUMO (Simulation of Urban MObility), example of running real-life graphical simulation, location is Riia-Turu streets' intersection in the city of Tartu.

and honest: to give people a free simulation software. For students and researchers, this is a possibility to implement their own research tasks in SUMO environment without re-inventing all the simulation steps again for the projects, which usually get lost in void eventually after author's graduation or end of research. Furthermore, SUMO allows for comparison and benchmarking against other technological approaches. [68] As an example of continuous research and development, Stefan Krauß developed time-discrete microscopic traffic flow model for his PhD thesis; this model is now being used for SUMO vehicle simulation [64], meanwhile other simulation specific maneuver: lane-changing movement was done during the development of the core simulation [62]. The scientific collaboration and coherence has expanded the simulator, while concentrating infrastructure, vehicular, and traffic behaviour related knowledge into different models to assist other researchers.

To understand the extent of SUMO community and its doings, the author of this thesis would like to point out this year's SUMO User Conference - "Towards Simulation for Autonomous Mobility (May 8-10, 2017)" [3], where people from numerous uni-

versities and vehicle manufacturers, like Volkswagen and Volvo, held talks on various topics relating to SUMO. A detailed wiki page [25] about the application package and an active e-mail mailing list also exist.

2.2.1 Vehicular communication

This thesis works on the cellular communication, setting a cellular device to every simulated vehicle (there could also be another simulation routable entity, like pedestrian, bicycle driver). Combining a vehicle with a mobile phone makes it possible to conduct more technical simulation scenarios from the cellular networking point of view, which opens us to Vehicle-To-Everything (V2X) research possibilities.

V2X basically means communication between vehicle and an entity, which might affect the vehicle or vice-versa. Vehicle-To-Vehicle (V2V) communication has been simulated with SUMO in various research scenarios with the help of third-party software. Bidirectionally coupled framework, which consists of network simulation and road traffic microsimulation, has been built to evaluate VANETs. VANETs are Vehicular Ad Hoc Networks, which were first introduced in 2001 as “car-to-car ad hoc mobile communication and networking” applications [74]. SUMO was used for the traffic simulation, while vehicle network simulation was done with OMNeT++ /citeomNet. The paper concluded that bidirectionally coupled traffic and network simulation provided major advantages and benefits to simulate Inter Vehicle Communication (IVC) cite-sommer2008need The bidirectional networking proved to be successful and gave important feedback to the researchers, resulting in more detailed future simulations where different VANET protocols were introduced and significant confirmation about IVC’s impact for the vehicles’ mobility was received [71].

The traffic simulation combined with different types of simulation software offers a new dimension for research, including the possibility to concentrate on more problems and problems in wider area that the tool was not at first intended to use for.

2.2.2 The SUMO Suite

The authors’ do not only classify SUMO as a traffic simulation, but rather, a bundle of applications which help to perform the traffic simulations to user’s specific scenarios. The main application has two versions: command-line based batch simulation program

and graphical interface powered live simulation.

The traffic simulation is not limited to road vehicles. There is also support for pedestrians and the community is currently working on bicycle, railway, and waterway simulations. All those subjects are good candidates for researchers to propose dynamic traffic flow models. Additional features of SUMO are logistics support, where the user has to define goods shipment stations, where the truck typed vehicles can stop to load/unload the merchandise, parking, public transport, emergency vehicles and electric vehicles support. Those mentioned features are not definitive number of features; there are more smaller features, plus vast configuration possibilities to all features to alter the traffic simulation scenarios to the user's specific needs.

The application bundle has multiple core programs along with the simulation and they vary from generating road network to creating meaningful mobility routes for the simulated entities. Below are the core programs, so-called main applications, needed for traffic researchers:

SUMO command-line software for simulating a defined scenario.

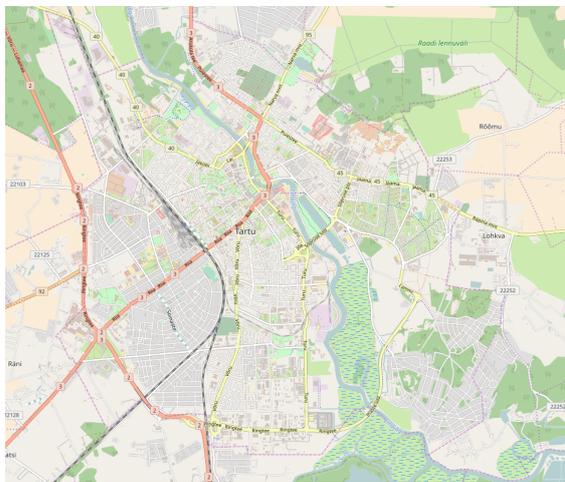
SUMO-GUI adds to SUMO program graphical user-interface with multiple view options to overview running simulation and make sure everything works according to the defined scenario.

MESO refers to mesoscopic simulation model which does not have all the features like SUMO microscopic simulation. Known limitations are missing TraCI support, not existing public transport, electric model, wireless model, opposite-direction driving, sublane-model.

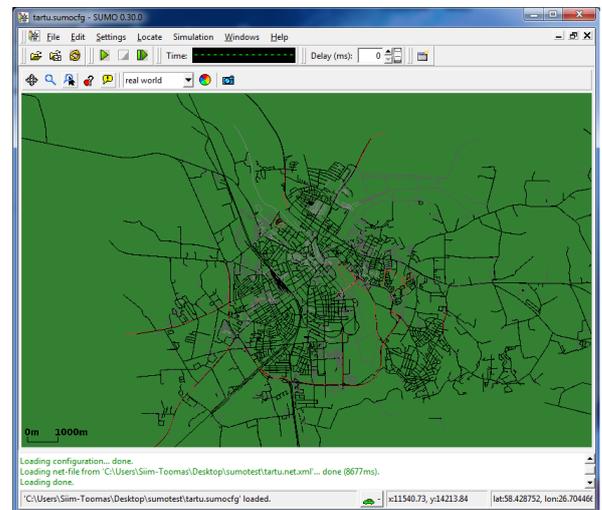
NETGENERATE support application to generate abstract traffic networks, which are usable by other programs. Has interesting solutions for generating grid networks, which reminds Manhattan's design, spider styled road networks, reminding like the name says spider web, which in turn reminds Moscow's road infrastructure on huge scale, and finally random network.

NETCONVERT is one of the most important tools of SUMO, because the community has put big effort to build multiple import styled integrations with other traffic simulators to enhance compatibility and collaboration. Currently, the importing

supports the following formats: SUMO's own XML format, OpenStreetMap's [18] XML file, third party traffic simulators network files from VISUM [22], Vissim [21], OpenDRIVE [15] and MATsim [13], ordinary shapefiles and some lesser known formats. NETCONVERT also supports exporting to some external formats, like the previously mentioned OpenDRIVE and MATsim. In the Figure 2a we can see selected area of city of Tartu, which we would like to import to the program, and in the Figure 2b we can see the successful result after the process. The city of Tartu with its vehicle roads, pedestrian ways, and even train tracks, have been imported to SUMO.



(a) Selected region, what we want to import to SUMO, the city of Tartu in OpenStreetMap.



(b) Previously selected area after the import process in SUMO GUI.

Figure 2. Selecting a region from OpenStreetMap to import into SUMO for simulation.

NETEDIT is a visual network editor to create and modify SUMO networks. NETEDIT is NETCONVERT's extension to debug and fix broken network attributes.

OD2TRIPS imports trip distribution related origin-destination matrices to generate trips, which can be converted into simulation ready vehicle routes.

DUAROUTER imports different demand definitions, which help to define vehicles' behaviours and how they will act (selecting paths) in the traffic on a microscopic scale.

JTRROUTER computes routes from the junction probability values and the traffic volume. Some junctions have probabilities to which road the vehicle proceeds.

DFROUTER generates routes from the statistics collected from high-way related entering and leaving flow data, which is being actively collected and therefore applicable to generate real-life like simulations.

MAROUTER calculates routes from the origin and destination matrices.

POLYCONVERT is being used for importing geometrical shapes to visualize SUMO-GUI simulation.

ACTIVITYGEN generates people's mobility in the simulation, for example, people going to work, school, having free time with various means of transportation like car, bike, bus, and even walking.

EMISSIONMAP tool produces matrices of emissions, which could be used in the simulation for emission simulation along the general traffic movement.

EMISSIONDRIVINGCYCLE is another emission tool, which computes emissions given a timeline of speeds and accelerations for a single vehicle.

TraCI is an abbreviation coined from "Traffic Control Interface" and behind that term is a tool which helps to access and control running traffic simulation. During the simulation, the user could debug the simulated entities and manipulate their behaviour to alter scenarios for research needs. [27]

SUMO also has additional tools. For example, this thesis used *randomTrips.py* tool for generating random trips with generated vehicle ids, input calculated departing times, random starting lane-locations, and destination lane-locations. With DUAROUTER program, the random trips generated file was converted into a route file, where vehicles' routes have been already calculated to save the performance during the simulation.

More details about SUMO and its applications' functionality can be acquired from its documentation page in wiki and its related articles. [27] [68] [63]

2.2.3 SUMO technologies

SUMO main applications have been programmed in C++ programming language, while some additional tools have been scripted in Python. The unit tests are done with C++ library Google test. The regression tests of the applications have been done by Python framework TextTest [28], which is an open source tool for functional testing.

The following frameworks are most notable for main simulation software:

FOX Toolkit [8] is a cross-platform framework for graphical user interface development.

Xerces-C++ [29] is a validating XML parser to read and write XML data.

GDAL [10] is an abbreviation for Geospatial Data Abstraction Library. It is a translator library for raster and vector geospatial data formats.

proj.4 [20] is a library which converts geographical coordinates into cartesian coordinates and in the opposite manner as well.

OpenGL [16] is an application programming interface for rendering two and three-dimensional vector graphics. It has a cross-language and cross platform support.

2.3 Cellular Communication

2.3.1 Mobility data

One of the results of the thesis project is simulated cellular mobility data logs. Mobility data is a set of different types of information entries recorded with telecommunications and ubiquitous computing where the locations of mobile user are continuously sensed. The regulations about data privacy are very serious; it has been regulated thoroughly by the legislators to protect people's' anonymity and privacy. The information is usually shared by the mobile operators under the user agreement, which impose limits to the research. That is the reason why there anonymity data processing is necessary before sharing the datasets. [32] Another possibility is to generate your own dataset, which relies on other science fields works to produce real-life like datasets to train and test your work.

Some mobility data examples:

- Mobility data can be produced from the mobile phone conversations, as the mobile operator knows your location in the cellular network through positioning techniques.
- Data recorded with GPS devices; modern smartphones in these days have many applications which rely on geospatial information.
- Data exchange between vehicles in VANET environment generates entries about the vehicular communication.
- Radio-Frequency Identification (RFID) systems generate logs, which have your location determined and timestamped by RFID reader.
- WiFi access points give information about user's position to service provider. [69]

2.3.2 Cellular networks

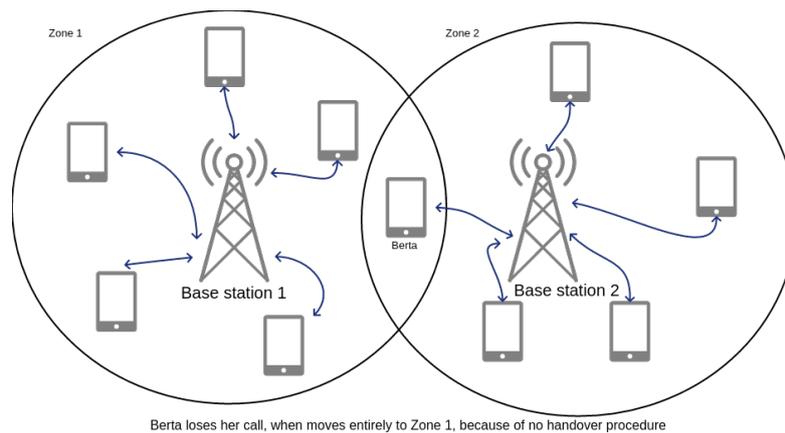


Figure 3. Example of pre-cellular mobile technology.

Modern mobile network, also known as a cellular network, is a communication network which connects the mobile - User Equipment (UE) to the infrastructure in the wireless manner. The name cellular comes from the network distribution over land: it creates cells of coverage to provide services in every location where needed. [56]

The cellular network concept was created in 1949 by Bell Labs with suggestions to fix pre-cellular mobile technology problems like handoffs between transmitter with

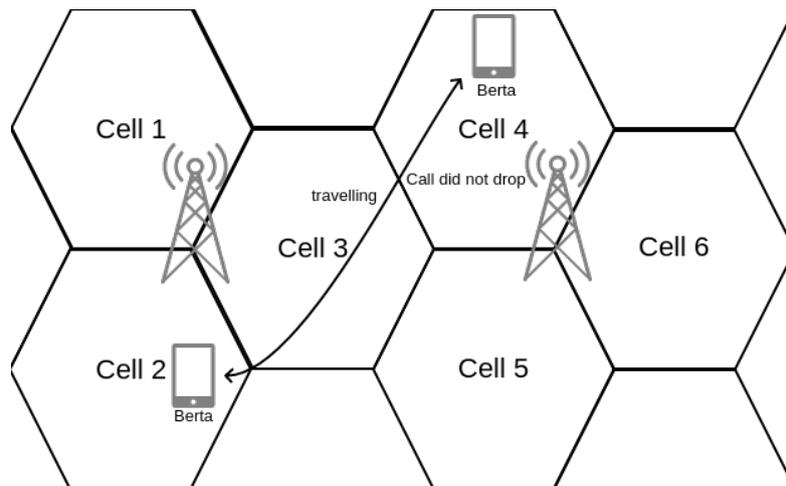


Figure 4. Example of cellular mobile technology.

reuse of the frequencies [70]. Before that, you had to re-initialize your connection when you moved to new base station's (BS) zone, which is visible in the Figure 3. In 1971, Bell Labs revealed a potential cellular system's design [65] [67] and 8 years later, the first commercially usable cellular system started to work in Tokyo in 1979, by Nippon Telegraph and Telephone [70].

The first cellular networks have been titled as *first generation (1G)* networks. In the Figure 4 we can see an example of cellular network, with two transmitters covering through sectorization technique 6 cells (more precisely they are called sectors and the cells in that case are the group of one transmitter's sectors). From 1979 onwards to 1991, when the first 2G call was made in Finland, lots of different 1G technologies were in use. Eventually, the Confederation of European Posts and Telecommunications formed Global System for Mobile Communications, a pan-European standard, because Europe was clustered between different analog and only voice call supporting standards and technologies, which caused problems to roam seamlessly from one country to another. GSM was meant to fix this problem.

2.3.3 Basic cellular characteristics

Network architecture is based on lots of cells covering large or small geographical regions, establishing the service region where the mobile operator offers its services. A

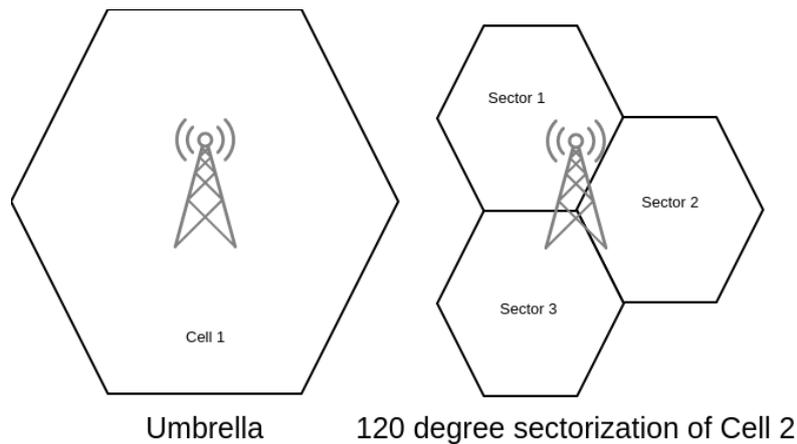


Figure 5. The visual presentation of umbrella cell and 120 °sectored cell.

cell is a resemblance of radio coverage emitted by the base station (BS) [35]. Cells are shaped as hexagons because the circular energy pattern emitted by the omnidirectional antenna from the base station depicts what is close to a hexagon shape [66]. While designing the hexagons help to avoid overlapping or non-filled areas. A group of cells produce a cluster. The cellular coverage has been achieved by using two approaches: omnidirectional and sectorized cells. Omnidirectional cells are usually huge umbrella cells covering large territories, classified as macro and boomer cells, defined in Table 1. Sectorization means that instead of 360 °omnidirectional umbrella cell, which covers all-around with one antenna, the 360 °around the tower is divided into, for example, three 120 °sectors. In the Figure 5 it visible that how Cell 2 is covering three sectors, while using directed antennas, while Cell 1 creates “umbrella” kind of coverage over the area.

Table 1. Non-standardized cell classification. [70]

Class	Radius	Typical deployment scenario
Boomer	> 100 km (up to 240 km)	Large, open space (e.g., plains)
Macro	Typically 1 to 20 km, sometimes up to 100 km	Rural and suburban areas
Micro	100 m to 1 km	Urban areas
Pico	Less than 100 m	Inside buildings
Femto	Few meters	Homes

For the second, the communication link between the Mobile Station (MS) and BS

over air by transferring radio waves.

Key characteristics is frequency reuse. Adjacent cells must have different frequencies, because if the neighboring cell provides the same frequencies, then that would cause co-channel interference. Therefore, frequency reuse means that adjacent cells have different frequencies in use, but further away residing cells can reuse the frequencies without worrying about negative effects, since the distance and low powered transmitting equipment are withdrawing the interference related problems [56]. The topic of how frequency spectrum is allocated between different technologies and finally between users is out of this paper's scope. If the reader is interested, then the topic could be explored in multiple books which cover the fundamental concepts of cellular networking [70] [56].

The last basic characteristic, which this paper will address is the concept of architecture, which will be described separately in the next chapter with GSM, *second generation* cellular network. GSM was picked because it was the first pan-European (even global, with some exceptions) standard on top of which upcoming generations have been built with backwards compatibility. Therefore, it is still active and fundamentally important to know, even though new technologies have introduced some figurative aggregative simplifications to the architecture and could be easier to explain and understand.

2.3.4 GSM architecture

Public Land Mobile Network (PLMN) is a network operated by a government owned administration or private mobile operator for providing collection of land mobile telecommunications service to the public. [35] PLMN of GSM has been described visually in the Figure 6, where we can see the radio network and fixed network (Network Switching Subsystem, abbreviated as NSS) with interface names from the user equipment to NSS.

PLMN may have one or more Mobile-service Switching Centres (MSC), which organize the communication between the radio system and the fixed networks as the interface. MSC performs all the signaling and switching to the mobile phone, which resides in its area. Main procedures are the location registration and handover between different cells. [40] [37]

Several MSCs may be required to cover an entire country, but MSC is just merely a

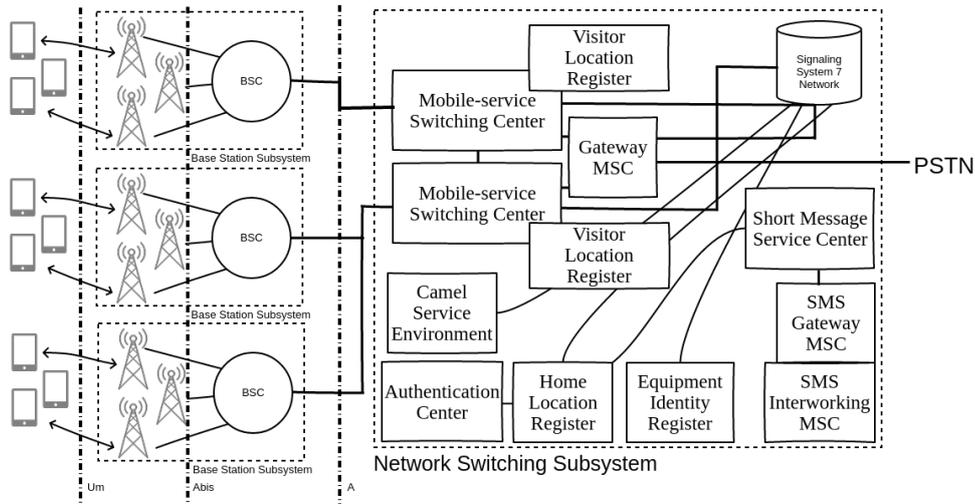


Figure 6. The basic architecture of GSM, second generation cellular network.

sub-unit in the communication between the radio system and the fixed network. Therefore, to achieve radio coverage of geographical areas, the mobile operator needs Base Station Systems (BSS).

Communication between MSC and BSS occurs through A-interface, which carries information about BSS management, call handling, and mobile mobility management. [38]

The base station system is a system containing vital equipment to build communication between MSC and Mobile Station (MS). Vital equipment consists of Base Station Control (BSC) and one or more Base Transceiver Station (BTS), which each serve one cell; therefore, BSC controls multiple cells through BTSs.

The Base station control keeps track of its one or more BTSs through Abis interface. This interface manages the services offered to the mobile stations, allows control of radio equipment, and allocates radio frequencies in the BTS.

The mobile stations are physical user equipment to interact in PLMN network. The physical user equipment consists of Mobile Equipment (ME) and the subscriber identity module (SIM).

Um-interface administers the communication between the mobile stations and the BSSs.

Inside of PLMN's fixed network, also known as network switching subsystem or

GSM core network, there are components which carry out the call switching and mobility management functions. MSC is one of them, but there are many other components:

Home Location Registration (HLR) is a central location register which stores subscriber information, location information to charge and route the call to appropriate MSC, where the MS is positioned. The subscriber information holds International Mobile Subscriber Identity (IMSI) and one or more Mobile Station International ISDN (MSISDN) numbers [39]. Integrated Service Digital Network (ISDN) is a technology which enabled to have simultaneous digital transmissions over the circuit-switched public switched telephone network [34].

Visitor Location Register (VLR) is location register closely bound to MSC, which oversees the user equipment's area transfers. MSC uses VLR for retrieve information. For example, it uses VLR to handle calls for the mobile stations situated inside of MSC's jurisdiction. VLR stores IMSI, MS ISDN number, Mobile Station Roaming Number (MSRN), Temporary Mobile Station Identity (TMSI), Local Mobile Station Identity (LMSI) and the location area of MS [39].

Gateway MSC (GMSC) is, during the delivery of the call to PLMN, the specific MSC which after the interrogation about the MS location with HLR, performs the routing to the actual location of the MS. [36]

Equipment Identity Register (EIR) is a logical entity which stores and shares the International Equipment Identities. It contains MS classification states. MS may be *white listed*, *grey listed*, *black listed* or *unknown*. [35]

Camel Service Environment is another logical entity which offers operator-specific services. [35]

Short Message Service Center (SC) stores, forwards, converts and delivers SMS messages. [35]

SMS Interworking MSC is between PLMN and SC, transferring SMSs from the MSs to SC. [35]

SMS Gateway MSC is a unit which determines suitable GSM PLMN that can send SMS from the short message service center to the MS. [35]

Authentication Center (AuC) handles and stores each mobile station's IMSI, responsible for authentication and ciphering communication over the radio path - cipher key management. During the authentication procedure, the ciphering key is set. [42]

2.3.5 Generations

We have already discussed the pre-cellular era, the shortcomings of the first radio telephone systems, and how the Bell Labs led research towards the first cellular networks. Pre-cellular era technologies are called 0G, since they are predecessors to the *first generation* technologies. 1G had issues as well; the roaming all over the Europe was made difficult due to following facts. At the time 1G was to be deployed, there were newer multiplexing and channel accessing technologies available. Therefore, the European Telecommunications Standards Institute (ETSI) was created with a focus on working out the unique solution with the latest technologies. As a result, 2G solution called GSM was declared the pan-European standard of mobile communication.

On top of GSM, multiple levels of technologies have been built, maintaining backwards compatibility. After decades, there have been numerous upgrades and currently there is ongoing work to release 5G commercially in 2020 [73]. As a result, we are exposed to different generations of mobile technologies, which have been or are being used by Estonian mobile operators :

1G is a code-name for the analog signal based telecommunications standards. 1G introduced the cellular network concept. A very important concept related to this topic is Frequency-division multiple access (FDMA). FDMA is a channel access method where the users get a channel which is divided into two frequency bands: one band for uplink (MS to BTS) and one band for downlink (BTS to MS). In FDMA, the unique channels are multiplexed into one carrier frequency using a modulation. Multiplexing is a method used to combine multiple analog or digital signals into one shared medium and, with multiple access methods makes it possible for numerous transmitters to connect to the same physical medium and share its capacity. This approach's main flaws are a small maximum data rate, and the use of multiple frequency bands and band guards to avoid interference limits overall capacity of users. [60]

2G went from analog signal to digital, bringing along a new multiplexing method to improve voice capacity and quality. GSM presented Time-division Multiple Access (TDMA). Time-division duplexing is a concept where at any given time happens only downlink or uplink transmission between two connected parties, but with TDMA the resource can be accessed by multiple parties. [60]

2.5G generation introduced two new technologies: General Packet Radio Service (GPRS) and Enhanced Data Rates for GSM Evolution (EDGE). Before, GPRS users accessed data services in a circuit-switched data approach, which means that two network nodes have to establish a dedicated channel for communication, known as a circuit. This approach has clear downsides, there are possibilities that the channel is under-loaded when the user is less active (the channel is allocated for the user). In contrast to that, the channel can be overloaded by a single active user. Old circuit-switched systems often limited themselves to manage only one circuit connection per device. GPRS established a new packet-switched core network to support packet data services in a more efficient manner than old circuit-switched solution. Packet-switching is a concept where data gets divided into packets and transmitted over the network independently. Channel dedication became obsolete and packets were transmitted through channel which were now shared by multiple simultaneous communication sessions. EDGE brought radio network enhancements to GPRS, which resulted in higher data rates while using packet data services. EDGE added no new NSS components. From 2G to 2.5G we saw better resource management and, over time, more efficient and improved packet data services. [41] [46]

3G managed to grow the capacity of the users and brought data rates in hundreds of kbps. ETSI consortium called this generation mobile technology as Universal Telecommunications System (UMTS). Another channel access method called Code-division Multiple Access (CDMA) was standardized. CDMA used wide-band transmission, introduced new universal frequency reuse, improved interference control, reliable handovers, and power settings. [52] [43] This spread-spectrum multiple access method allows several users to share a band of frequencies, since a special coding scheme assigns to each transmitter its own code [56].

4G has high data rates in Mbps, using yet another multiplexing technique called Orthogonal Frequency-Division Multiplexing (OFDM), an IP-based network architecture, beared down packet-switched core network. Efficient and cost-effective solution for modern high data demand. [52] OFDM encodes data on multiple carrier frequencies. Each sub-carrier uses modulation schemes like quadrature amplitude modulation or phase-shift keying [55].

5G is coming up in the future and has currently mapped out some of the requirements which the new standard should address.

2.3.6 Mobility management

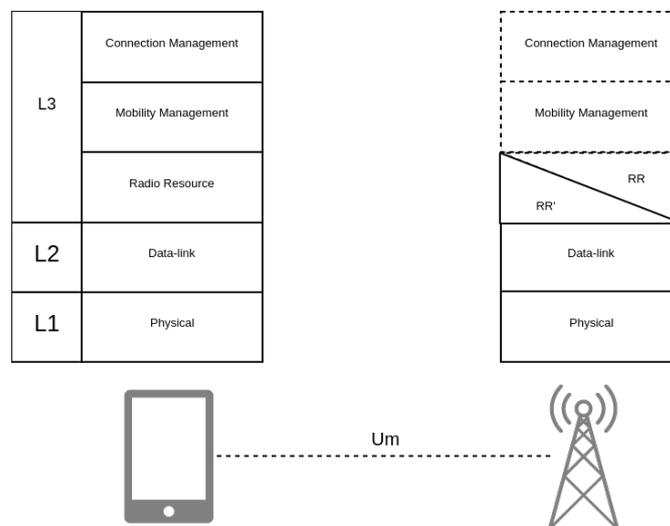


Figure 7. Overview of mobile station's communication layers with its another end (BTS).

Mobility management is an important concept. This section describes MS related mobility management (the network also has its own mobility management), because during the development of the simulation, the devices' behaviour was programmed according to GSM/GPRS mobility management, while GPRS related session management was not included.

As visible in the Figure 7, communication between MS and BTS through Um interface - GSM signaling protocol has been gathered into three layers:

Layer 1 is the physical layer, which offers transmission on a limited set of logical channels. Logical channels are multiplexed on physical channels. [44]

Layer 2 is the data-link layer. Layer 2 bases on Link Access Procedure on the Dm channel (LAPDm), which provides a reliable signalink between the MS and the network . [51]

Layer 3 has three sub-layers:

Radio Resource Control (RRC) manages the common transmission resources (physical channels) and the data link connections on control channels. It contains of procedures to establish, keep running, and release the dedicated radio connections. [50]

Mobility Management (MM) sublayer supports the mobility of MS; it informs the network about the location through location update (also periodic location update), manages authentication, identification, and IMSI related procedures etc. [54]

Connection management (CM) is a sub-layer, which also consists of three sub-layers Call Control (CC), short message services (SMS) and supplementary services. [49]

Mobility management depends on the RR connection. It calls out the establishment of RR connection to perform its own procedures. Import aspect is MM states, which were essential for the simulation. While RRC has two main states, IDLE MODE and CONNECTED MODE, the MM has many more states, which are dependent on the RRC states. During RRC IDLE MODE the MS mobility management resides in its own MM IDLE state. But when the mobility management conducts its procedures, it establishes RR connection and is maintained until the MM returns back to its IDLE mode or the user equipment gets turned off. Mobility management has following main states:

- NULL,
- LOCATION UPDATING INITIATED,
- WAIT FOR OUTGOING MM CONNECTION,

- MM CONNECTION ACTIVE,
- IMSI DETACH INITIATED,
- PROCESS CM SERVICE PROMPT,
- WAIT FOR NETWORK COMMAND,
- LOCATION UPDATE REJECTED,
- WAIT FOR RR CONNECTION (LOCATION UPDATING),
- WAIT FOR RR CONNECTION (MM CONNECTION),
- WAIT FOR RR CONNECTION (IMSI DETACH),
- WAIT FOR REESTABLISH,
- WAIT FOR RR ACTIVE,
- MM IDLE,
- WAIT FOR ADDITIONAL OUTGOING MM CONNECTION,
- MM CONNECTION ACTIVE (GROUP TRANSMIT MODE),
- WAIT FOR RR CONNECTION (GROUP TRANSMIT MODE),
- LOCATION UPDATING PENDING,
- IMSI DETACH PENDING,
- RR CONNECTION RELEASE NOT ALLOWED. [54]

MM IDLE has its own substates, which are important because they reflect the state of the device even more accurately, e.g., not having the connection or no valid subscriber data(SIM card missing). The substates are:

- NORMAL SERVICE,
- ATTEMPTING TO UPDATE,

- LIMITED SERVICE,
- NO IMSI,
- NO CELL AVAILABLE,
- LOCATION UPDATE NEEDED,
- PLMN SEARCH,
- PLMN SEARCH, NORMAL SERVICE,
- RECEIVING GROUP CALL (NORMAL SERVICE),
- RECEIVING GROUP CALL (LIMITED SERVICE),
- eCALL INACTIVE. [54]

2.3.7 Description of main mobility events

In this subsection, we are going to discuss project related mobility procedures (do not confuse with mobility management procedures, but rather think about overall mobility procedures), which were important for the simulating cellular network behaviour.

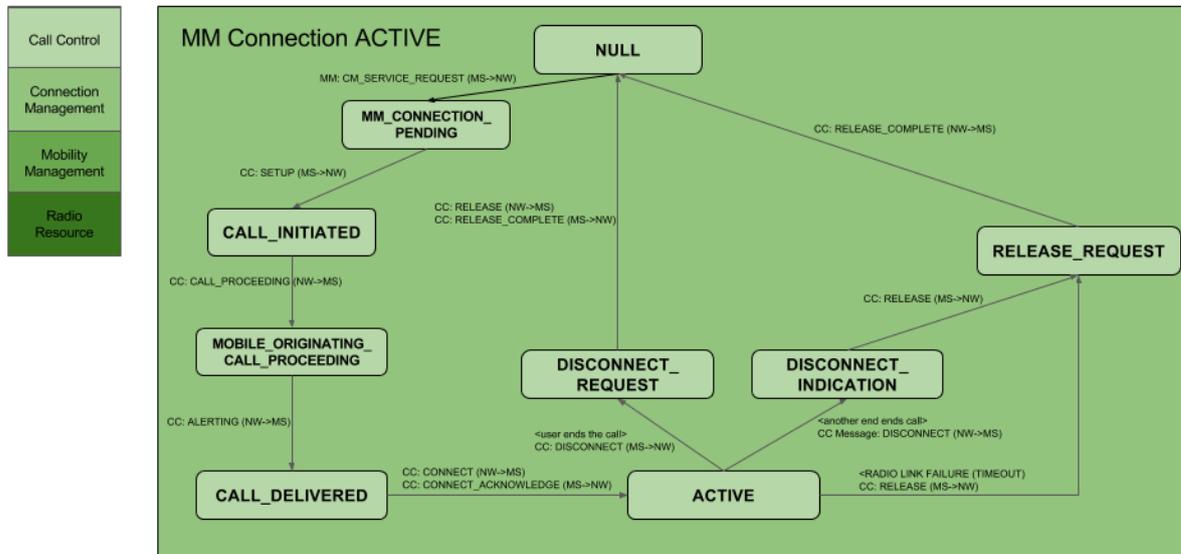
The simulation relevant mobility procedures:

Handover is an RRC specific procedure to change channels (intercell, intracell), determined by the MS or the network on the basis of signal level, connection quality, power level propagation delay, and the traffic reduction between the transmitters. [48]

Cell selection is an RRC function during which MS tries to find a suitable cell to camp on, while cell reselection is a process where the MS is periodically looking for a better cell to camp on. [45] In this thesis scope, the cell selection and the cell reselection are synonym since their workflows in the code level are the same.

Location update is MM specific procedure, which occurs during normal location update, other procedures, periodic location update, and when IMSI is attached (SIM card inserted). The location update registers or updates the mobile station's actual location area in the network. Periodic update procedure is controlled by a T312

Connection management, Call Control
 Mobile Station, Mobile originating call proceeding



Source: 3GPP TS 24.008, 3GPP TS 44.069, 3GPP TS 44.118

Figure 8. Simple overview about initiating a call and what happens on the connection management layer. [54] [47] [53]

timer in the mobile station. When the timer hits the constant with a counting, then it starts location update procedure. [54]

During the cellular network simulation, there are happening the mobile events like calling, sending SMS messages or web connection open over the communication management layer. On the top of the connection management, during the call, there is the call control managing the state, which handles the incoming and outgoing call events. In the Figure 8 we see the process when the call is originating from the MS. In the Figure 9 we can examine what is happening in the terminating end of the call.

2.4 Conclusion

In this chapter, we introduced the microscopic traffic simulator SUMO and the background of cellular networking, including some specific details about the architecture in order to familiarize the reader with the topic and to present the scope's boundaries to

3 Cellular communication network simulation

The intent of this chapter is to describe the core of this thesis; what, how and why something was done and eventually what kind of technical outcome was achieved in goal to produce cellular network behavioural data simulator.

3.1 Introduction

The extent of the work to understand the cellular network and to model it into SUMO traffic simulation had considerable difficulties. The learning curve to understand the basic concepts of both components demand lots of resources. Modelling architecture of over decade old code to realize the work flow and to design new features on top of that. Development was burdened with many project specific demands about documentation and C++ related robustness about writing; slow writing speed and poor stack traceless error management. Learning cellular network required sketching and visioning different systems and related technologies to grasp how the mobile technologies are working nowadays.

The vastness of cellular network made it clear that for the simulation of cellular behaviour there is a need to define a scope, which is achievable during the thesis limits. The NSS specific functionality has been left out or delegated into the mobile station. Primary goal of the thesis is to manage the mobile device to generate the cellular logs, similar to CDR through the symbiosis of traffic and cellular simulations.

3.2 Used technology

The technologies used in this thesis are primarily influenced by SUMO traffic simulator, in the perspective of the development task means that technologies described in the Chapter 2.2.3 were also used by the author.

For C++ there was used GNU Compiler Collection version 4.8.4 [9], project code was managed with Git [11] and integrated development environment was Eclipse CDT [5]. Some minor scripting was done in Python 2.7 [24] with PyCharm [23]

3.3 System architecture and design

3.3.1 Scope

In the Chapter 3.1, the scope was incoherently defined; in this subsection, the exact borderline will be explained. The central piece of this entire simulation is the user equipment, which produces CDR-like activity logs.

The network has been constructed in a manner that resembles a radio system, with one component from the NSS - mobile-service switching center. In the code level design, tasks of BSC and MSC are aggregated into one entity - towers control (SUMO architecture influenced buildup), while their main functionalities are removed or delegated to the mobile device class, where they have been also pruned and emulated in lower extent.

Mobile devices communicate with transmitters (called antennas in the code level). Transmitters are tied with physical tower entities, which are representing base station systems. BTSs are container type of classes to store tower's antennas and its location related information.

From the Chapter 2.3.6, we know that the mobile device and its communication with BTS over Um interface is vast and complicated due to the different protocol layers. The user equipment mainly has the mobility management layer, which is defined with some radio resource layer specific functions. The mobility management will be described in the Chapter 3.5.2.

Upgrading SUMO GUI to visualize the tower position and its transmitters coverage areas - hexagons. SUMO has since version 0.26.0 mesoscopic simulation model [26], but in this thesis, we want to produce individual entity level mobility logs about its cellular behaviour through the microscopic traffic simulation. Therefore, we are not utilizing mesoscopic model.

3.3.2 New models

Cellular tower is the presentation of the physical tower with the following attributes:
point of interest, which contains the cartesian and real life geospatial information;
list of cellular antennas; and connected mobile devices connected with tower's transmitters.

Cellular antenna , according to the cellular network definitions a transmitter, is a data structure that facilitates information about its coverage as polygon shape (e.g. hexagon, custom polygons are also supported). Antenna also have these following attributes: signal strength, frequency, radius, and reference to its tower. It also has an id value, which represents Common Gateway Identifier (CGI) value.

GPS device is defined as in-vehicle device producing data, known as geographical coordinates. The device and its subclasses have been explained more thoroughly in the Chapter 3.5.1.

Mobile device is another in-vehicle device which is used for the cellular network behaviour simulation. The device and its subclasses have been explained more thoroughly in the Chapter 3.5.1.

Mobility Event Data is a data structure model that contains a single mobility event which has been generated by Mobility Event Simulation Generator (MESGEN), which has been developed within this thesis. The model has three attributes: event start, event end and event type. The mobility event data has been labeled into following types:

NONE refers to error state, the event is in the void state;

CALL_INIT is an event for the call initiation process;

CALL_REC , incoming call event;

SMS_SEND - event describing that the device sends a SMS;

SMS_REC is for the device received a SMS;

WEB_COMM describes the ongoing web communication;

KEEP-ALIVE event is not generated, but it occurs during the simulation when the T312 timer initiates periodic location update;

RADIOLINK_FAILURE is another event which occurs during the simulation, bound to the signal propagation and losing the connection to the transmitter and no handover candidates were found. Radio link failure is an RRC layer technical term for losing the connection.

3.3.3 Visualization

The scope of this thesis aims to support the command-line and the visual simulation. Therefore, during the simulation, we needed to extend some core simulation models into the graphical user-interface supporting models.

GUI Cellular Tower depends on the SUMO graphical class *GUIPointOfInterest*. It contains the cartesian position coordinates and methods about visualizing the location. Therefore, GUI cellular tower basically wraps itself around the graphical user interface point of interest.

GUI Cellular Antenna has been implemented through wrapping itself around SUMO's *GUIPolygon*. SUMO's GUI class, polygon gives us the visualizing possibilities for drawing polygons in various styles.

3.3.4 Core mechanics

The main class of SUMO creates the central entity called *MSNet* and loads the program specific configuration options. Setting up the simulation network and performer requires simulation data and logical entities to manage them. The core SUMO simulation depends on the work of the Chapter 3.3.5, where we describe the data import into the traffic simulation program. Data import occurs during the road network loading, which is done by *NLBuilder* class. For SUMO GUI, the loading has been delegated to the thread *GUILoadThread*, where *NLBuilder* exists and does its work.

During the importing and network building, various entities with a wide area of functionality and commitment regions are created. There are control entities, which are created by their respective builders. The control entities are, e.g., logical control objects which administer simulation specific entities like road edges, lanes, junctions, detectors for triggering, traffic lights, and cellular towers. There are two types of dynamic controls for the routes and the mobility event data. The dynamic control means that they are loader, actively loading new routes and events during the simulation according to the step of the simulation. During loading the routes the vehicles are also created and added to *MSNet* related vehicle control unit.

Before the simulation, after building the network, the graphical SUMO builds up the visual presentation for the user while using given data and graphics library OpenGL. For

graphical version of SUMO *GUINet* is extending *MSNet*, adding attributes and methods for the visual simulation.

The simulation cycle is happening in *MSNet*. The simulation step has been, by default, defined to be equal with one second in real life (there is a possibility to configure to alter the value of SUMO time, but it has been noted that it might break some simulation models and, therefore, not advisable).

During the simulation step, *MSEventControl* entities are used for various simulation actions. In the Chapter 3.5.5, the essence of the mobility events simulation shall be explained more thoroughly. The problem is that certain actions have presumptions and dependencies on other resources, which need resources' states updated before their dependent events can work properly.

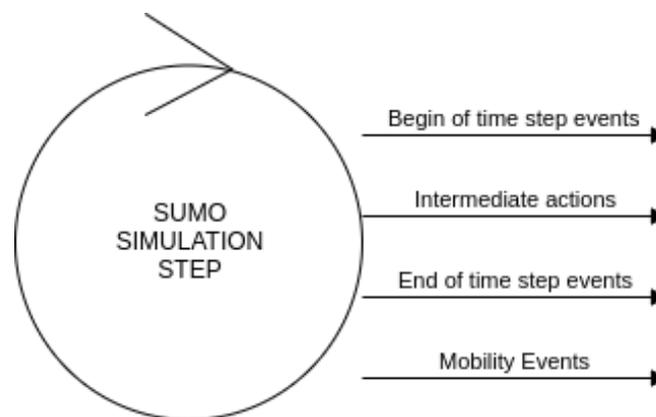


Figure 10. The simulation steps in the SUMO simulation cycle.

The simulation has been divided into four components (visualized in the Figure 10):

Begin of time step events is an event running cycle during which SUMO executes, for example, traffic light specific events, trigger type events (lane speed, rerouting), routing device, and pedestrian move events.

Intermediate actions are dependent on the previous event running cycle. During the time between two event running cycles, SUMO is detecting collisions, checking for the need of changing traffic lights, check if the edges are active (for simulation purposes the user can turn off the lanes), planning vehicle movements, executing vehicle movements, checking for the collisions again, run lane changes, check

if the previous movements created collisions, load new routes and mobile events for the cellular simulation, check if there are people or containers waiting for the transport, check and insert new vehicles to the network, insert new events, check about inserted vehicles. During the execution of vehicles movement, there is a case when the vehicle has arrived to the destination and, immediately afterwards, the vehicle shall be removed from the simulation. This function has been altered to have checks for existing mobility devices (GPS, mobile) to run their last events directly and then invalidating those events to stop them from recurring. This entire code alternation prevents losing data (otherwise, if the vehicle is removed before the mobility events, no arriving of the last event) and null pointer exceptions.

End of time step events is an event running cycle which runs the events dependent on the updated location of the routable elements (vehicles, pedestrians etc). There are also step by step logger events, vehicular devices' specific events (Bluetooth, GPS, mobility management related state update) and vehicle flow calibrator events.

Mobility events is a newly added event running cycle entirely for the purpose of mobility event simulation. The events are created by the mobile devices from the data, which has been attached to the vehicle after it has been loaded into the simulation during the intermediate actions.

3.3.5 Data import, output and feature configurations

Running the simulation requires its configurations and data to be imported into the program. For this, and altered multiple XML Schema Definition (XSD) files have been created [31]. XSD files specify how to build Extensible Markup Language (XML) files [6]. Describing how the importable elements should look like for being prepared for the reading process into SUMO.

There has been defined cellular network related XSD file, defining the cellular towers and their antennas. Mobility event data from MESSAGES has been also specified for the import purpose. To include new configurations for SUMO, alterations in existing XSD files were required to enable data import through files.

During the building of the network to perform the simulation with *NLBuilder* so-called handler elements are created for handling the XML data sources. For importing

the cellular physical network data and the mobility event data, base classes were created under the utils for importing the data. The creation of them into the utils is because the tools might need to import similar data and might have different data structures and workflows, so they can make their own extensions for those purposes. Cellular and mobility specific handlers are extending Xerces library class as a base class for XML importing. For SUMO, *microsim* extended versions were implemented to suit for the needs of the microscopic traffic simulation program.

Builder classes, which oversee the creation and populating of *MSNet* control entities, extract data from the XML files and transform them into the models governed by control entities. Our control classes covered command-line version and GUI version control creations. GUI cellular builder, for example, had to deal graphics centric classes like *GUIPolygon* and *GUIPointOfInterest* to be compatible with *MSNet* which operates with ordinary *Polygon* and *GUIPointOfInterest*. Upcasting techniques were required.

Output after the simulation has been written in the dynamic manner without XSD definition. The output has been built around C++ *std::ostream* (output stream) inside of SUMO's *OutputDevice*. *OutputDevice* has support for writing data inside of XML elements.

3.3.6 Extent of the development

In the Figure 11 is described this thesis contribution in the overall scale compare to SUMO. The white rectangles show the entities, which have not been altered during the development, orange that this section has new features and the green blocks mean that those are entirely new contributions.

3.4 MESGEN and HexagonGen tools

3.4.1 MESGEN

Mobility Event Simulation Generator is an independent program for preparing pre-requisites for the cellular network behaviour simulation. Some mobility user types have been defined to reflect real life. For example, the program describes casual user, idle user, and frequently SMS sending and receiving teenager. Each type has been described by three attributes: call frequency, SMS frequency, and web frequency. Each of the

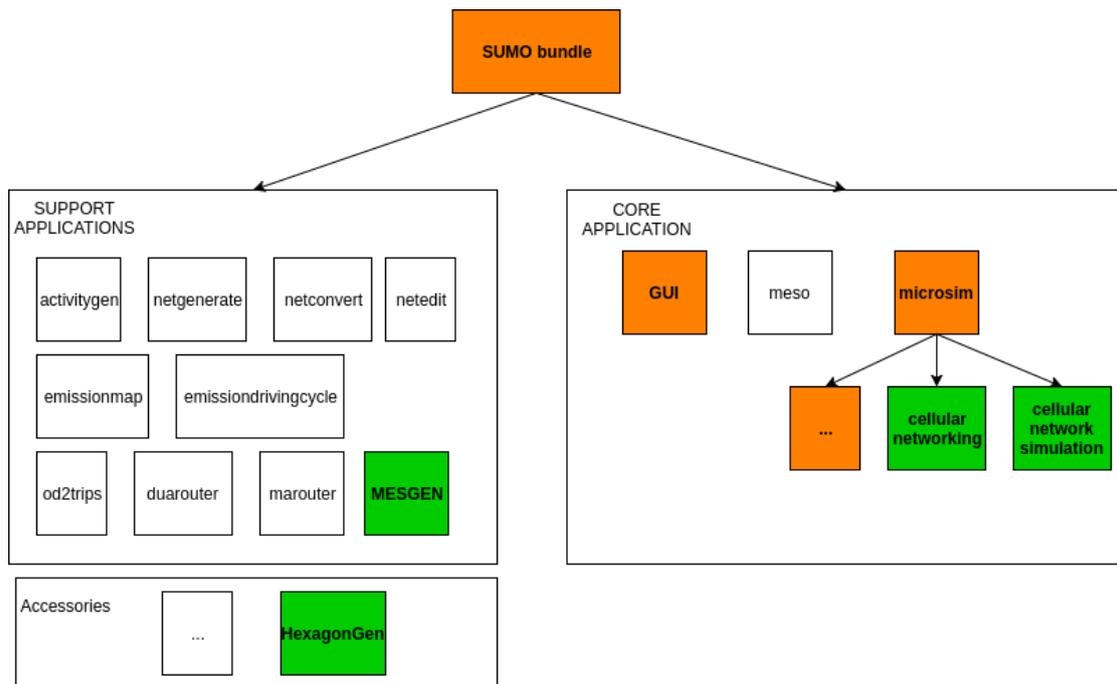


Figure 11. The state of the SUMO application bundle after the contribution from the author.

fields can be distributed into four categories, presenting the user’s activity: none, low, medium and high.

The mobile events which will be generated are similar to mobile event data model’s events, but with some reductions. The generated mobile event types are:

- SMS_REC,
- SMS_SEND,
- CALL_INIT,
- CALL_REC.

MESGEN takes as an input network file and route file to determine the vehicles and generate cellular mobility events. While reading in the routes, the network data is needed for validating purposes. The route file is important because of the vehicle id and

its departing time. The result file of MESGEN is an XML file which can be used as an input file for SUMO to simulate cellular network behaviour.

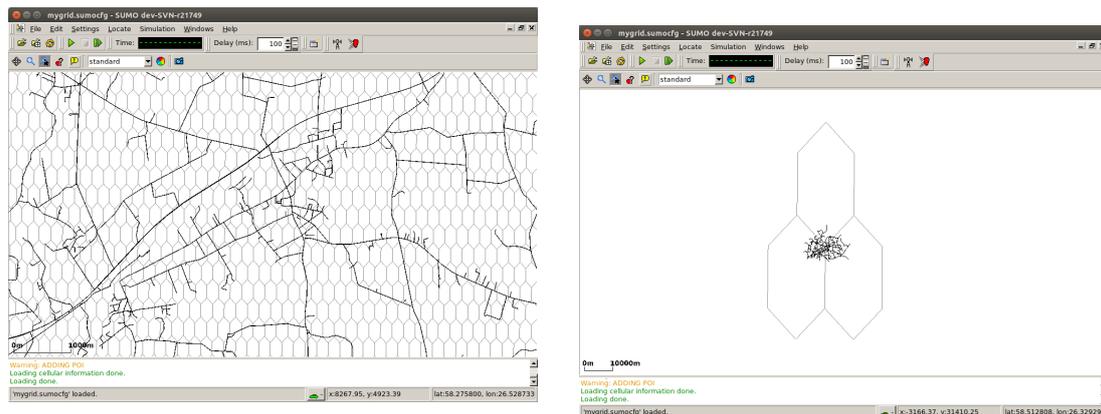
The program generates logs in the following manner:

1. Pick a random character (mobility user type).
2. After picking a character, the event type control sum will be calculated from the character mobility type properties (e.g., if high level calling property equals with 3, medium level SMS frequency equals with 2 and low level web communication frequency equals with 1, the sum will be 6).
3. Events for the person in the vehicle will be generated through simple while loop. The current time, which at the beginning is equal to the vehicle's depart time, will also be initialized. The will loop long until when the simulation end time minus our defined current time variable condition equals less than 0.
4. In order to decide the event, an arbitrary number between 0 and previously calculated sum. The random number will then fall into the category of mobility user type and then the occurring event has been decided.
5. The next step is to decide against the activity threshold if the chosen event happens or not. If not, then the event does not occur and a random unit of time will be added to the current time - narrowing the timeframe.
6. If the event happens, the length of the event will be calculated and the event will be saved. Finally, the timeframe gets a random event buffer time added to the current time - narrowing the timeframe between the simulation end time and the vehicle entity's current time.

Minimum length and maximum length of different events can be configured through variables. It is also possible to define time buffers between occurring events. The values of low and medium frequency events' thresholds and random number seed can also be set through the configuration file.

3.4.2 HexagonGen

HexagonGen is an accessory level script, written in Python, to generate 120° sectorized cells for the cellular network behaviour simulation. It takes an input of latitude and longitude coordinate values, size (length of hexagon's side), hexagon network width, and height values. The given coordinate pair is the center of the drawn network (tower position). The result of the HexagonGen is an XML file which can be directly fed into SUMO.



(a) Example of cellular network grid with lots of micro cells, where the hexagon side length has been defined as 200 metres.

(b) Example of macro cells with hexagon side length as 20km.

Figure 12. The village of Nõo with two types of cellular coverage with help of HexagonGen.

3.5 Mobility behaviour and management

This subsection is about mobility devices, their soul-life, and how they have been integrated with SUMO program.

3.5.1 Microscopic simulation devices

GPS device, also known as *MSDevice_GPS* in the code, is a test device, in the sense that it was created with the objective to study SUMO simulation cycle for the upcoming mobility device. Mapping out all the requirements for the development while researching through SUMO simulation core while engineering the

GPS device. *MSDevice_GPS* is extending *MSDevice*, which is SUMO's abstract in-vehicle device. Therefore, it already has some basic interfaces to communicate with the vehicle and abstract methods, which need to be implemented to develop a working device.

MSDevice_GPS has two subclasses called *RouteGPSInfo* and *GPSSignalUpdate*. *RouteGPSInfo* is a data structure that records geographical coordinates and the timestamp. *GPSSignalUpdate* represents GPS signal; it is an event by extending SUMO *Command* class (base SUMO *microsim* event class). During every simulation step, if the vehicle has a device, the event gets ran after the vehicle's movement.

Mobile device , is known as *MSDevice_Mobile* in the code. It is a very enormous class which simulates the mobile behaviour in multiple levels. It is emulating for the simulation multiple functions, which in real life are actually done by other logical units, or at least in cooperation with other entities (through chain of events, clarifying states, allowing procedures, etc). New SUMO mobile device's mobility management will be examined in the Chapter 3.5.2. Signal pinging during the simulation pinging will be discussed in the Chapter 3.5.3. Primary state changing procedures will be described in the Chapter 3.5.4. Events which direct the mobility behaviour will be elaborated in the Chapter 3.5.5. Finally, the result log object will be analyzed and discussed in the Chapter 3.5.6.

Mobile device is collecting its activity logs (the primary objective of the thesis - generation of logs), current signal strength, and boolean primitive flags if the mobility event is currently occurring and the connection with the transmitter has been interrupted. It also contains references to the mobility management entity and for the active mobility event (if the vehicle, and therefore the device, gets removed from the simulation, then through the both way reference we can invalidate an event in the event cycle).

In the Figure 13 has been described the mobile device in the SUMO program in the form of class diagram. It has defined the relationships with other mobility classes.

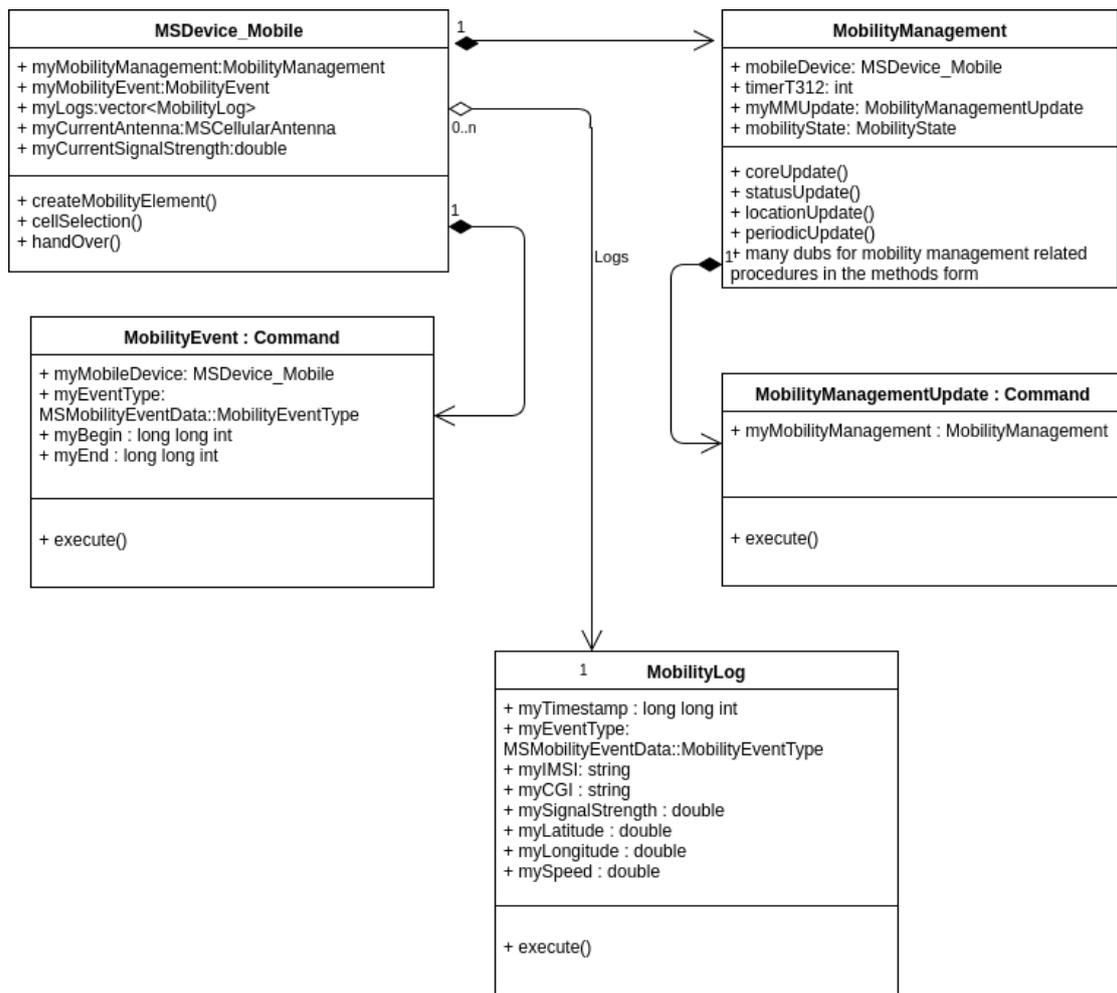


Figure 13. The class diagram of *MSDevice_Mobile* and its sub-entities.

3.5.2 Mobility management

In the Figure 14, we can see mobility management's reduced version. Layer 3's three sublayers, along with their actions, have been described in the context of this thesis.

In *MSDevice_Mobile* there is a subclass called *MobilityManagement* which governs mobile device's state during the simulation. The mobile mobility state is managed through the end of simulation step event cycle. During the creation of the vehicle into the simulation, it shall be equipped with a wide variety of devices, according to the configuration setup. If the vehicle has been equipped with a mobile device, then during

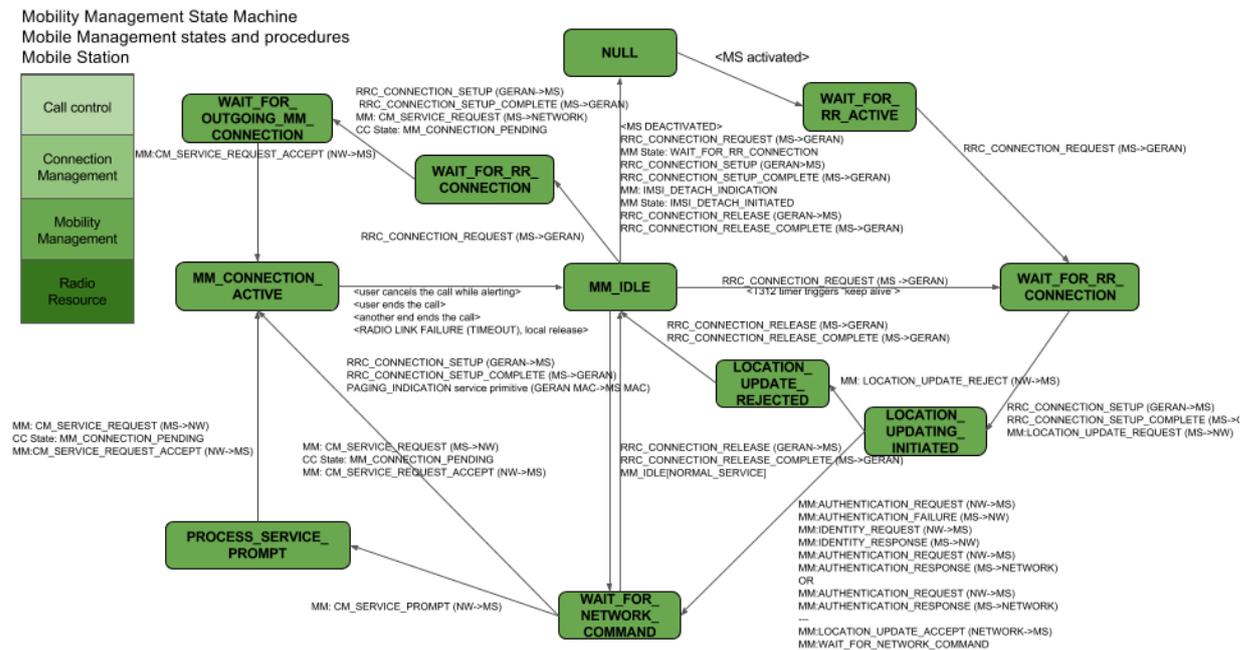


Figure 14. Reduced mobility management state machine. [54] [53]

the device creation, it will be created with mobility management entity. Mobility management entity has *MobilityManagementUpdate*, an entity which is extending *SUMO Command* base abstract event class. The mobility management update event entity will be added to a particular *MSEventControl* entity, which handles the end of the simulation step events' execution. Therefore, the mobility management is updated after the vehicle movement in the end of the simulation step. Well almost at the end of the simulation step, after mobility management update there will be the execution of mobility events which are dependent on the latest state of mobility management.

The mobility management has the following states:

- MM_NULL,
- MM_IDLE_NORMAL_SERVICE,
- MM_IDLE_SEARCH_FOR_PLMN,
- MM_IDLE_NO_CELL_FOUND,

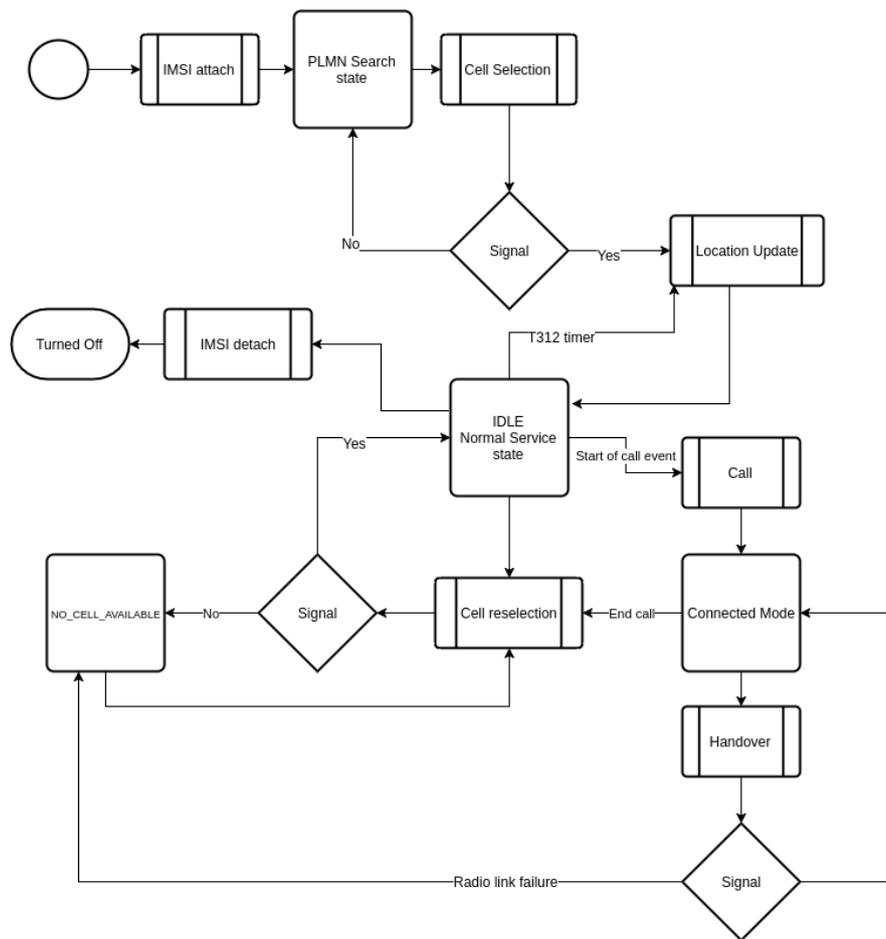


Figure 15. The simple work flow of the mobility management.

- MM_CONNECTION_ACTIVE,
- MM_WAIT_FOR_RR_ACTIVE,
- MM_WAIT_FOR_RR_CONNECTION,
- MM_LOCATION_UPDATING_INITIATED,
- MM_LOCATION_UPDATE_REJECTED,
- MM_WAIT_FOR_NETWORK_COMMAND,
- MM_PROCESS_SERVICE_PROMPT,

- `MM_WAIT_FOR_OUTGOING_MM_CONNECTION`.

Mobility management has a reference to the mobile device to call out mobility related procedures from the lower layers. Then there are mobility management state related variables; the T312 timer is for reminding a mobile device to make periodic location updates after every user defined steps (by default 10 steps). Some state related extra scenarios, define whether the mobile has been turned on, the SIM card inserted, the IMSI attached, and define whether the mobile has PLMN related information.

In the Figure 15 has been the mobile mobility life cycle described visually for better understanding.

At the start of the mobility management update, the creation of the current time step's mobility events for the simulation for this specific device takes place. The mobility events are explained in the Chapter 3.5.5.

During the mobility management update event, there will be called out management core's update function which increments instantly periodic location update and then, according to the previous state, goes through state machine to determine the next step:

If the mobility management state has been classified as *NULL*, which actually should be impossible, but nonetheless the core update will change mobile to the state as it was just physically turned on.

MM_IDLE_NORMAL_SERVICE starts mobile cell selection method on the device level. Described in the Chapter 3.5.4.

MM_IDLE_SEARCH_FOR_PLMN starts mobile cell selection method, but since the device has never been connected to the network, it also waits for the surrounding PLMN information.

MM_IDLE_NO_CELL_FOUND start mobile cell selection, since there is currently no connectivity.

MM_CONNECTION_ACTIVE starts the cell selection process for the handover purposes, if needed. Described in the Chapter 3.5.4.

As the reader could detect, in the state machine description there are less states described than in the previously outlined list. That is because many of the states are

actually intermediate states, and during the simulation, there are no meaningful purpose for them.

After cell selection, the workflow will move from the device back into the mobility management, where the state of MM will be updated and location update initiated. Location update is described in the Chapter 3.5.4.

If the state machine has done its work, then there is a successive check to determine if the periodic update is needed and, if it is, then it will be executed. The rule with the periodic location update is that if a mobile has not had any location updates during some certain amount of simulation steps, then the mobility management has to step in and remind the transmitter that it still exists and has not been lost during the commute.

3.5.3 Wave propagation from the transmitter

During the cell selection, the mobile device has multiple challenges. It has to determine the nearest polygons, determine in which one of them it resides, and determine the signal strength from the base station's transmitter.

The signal strength has been envisioned by the free space wave propagation model. It indicates the power received by the antenna under ideal conditions. The free space model predicts the power's decay to be the negative square root of the distance. In our case, we have not included the system loss factor. This could be included in the future work and allow real life like signaling in SUMO.

The signal strength is calculated with a formula described in the Definition 1. The distance in the formula has been calculated between the tower and the vehicle and for that, Haversine distance, defined in the Definition 2, was used. Haversine distance is used for distance calculation on the sphere (in our scope planet Earth) using two points longitudes and latitudes.

Definition 1. Friis free space equation:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L},$$

where $P_r(d)$ is the received power, P_t is the transmitter output power, G_t is transmitter gain, G_r is antenna gain, d is the distance between the MS and tower, and L is the system loss factor. [70].

Definition 2. Haversine distance:

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right),$$

where ϕ is the latitude, λ the longitude and r is the Earth's radius in meters (6 371 000 meters).

During the signal strength calculation, we use the configuration supplied tower transmitter output strength (dBm) and the transmitter's frequency. From the frequency, we calculate the wavelength. Wavelength equals the speed of light divided by the transmitter emitted frequency.

3.5.4 Location update, Cell selection and reselection, Handover

The procedures are key component of keeping track of the mobility management level state, this can be seen also in the Figure 15.

Cell selection is a mobile device level method for measuring the current signal strength value, determining the current antenna to which the mobile device is connected, and assessing the need for the handover procedure.

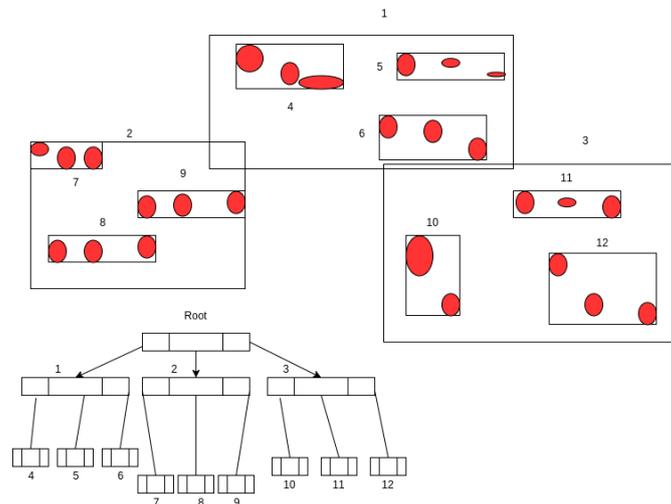


Figure 16. The example of R-tree after all the shapes have been inserted into the tree.

During the cell selection, the first step is to find surrounding cellular network cells. Those cells (polygons) have been structured into the R-tree entity during the creation of the mobility tower control unit, where they were added for a faster search. Otherwise, there would have been brute force like approach, where we would had to run through every single cell in the program with expensive point in polygon algorithm. As visible in the Figure 16, R-tree is a tree data structure to handle spatial data efficiently by indexing shapes for the future access. [57]

After searching the nearest polygons to the device, we check through the vector of cells and determine whether the vehicle is in the cell or not. The vehicle might be in multiple cells; therefore, we are calculating the best signal strength and accordingly, we pick the transmitter and cell to camp on. The check if the vehicle is in the polygon or not has been dealt with *the winding number algorithm*. In essence, *the winding number algorithm* is an algorithm which counts the number of times the polygon winds around the point of interest. If the point is not inside of the polygon, then the resulting winding number is 0. [61]

During the cell selection, the registration to the tower or the deregistration from the tower will be determined. This is basically emulating the registering the mobile device's location area into the visitor location register.

The final step is the status update of the mobility management.

Location update is the mobility management level procedure for updating the device's location in the network. Location update is called out during the status update, the periodic update procedure, and the handover. Location update resets the periodic update timer (sets the timer back to 0) and updates the mobility management state. To avoid spaghetti code, a simplification has been made, so that during the cell selection, the location of the device has been already updated in the network (added to the certain tower's private vector of connected mobile entities).

Handover occurs when a better quality cell has been found during a mobility event. The flags have been set and during the status update, there is a state machine level check about *MM_CONNECTION_ACTIVE* in regard to newly found high quality signal strength cell. If the statement is true, then there the handover process will occur. The handover in our reduced scope means calling out the location update.

3.5.5 Mobility events and the simulation cycle

We have covered the mobility management tied event object called *MobilityManagementUpdate*. In this section, we will discuss about the mobile device related event. *MobilityEvent* extends the SUMO *Command* abstract event base class. The mobility events have their own *MSEventControl* container-event queue and the events are being executed at the very end of the simulation step, after the mobility management update events.

Each mobility event has a type, which are the same types declared in the enum of *MobilityEventType* in the class of *MSMobilityEventData*. The event has the mobile device reference to let the mobile know about the radio link failure and trigger the log creation about the occurred event. Other attributes of the mobility event are the event's start time and end time. Calls and the web communication events depend on the timeframe variables.

3.5.6 Mobility CDR-like logs

Call detail record is the information about incoming and outgoing mobile activities, e.g., calls or SMS messages. The data in the CDR is about the event originated and the terminated parties (both sides' phone numbers), time of connection through the starting time and the call duration, call event type, unique generated id for the record, etc. [58]

MobilityLog class is a data structure that, to some extent, emulates the essence of CDR. The attributes of the mobile logs are the following (also visible in the Figure 13):

- timestamp,
- mobility event type,
- IMSI code, in our case usually known as the vehicle ID,
- Common Gateway Identifier (CGI) is the value indicating to the transmitter id (could be interpreted also as a cell ID)
- signal strength during the event,
- latitude coordinate,

- longitude coordinate,
- vehicle speed during the event (in m/s).

3.6 Conclusion

This chapter explained the practical part of the work with extensive overview about SUMO core mechanics. It was described how the new entities and their functions were integrated into the existing system. Different tools were presented which help to generate data for the cellular network behaviour simulation. Those tools created XML files which needed to be imported into the SUMO program. The entire work flow about that process was thoroughly presented, including the models, which were used for the simulation.

We took a deep look into the mobile device and its mobility management, as well as how it was installed and fit according to the needs and the scope of this thesis. The result of the cellular network behaviour simulation was the created mobility logs files: one of them was the practice and learning task (GPS logs) and the other was the primary task, which is required for vast area of research as an input. We will take deeper look into the logs and the other tangible result in the Chapter 4.

4 Results and Analysis

We will now examine the results of the practical task and give an overview of what has been accomplished. The mobility logs will be presented, along with a tutorial how to produce them. The second part of the chapter takes a glimpse into the performance benchmarking.

4.1 The mobility logs

4.1.1 GPS logs

```
...
<vehicle id="0" depart="0.00">
  <param key="has.gps.device" value="true" />
  <param key="has.mobile.device" value="true" />
  <route ... />
</vehicle>
...
```

Listing 1. Route file, an example how to add devices to the desired vehicle.

MSDevice_GPS related logs can be created in two ways. One way is to configure the main SUMO configuration file in a manner that all the vehicles have a device, as seen in the Listing 2. The output file must be defined into the SUMO configuration file. Otherwise, the output will not be delivered because of the lacking destination. Another possibility is to configure that specific vehicle has a device to the SUMO route file, which has been also shown in the Listing 1.

In the Listing 3 GPS signal events which have been described, in the Chapter 3.5.1 are exhibited. *RouteGPSLog*, as we can see from the Listing, it contains of the timestamp, longitude, and latitude.

```
<input>
  ...
  <mes files value="mes_output.xml" />
  <cellular files value="cellular_output.xml" />
</input>
...
<output>
```

```

    <gps output value="gps_output.xml"/>
    <mobile output value="mobile_output.xml"/>
</output>
...
<positioning>
    <device.gps.enable all value="false"/>
</positioning>
...
<mobility>
    <device.mobility.enable all value="true"/>
</mobility>
...

```

Listing 2. Sumo configuration snippet about activating the GPS devices in the all vehicles

```

<gps output>
  <vehicle id="1">
    <gps device id="gps_1">
      <signal time="1.00">
        <coordinates latitude="58.29626476" longitude="26.44804043"/>
      </signal>
      <signal time="3.00">
        <coordinates latitude="58.29628805" longitude="26.44810251"/>
      </signal>
      <signal time="5.00">
        <coordinates latitude="58.29622327" longitude="26.44831755"/>
      </signal>
      ...
    </vehicle>
    ...
  </gps output>

```

Listing 3. GPS coordinates (longitude and latitude) with a timestamp

4.1.2 Cellular network simulation logs

The primary objective, *MSDevice_Mobile* class generated logs can be created when the configuration values have been set true for the mobile device and the mobile data

events have been defined for the same device in the *mes-file*. If the user wants positive results, then the cellular network should be defined. Otherwise there would be no connectivity for the devices and the events cannot occur.

The configuration of the mobile devices is similar to the GPS devices. One must define the existing mobile device into the route file, as shown in the Listing 1. Otherwise you can set all the vehicles to carry a mobile device and do as show in the Listing 2. Also, the user must define the output file in the config file.

In the Listing 4 can be seen a snippet about the cellular network behaviour simulation from the resulting XML output file. The log entity has been defined in the Figure 13 as the class *MobilityLog* and in the Chapter 3.5.6.

```
<mobile output>
  <vehicle id="1">
    <event timestamp="92.00" eventType="KEEP_ALIVE" imsi="1" cgi=
      "t 438 1" signalStregnth=" 21.01" latitude="58.29"
      longitude="26.46292491" speed="13.68143750"/>
    <event timestamp="94.00" eventType="SMS_REC" imsi="1" cgi="t
      438 1" signalStregnth=" 21.70533381" latitude="
      58.29089506" longitude="26.46336419" speed="12.64468285"/>
  </vehicle>
  <vehicle id="27">
    <event timestamp="27.00" eventType="CALL_INIT" imsi="27" cgi=
      "t 852 1" signalStregnth=" 21.11894073" latitude="
      58.25022813" longitude="26.43376626" speed="0.00000000"/>
    <event timestamp="28.00" eventType="CALL_INIT" imsi="27" cgi=
      "t 852 1" signalStregnth=" 21.21370266" latitude="
      58.25021502" longitude="26.43379995" speed="2.45816080"/>
    <event timestamp="29.00" eventType="CALL_INIT" imsi="27" cgi=
      "t 853 1" signalStregnth=" 20.32984836" latitude="
      58.25019017" longitude="26.43386379" speed="4.65720239"/>
    ...
  </vehicle>
  ...
</mobile output>
```

Listing 4. CDR-like mobility logs

4.1.3 MESSAGES and HexagonGen

To generate mobility event data for the SUMO cellular network behaviour simulation, then there is needed a configuration file. In the Listing 5, there are visible two input files: network file and route file. There has been defined output file, start of the simulation and the end, last parameter enables random number generator in the SUMO tool.

```
<?xml version="1.0" encoding="UTF 8"?>
<configuration>
<input>
  <net file value="nyghen.net.xml"/>
  <route files value="router5000.rou.xml"/>
</input>
<output>
  <output file value="mes_output5000.xml"/>
</output>
<time>
  <begin value="0"/>
  <end value="1000"/>
</time>
<random_number>
  <random>true</random>
</random_number>
</configuration>
```

Listing 5. MESSAGES configuration file example

To run the MESSAGES there is needed to run command line tool called *mesgenerate*. The result of the generation is visible in the Listing 6. Vehicle with it list of mobile events.

```
<vehicles>
  <vehicle id="0" depart="0.00">
    <mEvents>
      <mEvent begin="114.00" end="137.00" type="CALL_REC"/>
      <mEvent begin="363.00" end="450.00" type="WEB_COMM"/>
      <mEvent begin="490.00" end="516.00" type="CALL_INIT"/>
      <mEvent begin="627.00" end="669.00" type="CALL_INIT"/>
      <mEvent begin="731.00" end="742.00" type="CALL_REC"/>
      <mEvent begin="829.00" end="856.00" type="CALL_INIT"/>
    </mEvents>
  </vehicle>
</vehicles>
```

```
</mEvents>
</vehicle>
...
</vehicles>
```

Listing 6. The appearance of the MESGEN output file

To generate the cellular network file, there is needed to run HexagonGen' Python file, visible in the Listing 7. The arguments are latitude, longitude, length of the hexagon side and how big the cellular network grid should be with height and width.

```
hexagonGen/main.py 58.2806 26.5187 0.02 10 10
```

Listing 7. Command-line invoke of HexagonGen Python script

In the Listing 8, there is visible the result with one cell tower and three coverage cells. Tower contains of cell coverages and point of interest, as the location of the tower.

```
<?xml version="1.0" encoding="UTF 8" ?>
<cellularTowers>
  <cellularTower>
    <coverage id="t 0 0" shape="26.3697436305,58.3356
      26.3680115797,58.3346 26.3662795289,58.3356
      26.3662795289,58.3376 26.3680115797,58.3386
      26.3697436305,58.3376 26.3697436305,58.3356" color="
      137,137,137" geo="true" layer="0">
      <param key="signalStrength" value="62"></param>
      <param key="frequency" value="800.0"></param>
    </coverage>
    <coverage id="t 0 1" shape="26.3714756814,58.3326
      26.3697436305,58.3316 26.3680115797,58.3326
      26.3680115797,58.3346 26.3697436305,58.3356
      26.3714756814,58.3346 26.3714756814,58.3326" color="
      137,137,137" geo="true" layer="0">
      <param key="signalStrength" value="62"></param>
      <param key="frequency" value="800.0"></param>
    </coverage>
    <coverage id="t 0 2" shape="26.3680115797,58.3326
      26.3662795289,58.3316 26.3645474781,58.3326
      26.3645474781,58.3346 26.3662795289,58.3356
```

```

26.3680115797,58.3346 26.3680115797,58.3326" color="
137,137,137" geo="true" layer="0">
  <param key="signalStrength" value="62"></param>
  <param key="frequency" value="800.0"></param>
</coverage>
<poi id="t 0" lat="58.3346" lon="26.3680115797" type="
cellular_tower" color="1,1,1" layer="0" width="12" height=
"16" imgFile="tower.bmp"></poi>
</cellularTower>
...
</cellularTowers>

```

Listing 8. The appearance of the cellular file

4.2 Graphical upgrade

This thesis produced two new visual entities: cellular tower and cellular network cell. In the Figure 17. The user-interface has been enriched with buttons for the cellular network manipulation purposes. Those buttons are also visible in the Figure 17.

Drawing the towers depend on the SUMO class *PointOfInterest* and the hexagons are drawn through *Polygon* class.

4.3 The benchmarking

There was created a little benchmark to see how much influence the cellular network behaviour simulation has on the SUMO program. In the Figure 18, we can see that the mobility data simulation with 1200 towers, 5997 cells and 200 vehicles resulted with 4.36s simulation. Meanwhile, when we removed the cellular towers and the respective cells then the simulation time fell from 4.36s to 3.19s, as we can see on the Figure 19.

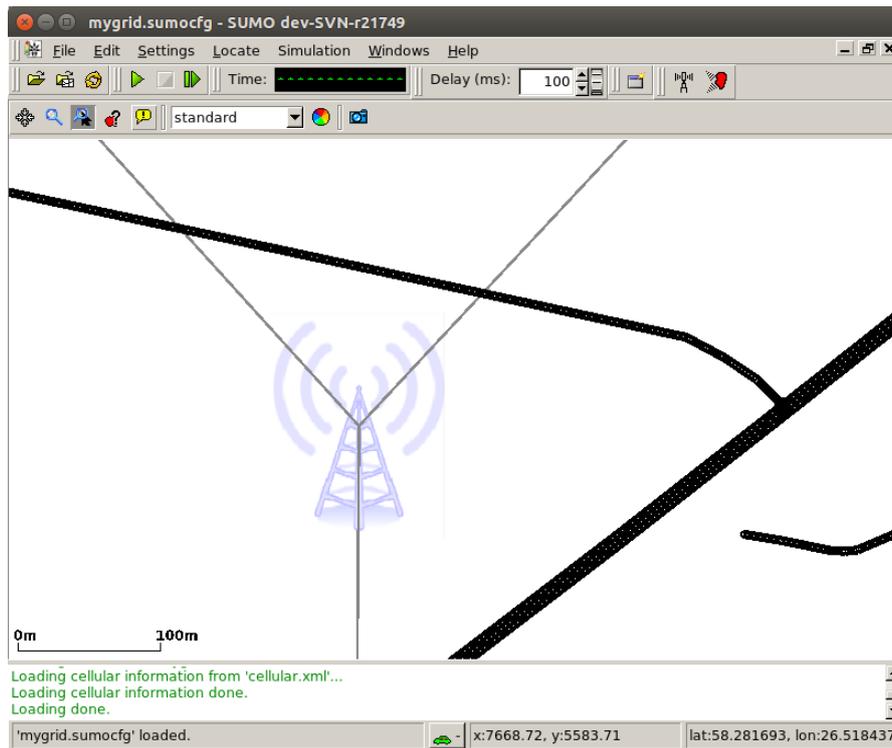


Figure 17. The SUMO simulation with the cellular tower with its cells.

5 Conclusion and Future Perspectives

This chapter concludes the thesis and presents the future perspectives in the research field, speculating about the possibilities and what extra work needs to be done, as well as the potential benefits from those upgrades.

5.1 Conclusion

The thesis aims at the possibility of simulating the mobile subscriber behaviour from the cellular network point of view. There are vast fields of research which depend on the mobile communication data. The work is often disturbed by the mobile operators' lack of interest and privacy of data constraints imposed by the law. The thesis investigates the SUMO traffic simulator and modern mobile technologies to try to blend these two components together to produce cellular network behaviour logs. The traffic simulator program offered the vehicular mobility and the cellular network behaviour was devel-

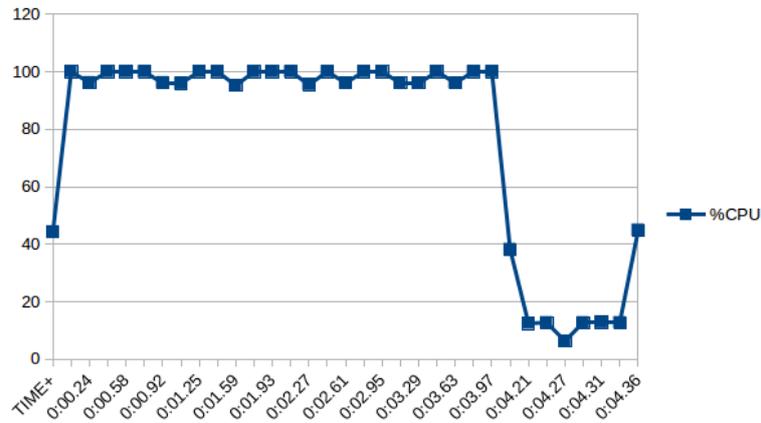


Figure 18. The CPU benchmark with the cellular network simulation.

oped during this thesis. The mobile technology and the concepts were derived from the 3GPP/ETSI specifications and industry recommended books.

The cellular network architecture and the models were produced from the documents and simplified for the implementing, reducing the scope and unwanted functionality which would hinder the development and shift the thesis from the main objectives. The mobile radio network was designed and developed, resulting in the mobile device, cellular network physical infrastructure, and their resulting network cells with wave signal propagation.

Finally, the graphical visualizations were made to understand the representation of the cellular network in the SUMO graphical version. Additionally, the generation of the mobility logs (mobile CDR-like and GPS) with storing was implemented.

5.2 Future work

The future perspective of this thesis work is to develop and visualize the wave propagation. There are many models that must be added into the created simulation such as: close electromagnetic properties to be emulated, for example, the path loss, reflection (diffraction, scattering), to have a proper real-life representation of the Received Signal Strength Indicator (RSSI) received at the mobile device in the simulation. The integration of wave propagation model, along with other cellular network behaviour models in the created simulation, will add more scientific value for writing a journal publication.

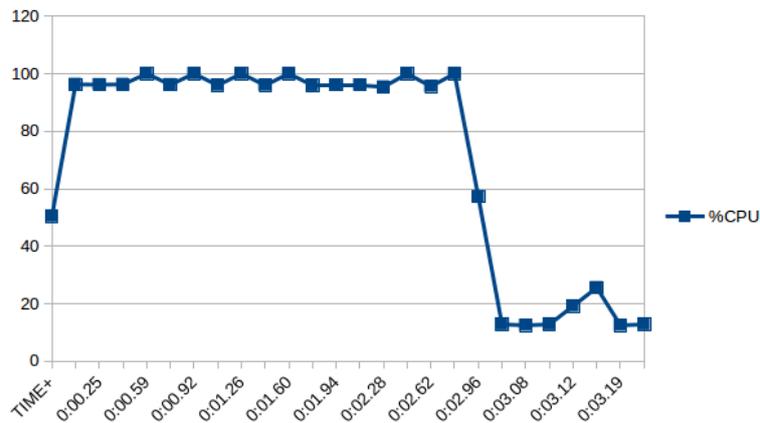


Figure 19. The CPU benchmark without the cellular network simulation.

One of the most important works which must be done is the integration of this thesis with other science projects. There is huge potential if we manage to integrate this thesis with projects where human behaviour models are used. This, would help simulate the vehicle movement and put the humans in vehicle communicate as they are in real life.

The human behaviour modelling in the communication could be one of the main tasks for the mobile event data generator, which currently relies on arbitrary events. Another accessory created during the development of cellular network simulation, *HexagonGen*, could be developed into a Bachelor thesis. During such a thesis, the objective would be to investigate and design a cellular network tool. Drawing different types of cellular networks with high functionality, e.g., angling hexagons and define sub-networks for more complicated scenarios. One of the main criteria would be the export of the data which should be compatible with SUMO and maybe with other tools in the market what could benefit from it.

On the SUMO level, there is a possibility to simulate the entire network's job. This requires analysis, design, and development of the fixed network - network switching subsystem. This could enable the base station level, MSC level, or even some other components log generation.

One of the bigger scale future works involves third-party developed tools. There are network technologies which simulate 2G, 3G, 4G networks. They could also be integrated with SUMO. In addition, their cellular network transmitter waves could be prop-

agated with open source gprMax [12], which specialises in the electromagnetic wave propagation. One of the network simulators is known as OsmocomBB [19], which is an open source GSM baseband software implementation. OpenBTS [14] is another open source project developing a new era of telecommunication protocols to substitute legacy proprietary code. This means that this project code could be fitted for the simulation software as well. An open source project also exists which is simulating 4G technology. OpenLTE [17] depends heavily on the 3GPP LTE specifications. Such projects enable researchers to take the cellular network simulation to next level, but the feasibility to simulate entire systems depends on the requirements and expected output.

Finally, there are many possible opportunities that can be integrated to the created system and how it could serve the scientific community.

References

- [1] 3GPP. <http://www.3gpp.org/>. Accessed: 09.05.2017.
- [2] DLR - DLR Portal. <http://www.dlr.de/dlr//en/desktopdefault.aspx/tabid-10002/>. Accessed: 10.05.2017.
- [3] DLR - Institute of Transportation Systems - SUMO User Conference 2017 - "Towards Simulation for Autonomous Mobility" (May 8-10, 2017). http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-3930/20125_read-47741/. Accessed: 10.05.2017.
- [4] DLR - Institute of Transportation Systems - SUMO – Simulation of Urban MObility. http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/. Accessed: 09.05.2017.
- [5] Eclipse CDT - Eclipse CDT (C/C++ Development Tooling). <http://www.eclipse.org/cdt/>. Accessed: 15.05.2017.
- [6] Extensible Markup Language (XML). <https://www.w3.org/XML/>. Accessed: 15.05.2017.
- [7] FCC ADOPTS RULES TO IMPLEMENT ENHANCED 911 FOR WIRELESS SERVICES. <https://www.fcc.gov/document/fcc-adopts-rules-implement-enhanced-911-wireless-services>. Accessed: 08.05.2017.
- [8] fox-toolkit.org. <http://www.fox-toolkit.org/>. Accessed: 11.05.2017.
- [9] GCC 4.8 Release Series - GNU Project - Free Software Foundation (FSF). <https://gcc.gnu.org/gcc-4.8/>. Accessed: 15.05.2017.
- [10] GDAL:GDAL - Geospatial Data Abstraction Library. <http://www.gdal.org/>. Accessed: 11.05.2017.
- [11] Git. <https://git-scm.com/>. Accessed: 15.05.2017.
- [12] gprMax, GitHub. <https://github.com/gprMax>. Accessed: 16.05.2017.

- [13] MATSim | Multi-Agent Transport Simulation. <http://www.matsim.org/>. Accessed: 11.05.2017.
- [14] OpenBTS | Open Source Cellular Infrastructure. <http://openbts.org/>. Accessed: 16.05.2017.
- [15] OpenDRIVE - Home. <http://www.opendrive.org/>. Accessed: 11.05.2017.
- [16] OpenGL - The Industry Standard for High Performance Graphics. <https://www.opengl.org/>. Accessed: 11.05.2017.
- [17] openLTE download | SourceForge.net. <https://sourceforge.net/projects/openlte/>. Accessed: 16.05.2017.
- [18] OpenStreetMap. <https://www.openstreetmap.org/>. Accessed: 11.05.2017.
- [19] Overview - OsmocomBB - Open Source Mobile Communications. <http://osmocom.org/projects/baseband>. Accessed: 16.05.2017.
- [20] proj.4 — proj.4 4.9.3 documentation. <http://proj4.org/>. Accessed: 11.05.2017.
- [21] PTV Vissim. <http://www.ptvgroup.com/en/solutions/products/ptv-vissim/>. Accessed: 11.05.2017.
- [22] PTV Visum. <http://vision-traffic.ptvgroup.com/en-us/products/ptv-visum/>. Accessed: 11.05.2017.
- [23] PyCharm. <https://www.jetbrains.com/pycharm/>. Accessed: 15.05.2017.
- [24] Python 2.7.0 Release | Python.org). <https://www.python.org/download/releases/2.7/>. Accessed: 15.05.2017.
- [25] Simulation of Urban MObility - Wiki. <http://sumo.dlr.de/wiki>. Accessed: 09.05.2017.

- [26] Simulation/Meso - Sumo. <http://sumo.dlr.de/wiki/Simulation/Meso>. Accessed: 15.05.2017.
- [27] SUMO User Documentation. http://sumo.dlr.de/wiki/SUMO_User_Documentation. Accessed: 09.05.2017.
- [28] TextTest.org. <http://textttest.sourceforge.net/>. Accessed: 11.05.2017.
- [29] The Apache Xerces™ Project - [xerces.apache.org](http://xerces.apache.org/xerces-c/). <https://xerces.apache.org/xerces-c/>. Accessed: 11.05.2017.
- [30] The GNU General Public License v3.0 - GNU Project - Free Software Foundation. <https://www.gnu.org/licenses/gpl.html>. Accessed: 10.05.2017.
- [31] W3C XML Schema. <https://www.w3.org/XML/Schema>. Accessed: 15.05.2017.
- [32] Esteban Zimanyi Chiara Renso, Stefano Spaccapietra. *Mobility Data: Modeling, Management and Understanding*. Cambridge, 2013.
- [33] Craig Scott Christopher Drane, Malcolm Macnaughtan. Positioning gsm telephones. *IEEE Communications Magazine*, 36(4):46–59, April 1998.
- [34] M. Decina and E. Scace. Ccitt recommendations on the isdn: A review. *IEEE Journal on Selected Areas in Communications*, 4(3):320–325, May 1986.
- [35] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.02: Digital cellular telecommunications system (Phase 2+); Network architecture*, 5.1.0 edition, May 1996.
- [36] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.04: Digital cellular telecommunications system; Signalling requirements relating to routeing of calls to mobile subscribers*, 5.0.0 edition, November 1996.

- [37] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.12: Digital cellular telecommunications system; Location registration procedures*, 5.0.0 edition, November 1996.
- [38] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 08.02: Digital cellular telecommunications system; Base Station System - Mobile-services Switching Centre (BSS - MSC) interface; Interface principles*, 5.0.0 edition, December 1996.
- [39] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.08: Digital cellular telecommunications system (Phase 2+); Organization of subscriber data*, 5.1.0 edition, April 1997.
- [40] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.09: Digital cellular telecommunications system (Phase 2+); Handover procedures*, 5.1.0 edition, August 1997.
- [41] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GPRS 01.60: Digital cellular telecommunications system (Phase 2+); General Packet Radio Service (GPRS) Requirements specification of GPRS*, 6.0.0 edition, April 1998.
- [42] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *GSM 03.20: 3rd Generation Partnership Project; Digital cellular telecommunications system (Phase 2+); Security related network functions (Release 1999)*, 8.1.0 edition, October 2000.
- [43] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 25.401: Universal Mobile Telecommunications System (UMTS); UTRAN Overall Description*, 4.2.0 edition, September 2001.
- [44] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 04.04:3rd Generation Partner-*

ship Project; Technical Specification Group GSM/EDGE Radio Access Network; Layer 1; General requirements, 8.1.2 edition, May 2002.

- [45] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 25.304: Universal Mobile Telecommunications System (UMTS); User Equipment (UE) procedures in idle mode and procedures for cell reselection in connected mode*, 8.7.0 edition, September 2009.
- [46] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 43.051: Digital cellular telecommunications system (Phase 2+); GSM/EDGE Radio Access Network (GERAN) overall description; Stage 2*, 8.0.0 edition, January 2009.
- [47] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 44.069: Digital cellular telecommunications system (Phase 2+); Broadcast Call Control (BCC) protocol (3GPP TS 44.069 version 9.0.0 Release 9)*, 9.0.0 edition, January 2010.
- [48] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 23.009: Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Handover procedures*, 10.0.0 edition, April 2011.
- [49] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 24.007: Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Mobile radio interface signalling layer 3; General Aspects*, 10.0.0 edition, March 2011.
- [50] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 44.018: Digital cellular telecommunications system (Phase 2+); Mobile radio interface layer 3 specification; Radio Resource Control (RRC) protocol*, 10.3.0 edition, June 2011.
- [51] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3rd Generation Partnership Project;*

Technical Specification Group GSM/EDGE Radio Access Network; Mobile Station - Base Station System (MS - BSS) interface; Data Link (DL) layer specification, 8.4.0 edition, January 2012.

- [52] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 23.002: Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Network architecture*, 12.5.0 edition, October 2014.
- [53] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 44.018: Digital cellular telecommunications system (Phase 2+); Mobile radio interface layer 3 specification; Radio Resource Control (RRC) protocol; Iu mode*, 13.0.0 edition, January 2016.
- [54] European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE. *3GPP 24.008: Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Mobile radio interface Layer 3 specification; Core network protocols; Stage 3*, 13.8.0 edition, January 2017.
- [55] Flarion. OFDM for Mobile Data Communications. <https://www.scribd.com/document/7352160/OFDM-Mobile-Data-Communications>, March 2003.
- [56] Ki Won Sung Slimane Ben Slimane Guowang Miao, Jens Zander. *Fundamentals of Mobile Data Networks*. Cambridge, March 2016.
- [57] Antonin Guttman. R-trees: a dynamic index structure for spatial searching. *SIGMOD '84 Proceedings of the 1984 ACM SIGMOD international conference on Management of data*, pages 47–57, 1984.
- [58] Ray Horak. *Telecommunications and Data Communications Handbook, 2nd Edition*. Wiley, August 2008.
- [59] Jean Bolot Hui Zang, Francois Baccelli. Bayesian inference for localization in cellular networks. *IEEE INFOCOM'10 Proceedings of the 29th conference on Information communications*, pages 1963–1971, 2010.

- [60] Simone Frattasi João Figueiras. *Mobile Positioning and Tracking: From Conventional to Cooperative Techniques*. Wiley, June 2010.
- [61] Alexander Agathos Kai Hormann. The point in polygon problem for arbitrary polygons. *Computational Geometry*, 20(3):131–144, November 2001.
- [62] Daniel Krajzewicz. *Traffic Simulation with SUMO – Simulation of Urban Mobility*, pages 269–293. Springer New York, New York, NY, 2010.
- [63] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012.
- [64] Stefan Krauß. *Microscopic Modelling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics*. PhD thesis, Deutsches Zentrum für Luft- und Raumfahrt e.V. (Hauptabteilung Mobilität und Systemtechnik), 1998.
- [65] Bell Laboratories. High-capacity mobile telephone system. *Technical Report*, December 1971.
- [66] W. C. Y. Lee. Elements of cellular mobile radio systems. *IEEE Transactions on Vehicular Technology*, 35(2):48–56, May 1986.
- [67] V. H. MacDonald. The cellular concept. *Bell System Technical Journal*, 1(58):15–42, January 1979.
- [68] Jakob Erdmann Daniel Krajzewicz Michael Behrisch, Laura Bieker.
- [69] Yannis Theodoridis Niklos Pelekis. *Mobility Data Management and Exploration*. Springer, 2014.
- [70] Jeffrey H. Reed Nishith Tripathi. *Cellular Communications: A Comprehensive and Practical Guide*. Wiley, October 2014. 1032 pages.
- [71] Christoph Sommer, Zheng Yao, Reinhard German, and Falko Dressler. Simulating the Influence of IVC on Road Traffic using Bidirectionally Coupled Simulators. In *27th IEEE Conference on Computer Communications (INFOCOM 2008)*: IEEE

Workshop on Mobile Networking for Vehicular Environments (MOVE 2008), pages 1–6, Phoenix, AZ, April 2008. IEEE.

- [72] V. Pelishok T. Maksymyuk, M. Brych. Stochastic geometry models for 5g heterogeneous mobile networks. *Smart Computing Review*, 5(2):89–101, April 2015.
- [73] 5G Initiative Team. *5G White Paper*. https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf, February 2015.
- [74] Chai K. Toh. *Ad Hoc Wireless Networks: Protocols and Systems*. Prentice Hall PTR, 2001.
- [75] F. Calabrese F. Morlot V.A. Traag, A. Browet. Social event detection in massive mobile phone data using probabilistic location inference. 2011.

Appendix

I. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Siim-Toomas Marran**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:
 - 1.1 reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and
 - 1.2 make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

of my thesis

Simulating Cellular Network Behaviour in the Context of Urban Environment

supervised by Amnir Hadachi and Artjom Lind.

2. I am aware of the fact that the author retains these rights.
3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 18.05.2017