POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Automotive Engineering

Tesi di Laurea Magistrale

Development and implementation of a fleet simulation in SUMO

Relatori
prof.ssa Daniela Misul
prof. Giovanni Belingardi

Candidato
Stefano Niero

firma dei relatori
....

firma del candidato
....

A.A. 2018

Per fare tutto ci vuole un fiore

Gianni Rodari

Acknowledgements

I would like to thank all the people that helped me in the realization of this project.

My family made the experience possible and supported me in my choices sharing their opinions, helping me in the solving of doubts and stimulating ideas. Special emotional support has been provided by them, by Alice and my closest friends.

A special thank to my tutor Dipl.-Ing Manfred Klöppel that followed my work carefully and guided me in finding problem solutions giving me precious advices and sharing his research experience. His tutoring left me the autonomy to make choices and sometimes mistakes but never let me down, showing an availability that I really appreciate.

Garching, 26 September 2016

Stefano Niero

Table of contents

Abbreviation Directory

AGV	Automated Guided Vehicle
API	Application Programming Interface
AV	Autonomous Vehicle
CO2	Carbon Dioxide
CSV	Comma Separated Values
DLR	Deutsches Zentrum für Luft- und Raumfahrt e.V. – German Aerospace Institute
EU	European Union
EV	Electric Vehicle
GUI	Guided User Interface
IVC	Intra Vehicle Communication
LuST	Luxemburg SUMO Traffic
MAVEN	Managing Automated Vehicles Enhances Network
MMA	Munich Metropolitan Area
MOD	Mobility On Demand
O-D	Origin-Destination
OSM	Open Street Map
PM2.5	Particulate Matter (radius smaller than 2.5 µm)
SAV	Shared Automated Vehicle
SUMO	Simulator of Urban MObility
TAZ	Traffic Assignment Zone
TCP	Transmission Control Protocol
TraCI	Traffic Control Interface
UK	United Kingdom
UN	United Nations
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VGI	Volunteered Geographical Information
VISSIM	Verkehr In Städten, SIMulationsmodell – Traffic in cities, simulation model

VM	Virtual Machine
XML	Extensible Markup Language

1 Introduction

1.1 Current situation and problems

Cities are rapidly growing all over the world and along with them, individual mobility demand. According to UN studies, 55% of the world's population lives today in urban areas and this percentage is expected to grow up to 68% by 2050 with an overall adding to world population of 2.5 billion people. It is also expected to see the presence of 43 megacities with more than 10 million inhabitants by 2030, an increase of 10 with respect of today [1].

A major concern, particularly in developed and expanding economies, is constituted by traffic congestion, widely recognised as a significant contributor to air pollution and global warming. In fact it is estimated that the transport sector in the European Union, is the cause of more than a quarter of the total greenhouse gas emissions and it has been registered in 2014, that a contribution of more than 73% to these emissions came from road transport. In view of these figures, it is evident how acting on transports constitutes one of the keys for the compliance with environmental targets [2, p. 29].

Traffic congestion, it is not only an alarming environmental problem, but also an economical one: it is a major cause of substantial delays in urban areas negatively affecting journey times and the convenience of road travel. Some estimates for example, have calculated a negative impact of congestion to the UK economy as big as £13-£20 billion per year [3, pp. 73-74].

Solutions to this problem are not likely to come from an increase of road capacity: efforts to do so on a large enough scale, often result economically unaffordable and environmentally unsustainable. Moreover, the building of infrastructures often implies too long-term projects in contrast with the needs of the most rapidly growing cities and mega cities for an immediate solution. Therefore, the need for an affordable, green, socially inclusive and rapidly implementable mobility solution, is becoming ever more strong and urgent.

Solutions must then be focused on increasing the efficiency of existing transportation infrastructures, as they require a lower intervention and can be implemented in a shorter term.

Autonomous Vehicles (AVs) and on-demand mobility can be one of these solutions as such a technology promises to strongly reduce traffic, accidents and polluting emissions [4]. This will also give the possibility and will require the necessity to rethink mobility: from a privately owned car system to a car sharing one.

An ever increasing number of manufacturers are aiming their interest towards Autonomous Vehicles and an ever increasing number of countries are starting legislative processes in order to take into account their presence in the streets. Recent openings even in Europe involve their testing: as an example the city of Turin announced the legalization and regulation of AV tests starting from September 2018 [5]. It is becoming evident that younger generations such as millennials are less interested in the ownership of a car, showing a lower rate of ownership with respect to previous generations at their age [6]. Thus, there is more openness towards car-sharing and even peer-to-peer car sharing concepts. In addition, a private car is mostly used only few hours a day, standing parked for the rest of the time. This results in an important amount of space in cities that is required for parking. On-demand mobility services, although without the use of AVs, have already started to spread: they consist in Car and Bike sharing.

Despite all these measures, the implementation of AVs on a large scale is likely to need several years if not decades in order to take place: the arrival of AVs in cities is expected to begin in 2020s or 2030s but the real traffic flow improvement effects will occur later, possibly from 2040s to 2060s [7, pp. 81-82].

Besides all the aforementioned benefits that AVs would bring, if these are considered singularly they share a non-negligible limitation with human beings: the sensors they rely on, have a limited field of view. However, they can overcome this limitation through the sharing of information, exploiting networks and communication systems [8, p. 51]. This has become possible thanks to technological improvements related to Intelligent Transportation Systems (ITS): Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communication. Such technologies are gaining more and more attention, particularly for the improvements offered in traffic network management and accidents reduction.

To successfully implement new traffic solutions, traffic simulation is an essential step since it offers a controlled environment in which it is possible to perform tests without any need of modification of the real infrastructure thus cutting dramatically necessary time and costs.

The higher is the level of detail, the more accurate is the scenario and therefore the more realistic are the simulations of traffic management strategies to be tested. Microscopic traffic models offer such a level of detail by simulating each vehicle unit: using car following models for longitudinal behavior and lane change models for lateral one.

Many software packages exist for such a kind of traffic simulation, among them, there is SUMO (Simulation of Urban Mobility), an open source simulator developed mostly by DLR (Deutsches Zentrum für Luft- und Raumfahrt e.V.) [9]. Being open source, this simulator has the big advantage of being supported and continuously updated by a large community. Furthermore, the software is characterized by a high integration possibility with external platforms through a series of interface tools, the most important of which being TraCI (Traffic Control Interface) [10, p. 71]. This consists in an interface that allows the use of a series of python commands that interact on line with the simulation.

1.2 Objectives of the study

The objective of the study is to explore the possibilities and the way of working of SUMO software. The program is used to reproduce the taxi traffic in the city of Munich through an agent based simulation. Input data have been provided by Isarfunk, these are related to approximately 400 connected vehicles (11.7% of all 3400 taxis in Munich) [11, p. 12].

After the building of the scenario for an average weekday, a first study phase consists in the validation of the simulation, comparing the results of distances covered and times needed for the service of the customers using a fleet of 428 taxis.

Consequently the fleet size is tuned to investigate how its ability to serve the customers changes and to find an optimum size based on the evaluation of some KPIs, namely waiting times and utilizations of the vehicles. Effects of distribution of the vehicles in the network and its impact on aforementioned KPIs are also investigated.

Finally, the verification of the assumptions made on a weekday is performed on a weekend day, characterized by a different demand pattern and by higher demand peaks.

A cost model has then been developed in order to compare the solutions selected and provide a brief analysis also from an economical point of view.

1.3 Outline of the structure

After this first chapter that presents motivations and objectives of the thesis work, Chapter 2 provides a brief introduction to agent base simulation and the description of the software package used for the simulations. Subsequently the state of art of mobility researches is described, focusing on scenario generation and studies based on the use of agent based simulation softwares. At the end of the chapter, a summary of real mobility service KPIs is illustrated, to provide comparisons with the results obtained from the simulation.

Chapter 3 describes the approach to the work and the building of the simulation explaining the reasons of the choices made, the input data and all the procedure to build the scenario. A section is also dedicated to the analysis of the program and its way of working. Different script alternatives are also briefly described and compared with the final choice. Strategies to retrieve useful data are illustrated as well as tools to perform data analysis once the simulation has been performed.

Results based on data collected by the simulation and subsequently elaborated, are shown in Chapter 4. This section initially deals with the simulation validation, to continue with the illustration of results of the simulations performed on an average weekday with different fleet sizes and distribution. Finally results of the simulation performed on a more demanding weekend day are illustrated to validate assumptions previously made.

Chapter 5 discusses and comments the results obtained, showing advantages and disadvantages of the different fleet sizes and advancing and motivating a fleet selection hypothesis. A cost model is also developed for the purpose providing an economical point of view in addition to the operational one.

Chapter 6 draws final conclusions of the work and advances hypothesis about possible next steps and research opportunities that could be undertaken as a prosecution of the study.

A series of appendix chapters integrate Chapter 4 with additional data.

Figure 1.1 provides a graphical representation of the thesis structure.

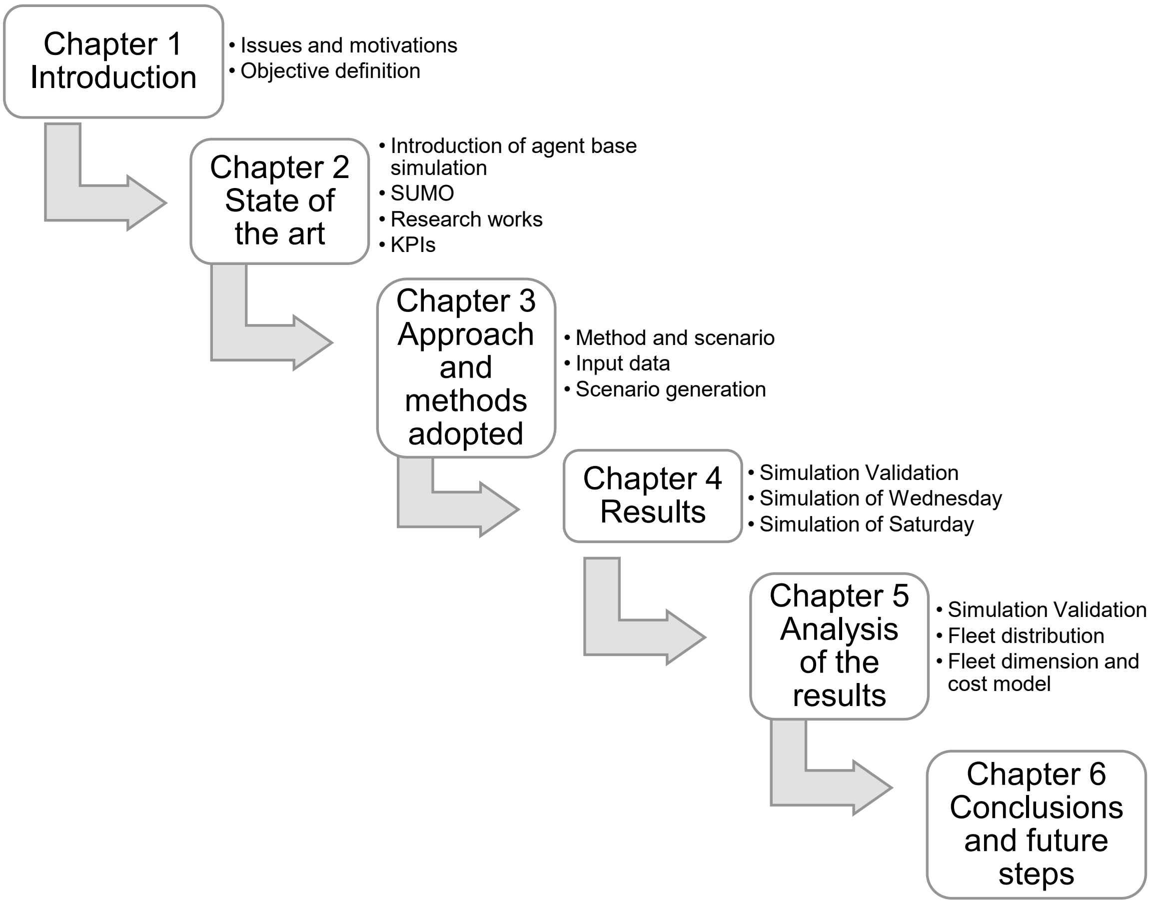

Figure 1.1 Graphical representation of the structure of the Thesis

2 State of the Art

2.1 About traffic simulations

The use of simulation methodologies in the field of System of Transportation has been a common practice for decades. It is true that the transportation in urban scenarios is requiring an ever increasing importance of simulations due to high traffic saturation, an increasing demand and an evident unoptimized planning [12, p. 197].

Traffic simulation tools are meant to generate scenarios, optimize control, predict network behavior as well as cope with exceptional events that can attract a lot of people such as the World Cup or the Oktoberfest in Munich [13]. A virtual scenario offers the possibility to easily modify the network topology or the traffic control strategies, so to quickly validate new models with a minimum expense and saving interventions on the physical structure before the final solution is found.

Simulators are constantly evolving and are able to measure with an ever increasing precision an ever increasing number of data such as emissions or state of charge of the battery for EVs. Moreover the possibility of more complex scenario such as inter-vehicular communications (IVC) and multi-modal simulations has been opened.

It is possible to classify traffic models according to different criteria [14, p. 5]:

- Scale of the independent variables: time scale must be taken into account since the system described is usually dynamic. A discrete model represents changes in a discontinuous way, after regular intervals or after some events. Instead, a continuous model describes traffic changes continuously over time. Time is not the only dimension that can be either discrete or continuous: examples are given by speed or position.
- Representation of the processes: the discriminant in this kind of classification is whether these processes are deterministic or stochastic. In the former case, if two simulations are run with the same input parameters, the output will be exactly the same, meaning that the simulation is fully predictable. On the other hand, simulations based on stochastic models are unpredictable due to the presence of some random factors.
- Scale of application: The way the dynamics of the single entities such as networks, links and intersections and are represented
- Level of detail: whether the simulation is macroscopic, microscopic, mesoscopic or nanoscopic

It is worth to focus on this last point. Table 2.1 summarizes the characteristics of the different levels of detail of a simulation.

Macroscopic simulators are based on mathematical models that describe the flows of vehicles through the network with an approach similar to the fluid dynamics. This is usually applied to large scenarios, where a high level of detail is not required and would make the simulation dramatically heavier.

The microscopic simulation consists in a set of models, each representing an individual vehicle whose behavior needs to be calibrated in order to follow the macroscopic flow patterns.

In between the microscopic and the macroscopic simulations, is the mesoscopic one. The aim of such a level of detail is to get the advantages of both the macro (lightness) and the microsimulations (detail). A way to do so is to describe the interactions between entities in probabilistic terms. Approaches can be different:

- vehicles grouped in packets routed through the network: characterized by a speed on each link (road) derived from a speed density function and the density on the link at the moment of entry;
- individual vehicles grouped into cells that manage their behavior: a vehicle can enter or leave the cell when needed but cannot overtake, speed is determined by the cell and not the driver [15, pp. 17-19].

As a fourth kind of model, there is the nanoscopic or submicroscopic one. This describes not only the time-space behavior of the vehicles in the system but also the way of working of some specific parts and processes. To do so, it is important to model in detail the driving and the vehicle control behavior (e.g. inertia, tire deformation).

Table 2.1 Characteristics of the different levels of detail of traffic simulations

Detail	Nanoscopic	Microscopic	Mesoscopic	Macroscopic
Main feature	Describes the way of working of specific vehicles component	Each vehicle is represented individually	In between level: different strategies to group vehicles and handle them	Describe flows of vehicles with an approach similar to fluid dynamics
Advantages	Possibility to perform studies on vehicle parts	High level of detail	Good level of detail reducing sensibly simulation effort compared to microscopic case	Possibility of application on large scale scenarios reducing simulation effort
Disadvantages	Dramatically high simulation effort. Adopted only for very small scenarios focused on one or few vehicles	Heavy simulation, not suited for large scale scenarios	In the case a high level of detail is required, this cannot substitute microsimulation	Low level of detail

Microscopic simulators are used for the analysis of complex systems. In order to foster low demanding simulations, complex systems are divided in several entities interacting with each other and the surrounding environment. These kinds of system are also known as multi-agent systems. Since general approach is to treat driver and vehicle as a single unit, there are several rules also known as behavioral models, that support the specific interactions. Most important behavioral models are the one that regulates longitudinal and lateral interactions. The former, more important, is called car-following model while the latter is called lane-changing model. Other sub-models can be gap-acceptance, overtaking, ramp metering or speed adaptation [11, pp. 6-7]. Figure 2.1 schematizes the way of working of an agent base simulation.

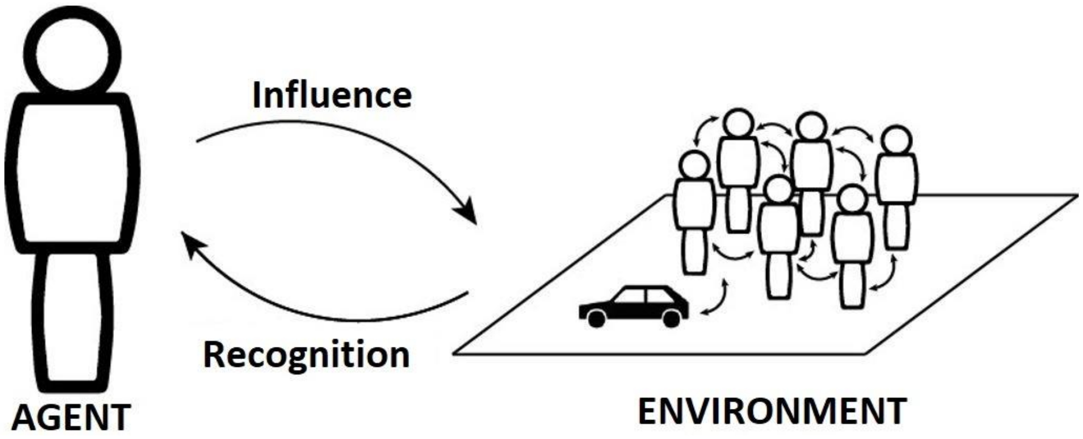

Figure 2.1 Agent base principle according to Wittmann [11]

Thanks to the possibility to handle complex interacting systems, agent-based simulations are used in different fields of application such as Bioinformatics, Economics or Social Sciences.

2.2 SUMO

SUMO stands for "Simulator of Urban Mobility" and it is a program used for multi-modal traffic simulations. The main features of this program are that it is open source and microscopic, thus allowing to simulate how each single vehicle moves through a given road network.

The development of SUMO platform started in year 2000 by DLR (Deutsches Zentrum für Luft- und Raumfahrt e.V.) with the aim of supporting the traffic research community providing a tool able to implement and evaluate own algorithms. In order to gain additional help from other contributors, the program has been made open source and expandable by the user community, although it is developed mostly by DLR itself. Another main need that the creation of SUMO wanted to provide for, was the implementation of more comparable algorithms by using a common architecture and model base [16]. Thus, SUMO is supported by a very detailed and constantly updated online user documentation that provides tools and content descriptions, FAQ, installation instructions, tutorials, publications and as well as a developer documentation [17].

As a result, over the years the simulator have reached the 33rd release and has seen its community expand with important contribution by universities all over the world such as UCLA, Carnegie Mellon University, TUM, HU Berlin, Friedrich-Alexander University Erlangen-Nürnberg, Uni Lübeck, IIT Bombay, Universität Innsbruck and Politecnico di Torino [16]. Thanks to these improvements the simulator can be considered a valid alternative to other available commercial solutions.

The major goals achieved by the simulator are two: speed and portability. In order to approach such goals, the first versions of the program have been developed to be run only from the command line. Successively, a GUI has been implemented and developed too but the possibility to run the simulation without it is still available.

Another consequence of these goals is that the software is split into several parts, each of them with a certain purpose and to be run individually. This solution facilitates the extension of the applications and allows the usage of faster data structures each optimized for the purpose. Conversely, other packages are much more integrated, making for example the dynamical user assignment within the simulation itself while in SUMO a dedicated application is present

(DUAROUTER). This can make the usage of SUMO a little bit more uncomfortable but allows to obtain a package significantly smaller than a monolithic integrated application [16].

The results of the strategies adopted are the possibility to perform space-continuous, time-discrete simulations managing networks with hundreds of thousands edges and different vehicle types at a fast execution speed (up to 100,000 vehicle updates on a 1GHz machine). Outputs produced can be either network-wide, edge-based, vehicle-based or detector-based [16]. SUMO's simulations are deterministic by default although there are various options that allow randomness introduction.

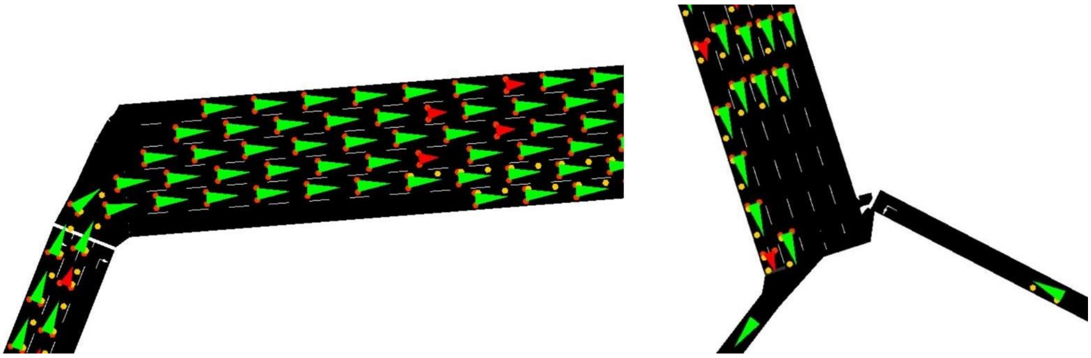

Figure 2.2 Example of conservative behavior of vehicles

A possible criticality to be taken into account while using SUMO in particularly complex and crowded scenarios is that the simulator is collision-free. As a consequence, vehicles require a large gap between them in order to decide to move forward, making their behavior more conservative than reality [18, p. 32]. Still it is possible to act on this gap adopting particular solutions in order to reduce it or even cause accidents [19]. An example of this conservative behavior is represented in Figure 2.2 extracted from the modification of a tutorial present in the user guide: here it is possible to notice two situations in which the cars do not fill all the spaces available; since they are trying to reach a precise lane they may tend not to move forward. Such a behavior is very unlikely to happen in reality.

2.2.1 Important projects that relied on SUMO

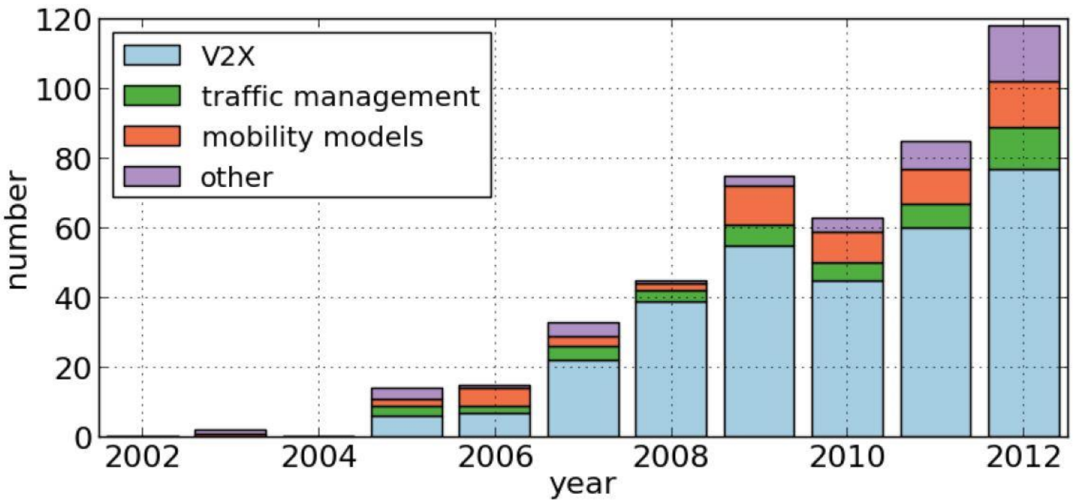

Figure 2.3 Development of the publications' major topics [20]

Opportunities offered by this package are several. The easy access to almost all of the parameters during runtime, opens the possibility for different research projects. Many project applications of SUMO involved V2X (V2V and V2I) communications, AV simulations and traffic testing techniques such as platooning [8]. In Figure 2.3 it is possible to see the growth of the number of researches based on SUMO from 2002 to 2012. From the picture it is possible to notice that the most studied topic with the use of SUMO is V2X communications systems simulation with 70% of the works [20, p. 18].

One of the most important projects in this field is the COLOMBO project, funded by European Union. The project works on traffic surveillance algorithms for a cooperative Self-Organizing System for low Carbon Mobility at low Penetration Rates exploiting V2I and V2V communications integrating it with the ns-3 communication simulator [21, p. 35].

In the logic of Horizon 2020, EU funded also MAVEN project, acronym for Managing Automated Vehicles Enhances Network. This SUMO based project aims at providing solutions for the management of automated vehicles in urban environments by developing algorithms for the organization of the flow in the infrastructures, for example using Platooning. As an outcome of the project, a prototype system will be built and used for tests and a roadmap for the introduction of road transport automation will be developed in order to support road authorities in understanding potential future changes in their role also providing guidelines that will help the smart cities implementation [22].

Another important large scenario project that implies the use of SUMO is ITS Austria West. It consists in a long-term Austrian project in which real-time traffic data is continuously generated and published. The traffic monitoring application exploits data coming in real time from sensors, SUMO is used to generated and maintain the road network and simulate the traffic in order to estimate values for the roads not covered by the sensors [23, p. 157].

SUMO is also starting to be used to investigate emission scenarios (CO2 and PM2.5) although this is more at an embryonal stage [24]. The simulator is also regularly used for the forecast and management of anomalous traffic conditions, particularly in the case of big events expected to attract a lot of people. Example of this usage are yearly simulation of the Oktoberfest Scenario in Munich, the Pope's visit to the city of Cologne in 2005 or the Soccer World Cup of 2006 [13]. From this perspective, it is worth to mention also project VABENE++ that exploits SUMO to simulate and predict large-scale traffic for disaster and public event management. The objective is to efficiently manage the required rescue logistics and the nearby traffic flow even under extreme conditions so to enable a quick response of teams that can rapidly reach the places where they are most need [25].

In order to support the community and foster the spread of results of researches, DLR organizes yearly conferences in which the participants expose their research projects to the community. From these conferences a journal collecting all the papers of the participants is then published. This way, the conference proceedings are able to offer an overview of the applicability of the SUMO tool suite as well as its universal extensibility thanks to the availability of the source code.

2.2.2 Contents of SUMO package

SUMO, as already mentioned consists in a package of all the different applications needed to prepare and perform a traffic simulation. Many of these applications have not been used for the scope of the work so will not be mentioned any further. Table 2.2 summarizes the functions of each the these application [17].

Table 2.2 Contents of SUMO package

Tool's name	Description
SUMO	microscopic simulator with no visualization, to be run directly by the command line
SUMO-GUI	microscopic simulator equipped with a Graphical User Interface
NETCONVERT	Network importer and generator. It reads and convert road networks from different formats into the SUMO-format
NETEDIT	graphical network editor
NETGENERATE	generator of abstract networks for the SUMO-simulation
DUAROUTER	computes the fastest routes through the network importing different kinds of demand description. The name comes from Dynamic User Assignment.
JTRROUTER	computes routes using junction turning percentages
DFROUTER	computes routes from induction loop measurements
MAROUTER	performs macroscopic assignment (used for mesoscopic simulations)
OD2TRIPS	decomposes O-D matrices into single vehicle trips
POLYCONVERT	imports points of interest and polygons from different formats and translates them into a description that may be visualized by SUMO-GUI
ACTIVITYGEN	generates a demand based on mobility wishes of a modelled population. Receives as input detailed statistical data concerning the population, the points of interest in the map, working hours, school hours, leisure time and other statistics related to activities in a given network.
EMISSIONSMAP	generates an emission map
EMISSIONSDRIVINGCYCLE	calculates emission values based on a given driving cycle

2.3 TraCI

TraCI stands for "Traffic Control Interface". It is an interface that consists in a series of commands that enable the user to have access to the running traffic simulation. It is therefore possible to retrieve values of simulated objects and manipulate their behavior online [26]. TraCI can be used either as an interface for other programs or by means of python scripts that interact with the simulation for a given scope.

The way the interface works is through the usage of a TCP based client/server architecture, this provides access to SUMO. This way, SUMO acts as a server started with additional command line options. The --end <TIME> option commonly used for SUMO simulations is ignored in the case it runs as a TraCI server and it is up to the client to decide when the simulation ends [26].

As already mentioned, TraCI works through a series of commands split into 14 different domains corresponding to the individual modules: gui, poi, simulation, lane, edge, route, traffic light, junction, induction loop, multi entry-exit, polygon, person, vehicle and vehicle type [27]. Largest use of TraCI is through Python language which library is tested daily and supports all TraCI commands; nevertheless, alternatives exist [26]:

- TraaS (TraCI as Service): web-service adapter for TraCI which allows automatic API generation for multiple languages. API is complete but usually lags behind the python client.
- TraCI4Matlab: supports Matlab platform, not all TraCI commands have been implemented
- TraCIAPI: client C++ library that is part of the SUMO-source tree.
- TraCI4J: outdated Java implementation of TraCI; It only supports a small portion of the API.
- TraCI.NET: client library with partial API coverage.

The possibilities offered by TraCI commands are different:

- Control-related commands: perform a simulation step, close the connection, reload the simulation.
- Value Retrieval: allow the retrieval of simulation data such as information about:
- State Changing: allow the change of state of the different objects, in particular of

Table 2.3 provides information about which are the elements of the simulation on which value retrieval and state changing commands can act.

The scenarios that the use of TraCI opens, are several. One of the largest use that has been done is the interaction with traffic light systems and signal optimizations since triggered and dynamic systems can be generated only in this way. An example is provided by study “the Stochastic Optimization of Signal Plan and Coordination using Parallelized Traffic Simulation” by Junchen Jin and Xiaoliang Ma [28]. The motivation of this study is that the signal optimization can reduce congestion. The idea behind it is to interact through an interface with the signal controller while the simulation is running. SUMO is here used as the simulator engine and the logic signal control is implemented using Python language thanks to TraCI.

One major drawback of TraCI that must be taken into account is that it slows down the simulation. The slow-down depends on various factors, mainly:

- number of TraCI function calls per simulation step
- types of TraCI functions being called (some being more expensive than others)
- computation within the TraCI script
- client language

A practical example of such a slowdown is provided by the studies of the Bologna scenario [29]. This scenario generated using SUMO contains 9,000 vehicles and it lasts 5,000 simulation steps. Different running times have been recorded [26]:

- without TraCI 8 s
- plain position retrieval 90 s
- retrieval using embedded python 46 s

It must be considered that usually value retrievals are less expensive functions with respect to state changing ones.

Table 2.3 Elements on which value retrieval and state changing commands can act

Simulation elements	Value Retrieval	State Changing
induction loops	X	
lane area detectors	X	
multi-entry/multi-exit detectors	X	
traffic lights	X	X
lanes	X	X
vehicles	X	X
persons	X	X
vehicle types	X	X
routes	X	X
points-of-interest	X	X
polygons	X	X
junctions	X	
edges	X	X
simulation in general	X	X
simulation visualization	X	X

2.4 About scenario generation

Besides the simulator availability, a very delicate aspect of a traffic simulation is related to the generation or availability of the scenario. Often SUMO is used to test particular situations and solutions that do not require a scenario bigger than some blocks or one or few junctions. For small scenarios like these ones, there is usually no particular problem in fixing the map and generating a synthetic traffic for the scope of the study.

When instead the study necessitates to deal with a big scenario the preparing of the simulation gets much more complicated. The importance and criticality of this aspect is underlined by the fact that many of the works and researches that relied on SUMO were focused more or less exclusively on the scenario generation. Since many of the problems encountered and solutions found have been useful for the scope of the thesis, it is worth to rapidly discuss about the most significative ones. Most of the works dedicated to the big scenario generation were focused on the gathering of input data in order to build a realistic simulation, particularly they were focused on map and traffic generation. Table 2.4 summarizes the important aspects of the projects described in the following paragraphs.

In 2013, a research study with the name “A Use Case for SUMO: Simulating Traffic around the Port of Duisburg” by Gerhard Hermanns, Thomas Zaksek, and Michael Schreckenberg for the University of Duisburg-Essen [30], has been dedicated to the census of the traffic in the area around the port of Duisburg. In the work not only the number of vehicles in the main roads were counted, but also origin and destinations were determined measuring turning ratios at the junctions. In order to build the simulation, the network was imported from OperStreetMap [26], converted using NETCONVERT and edited adjusting number of lanes and speed limits. In this case, information about the cycles of the traffic light system was provided by the city of Duisburg. JTRROUTER module was then used in order to generate the trips that compose the traffic in the area, starting form the turning ratios information. This solution was adopted due to a lack of information that made an O-D matrix approach not possible. A problem due to this solution was the generation of a non negligible number of routes that did not have any sense, such as vehicles entering and leaving the network at the same edge. To overcome this problem and obtain a more realistic scenario, a list of allowed routes had to be developed and used to substitute those non sense routes.

Another research work that treats some problematics related to big scenarios is “DFROUTER —Estimation of Vehicle Routes from Cross-Section Measurements” by TeRon V. Nguyen, Daniel Krajzewicz, Matthew Fullerton and Eric Nicolay of 2015 [31]. The work offers a discussion about traffic generation using O-D matrixes. It analyses and shows the problematics of the scenario generation of the Pope’s visit to the city of Cologne in the year 2005, where about 1 million people were expected to participate at the event on a green field near the city. Project’s objective was to support the police and the local traffic management with on-line information about road conditions. Information was collected by means of an airborne camera based traffic surveillance system mounted on a zeppelin and by means of induction loops. These data were used to calibrate a mesoscopic traffic simulation able to predict the situation half an hour in the future. A mesoscopic simulation have been performed also because a microscopic traffic demand generation was too demanding from time and power points of view. To achieve the goal, an initial demand to be calibrated using on-line measurements was necessary. These input data included to O-D matrixes: one describing a usual working day, the second one describing an estimation of the road traffic during the Pope’s visit.

O-D matrixes are a form of traffic input data that describes the demand by dividing a given area into “taffic assignment zones” or TAZs. For each TAZ at which vehicles start, the number of participants approaching a given destination is provided. Two major types of estimation exist:

- Static method: assumes that O-D flows are constant over time and determine an average, used particularly for long time transport planning and design purposes
- Dynamic method: considers time variation, used for short-term strategic traffic control management

Matrixes used for the Cologne scenario were static containing the demand of a complete day.

Criticalities encountered in the work concerning the visit of the Pope to Cologne, were mostly related to the uncertainty about the applicability of the demand description since the necessary traffic assignment activity is very sensitive to both the road network representation and to the demand itself.

Very important are the motivations that led one of the major works, exclusively focused on a big scenario generation that has been made available to the public: the Bologna Scenario. This is described in the work: “Traffic simulation for all: a real world traffic scenario from the city of Bologna” by Laura Bieker, Daniel Krajzewicz, Antonio Pio Morra, Carlo Michelacci and Fabio Cartolano, in a collaboration between DLR and city of Bologna [29]. The initial hypothesis advanced by the authors and that justified the necessity of such a work are that it is time consuming to set up a realistic traffic simulation scenario. Even with the availability of necessary data, an important effort is still required to gather, convert and adapt them in order to replicate a part of a real road network. Issues concern both the map generation and adaptation and the demand generation based on the given measurements. For these reasons, three real world traffic simulations of two areas located east to the inner city ring of Bologna have been created and made available to the public so that researchers are able to start their investigations with a minimum preparation effort and can concentrate on their research questions. The possibility to improve the quality of scenarios with corrections and enhancements is also available.

Authors claim the importance of the quality of input data in order to obtain a realistic simulation, particularly for what concerns the representation of the road network, of the demand and of the real traffic lights logic. Besides the time consumption of the processing and validation of such information, the actual difficulty in the gathering of real world data is underlined as for example traffic lights signal plans are very seldom made available to the public.

The scenario models have been supported by the use of VISSIM ("Verkehr In Städten - SIMulationsmodell", german for "Traffic in cities - simulation model"), a microscopic multi-modal traffic flow simulation software package developed by PTV Planung Transport Verkehr AG in Karlsruhe [28]. Passenger vehicles are described in aggregated way, e.g. the number of vehicles to insert for certain roads located at the network’s border. Following their initial route, the vehicles go through “routing decision points” where they are randomly assigned with a new route according to a given distribution. This method reproduces the turn percentages at intersections that have measured in reality. It is important to highlight, as a proof of the complexity of the task, that these scenarios describe the traffic a the morning peak hour, between 8:00a.m. and 9:00a.m.

Datasets containing detector measures were made available by the municipality of Bologna. These contains the measures 636 detection sites from the days 11.11.2008-13.11.2008, Tuesday to Thursday. The choice of these days is due to the fact that these are usually the only “regular” weekdays speaking of traffic: Mondays and Fridays have different shapes due to various factors mostly related to the proximity of the weekend. Despite the aim and the validity of the work, even in this scenario there are still some open issues that should be improved: the multi-lane roundabouts produce unrealistic traffic jams. The reason of this behavior is likely to be related to the already mention conservativeness of the simulator. Other points of improvement are related to the absence of pedestrians [29, p. 25].

One last project dedicated to the problematic of the big scenario generation that is important to mention is the LuST (Luxembourg SUMO Traffic) project, described by Lara Codeca, Raphael Frank, Thomas Engel in the work: “LuST: a 24-hour Scenario of Luxembourg City for SUMO Traffic simulations” [32]. Motivations of this work are the same advanced for Bologna scenario creation but with the pledge to overcome one of the main limit of the Italian work: the duration. Another limitation in that scenario was the relatively small size, not covering the whole city.

The city chosen in this case is the City of Luxembourg since it is claimed to have a topology that is common in mid-size European cities, i.e. a central downtown area surrounded by different neighborhoods linked by arterial roads, and real information concerning traffic demands and mobility patterns was available for the project. LuST scenario covers an area of 156 km2 and 932 km of roads, it is big enough to show standard congestion patterns but small enough to execute simulations in a reasonable amount of time. Network has been extracted from OSM and ACTIVITYGEN tool has been used for the generation of activity demand. All the statistical data (e.g. population, age distribution, schools etc) necessary as input for ACTIVITYGEN where provided by the government that published them on the Internet site of the Luxembourg National Institute of Statistics and Economics studies. A traffic demand around 140,000 vehicles per day and 38 bus lines with 563 bus stops, has been generated. Trips generated by the traffic demand have then been optimized using DUAROUTER tool. Different uses have then hypothesized for the LuST scenario such as the evaluation and testing of network protocols or the applications for intelligent transport systems. This scenario is supposed to require only a regular maintenance as new SUMO versions are released with the implementation of possible additional tools and has been made freely available to the community.

Table 2.4 Summary of the characteristics of the works concerning big scenario generation

City	Duisburg	Cologne	Bologna	Luxembourg
How input was collected	Traffic Census	Airborne camera and induction loops	Detectors installed by the Municipality	Government publications about statistic of Luxembourg
How traffic was generated in the simulation	Use of JTRRouter providing turning ratios and vehicle counts as input	O-D matrix for calibration of mesoscopic traffic demand	Insertion from network's borders and rerouting based on turning ratios at the junctions	ActivityGen tool: based on networks statistical data the tool generates the traffic
Limits	Lack of information mad the use of O-D matrixes not possible	Uncertainty of applicability of the traffic demand description	Scenario covers Only one hour and network does not contain the whole city	No limits have been described, only regular maintenance is required
Problems	Some unrealistic routes have been generated	High sensitivity of TAZs to networks	Multilane roundabouts cause unrealistic jams	No problems have been mentioned

2.5 Comments on OSM

After some exploration of the complexity of a big scenario generation, particularly related to the traffic, it is necessary to make a few comments about the map generation.

OpenStreetMap or OSM is one of the most used data sources for road network generation for traffic simulations, thanks to its extensive world-wide coverage and the fact that it is continuously updated and provided for free being open source [33, p. 24]. Moreover, OSM is supported by a very detailed and complete user and contributor guide [34].

SUMO is not the only simulator that can rely on conversion of OSM data, other simulators can be VISUM, Vissim or MATSim. These networks describe a logically linked road network with attribute data such as the number of lanes, speed limits, traffic lights or other elements [33, p. 24]. Among the mentioned simulators, SUMO is the only one that considers intersections as separate complex entities with their own shape [21, pp. 37-38], this can provide some more detailed information, for example the distance covered in a junction that depends on the street the vehicle is coming from and the street the vehicle is going to.

Signal groups are by definition at the end of the edge in SUMO and cannot be in the middle. This could cause some problems of inaccuracy when intersections have complex shapes, giving rise to unrealistic signal system. Figure 2.4 shows an example of such a problem where the traffic light signals are grouped in the middle of the rectangular intersection. It has to be stated though, that usually such problems of conversions of intersections happen really seldom for highway junctions as the one reported in the figure since these most of the times contain information about the ramp connections at different level, only in the absence of these information such a problem appear. Figure 2.5 shows good conversions of different level intersections, on the left one can see a highway intersection, on the right one can see the intersection with an underground train, no interference is present.

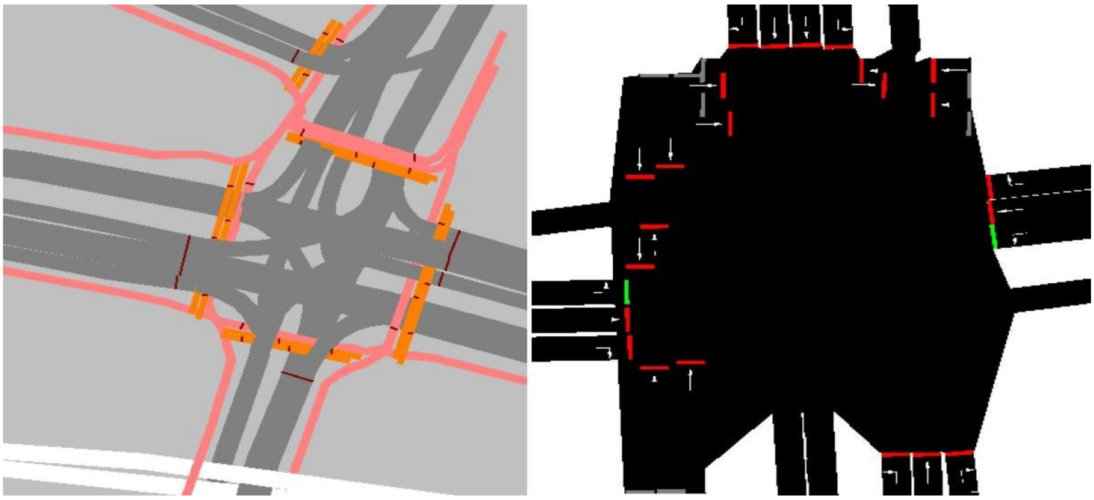

Figure 2.4 problems of roughness of conversion for complex intersections [21]

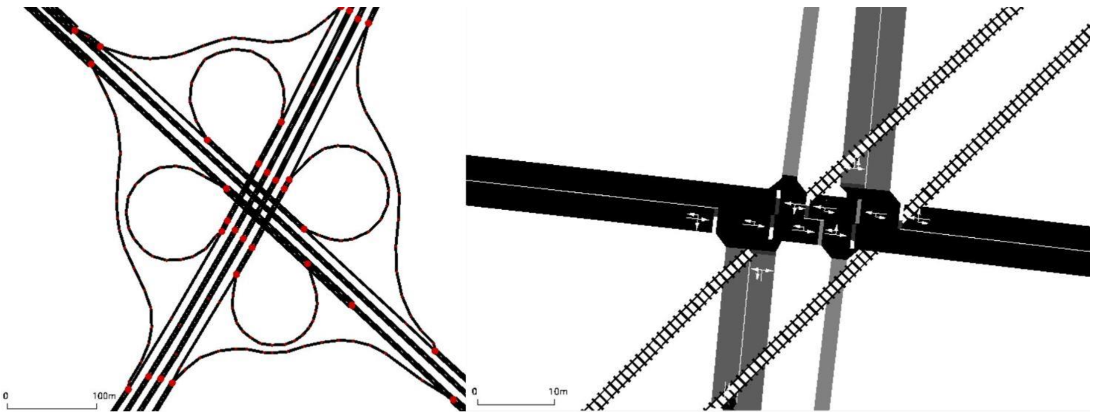

Figure 2.5 Info about different levels of the streets present in the map avoids conversion problems

2.5.1 Translation of the data

OSM information about geometry and descriptions are stored in an XML formatted text file where each feature is described with tags. A tag is composed by a key and a value. Code 2.1 reports an example of an edge (road) representation.

```
<lane id="xxx" speed="yyy" length="zzz"/>
```

Code 2.1 Edge representation by means of tags in XML format

Usually tagging follows certain guidelines in order to describe certain features but these are not binding and that is what makes it easier for crowd sourcing.

Crowd sourcing allowed a growth of data about road infrastructure and lanes in the years. This increase of available information though, is not always able to cope with the ever increasing need of details about lane-level and road infrastructure required by the simulations aimed at the development of driver assistance and automation systems where traffic simulations are coupled with driving simulators. This information may be missing as already seen in the previous section in figure 2.4 or modelled in an inappropriate way.

Another issue is related with tags themselves: for information regarding [lanes] or [turns] tags are correct in more than 99% of the cases but for other information like [width] information often includes units of measure, leading to errors of interpretation. Furthermore, being tags not bindingly regulated, can be cause of confusion when it is necessary to state which kind of information can be retrieved. Examples of these tags are different such as [turn:lanes], [turn:lanes:forward] or [turn:lanes:backward]; these are rare but still they exist. To overcome or at least to minimize most of these issues, DLR is working together with major German car manufacturers to develop more precise guidelines to survey road networks in a simplified but improved way in order to meet simulation requirements. Guidelines have to cover also points such as the collection of attribute data so to support a successive transformation into specific driving simulator or navigation formats [33, pp. 24-25].

2.5.2 Reliability of the data

OSM is an open source project aiming at the creation of a free world digital map. The pillar of such a project is the engagement of the participants: information is collected by members, put in a central database and distributed through the World Wide Web. This process has been classified as “Volunteered Geographical Information” (VGI). Mordechai Haklay, from the department of Civil, Environmental and Geomatic Engineering of the University College London has dedicated a research work to the investigation of the validity of VGI data analysing opportunities and threats offered by crowdsourcing [35].

The biggest opportunity offered by crowdsourcing, is that the large scale community opens the possibility to new activities of difficult or impossible alternative implementation, thanks to a larger ability to collect and share information. SUMO community is a perfect example of utilization of these opportunities.

The issues related to crowdsourcing are that large groups of volunteers are working without coordination, each of them usually concentrated on his own interests. Moreover participants are likely to be amateurs instead of professionals, therefore they may not follow common standards in data collection, verification and use.

Different expedient though are adopted in the effort of keeping the quality of data as high as possible: unlike other sourced database, where contents are generated in disparate locations, OSM community organizes a series of local workshops. These workshops aim at the creation and annotation of content for localized geographical areas and at the introduction of new contributors to the community. In addition, there is also a very detailed user and contributor guide available in the wiki dedicated to OSM [34]. Comparing a series of quality standards, the work concluded that a very good, although not constant, spatial data quality is achieved.

2.6 Researches on Mobility

2.6.1 Researches based on the use of SUMO

After a rapid excursus on researches aimed at the scenario generation and a brief discussion about OSM, it is now important to focus on mobility researches dealing with autonomous vehicle transportation.

Rainer Gasper, Stephan Beutelschieß, Mario Krumnow, Levente Simon, Zoltan Baksa, Jochen Schwarzer, investigated the roboshuttles use in the campus of the Robert Bosch GmbH in Renningen and described their researches in the paper "Simulation of Autonomous RoboShuttles in Shared Space" presented at the SUMO conference of 2018 [36]. RoboShuttles here are considered as a solution to solve the so called "last mile problem" i.e. the gap from the transportation network to the final destination. Roboshuttles have a capacity between 4 and 12 people and a maximum speed of 20-40 km/h depending on the operating area. The area considered in this study is closed and RoboShuttles must share it with pedestrians so SUMO had to take into account the interactions with them. Pedestrians present some peculiarities with respect to other vehicles as they can turn on the spot and can be simulated by SUMO by activating some on-purpose options. RoboShuttles instead behave in a similar way to regular vehicles but being them automatic, the parameter sigma that describes the driver imperfection is set to zero and reaction time is assumed to be 0.1 s being it the minimum possible as the simulation step size is 0.1 s. By defining a ride stage in the trip definition and sending it to the simulation via TraCI, SUMO allows pedestrians to board the vehicles and be driven around as passengers.

Since the modelled area is closed, the only vehicle traffic is the one generated by RoboShuttles while the pedestrian one is modelled taking in account distributions of typical flows among the points of interest in the campus (entrance, working places, cafeteria etc). Vehicles have been inserted in the simulation via TraCI and assigned a fixed route. In the scenario, a pedestrian sends a request to the operation strategies with actual position, time to destination is then analysed and operation strategy checks whether there is at least one shuttle with a free seat. Most advantageous solution is found in such a way to minimize the time to destination for the pedestrian and to avoid that the RoboShuttles wait for him at the stop (so pedestrian has to arrive first at the picking point). Best distribution of RoboShuttles has then been found and results have been analyzed at the different times of the day. In such a case study, it has not only been applied an on demand mobility model, but it has also been demonstrated that a simulation of multi modal trips of pedestrian with SUMO is possible and that the program is suitable for studies of the traffic systems in shared space.

Another study to be mentioned is "Simulating the Impact of Shared, Autonomous Vehicles on Urban Mobility – A Case Study of Milan" by Sabina Alazzawi, Mathias Hummel, Pascal Kordt, Thorsten Sickenberger, Christian Wieseotte, Oliver Wohak, as the previous one, presented at SUMO conference 2018 [37]. This work was aimed at the usage of autonomous, on-demand vehicles to reduce vehicle counts in urban environments taking as an example the city of Milan. The concept has been implemented through the usage of so called robo-taxis, very similar to the already mentioned RoboShuttles, autonomous electric vehicles with a capacity of 6 people each. Input traffic data has been inserted using an O-D matrix derived by publicly available mobile phone usage data. SUMO has been used as the simulation framework while TraCI has been used for the introduction of the robo-taxis and the management of ride assignment and ride sharing algorithm. Via TraCI interface it has been possible to access the simulation each simulation step and to distinctly steer each vehicle and person. Robo-taxis never left the simulation and drove continuously, even without any passenger on board, going in that case towards areas where the mobility demand is higher. A matching algorithm has been implemented to pair passengers to a robo-taxis in the nearby and with an already similar route. The mobility demand is coming from pedestrians introduced in the simulation with "unassigned" status, as soon as a robo-taxi has been matched, the status is changed to "assigned" until the vehicle actually arrives and picks the passenger, once the pedestrian reaches its destination it is dropped and removed from the simulation. Similarly robo-taxis in turn change their status from unassigned, to assigned once the passenger is matched, they have also occupied status once a passenger is already in. Conclusions of such a study have been that the introduction of 9,500 6 seated RoboTaxis in Milan would free the city from congestion thus drastically reducing emissions and energy consumption.

Table 2.5 Summary of characteristics focused on mobility studies with SUMO

Object of study	RoboShuttles in shared space [36]	Autonomous mobility in Milan [37]
Ride sharing implementation	yes	yes
Type of route	Fixed	Depending on the demand
Scenario	Closed research campus	City of Milan
Conclusions	SUMO validity for multi modal simulations	SUMO validity using TraCI 9,500 shared AGVs can substitute private traffic in Milan

2.6.2 Researches not based on the use of SUMO

Outside the framework of SUMO, several researches have also been performed for what concern mobility. Similarly to the Milan case study, many are the works that aimed at the investigation of the substitution of private demand with a Mobility On Demand (MOD) system.

One of these works, described in "Simulation of Citywide Replacement of Private Cars with Autonomous Taxis in Berlin" by Bischoff and Maciejewski [38]. In this work, MATSim (Multi-Agent Transport Simulator) has been used in order to simulate the replacement of private traffic in the city of Berlin with an autonomous taxi fleet. Simulation implemented was microscopic based on a synthetically generated dynamic demand. Taxis operated without the use of the ride sharing and once the customer was dropped, instead of going to a taxi stand, the vehicle was parked in the same place. Results based on the area of 891 km^2 with 3.5 milion inhabitants revealed a need of 100,000 autonomous vehicle in order to completely satisfy the private demand with an acceptable service level. Based on the actual number of vehicles in Berlin, the research concluded that each taxi could replace 11 private cars.

A similar work is “Study of a Shared Autonomous Vehicles Based Mobility Solution for Munich” done by Fares Mrad Agua at TUM in 2016 [39]. This aimed at the investigation of the feasibility and performance of an electric SAV (Shared Autonomous Vehicle) based Mobility On Demand (MOD) service that would cover the whole intra urban mobility demand originally performed with private cars. The city of Munich has been chosen as the basis of the study because of the availability of individual mobility demand and because of the pledge the administration made to have 100% of the energy generated by renewable and environmentally compatible sources by 2025 thus, improving positive effects from emissions point of view with the application of electric mobility system in the city.

The major steps forward this study claims involve the implementation of an electrification system with charging station and a high quality of traffic demand input data being these provided by the municipality which operates and maintain a multi-modal transportation model. The simulation performed is a large scale agent based one, implemented using Java and JADE (Java Agent Development Framework) and main goal of it is to determine the minimum required fleet size to cope with the demand with a good service level. In the study, the simulation is also combined with a cost model in order to estimate the cost per kilometre of the service. Choice of adoption of the model were influenced by the possibility of reuse of the code of previous projects within the FTM (Lehrstuhl für Fahrzeugtechnik, german for Institute of Automotive Technology) investigating electrification potentials of the city of Munich.

Within the simulation, each SAV is represented as an individual agent and so is the central taxi office. Taxi agents are then responsible for the fulfilling of customer rides. They can decide individually about their actions and send proposals or reject customer requests. The study concluded that in order to fulfil the private traffic demand of Munich, consisting in 1,061,183 daily trip requests a minimum fleet of 34,000 SAVs and 1,370 charging stations are necessary. Adopting such a solution, each SAV serves on average 32 customers covering 234 km a day.

Table 2.6 Summary of characteristics focused on mobility studies with other simulators

City studied	**Berlin [38]**	**Munich [39]**
Simulator used	MATSim	JADE
Type of simulation	microscopic	microscopic
Ride sharing	no	no
Inhabitants	3,500,000	1,500,000
Vehicles needed	100,000	34,000

2.7 KPI: waiting time

As it will be explained in section 3.1, one of the KPIs identified to evaluate fleet performance is the waiting time. In order to evaluate the results of the research, it is necessary to compare waiting times with the ones of existing alternative services.

Main alternative service to taxi one has been considered the public transport system. Munich's public transport system integrates different networks allowing the multi modal transportation with just one ticket. Means of transport available are: busses, tram, S-Bahn (urban rail) and U-Bahn (underground) [40]. The whole network relies on 245 stops for the mass transport (S-Bahn and U-bahn) integrated with the bus one counting more than 400 busses and 900 stops and the tram network with 106 vehicles and 165 stops [41].

Moovit is a smartphone application used to plan itineraries and routes in the cities using public transport. With more than 200 million users, it is the most downloaded app for this kind of purpose, accounting the availability of more than 2,500 cities in 82 countries [42]. It even accounts partnerships with local transport companies as it is the case of the city of Turin, where the local operator GTT suggests the use of this application as a valid alternative to the one developed by the company itself [43]. The aim of such an application is to provide a sustainable and integrated mobility platform in order to empower smart cities, improving its precision by exploiting the collection of data from the large community, making it the world's largest repository of transit data [44].

This collection of data allows the development of some summary analysis concerning the statistics of public transport of some of the cities served by the application. These statistics are called Moovit Public Transit Index and are published in a dedicated section of the website [45]. Main data concerning Munich's public transport statistics are summarized in Table 2.7.

These figures can be key for a comparison with waiting times collected in the study. It is also important to consider the fact that data refer to an average week day, on which most of the studies usually concentrate, but it is reasonable to expect that, because of a variation of the demand pattern, situation during the weekend can be different. As exposed in section 3.4.3, the highest peaks of the week are obtained on Friday and Saturday nights in Munich. Such a phenomenon is quite common in many countries and cities around the world as many studies, guides and interviews highlight the elongation of waiting time (and thus a decrease of service quality) during weekend nights due to a higher saturation of the fleet [46–48].

Table 2.7 Statistics about Munich's public transports according to Moovit Public Transit Index [45]

Type of data	Unit	Measure
Average commute time	min	56
Riders that commute more than 2 hours every day		11%
Average waiting time at the station on a weekday	min	10
People that wait longer than 20 minutes every day		6%
Average trip distance	km	9.2
People that travel for more than 12 km in a single direction		21%

Particularly interesting from this point of view are figures reported by Lisa Rayle from UC-Berkeley in her studies concerning ridesourcing services such as Uber or Lyft [49]. The study was based on the collection and analysis of information about mobility by means of surveys in San Francisco in June 2014. Two kind of people have been interviewed: individuals who just completed a ridesourcing trip and individuals who used ridesourcing services in the past two weeks. Some results of this research are reported in Table 2.8.

From these data, not only the variations in terms of waiting times can be noticed, but also a substantial resilience of waiting times related to ridesourcing services making this last, more competitive with respect to the taxi one. The reasons addressable for this phenomenon are two:

- Technological efficiencies: ridesourcing enjoys a smartphone enabled matching while taxi enjoys a telephone dispatch
- Vehicle supply: ridesourcing is not subjected to any regulation that restricts the supply

These analysis turned out useful in the interpretation of data exposed in chapter 4.

Table 2.8 Comparison of waiting times of ridesourcing and taxi service (dispatch and hailing) [49]

Period considered	**Waiting**	**Ridesourcing percentage**	**Taxi dispatch percentage**	**Taxi Hail per-centage**
Weekday daytime (4:00-18:00)	< 10 min	93%	35%	39%
	10 – 20 min	7%	41%	29%
	> 20 min	0%	23%	32%
Weekday nighttime (18:00-4:00)	< 10 min	92%	16%	33%
	10 – 20 min	6%	47%	31%
	> 20 min	1%	37%	36%
Weekend day	< 10 min	88%	16%	25%
	10 – 20 min	12%	39%	35%
	> 20 min	0%	45%	39%

3 Approach and methods adopted

3.1 Approach description

Section 1.2 described the goals of the thesis, i.e. to explore the possibilities offered by SUMO software and to use it in order to simulate the taxi traffic in Munich. In the work the investigation of the ability to serve customers tuning fleet size and vehicles distribution is performed in order to identify an optimum fleet size based on results reported by two KPIs, namely waiting time and fleet utilization. To do so, the following approach has been adopted.

A first phase have been dedicated to the exploration of the possibilities offered by SUMO program with the use of both literature and available tutorials. The need of deeper analysis have brought to the code exploration in order to find the best way to perform the simulation.

After a study phase, a simulation scenario of the city of Munich has been prepared and an average week day has been chosen to focus the study on: Wednesday. TraCI has been used to intervene on the simulation introducing the vehicles, setting taxi stands distribution and implementing the trip demand.

Not knowing the effect of the initial fleet distribution and their duration, it has been considered more accurate to include the day before in the simulation. This two day scenario has been used to observe the effects of the fleet size on the operational times and distances covered by the fleet. At first, no limits of stand capacities have been contemplated to let the fleet free to naturally distribute and to detect activity attractors in the network.

Restricting the focus to a smaller number of fleets, the effect of the distribution on waiting times have been investigated. For this purpose, stand capacities have been taken into account as a constraint for the fleet moves. Main KPIs have then been evaluated, namely waiting times and fleet utilizations in order to choose an ideal fleet able to satisfy the demand with a good trade-off between service level and utilization.

Finally, once identified possible fleet dimensions for a regular day, the study focused on testing a weekend day: Saturday. For the same reasons that led to the choice of a two day scenario in the week, Friday has also been included. The weekend, being characterized by a more extreme demand pattern and higher peaks represents a good test for the analysis of the variation of KPIs with a change in the demand pattern, i.e. how much are the KPIs susceptible to demand variations.

To help in the evaluation of the solution to adopt, a cost model has been developed and used to compare and evalutate the results of previous analysis. This would allow more precision in final considerations.

3.2 Munich

3.2.1 The City of Munich

The city of Munich is the capital of the German state of Bavaria. The city is located along the river Isar, north of the Bavarian Alps with the city center positioned at N 48° 8‘ 22“ latitude, E 11° 34‘ 28“ longitude (Rathaus Turm – Town Hall tower) [50]. It has a city area of 310.7 km^2, counting a population of 1,542,860 inhabitants (85.2% above 18 years). It is the most populated city of the state and the third one in Germany after Berlin and Hamburg but it registers the highest density with 4,966 people per square kilometer.

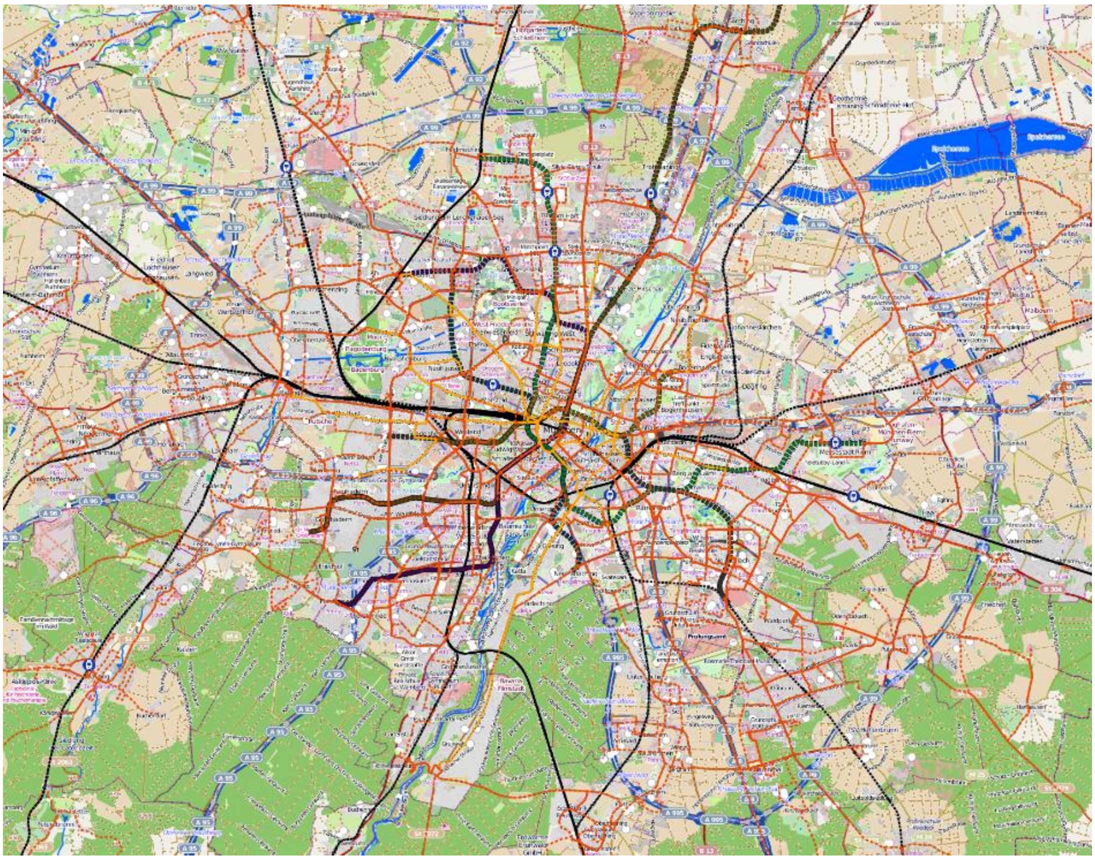

Figure 3.1 Map of the city of Munich with main public transport lines and highways

The city of Munich, represented in Figure 3.1 includes 2,382 km of motorized vehicles road and 943 km of cycling paths. Public transport infrastructure instead includes 434 km of S-Bahn (inter-urban railway) network distributed among 8 lines (black lines in figure 3.1, they cover the metropolitan area), 95 km of U-Bahn (underground) network distributed among 6 lines (bold dotted lines in figure 3.1) , 82 km of tram network distributed in 13 lines (yellow lines in figure 3.1) and 495 km of bus network distributed in 73 lines (red lines in figure 3.1). The registered vehicles are 813,592 of which 701,131 are cars (the rest being motorcycles and lorries) with a quantity as high as 527 vehicles per 1,000 inhabitants [51].

3.2.2 The Metropolitan Area

Munich is positioned at the center of its metropolitan region (MMA, Munich Metropolitan Region) represented in Figure 3.2. This region registers a population of 5,991,144 inhabitants distributed in 25,548 km^2 including 33 districts (city included) [52]. The surroundings of the city include the airport that registers 394,430 departures and landings a year with a count of 42,261,309 passengers [51].

3.2.3 Reasons for the choice of Munich

The city of Munich has been chosen as the location of this study mainly because of the availability of input data regarding the traffic and taxi mobility demand and a higher availability of information regarding the network. In particular, previous research work done by Michael Wittmann [11] collected important data concerning the taxi demand data that will be presented in section 3.4.2.

Another reason that favours the choice of this city is that the municipality of Munich, has made the pledge of being able by 2025 to produce as much green energy as needed by the city itself [53]. Thus, the implementation of an electric vehicle fleet in the city would have a very low impact, counting only on clean energy.

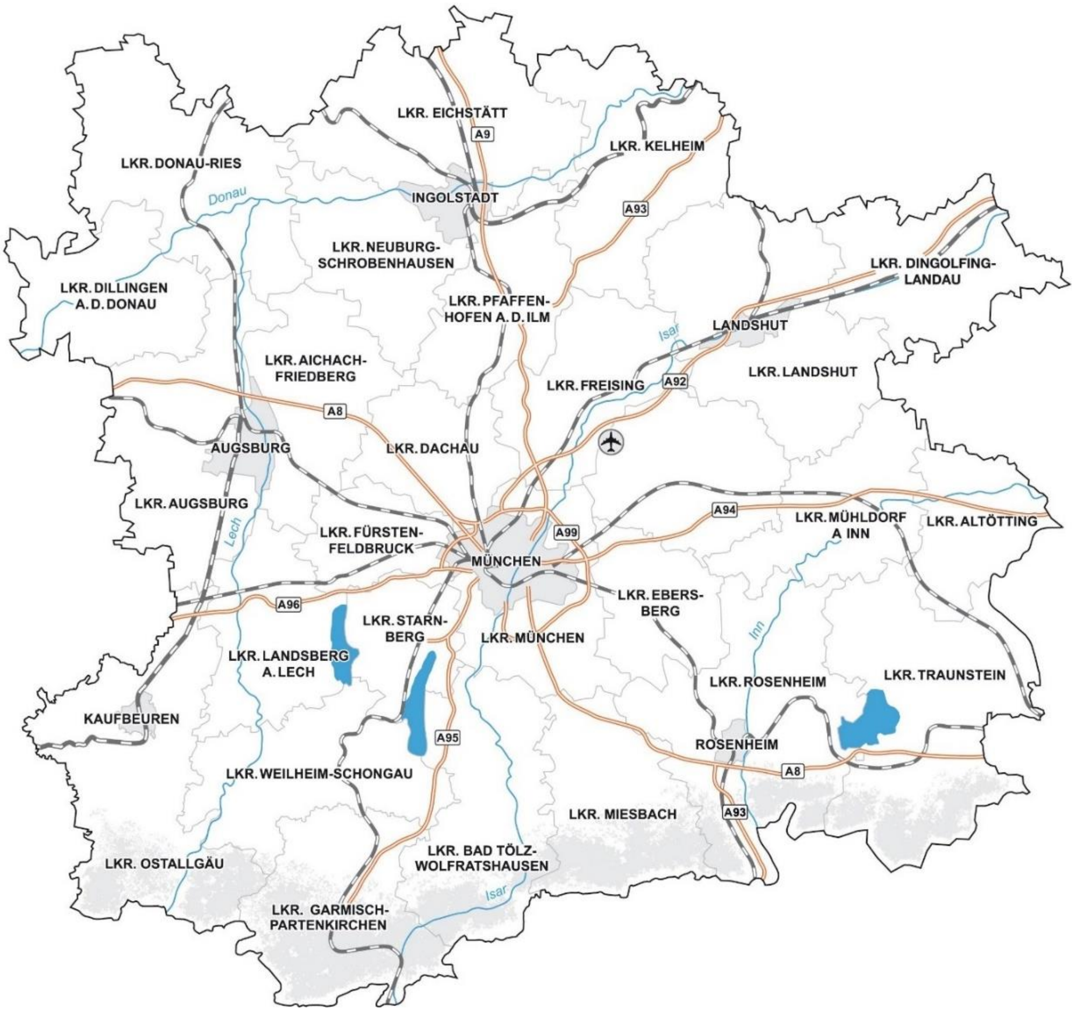

Figure 3.2 Munich Metropolitan Area

3.3 Reasons for the choice of SUMO as a simulator

Opportunities offered by SUMO have already been described along with the program itself in section 2.2. In view of the above, the absence of a study performed with this simulator within the Institute of Automotive Technologies of TUM, required the need of a research work that would explore its functionalities and would validate it for a future adoption for the prosecution of the study, object of this thesis or for new studies within the field of smart mobility.

SUMO has been selected among the other “unexplored” simulators because of its high level of maturity: its widespread use in an ever increasing number of studies has allowed a rapid development and the achievement of high reliability in terms of results.

Another very important reason for the choice of this simulator is that it provides a very detailed user guide that allows a learning and knowledge of the program within a reasonably short time. The wide wiki-like guide available contains step by step instructions, tutorials, cases, limitations and common problems, allowing a higher focus on the subject of the research. In addition, a wide literature describing a wide variety of research studies performed with this simulator is available on line, with most of the titles reported in a dedicated section of the wiki itself.

Finally, the already mentioned advantage that made SUMO particularly competitive among the different traffic simulators is the dimension and availability of the user and developer community that supports its open source nature.

3.4 Input data

Two kinds of input data are necessary for the performance of a traffic simulation: the network where simulated vehicles must move and vehicles themselves, i.e. the traffic demand. This section will describe how these have been collected.

3.4.1 The Network

Difficulties and criticalities linked to the generation of a big scenario have been described in section 2.4 by means of examples provided by previous studies. It is important to state that most of these limits and difficulties could not be overcome in the time this research study has been executed but they have been carefully evaluated and taken into account and will be described in this text.

The area considered is not just the city of Munich because such an area would be too limited as the traffic demand often has as origin or as destination places located out of the highway ring. In particular, an important part of the taxi trip demand involves the “Flughafen München”, i.e. the airport area located north of the city with 4.62% of total destinations for taxi trips demand. Therefore, a key necessity was to include this point of interest in the considereded Network. Different evaluations has been made about the coverage of the network in order to find the smallest possible area so to work wiht files as small as possible but still without losing trips to be requested because either the starting point or the destination are out of the map. Figure 3.3 shows the trips covered by taxis within the period of collection of the data performed by Wittmann [5]. In order to simplify the readability of the image, only the trips between 30 and 46 km have been represented as the majority of the trips (73.5%) is below 5 km and usually within the city.

It can be seen that these trips cover an area much larger than the city but the quantity of these journeys is less relevant as they go farther from it. A confirmation is provided by Figure 3.4 that shows that the quantity of trips within the city in a week is so important that they cover virtually every street around the city center but coverage gets visibly more scarce towards the outskirts. From Figure 3.3 it is also possible to notice the relevance of the traffic between the city and the airport.

After these evaluation the criteria adopted for the selection of the network where to include the airport and the major towns around Munich such as Freising, Dachau or Erding. Final selection is represented in Figure 3.5 and has kept the average trip loss below 1.08%.

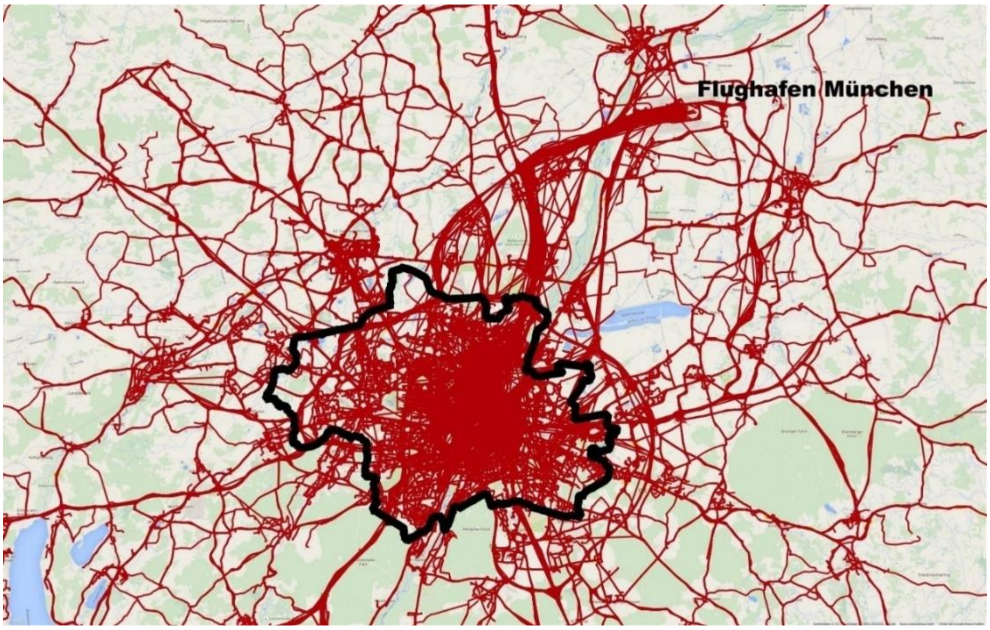

Figure 3.3 Coverage of the taxi journeys in the area of the city of Munich in the whole recording period according to Wittmann [11, p. 18]

Figure 3.4 Coverage of Taxi journeys in the inner city area in a single week according to Wittmann [11, p. 17]

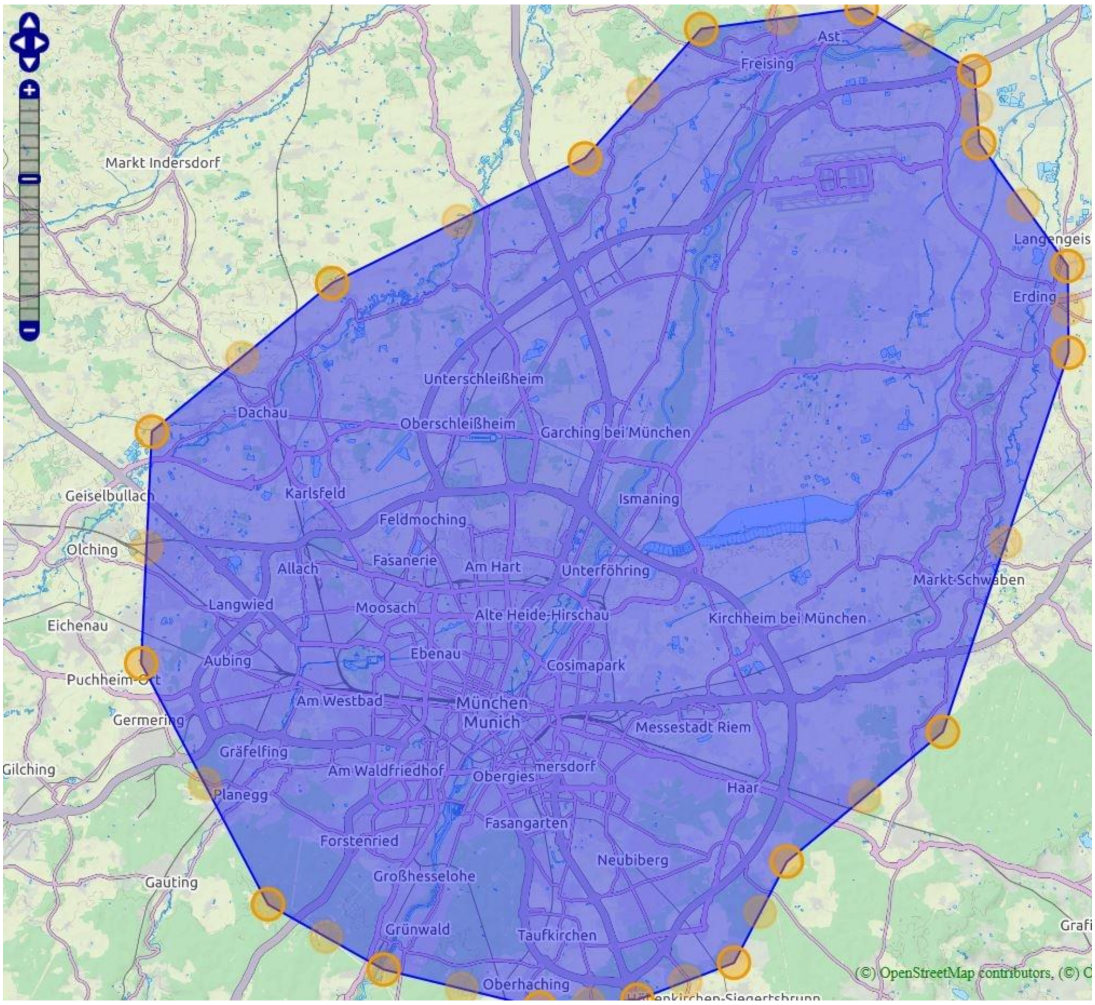

Figure 3.5 Area selected for the study

3.4.2 Taxi Demand Data

For what concerns taxi demand data the work relies on the data already collected by Michael Wittmann in his work [11, pp. 12-31]. In this section a summary about their collection and presentation is proposed.

The source of these data is the Munich Taxi Centre IsarFunk. The data regard approximately 400 connected vehicles (11.7% of all 3,400 taxis in Munich). A Java application written specifically for the purpose of the retrieval of shared data sends periodically a request to the server of the fleet management system of IsarFunk which returns to the institute server requested data in XML format. These data are then converted and stored in Institute's Postgre Database.

Data stored refer to a period of 19 weeks, between the 09.03.2015 00:00:00 pm and the 20.07.2015. During this period a total of 2 million trips have been recorded for an aggregated distance of 9.9 million kilometers. Figure 3.6 summarizes the total coverage of all the journeys in the considered period. It is possible to notice that there are journeys that reach distant destinations such as Frankfurt, Halle, Passau, Saarbrücken, Salzburg and Konstanz. Despite these exceptions, as already discussed in section 3.4.1, the majority of the trips takes place within the area of Munich. In the picture there are also a few straight lines, these are due to an error in the registration of the GPS position in a distant place.

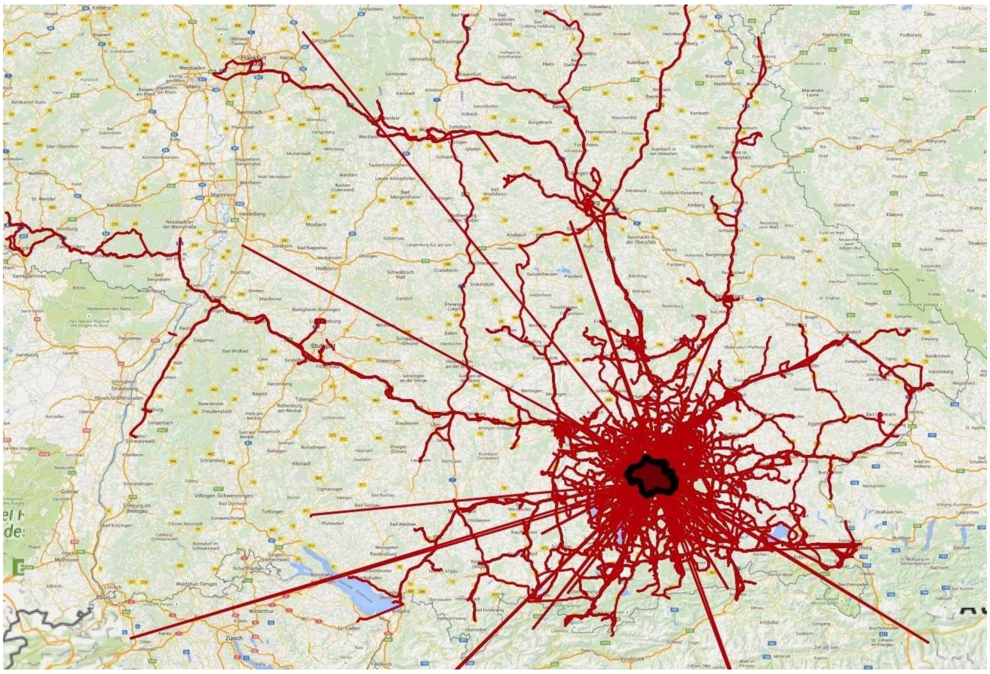

Figure 3.6 Coverage of all trips in the period considered according to Wittmann [11, p. 16]

From data collected, a model of a single week has been obtained through the use of statistical distributions. Detailed description of such a model can be found in Chapter 2 of Wittmann's work [11, pp. 32-41]. The final result is the demand pattern of an average week in Munich, with a total of 40,955 trips performed.

Data have then been retrieved for the execution of the SUMO study in CSV format. Table 3.1 reports data contained in each record of the input file.

Since no information was available about the time in which the request has been issued, the starting time has been used in the simulation as the time of issuing of the trip request.

Not all the week has been simulated but an average day has been selected. Figure 3.7 shows the trip demand during the week represented in trips requested every 15 minutes. During the weekend the demand pattern is completely different compared to the other days, with a quite low request during the day and the highest peaks of the week during the night hours. Monday also shows a different pattern because many activities are not operating at regular regime being some of them closed. Friday instead, has a particular pattern because it is characterized by some earlier activity times and by a high demand peak at night due to the start of the weekend. The only average days left are then Tuesday, Wednesday and Thursday, confirming the thesis exposed in the description of the Bologna scenario in section 2.4. Among these three days, Wednesday has been chosen as it is in between of this period and because a lower demand characterizing Tuesday could speed up the phase-in of the simulation.

Black Rides

According to company Linne+Krause a shadow economy is regularly present in the taxi industry. It is estimated that some 30-40% of the sales revenue generated by taxis is hided to tax authorities. For this reason it is important to keep in mind that in reality more journeys have been performed than the ones actually recorded in the given input data. However, since no more information is available about this additional hided demand, this can not be considered [11, p. 22].

Table 3.1 Data about trip demand stored in CSV file

Type of data	Unit	tag in the file
Trip id		track_id
Day of the week (progressive number)		day
Hour at which the trip starts		hour
Minute at which the trip starts		minute
Duration of the trip	sec	duration
Distance covered within the trip	m	distance
Code referring to the starting area		start_area
Starting point geocoordinates		start_x start_y
Code referring to the destination area		stop_area
Destination point geocoordinates		stop_x stop_y

3.4.3 Input data analysis

A brief presentation of duration and length distribution of the trips obtained with the model and used as input for the simulation is proposed in this section. First the whole week is considered, secondly the two days on which the simulation focus are presented. Since the peak of demands are obtained in the hours around midnight, some extra hours of the day after are in included, namely:

- 28 hours for the Wednesday case, until 4:00 of Thursday
- 31 hours for the Saturday case, until 7:00 of Sunday

Whole week

Comparing Figure 3.8 and Figure 3.9 it is possible to notice that the distributions are not fully coinciding and that the outlier visible for distances between 30 km and 45 km has no correspondent in the graph describing times. This is explainable with the fact the average speed in longer journey is disproportionately higher compared to the one of inner city routes mostly due to the usage of highways.

Table 3.2 and Table 3.3 summarize characteristics of the distibutions of distances covered and times needed to bring customers from their origin to destinations.

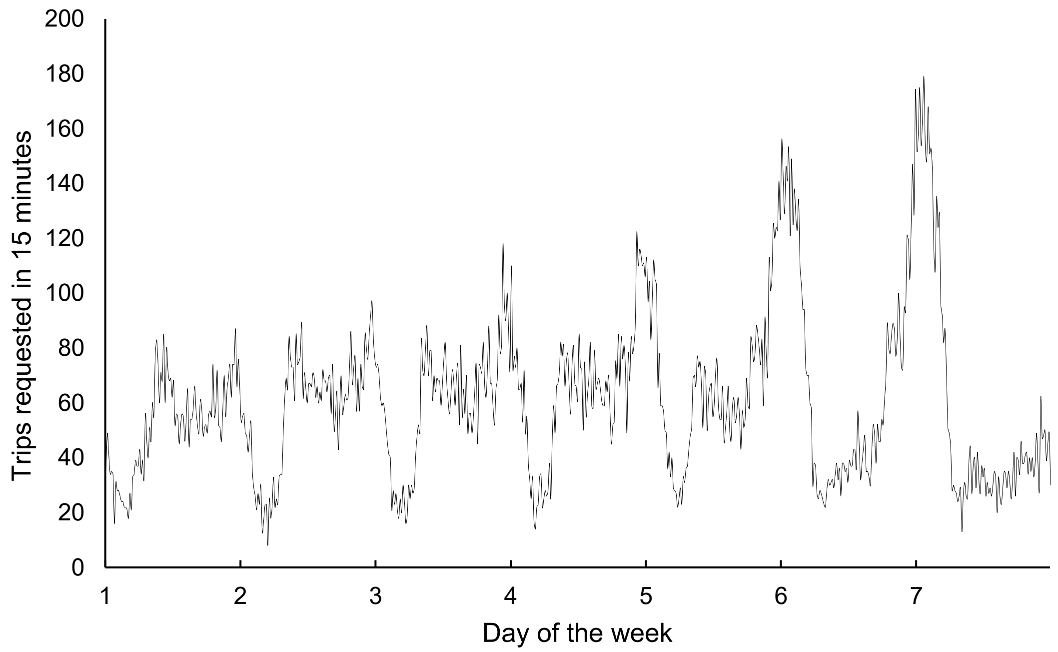

Figure 3.7 Taxi trip demand during the week

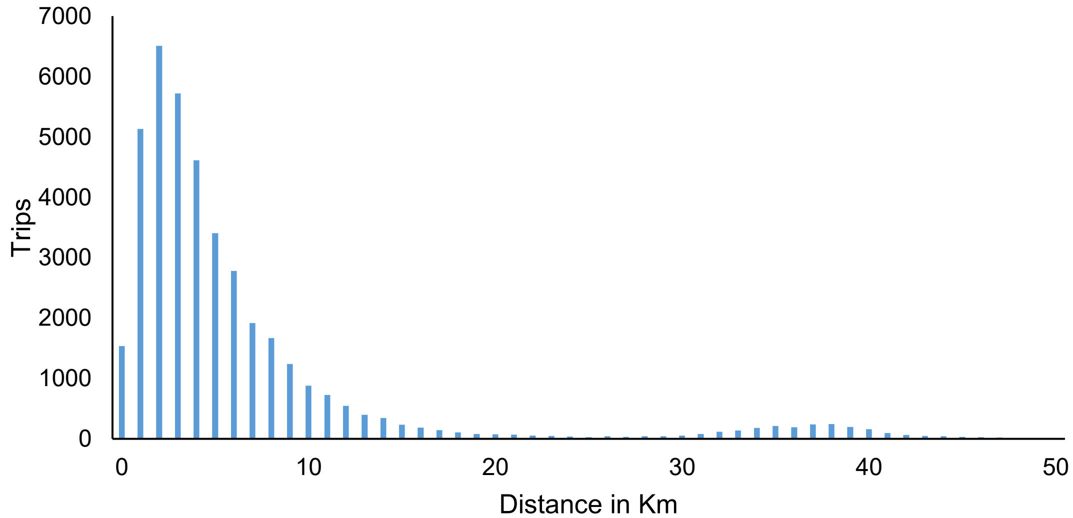

Figure 3.8 Distribution of distances covered per trip in input model

Table 3.2 Summary data concerning the covered distances per trip in a week

Data	Unit	Value
Total trip requested		40955
Average Trip Length	km	7.17
Median Trip Lengths	km	4.32
Standard Deviation in km	km	9.43
Maximum Distance Covered	km	297.82
Minimum Distance Covered	km	0.25
Trips above 30km		5.62%

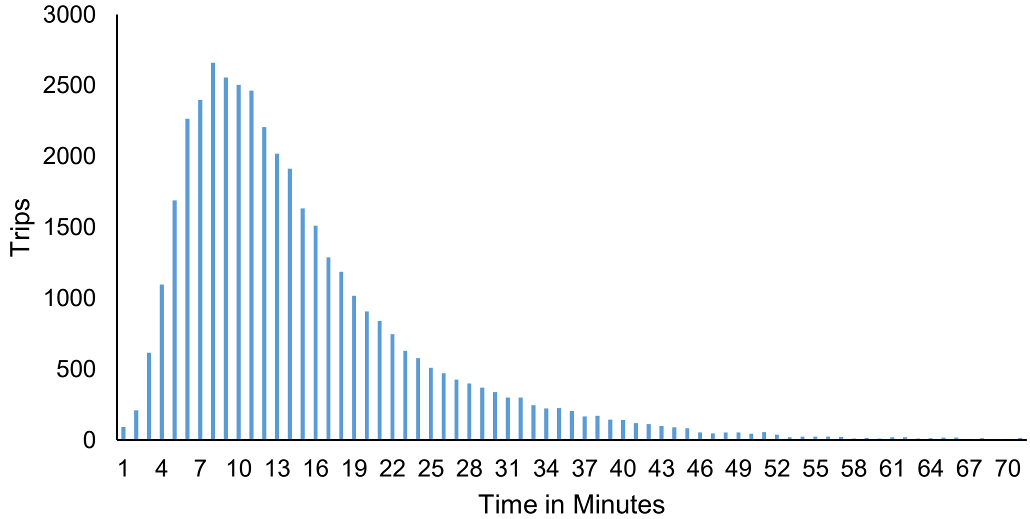

Figure 3.9 Distribution of duration per trip in input model

Table 3.3 Summary data concerning the duration per trip in a week

Data	Unit	Value
Total trip requested		40955
Average Trip Duration	min	14.79
Median Trip Lengths	min	11.85
Standard Deviation	min	11.62
Maximum Duration	min	436.67
Minimum Duration	min	0.28
Trips above 30 minutes		8.35%

Wednesday

Figure 3.10 shows the distribution of the demand over time on Wednesday, representative of an average day. Durations and distance data on Wednesday are comparable to those of the whole week. A higher percentage of long trips for what concerns both distance and duration is present compared to the whole week.

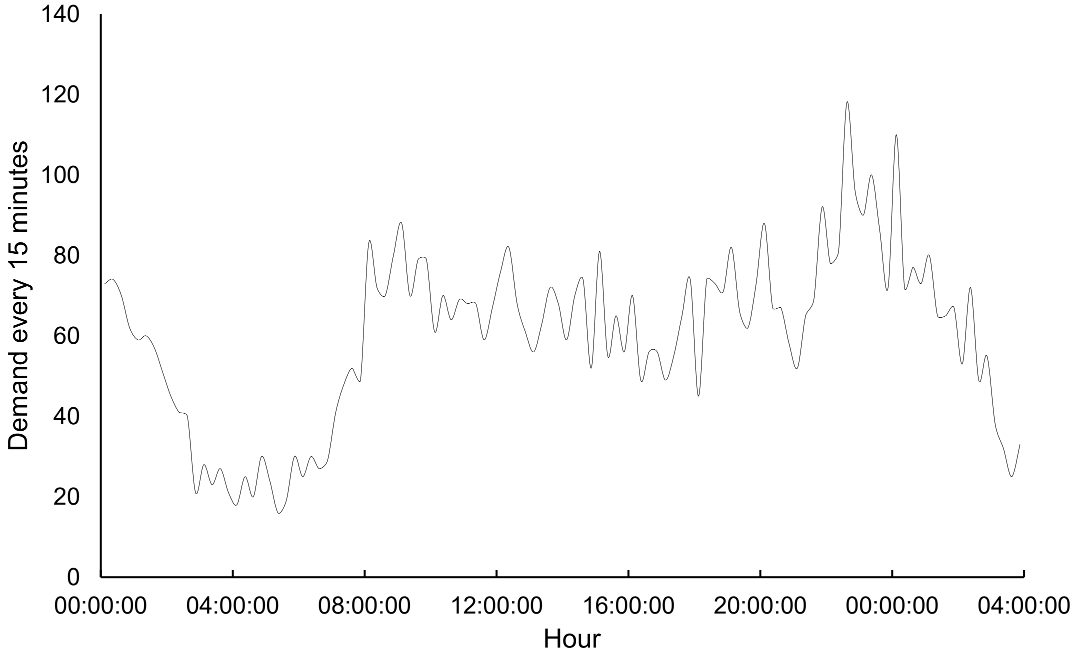

Figure 3.10 Taxi trip demand on Wednesday

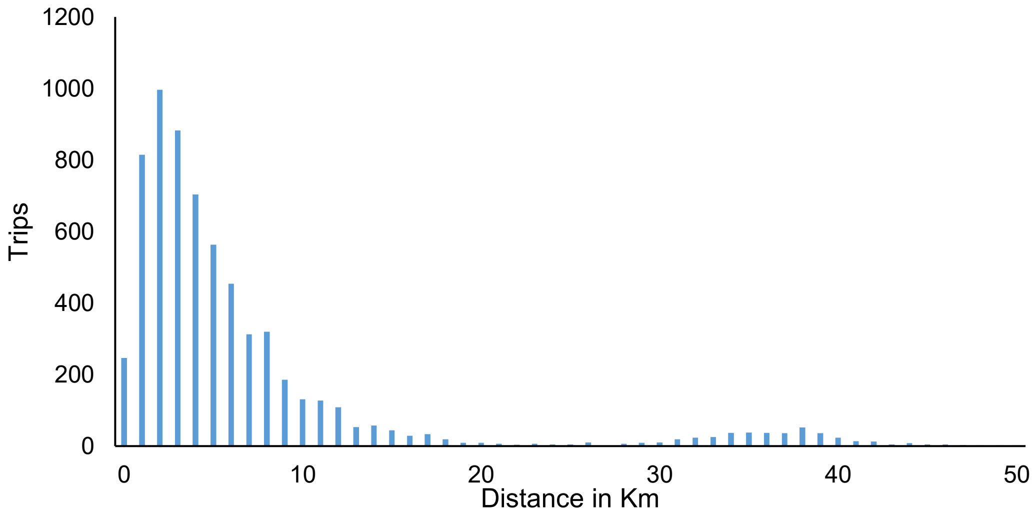

Figure 3.11 Distribution of distances covered per trip on Wednesday

Table 3.4 Summary data concerning the covered distances per trip on Wednesday

Data	Unit	Value
Total trip requested		6576
Average Trip Length	km	7.52
Median Trip Lengths	km	4.48
Standard Deviation in km	km	10.13
Maximum Distance Covered	km	229.08
Minimum Distance Covered	km	0.272
Trips above 30km		6.39%

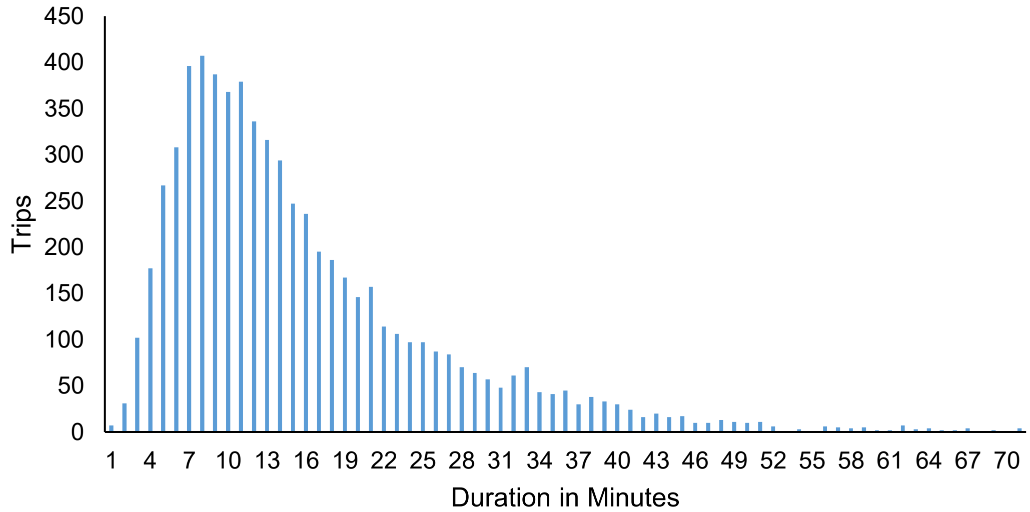

Figure 3.12 Distribution of duration per trip on Wednesday

Table 3.5 Summary data concerning the durations per trip on Wednesday

Data	Unit	Value
Total trip requested		6576
Average Trip Duration	min	15.70
Median Trip Lengths	min	12.30
Standard Deviation	min	12.89
Maximum Duration	min	419.35
Minimum Duration	min	0.28
Trips above 30 minutes		10.51%

Saturday

Figure 3.13 shows the demand pattern along the day of Saturday. Week demand peaks are reached during the night between Saturday and Sunday. To this, residual demand of Friday night must be added where second to highest peak is reached. The result of these factors is a very high demand during the 31 hours analyzed: 24.4% of the whole week demand. Conversely to what seen in the Wednesday case, a lower percentage of long trips is characterizing Saturday demand with respect to the aggregated week average. In fact, even average length and duration are lower and their distributions are more compact. This can be explained with the fact that during the weekend nights there is a lower propension of using private car because of the difficulties in finding a parking place around nightlife attractors, mostly located around the city centre and also because people is more likely not to be in the legal conditions for driving at the return. In addition, the lower availability of public transport causes a higher taxi demand of journeys within the inner city with respect to any other working day.

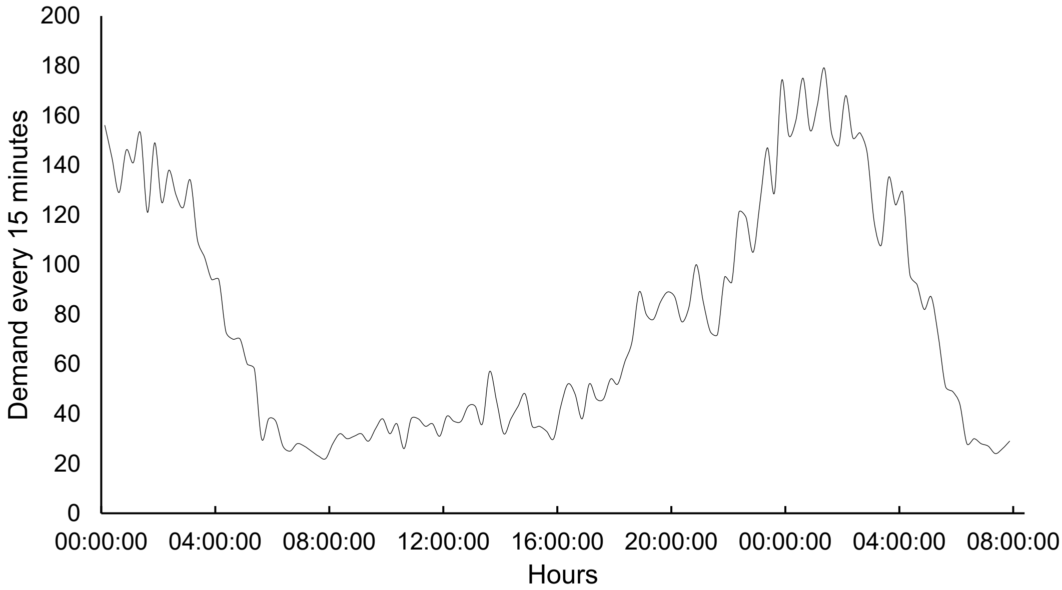

Figure 3.13 Taxi trip demand on Saturday

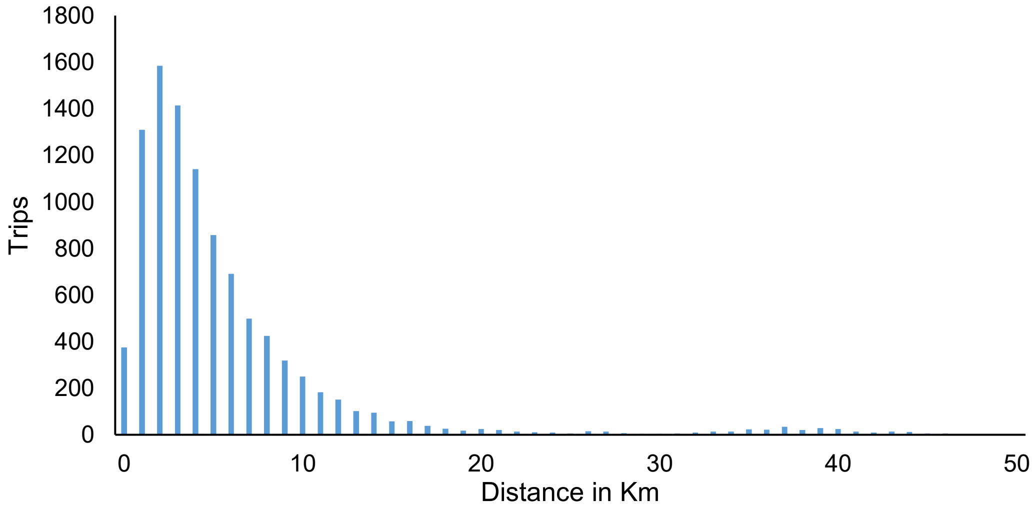

Figure 3.14 Distribution of distances covered per trip on Saturday

Table 3.6 Summary data concerning the covered distances per trip on Saturday

Data	Unit	Value
Total trip requested		9794
Average Trip Length	km	6.21
Median Trip Lengths	km	4.23
Standard Deviation in km	km	7.21
Maximum Distance Covered	km	169.70
Minimum Distance Covered	km	0.25
Trips above 30km		2.57%

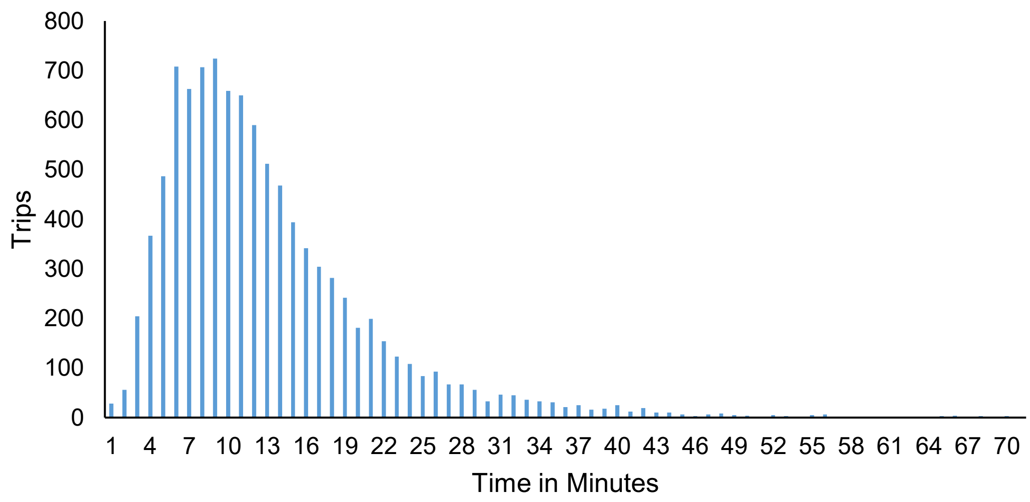

Figure 3.15 Distribution of duration per trip on Saturday

Table 3.7 Summary data concerning the durations per trip on Saturday

Data	Unit	Value
Total trip requested		9794
Average Trip Duration	min	12.65
Median Trip Lengths	min	10.55
Standard Deviation	min	9.52
Maximum Duration	min	323.72
Minimum Duration	min	0.28
Trips above 30 minutes		4.31%

3.5 SUMO and TraCI investigation

Because of the mentioned slowdowns (section 2.3) caused by TraCI, SUMO code has been first explored in order to look for a way that would allow to avoid to interface online through an external application, in order to keep the simulation rapid.

3.5.1 SUMO source code

Besides the user guide, a developer guide is also available online [54]. This contains very detailed instructions for the building of the source code on an IDE that would allow the debugging. Microsoft Visual Studio 2017 has been used for the building of the source code as suggested by the guide. After the downloading of the necessary libraries the source code could be built and debugged. A minimal scenario has been simulated in order to analyze the functionality of the different variables and using also the description of SUMO's architecture present in Pereira's work [55, pp. 37-38] a deeper understanding of the program has been possible.

The simulator consists of around 450 classes and more than 200.000 lines of code developed in C++. Figure 3.16 reports a general hierarchical scheme of the architecture of SUMO program.

From the scheme it is possible to individuate a series of modules, from top to bottom they are:

- GUI: is the Graphical User Interface module, it controls the microsimulation parameters and deploys them;
- MSNet: this module contains the simulated network, the simulation performer and all the objects related to the microsimulation;
- MSEdgeControl: is the module that stores and manages the edges and the lanes performing the movement of the vehicles;
- MSVehicleControl: is the module that introduces and removes the vehicles from the simulation;
- MSEdge: the detailed module that represents a street connecting two junctions (node) and consisting in one or more lanes;
- MSLane: the module that represents a single lane;
- MSLaneChanger: the module that performes the lane changing of the vehicles;
- MSVehicle: is the class responsible for the representation of a vehicle in the microsimulation describing vehicle type, current speed, lane, angle and other details;
- MSVehicleType: module that includes and manages all the vehicle parameters that can be associated with a vehicle type such as vehicle shape, car-following model, maximum speed, emission class etc;
- MSCFModel: sub module implemented by a car-following model;
- MSRoute: module that describes the vehicle route.

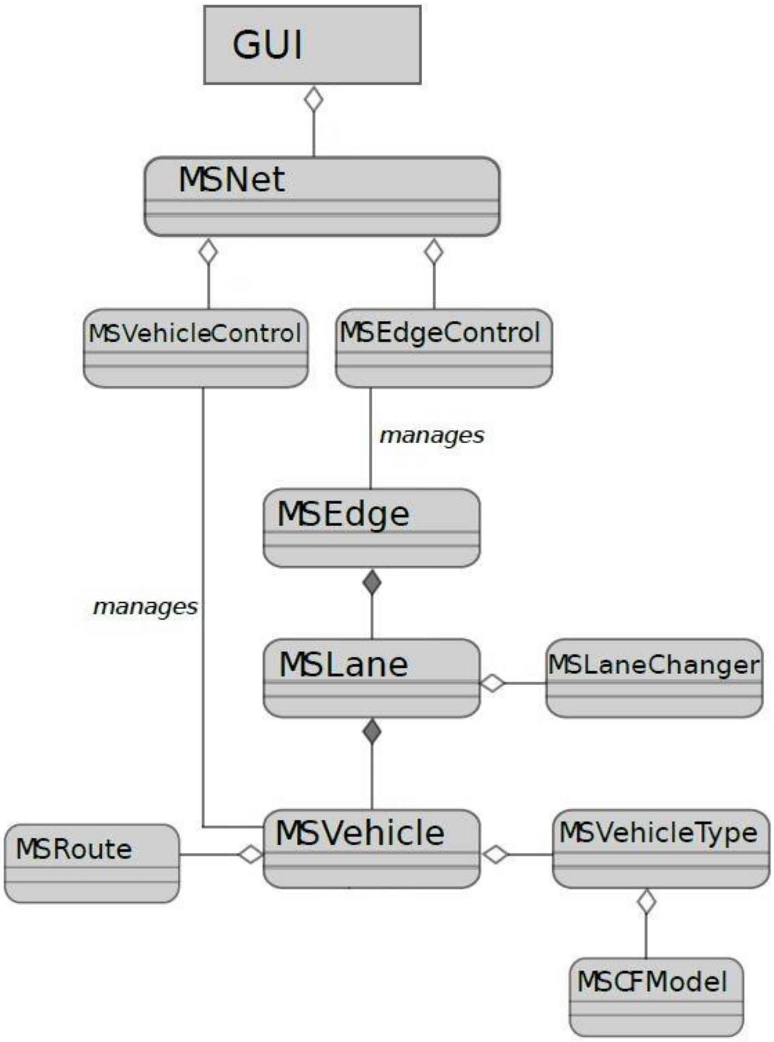

Figure 3.16 Scheme of the architecture of SUMO according to Pereira [55]

As expected, the program revealed itself very complex and mature. It has been concluded that any modification would require a major effort and time consumption.

Based on information collected from other research studies, particularly those described in section 2.6.1 and after a consultancy of the community it has been concluded that the direct modification of SUMO source code would overcomplicate the work and its success would have been too uncertain to be viable. Consequently, in order to generate the simulation the use of TraCI has been accepted and a building of the simulation by steps has been adopted as strategy, for a deeper comprehension to the tool itself.

3.5.2 TraCI steps

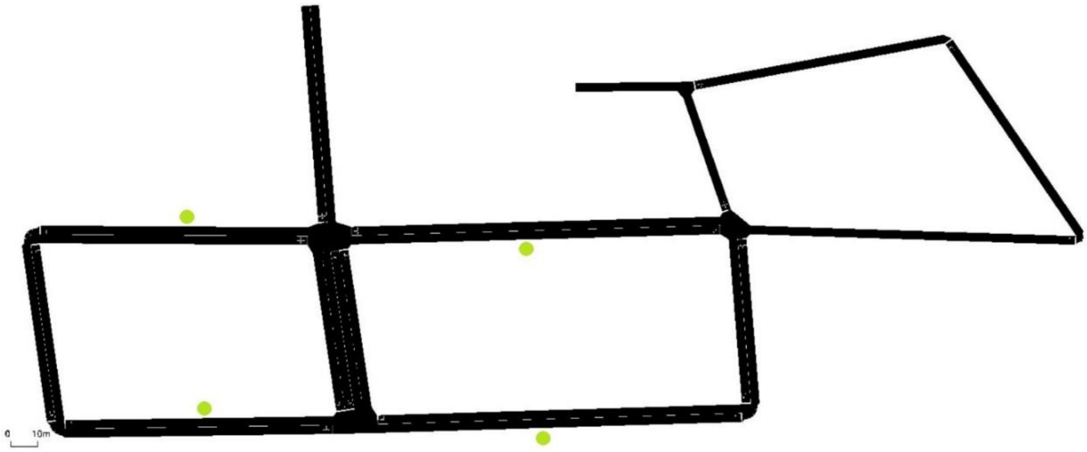

Figure 3.17 Network adopted for the first steps of the scenario

In order to have full control of variables used and to allow a better comprehension of TraCI, a small simple scenario, represented in Figure 3.17 has been adopted. The shape of the net is the result of the necessity of simulating all the possible cases that could cause errors or abortion of the simulation like unconnected roads, or to allow a better observation of the movement of the cars, in order to have an exhaustive comprehension of the effects of the commands. Shape of the basic scenario's network has then evolved through the different steps and the one represented is the final version. Green dots represent the taxi stands, where vehicles park while they are waiting for a new request.

With the small network generated with the use of NETEdit, a python script has been written and integrated with an ever increasing number of functionalities, from the simple insertion and redirection of vehicle up to the final behavior, with the return to the stand waiting for a new request and the possibility of being redirected an indefinite number of times. In particular steps were:

- STEP 1: Insertion of the vehicles through TraCI, investigation of the behavior with the assignment of stops and new targets;
- STEP 2: Insertion of target inputs through a CSV format file that is read and managed by the python script that sends the orders at the right taxi at the right time;
- STEP 3: Implementation of the assignment process, i.e. the detection of the closest free taxi to the customer (starting point of the trip). Implementation of different vehicle states (free, reaching the customer, busy) and of different parking states (side of the road if parked at the stand, on the road if loading or dropping the customer);
- STEP 4: Full working scenario with handling of problems such as unconnected roads or no available taxis at the moment of the request;
- STEP 5: Generation of output files with simulation data (assignments, states and positions over time).

Final result of such a process is explained in detail in dedicated section 3.6.

3.6 Building of the Scenario

After the description of the different input and the preliminary steps, the building of the actual simulation scenario is here described, starting from the preparation of input files such as network and trip demand filtering, up to the working of the python script that would interact with SUMO through TraCI in order to insert inputs and bring the simulation to completion.

3.6.1 Input data preparation

Network

Chosen network has been downloaded in OSM format from BBBike extracts [9] after a brief comparation of maps downloaded from different sources. Once obtained the certainty of their equivalence, the aforementioned website has been judged as the best one for the task since it allows the accurate selection and downloading of a map form Planet.osm [56]. The advantage of such a website is that the area to be selected can be as big as 24,000,000 km^2 and 512 MB of dimension. Area selected is shown in Figure 3.5, it covers an area of 1,150 km^2 large and OSM format file downloaded is 431.4 MB big.

After being extracted, the file has been converted into the input format necessary for SUMO. NETCONVERT is a tool included in SUMO package that allows the conversion from OSM format to XML one. The program is called from the command line by means of Code 3.1.

```
netconvert --osm-files GreaterMunichFreising.osm -o munichmap.net.xml
```

Code 3.1 Command line call for NETCONVERT

```
--geometry.remove --roundabouts.guess --ramps.guess --junctions.join
--tls.guess-signals --tls.discard-simple --tls.join
```

Code 3.2 Conversion options

Table 3.8 List of options used for map conversion and their explanation

Option	**Purpose**
--geometry.remove	simplifies the network keeping the topology unchanged
--roundabouts.guess	sets the right-of-way rules at roundabouts
--ramps.guess	since Acceleration/Deceleration lanes are usually not included in OSM data, this option identifies likely roads that have these additional lanes and adds them
--junctions.join	during the conversion, two edges forming a single street but separated by an obstacle such as a tram line, form to separate junctions when crossing another street. This option allows the automatic merging of these junctions.
--tls.guess-signals --tls.discard-simple --tls.join	traffic lights system optimization of the interpretation of the position of traffic lights, and unification of the controller in case of different traffic lights inserted in two adjacent junctions.

First converted file's dimensions were of 970 MB. In order to automatize some standard fixings of the map and reduce its dimensions, options listed in Code 3.2 have been used. Table 3.8 list what is the purpose of each option.

With the implementation of these options, dimensions of the file reduced to 895 MB. In addition, the conversion from OSM to XML allows the conservation of geocoordinates information so the final map contains both an internal xy coordinate system and geocoordinates.

A detail of the network obtained is illustrated in Figure 3.18.

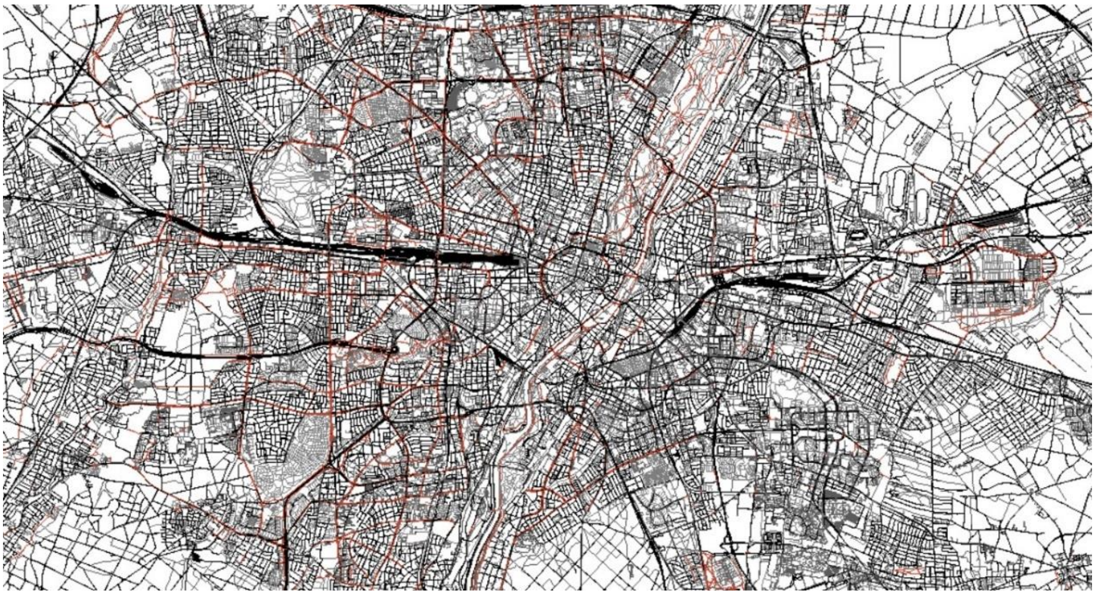

Figure 3.18 Central part of Munich right after conversion from OSM visualized in SUMO-GUI

From the figure it is possible to notice the presence in the map of railways (bolder black lines), cycling paths (red lines) and walking paths (grey lines), particularly evident in correspondence of the parks. These roads are not used in the simulation since only cars are to be inserted, for this reason, in order to reduce file dimensions they have been eliminated. The map has then been edited with the use of NetEdit. The program allows different modes of selection of edges and junctions also depending on some aggregated characteristics such as speed limit higher than certain values or type of edges.

In order to understand what were the types of edge to be eliminated the XML file has been read in order to find the list of kinds of edges and the vehicles allowed in them. XML file's content is shown in Code 3.3, Code 3.4, Code 3.5, Code 3.6 and Code 3.7.

```
<type id="…" priority="…" numLanes="…" speed="…" allow="…" disallow="…" oneway="…"
width="…"/>
```

Code 3.3 Type of edge representation in SUMO net.xml file

At the top of the code, there is a list of types of edge present in the map. From the reading of these lines, at the tag "allow" or "disallow" (the presence of these tags is mutually exclusive), it is possible to understand which kinds of edges allow vehicles and which of them do not being dedicated to pedestrians, bicycles, trams or trains. The "id" corresponds to the name that is used by NetEdit for the multiple selection.

```
<edge id="…." function="….">
        <lane id="…." index="…." disallow="…." speed="…." length="…." shape="…."/>
</edge>
```

Code 3.4 Edge and lane representation in SUMO net.xml file

Follows Code 3.4: a list of all the edges contained in the map with specific information with the possible presence of repetition of some tags to specify exceptions with respect to information provided in the edge type description.

```
<tlLogic id="…." type="…." programID="…." offset="….">
        <phase duration="…." state="GG"/>
        <phase duration="…." state="yy"/>
        <phase duration="…." state="rr"/>
</tlLogic>
```

Code 3.5 Traffic light logic representation in SUMO net.xml file

```
<junction id="…." type="…." x="…." y="…." incLanes="…." intLanes="…." shape="….">
        <request index="…." response="…." foes="…." cont="…."/>
        <request index="…." response="…." foes="…." cont="…."/>
</junction>
```

Code 3.6 Junction representation in SUMO net.xml file

```
<connection  from="…."  to="…."  fromLane="…."  toLane="…."  via="…."  dir="…."
state="…."/>
```

Code 3.7 Connection representation in SUMO net.xml file

After edge list, other descriptions are contained such as the traffic light logic Code 3.5, the junctions Code 3.6 and the connections Code 3.7 i.e. the description of which outgoing lanes can be reached from an incoming lane.

In the selected map the traffic light logic is of the static kind since no information is available about it.

After the elimination of the edges that were not useful for the simulation many correction has been necessary due to the appearance of some unconnected edges. Some edges that allowed the transition of cars were isolated from the other or connected to the rest of the network by some roads that for some reason were recorded as one of the types that do not allow the passage of vehicles as shown in Figure 3.19.

The elimination of all the pedestrian and cycling edges made the presence of these unconnected edges more evident. Due to the fact that the trips demand input preparation was running parallel to the network preparation, none of the edges that allowed vehicle transit could be eliminated since there was the possibility of being assigned as the starting or destination point of a trip. Therefore, these have been reconnected with the rest of the network, first with a visual inspection of the map, secondly with the generation of a large quantity of cars following different sample trips along the network. The larger the quantity of trips used for this operation, the higher the possibility to cover the whole network: every time an unconnected edge was present the simulation sent an error stating that the route connecting two edges could not exist providing also the two edges that could not be linked. This way, a further fixing of the network has been possible. Figure 3.20 shows an example of unconnected edge with consequent corrective action: the relevance of the actual connection to the network and its length is of negligible importance due to the large scale of the scenario, no comparison with real layout has then been considered necessary.

Figure 3.19 Example of error in the assignment of the type to some edges

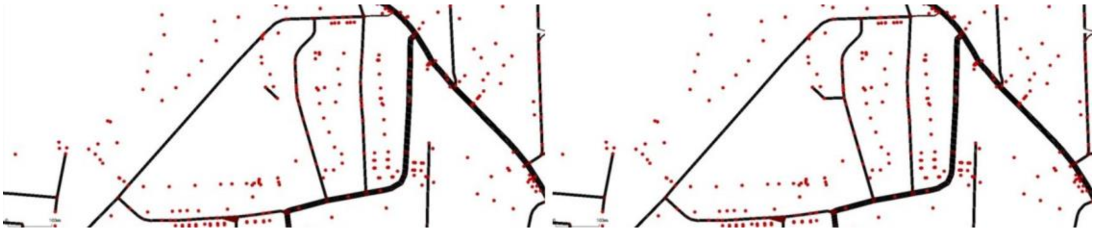

Figure 3.20 Example of reconnection of isolated edges in the map

The elimination of not-for-vehicle edges has left a large quantity of unconnected nodes, visible ads red spots in Figure 3.20. These have been eliminated thanks to the presence of an automatic function activated pressing F6 button of the keyboard.

Finally, the working network represented in Figure 3.21 has been obtained. Final file's dimension after all the cuttings and the correction is of 718 MB.

Performing some checks inserting trips and analyzing data such as distances covered and times needed, and comparing them with inputs, anomalous speeds have been detected between the city and the airport. The use of NetEdit allowed to find the cause: highways in the scenario had speed limits of 200 m/s to simulate the absence of limits. The simulation regulates the vehicle speed's according to the present limits, if no limit is present the vehicle reaches its maximum speed, leading to an unrealistic behavior since most of the vehicles in reality tend not to reach the maximum speed in absence of limit but tend to keep a reasonable cruise speed. Moreover, a speed limit in the highways included in the network is actually present, so this has been reduced to 36.11 m/s, i.e. 130 km/h. Figure 3.22 shows a simplified version of the simulation network that highlights the highways in which the speed limit has been reduced.

Figure 3.21 Overview of the final network visualized on SUMO-GUI

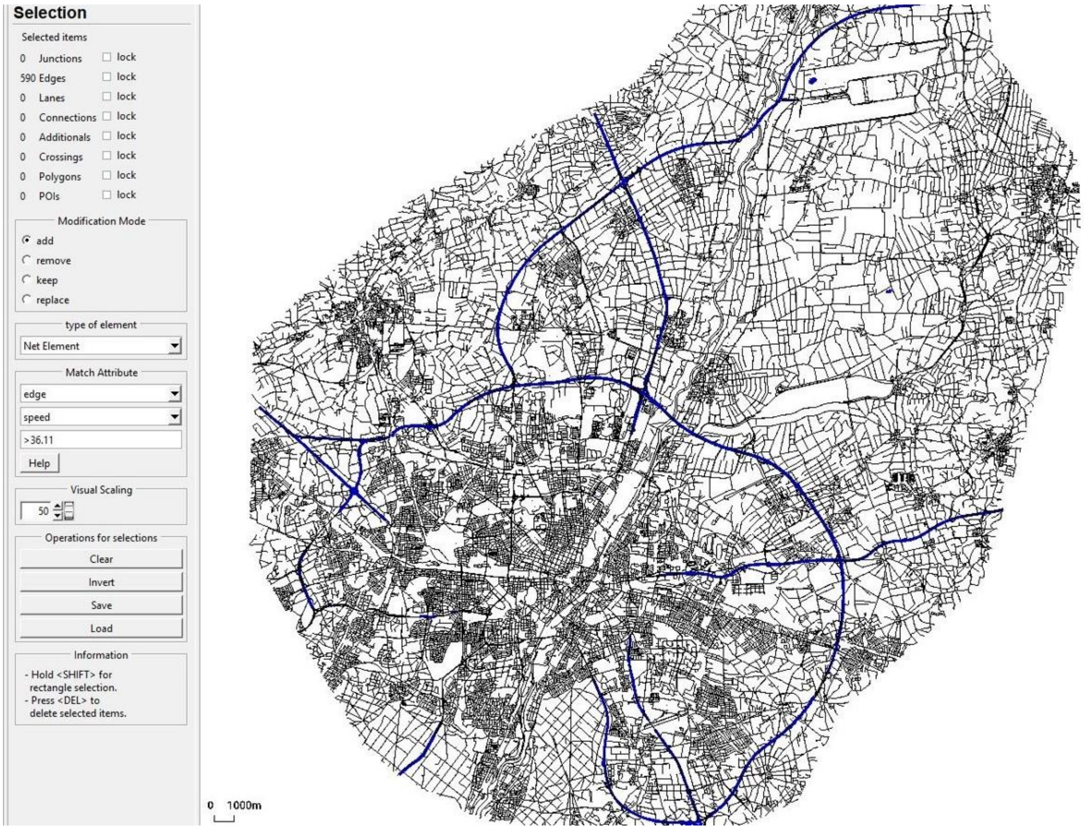

Figure 3.22 Visualization of roads that showed an unrealistic speed limit

Trip demand

Trip demand file has been obtained from Wittmann's work in CSV format. This file expresses starting and destination points in terms of geocoordinates, whose values are not readable neither by SUMO nor by TraCI. It has then been necessary to write a python tool that by reading the map would find the point identified by the geocoordinates and associate it to an edge of the map. Edge IDs are the only acceptable input for SUMO and being them assigned by program's rules during the conversion from OSM, it is a common problem not to have the input in the right format. For this reason, in the user guide a precompiled python procedure that enables the user to locate nearby edges based on the geocoordinates is provided. This procedure has been integrated in a tool that opens the input file and the network, defines an initial radius of 10 m around the spot identified by the coordinates and looks for an edge to associate, if no edges are found, radius is increased of 10 m more up to a limit of 200 m. This solution is necessary because there is not absolute precision in the map and some positions might be slightly different. The limit of 200 m is necessary to avoid infinite loops and to exclude those rare trips that have start or destination out of the considered network.

After this first treatment a second tool has been built in order to perform a first filtering. If in the file, the same edge has been assigned both as start and destination, it causes not only an unrealistic request, but also an error to the simulation that would crash in presence of such trips. The reason of the existence of these trips can be attributable to the combination of two factors: a very short trip and some imprecisions in the map that caused the radius of research of an edge to increase enough to spot the same edge for start and destination. After these two processes, 1.08% of the trips have been removed.

Due to some more crashes of the simulation caused by errors related to very short trips, a second filter has been necessary. The python code of the TraCI simulation, in the case a too short edge (below 10 meters) is either the start or the destination of a journey, substitutes it with the following (in the case it is the start) or the previous (in the case it is a destination) edge. If the route is 2 edges short an error occurs because the following or previous edge is the destination or the origin of the route, causing the same error that required the first filtering, with the consequent abortion of the whole simulation. A second filter that spots all these cases has been written and a further reduction of the 0.02% of the trips has been caused.

Taxi stands

Along with the trip demand, the list of taxi stands with relative capacity and position has been obtained. A total of 1,543 parking lots are distributed among 216 station; Figure 3.23 shows their position on the map with Figure 3.24 zooming in the central part of the city. Positions for each of the 216 stations was expressed in geocoordinates. Another tool has then been written in order to associate an edge to each stand.

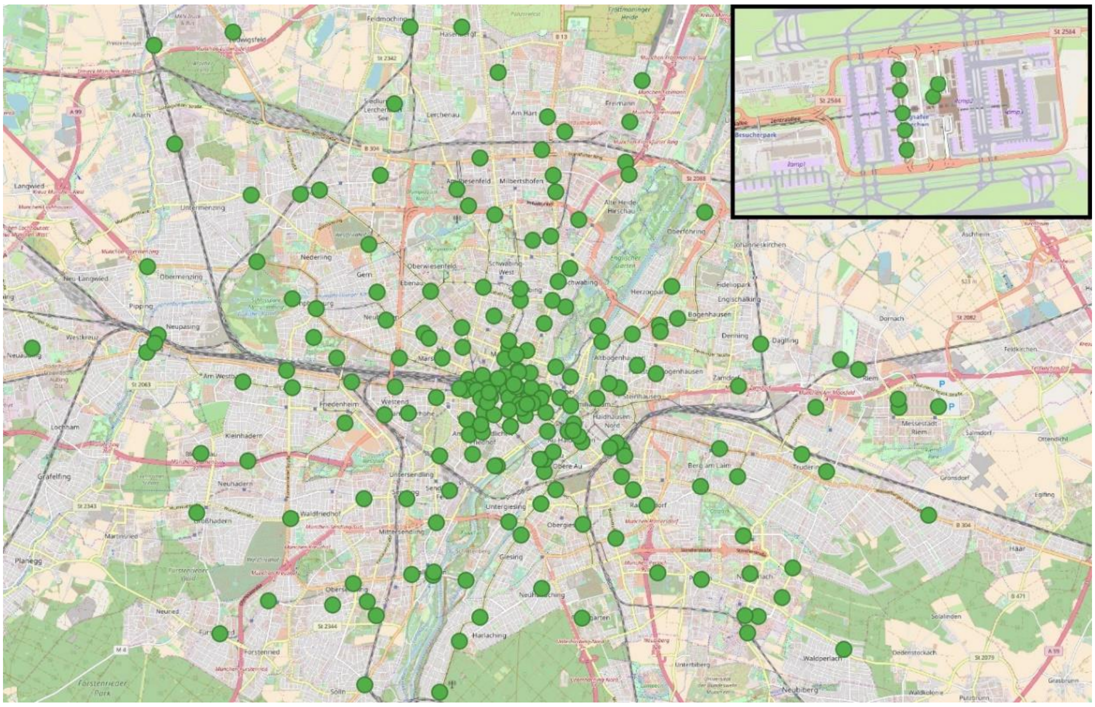

Figure 3.23 Positions of the stands in the city of Munich, airport zone in top right corner

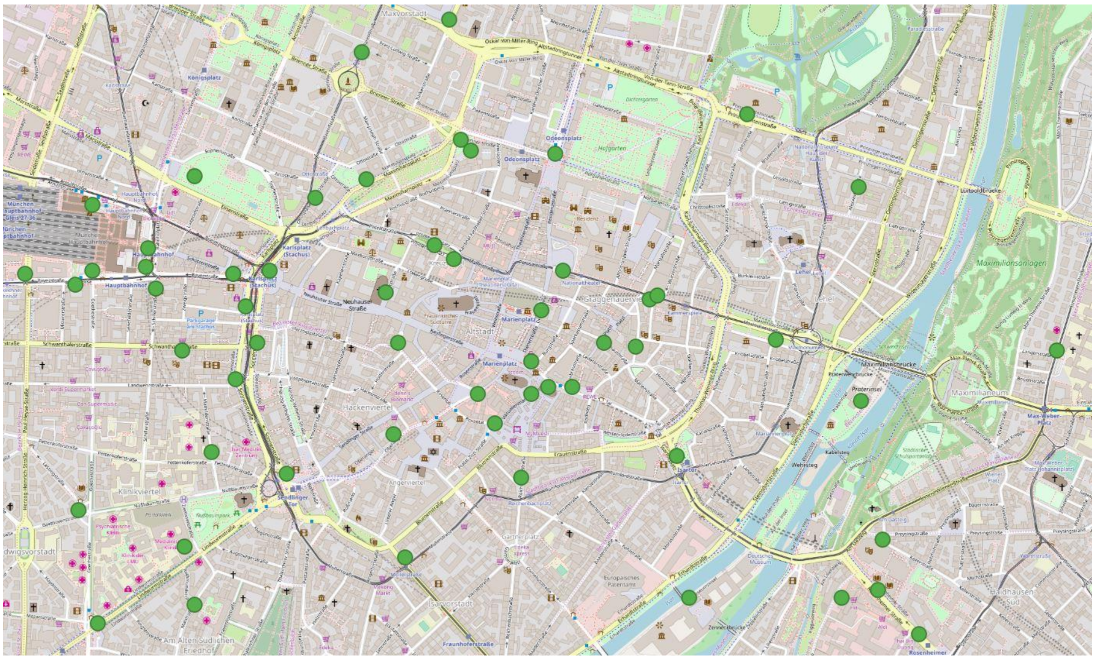

Figure 3.24 Taxi stands in the central part of Munich

3.6.2 Python script of the simulation

Final script is a python file named runner.py containing 425 lines of code. The file opens SUMO application and interacts online with the simulation, starting it and also stopping it. In order to do so, the file necessitates the presence in the same folder of the network file (Munichmap.net.xml) and configuration file (bus.sumocfg) containing the list of the files used for the basic simulation without TraCI intervention. The configuration file in turn requires the presence of all the files listed in it, namely:

- The net file: same file required by runner.py
- The route file (bus.rou.xml): it contains a two edge route assigned to two busses at the very beginning of the simulation. The first vehicles are inserted by TraCI at step 1 (a simulation step corresponds to a second) but if no vehicle is present at step 0 the simulation aborts immediately since it is supposed to stop as soon as all the vehicles have left the simulation. The busses leave the simulation as soon as they reach the end of the second edge of their route, this happens after 11 simulation steps.
- The additional file (bus.add.xml): this file contains the characteristics of the vehicles inserted in the simulation by TraCI, i.e. the taxis. Being there 4 possible states that a taxi can assume, 4 different descriptions exist, they differ only for the name and the color associated to them. Table 3.9 lists all the characteristics of the taxi vehicles, the values correspond SUMO's default values associated to a passenger car [57], the only exception to these values is the vehicle length, set as an average value obtained comparing different C-segment cars; the value of length of the car is of negligible influence to the results of the simulation. Table 3.10 instead reports the four different states that can be assumed by a taxi with associated color in the GUI.

Table 3.9 Description of taxis in additional SUMO file

Characteristic	Unit	Value
Type		passenger
Length	m	4.3
Acceleration	m/s^2	2.6
Deceleration	m/s^2	4.5
Maximum speed	m/s	55.0
Sigma (drivers imperfection)	-	0.5
GUI Shape		passenger/hatchback

Table 3.10 Different possible states of a taxi

Name of the state	Associated Color
taxi_at_the_station	Yellow
taxi_going_to_customer	Light Blue
taxi_busy_with_customer	Red
taxi_going_to_station	Green

Sigma value is the car following model parameter that represents driver's imperfection in driving. This could be the discriminant between a human driver and an AV driver being the latter more able to keep a constant speed. A study by François Vaudrin, Jakob Erdmann and Laurence Capus reported in SUMO Proceedings 2017 about the "Impact of Autonomous Vehicles in an Urban Environment Controlled by Static Traffic Lights System" [58, p. 82] reported that this factor was of no influence to the simulation results. For this reason no changes have been applied to this value.

Preliminary operations

In order to run the simulation, runner.py must first perform some operations in order to load the necessary files and open SUMO.

First lines of the code are dedicated to the import of the necessary libraries: TraCI for the python commands to the simulation, CSV for the handling of input and output files or the python modules for the loading of the configuration.

Secondly some lines of instruction are dedicated to the preparation of output CSV files, writing the tags, i.e. the first line that specifies the contents of each column.

After these instructions, the input files are loaded, namely the file containing the demand and the file containing positions and capacities of taxi stands. Last input inserted is the map file that is opened and read by the script: values of length of each of the edges is stored in a dictionary generated in this phase.

Finally, main is started with the collection of the options to start sumo with or without GUI and with the loading of SUMO configuration. With the configuration open, trip demand data are stored in lists and stations are inserted in the network with the information about their capacity stored in a dictionary. Eventually at simulation step 0 all the vehicles are inserted to the simulation, evenly distributed among the stands. Figure 3.25 depicts a scheme that summarized all the preliminary operations.

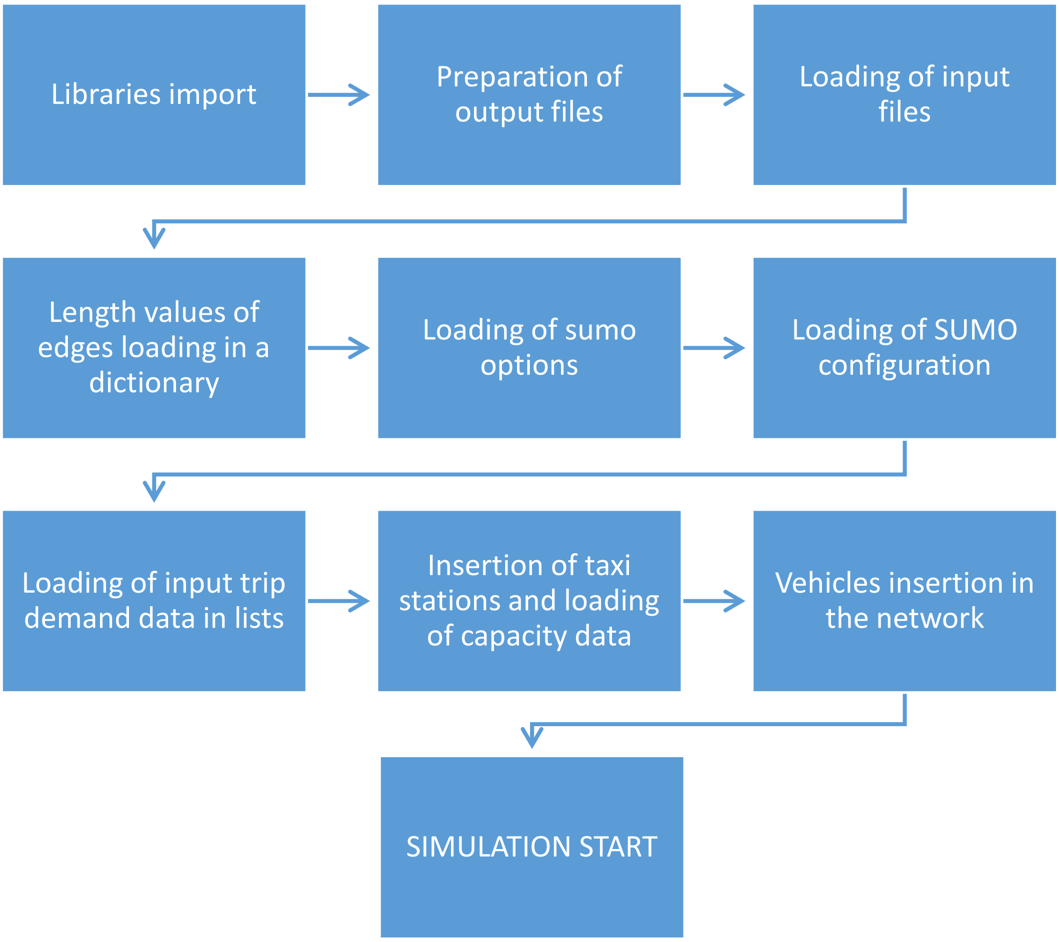

Figure 3.25 Sequence of preliminary operations performed by the script

Simulation mechanism

Each value of time step of the simulation, i.e. each second, as the simulation proceeds, is compared with the list of assignment timesteps created from the reading of the input file. If a correspondence is found, i.e. there is a trip request at the same timestep of the simulation, a parsing of available taxis around the requested starting point is performed. Taxis available are not only those waiting at the stand (yellow color), but also the ones that, once dropped the customer are directed towards a taxi stand to park (green color). Each taxi in the simulation is checked whether it is in one of the two mentioned states, if so, its position is retrieved with a TraCI command.

The distance from the requested starting point for each taxi examined is measured as the crow flies. If this value is lower than a given radius of research, the travel time for that taxi is calculated and the car-id and travel times are stored in a list. Travel time can be retrieved with a python TraCI command: it is the estimated time needed to reach the point in which the customer is, based on the route to be covered, lengths of the edges in this route and relative average speeds. The searching radius is initially set to 400 m, if no car is found, this is increased of 400 m more until a limit of 50 km is reached. Initially a limit radius of 6 km has been set but first results of the simulation demonstrated that this was not wide enough and that many trip requests remained unsatisfied until a car by chance passed close enough to be assigned. Different attempts with ever increasing limit radius did not solve completely the problem so 50 km has been set. Such value guarantees that the whole map is covered in extreme cases. If no car available is found after this operation it means that all the taxis in the network are busy with other customers and the trip request is postponed of one minute. Postponement is repeated until a taxi is assigned to that trip.

Once a list of candidate taxis is obtained, the one corresponding to the minimum travel time from its current position to the customer is found and selected. A TraCI command is sent to recover the vehicle from its parking state and another assigns the edge where the customer is, as its new target. Taxi is also ordered to stop at this target edge to load the customer. At the same time, information containing customer destination is stored in a dictionary, associating the destination edge with the car-id. The car then is set in the “going_towards_customer” state (blue color) and starts its trip. Figure 3.26 schematizes the assignment procedure.

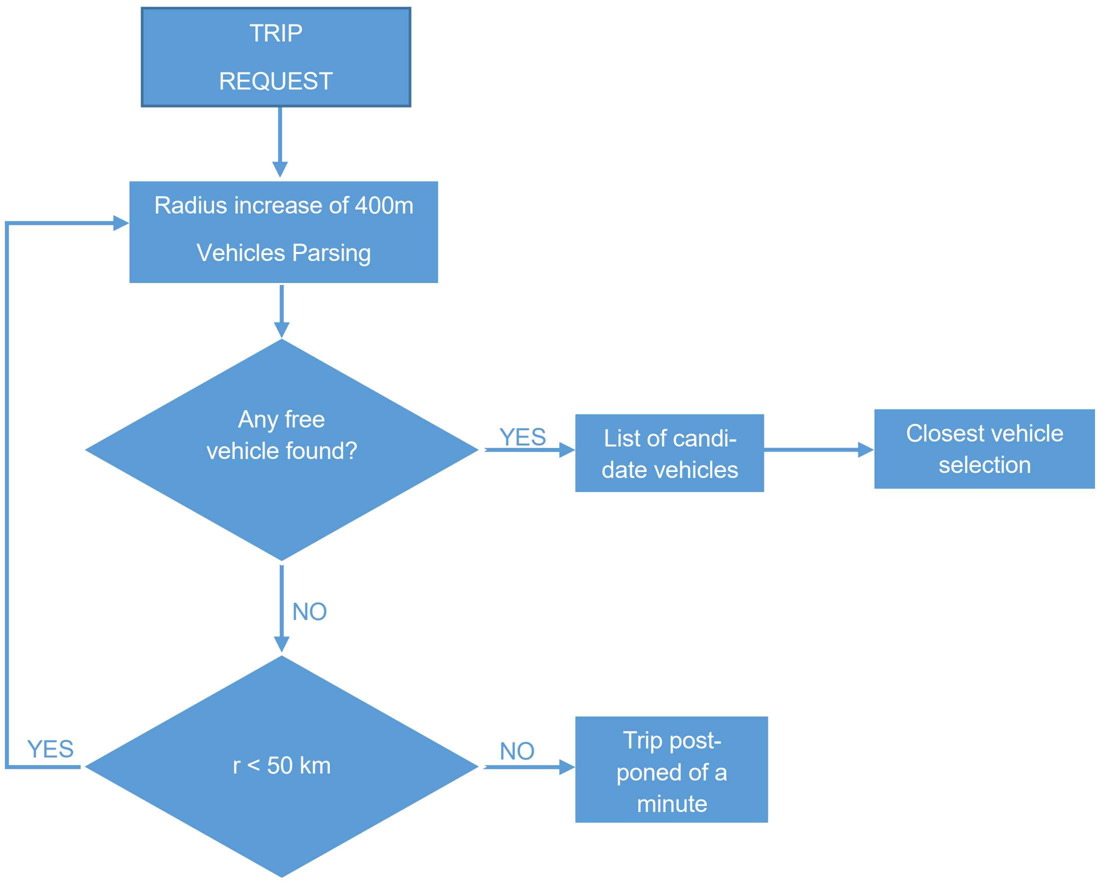

Figure 3.26 Scheme of assignment process

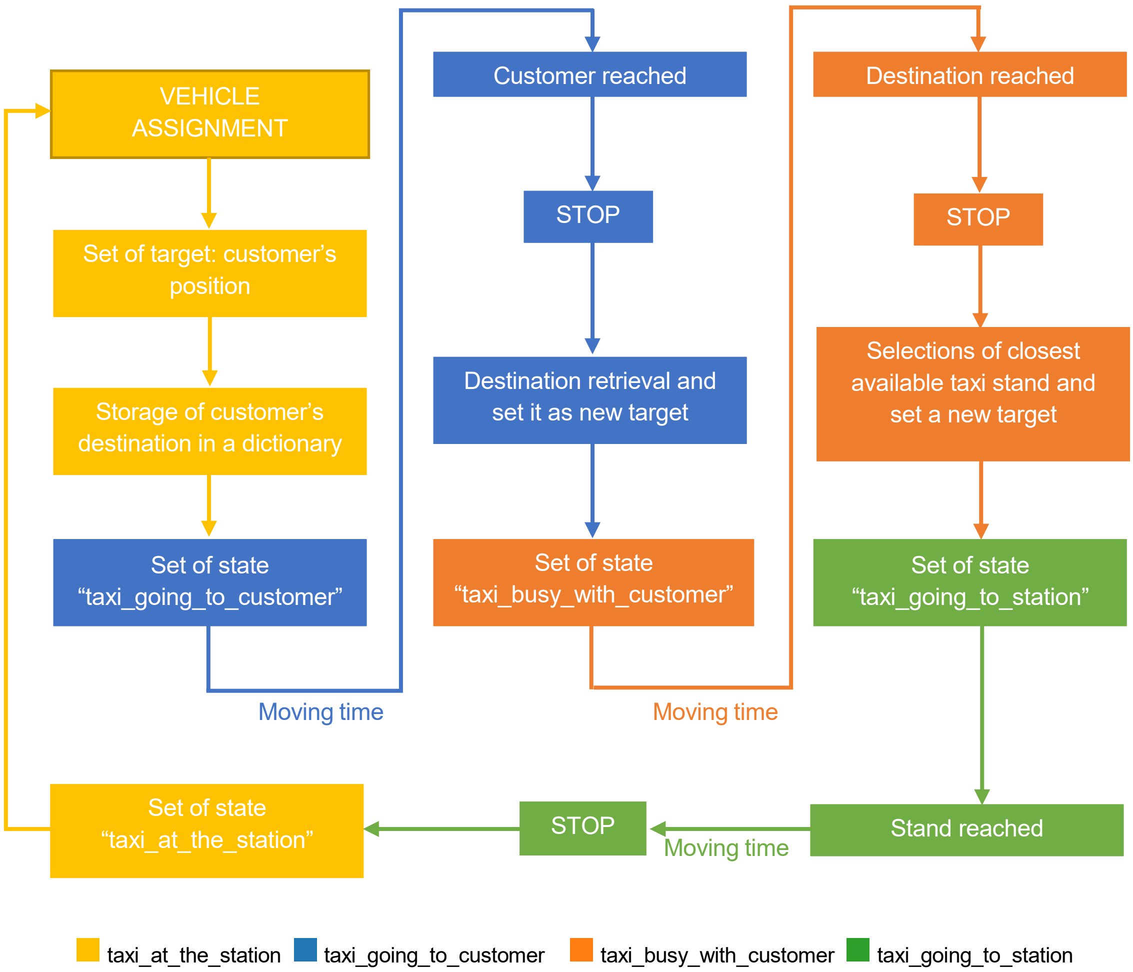

Figure 3.27 Scheme of the sequence of states if no trip is assigned before taxi reaches the stand

The length of the edges is a critical factor since the position where to stop the vehicle must be provided. This is calculated as the middle point of the edge to avoid to stop at the very beginning of the edge, thus not congesting the previous junction. This system avoids also that the stopping position is beyond the edge end since such situation would cause an error. Values of the lengths of each edge can be quickly retrieved from the dictionary created in the preliminary steps (see section 3.6.2 – Preliminary Operations). In the case an edge is too short, some errors related to the minimum length that can be read by SUMO, can appear. In order to give more robustness to the simulation, if an edge is shorter than 10 meters, another one close to it is selected for the stop: the following one in the route in the case it is the starting point of the trip, the previous one in the case it is the destination edge. This requires the second filtering of the trip demand file described in section 3.6.1.

Once the taxi in the "taxi_going_to_customer" state stops to pick the customer up, it is detected by the program through checks on the vehicles performed every second. Value of destination edge previously stored is retrieved and set as a new target. State "taxi_busy_with_customer" is activated and the trip starts.

Checks performed on the vehicles detect also the moment in which the customer is dropped. At this point the taxis station are parsed and the ones that have available places are added to a list for which travel time is estimated. Once the closest station is selected, the taxi is turned into "taxi_going_to_station" state and sent there. This new state makes it available for new trip assignments. In order to save some time, after 20 stations reachable within 400 seconds have been added to the list, parsing stops.

Script alternatives

Two script alternatives have been elaborated in order to compare the results and to select the fastest one. The other solution sets the customer's destination directly as the vehicle target instead of storing the value in a dictionary. The vehicle is then ordered to pass via the edge the customer is and make a stop there before reaching the target. This solution requires an instruction called rerouteTravelTime that makes the recalculation of the route after the via is set. The instruction is illustrated in Code 3.8. If currentTravelTime option is set to True, it means that the rerouting calculation will take into account the current traffic conditions in order to detect the fastest route, otherwise it will base its choice only on speed limits and priorities of the streets as done in the final script used. The activation of the option causes a major slowdown to the simulation making it 4-5 times slower than real time. Due to the dimension of the map, the relatively small number of vehicles inserted is not causing any congestion, after the comparisons of performance and results of the two script alternatives and a consultancy of SUMO developers community, this solution has been discarded. Script described in section 3.6.2 has also proved to be leaner from the coding point of view.

```
TraCI.vehicle.rerouteTravelTime(self, vehID, currentTravelTimes=True)
```

Code 3.8 route recalculation command

3.6.3 Output recording and elaboration

TraCI scripts compiles five different files:

- Assignment file: is the record of all the trips that are assigned, to each trip is associated a car ID and a step of assignment;
- Available taxis: record of the vehicle counts, listing the number of vehicles per state;
- EdgesTowardsCustomer: record of every edge covered each second by each car in the state "taxi_going_towards_customer";
- EdgesBusyWithCustomer: record of every edge covered each second by each car in the state "taxi_busy_with_customer";
- EdgesToStation: record of every edge covered each second by each car in the state "taxi_going_to_station".

The purpose of these files was to record every possible useful data and to keep the recording as simple as possible in order not to be cause of slow downs in the simulation, even if this implied a higher complexity of elaboration tools.

With the use of ad hoc python tools, information is analyzed in order to obtain information about the percentages of vehicles in time, trips assigned to each car, distances covered to reach customers for each trip, distances covered to bring each customer to destination, times required for these operations, total of kilometers covered by each car and total of trips assigned to each of them. This information has then been analyzed and results are presented in chapter 4.

3.6.4 Traffic limitations

In section 2.4, difficulties and criticalities linked to the aspect of the traffic in big scenarios have been described. In this section, limitations that made the insertion of traffic impossible in the simulation are briefly described.

Some private traffic input data were available from Agua's work [39]. The file containing the traffic data made use of geocoordinates to define origin and destination points of each trip, so a tool similar to the one described in section 3.6.1, has been written. The quantity of trips is of 1,061,185 so conversion of geocoordinates has been very time expensive, taking three weeks to provide a complete output. These data have been used for the checks of the network integrity described in section 3.6.1 but have not been considered reliable enough for their use in traffic preparation. The reason is that they consider only trips with origin and destination within the city of Munich, discarding any other trip with either start or destination outside of the city. In addition to the fact that information is not complete, no traffic would have been generated out of the city, for example between the city and the airport.

An alternative solution for input data generation would have been the use of SUMO tool ActivityGen [59] that generates a list of routes specifying origin and destination for each of them. The problem with the use of such a tool is again a lack of input data: the program takes as input a file containing very detailed statistical information about the network such as population, working hours, schools' positions and times etc. Population and Activity distribution must be described at street level and such information was not available [60].

Another limitation is given by the fact that origin and destination for each trips must have been elaborated successively by another SUMO tool called DuaRouter [61]. This program compiles a SUMO input file with the list of edges to be covered by each car. The issue is that DuaRouter considers each car as it was the only one in the map and the result is that the output file for a city is very congested as no car makes changes to its route due to the presence of traffic. To optimize traffic, a python script named duaIterate.py exists [62]. This script iterates DuaRouter many times until a certain fluidity in the traffic is obtained and a distribution of different routes per trip with related probability is obtained as output. The limitation is that the program needs a number of iterations which cannot be stated in advance, before it produces an acceptable output. Moreover, in city scenarios congestion in the first iterations is so high that the program does not find an optimization direction quickly. Taking an iteration of the Agua's traffic file around 17 hours, this has not been considered as a viable solution.

Finally, one last issue is related to simulation time: also in the simulation each taxi is directed to its destination as it was the only one in the map. To make it calculate the fastest route based on traffic each time, the option RerouteTravelTime described in section 3.6.2, must be activated, bringing the simulation time of a 24h scenario from around 15 hours to 4-5 days.

4 Results

A series of 16 python tools have been developed for the data analysis for a total of around 2,300 lines of code. First operation to be performed is to elaborate output data in order to obtain start and end time, duration and distance covered for every single trip and vehicle state.

Other measures obtained from the data elaboration with python scripts are the distributions over time of taxis available by means of 100% stacked charts, waiting times and ability to satisfy the demand.

Before any investigation of performance of the different cases (fleets and days), a validation of the simulation has been necessary. Input data provided not only time of picking of the customer and coordinates for the origin and destination of each trip, they also provided distance covered and time needed to bring the customer to destination. This way a comparison has been made available in order to validate the simulation. It has to be taken into account that input data are characterized by the presence of traffic.

4.1 Simulation evaluation

Input data did not contain any information about times and distances covered in the approaching of the vehicle to the customers, for this reason evaluations of times and quality of the simulation have been made considering a 24 h scenario with a fleet of 428 taxis that corresponds to the number of vehicles connected for the gathering of input data. Only the trips with customers on board are compared and fleet size and taxis distributions do not have any influence on this output.

Codes comparison

First comparisons performed, have been among the different codes and between the codes and the input. The three codes compared are identified with the names “script 6 rerouteTravelTimeTrue” and “script 6 rerouteTravelTimeFalse” for the alternative codes mentioned in section 3.6.2, and “script 7” for the solution adopted and described in the same section. The difference in the values of duration and distance of the trips with the costumer on board has been calculated trip by trip, the values reported in Table 4.1 consists in the averages of all the single tip differences with the values provided in input.

Table 4.2 instead reports the aggregated data concerning the time needed to reach the customers and the kilometres covered in this phase. Total time consists in the amount of hours obtained summing all the time spent for each trip by a car to reach the customer from the position they were at the moment of assignment. Total distance instead is the corresponding value of kilometers covered in this phase.

Table 4.1 Summary of comparisons of the alternative scripts (input data are the reference)

Comparison	Data compared	Average difference
Input – Script 6 rerouteTravelTimeTrue	Time	-37.3%
	Distance	-3.6%
Input – Script 6 rerouteTravelTimeFalse	Time	-37.8%
	Distance	-3.8%
Input – Script 7	Time	-37.7%
	Distance	-3.9%

Table 4.2 Aggregated data concerning the waiting time

Script version	Data	Unit	Value
Script 6 rerouteTravelTimeTrue	Total Cumulated Driving Time	hrs	606.1
	Total Cumulated Driving Distance	km	25,062
Script 6 rerouteTravelTimeFalse	Total Cumulated Driving Time	hrs	588.2
	Total Cumulated Driving Distance	km	24,279
Script 7	Total Cumulated Driving Time	hrs	584.1
	Total Cumulated Driving Distance	km	23,846

In view of these results, the solutions can be considered as equivalent in terms of results. As already mentioned in section 3.6.2, the code “script 6 rerouteTravelTimeTrue” has been discarded due to the long times needed for the simulations. Results confirm the fact that no particular influence can be attributed to this option.

Sampling interval

Once identified the code concept to be adopted, a brief analysis has been dedicated to sampling interval in the attempt to further decrease the simulation time. Instead of recording the positions of each car each second, the operation has been performed every five seconds. This solution led to no gain in time and to a high loss of information as Table 4.3 shows. The loss of information is due to the fact that there are a lot of short edges and junctions for which the car necessitates very few seconds to cross them (in many cases 1 second is enough since these can be part of main roads characterized by sustained average speeds) so these are not recorded in the output file. Measuring has then been performed every second.

Table 4.3 Information loss collecting data every 5 seconds instead of every second

Data	Loss
Aggregated time to reach the customer	3.8%
Aggregated distance to reach the customer	23.4%
Aggregated time busy with the customer	1.2%
Aggregated distance busy with the customer	20.0%

4.2 Wednesday with unlimited stand capacity

First scenario simulation has been run with unlimited stand capacity. Taxis have been distributed evenly among the 216 stations, for this reason, taxis inserted are not exact multiples of 100. Fleets tested in this scenario are: 107 – 161 – 214 – 321 – 428 and 535. 100% stacked area chart have been calculated and reported in Figure 4.2, Figure 4.3, Figure 4.4 for the first three fleet cases. The charts describe the percentage of vehicles distributed in time among the different states reported in the legend. For the bigger fleets, it is necessary to consult Appendix A: the low utilization of the fleet appears evident, with a major percentage of the taxis waiting at the stand (blue state). This provides an immediate understanding of the excessive dimensions of these fleets for the task. For this reasons and in order to keep graphs more readable, data concerning bigger fleet cases are not always reported in the following analysis.

Table 4.4 and Table 4.5 compare respectively the average difference in times and distances covered to bring the customer to destination, and aggregated times and distances covered to reach the customers, in a similar way of Table 4.1 and Table 4.2. The case with 428 taxi fleet has been considered as reference since it guarantees that no saturation of the fleet is reached.

Among the data recorded, some trips are characterized by 0 distance covered with the customer on board, these trips have been defined as faulty trips. During the simulation the program fails to detect the position of the vehicle for some seconds: if these failures last a second or few the reason is attributable to the fact that if the vehicle is between two edges, SUMO is not able to provide a position. The reason of the longer failures instead, have not been completely understood. In order to compensate for these errors, in the output analysis the possibility to accept a maximum of 30 seconds of delay in the start of recording have been considered. This solution limits the number of faulty trips, particular attention has been paid to the fact that it would stay as low as possible. The percentage of faulty trips in 107 fleet case is of the 5.7% of the total while in all the other cases it ranges between 4.6% and 5.0%. Since faulty trips percentages have decreased in the following simulation scenarios, these figures have been considered acceptable.

Table 4.4 Average differences in times and distances concerning trips with customer on board

Comparison	Data compared	Average difference
428 fleet – 107 fleet	Time	1.1%
	Distance	0.0%
428 fleet – 161 fleet	Time	-0.2%
	Distance	0.0%
428 fleet – 214 fleet	Time	-0.4%
	Distance	0.0%
428 fleet – 321 fleet	Time	-0.2%
	Distance	-0.2%

In Table 4.5, it is also possible to notice a decreasing trend in distances and times needed for the taxis to reach the customer position with a very important difference between the 107 fleet case and all the others.

Figure 4.1 shows the trip request every 15 minutes, represented with the black dots and the curves of trips satisfied by each fleet. All the curves with the exception of the 107 fleet case, pass through almost all the black dots and are superimposed, so only the green curve (321 fleet) is visible. Figure 4.2, Figure 4.3 and Figure 4.4 describe the fleet activity evolution in time by showing the percentages of vehicles unavailable, specifying which of them are going to the customers (green) or busy with them (red), and the percentages of available ones specifying which are waiting at the stand (blue) or which instead are going to one (orange). From the graphs it is possible to see that the 107 taxi fleet is the only one that reaches saturation with a percentage of vehicles available for new trips assignments very close to or at 0% during rush hours and in the evening: in the morning between 8:00 and 9:30, in the afternoon for a briefer period around 18:00 and at night between 23:00 and 00:00. During the day instead, the percentage of available vehicles fluctuates between 20% to 60%. Fleet saturation is the reason why its demand satisfaction curve does not coincide with the others in Figure 4.1. For the other fleets the occupation pattern is very similar to the smaller one: at night, before the morning rush hour, all the fleets tend to cumulate at the stands until more than 95% of the fleet is waiting at the stand at 4:00. The differences between the smallest and the bigger fleets is that maximum occupation reaches 80% at 8:00 and at 23:00 for the 161 fleet and 60% at the same times for the 214 fleet. 18:00 rush hour is less demanding and occupation at this time is around 65% for the 161 fleet and 55% for the 214 fleet. Maximum occupation times are much shorter for the two bigger fleets compared to the smaller one due to the fact that trip postponement extends the peak demand period to the moment in which all the orders have been satisfied. The portion of taxis on the way to the customer (green) grows in consistency as the demand grows until it even overcomes the percentage of taxis busy bringing the customer to destination (red) during peaks.

Some trips present in the demand have not been executed due to some residual incompleteness in the map. The number of unexecuted trips is 9 and represents the 0.13% of the total demand in the 28 hours considered.

Table 4.5 Aggregated comparison of time and distances characterizing the trips necessary for the taxis to reach customer position

Fleet case	**Data**	**Unit**	**Value**	**Comparison with 428 fleet case**
107 taxi fleet	Total Cumulated Driving Time	hrs	712.4	18.6%
	Total Cumulated Driving Distance	km	30,798	29.3%
161 taxi fleet	Total Cumulated Driving Time	hrs	619.0	3.0%
	Total Cumulated Driving Distance	km	26,360	10.7%
214 taxi fleet	Total Cumulated Driving Time	hrs	638.8	6.3%
	Total Cumulated Driving Distance	km	27,300	14.6%
321 taxi fleet	Total Cumulated Driving Time	hrs	634.3	5.6%
	Total Cumulated Driving Distance	km	26,939	13.1%
428 taxi fleet	Total Cumulated Driving Time	hrs	600.7	0%
	Total Cumulated Driving Distance	km	23,814	0%

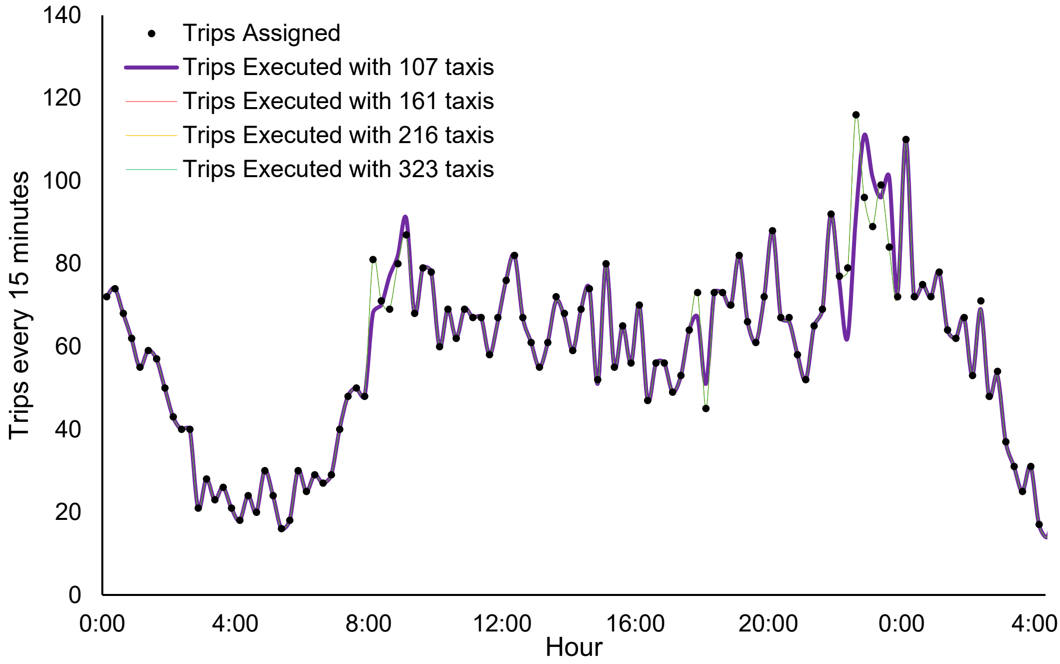

Figure 4.1 Evolution of the demand on Wednesday and the ability of the different fleets to satisfy it

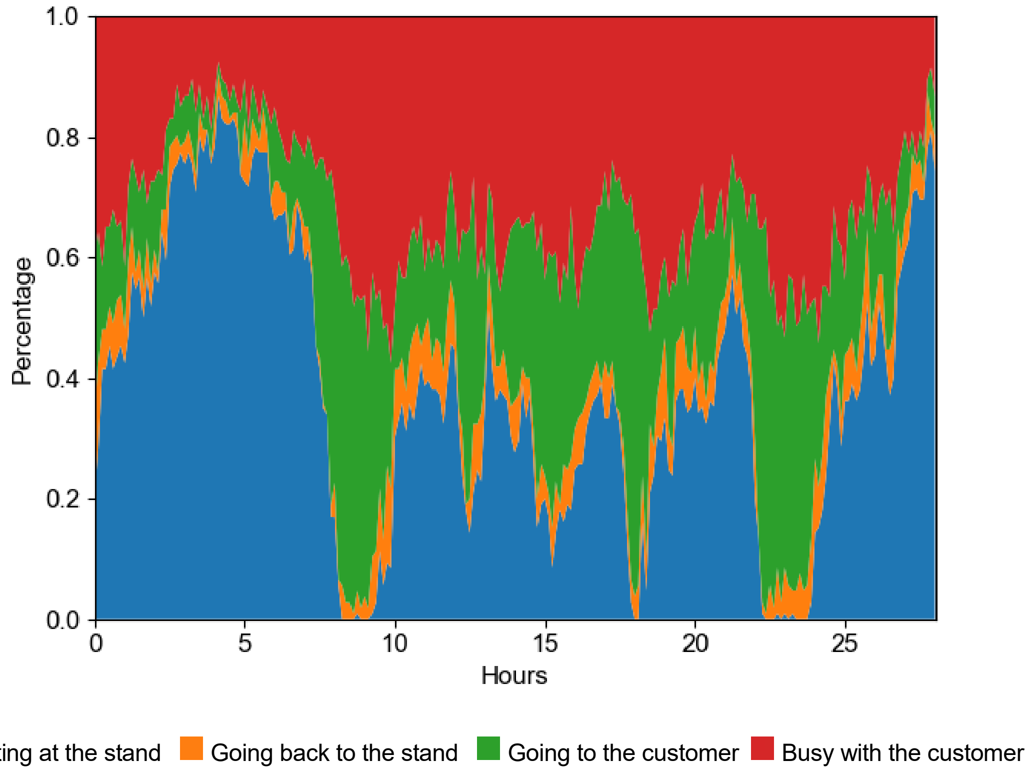

Figure 4.2 100% stacked area chart describing the activity of the 107 taxi fleet with no limit in stand capacity

Figure 4.3 100% stacked area chart describing the activity of the 161 taxi fleet with no limit in stand capacity

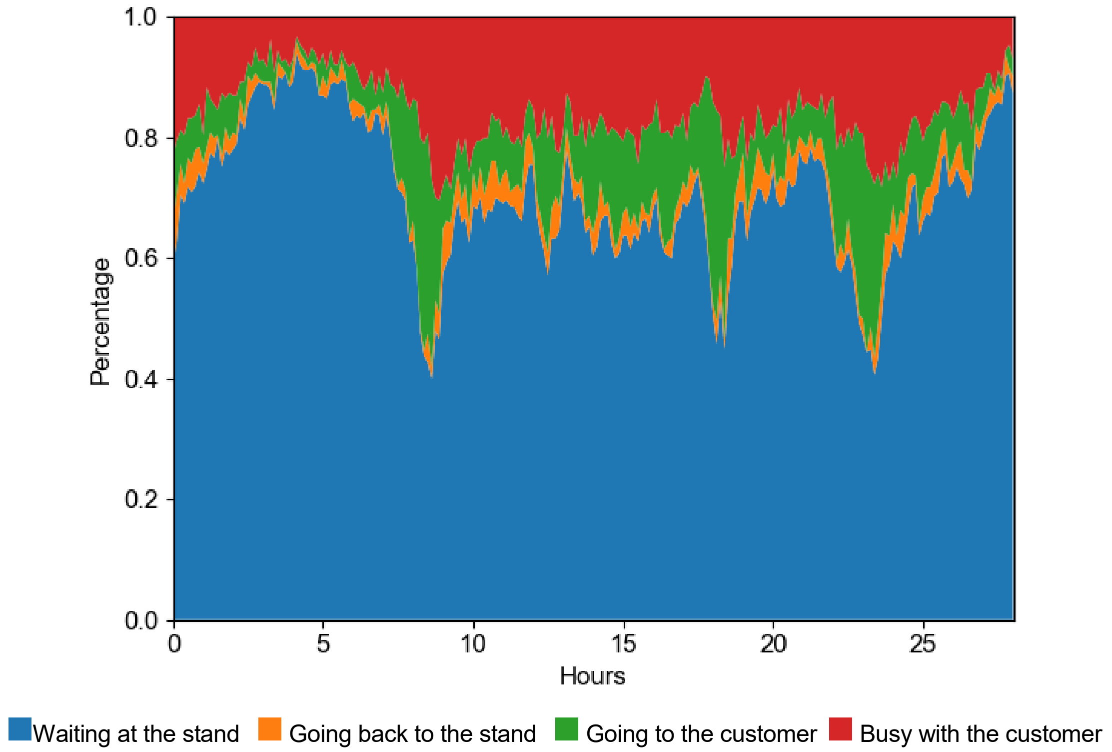

Figure 4.4 100% stacked area chart describing the activity of the 214 taxi fleet with no limit in stand capacity

Having considered the taxi fleet states and ability to satisfy the demand, it is now necessary to focus on the waiting times: Table 4.6 reports some aggregated data characterizing the whole day. Figure 4.5 and Figure 4.6 describe the waiting times evolution during the day, they show respectively the average and the median of the waiting times, calculated every half of hour.

Important increases in waiting times can be noticed for all the fleets under study whenever there is an increase of the demand.

Table 4.6 Aggregated data concerning the waiting time

Fleet size	Average in min	Median in min	Waiting Time above 15 min		Waiting Time above 25 min		Waiting Time above 30 min	
107	6.41	4.57	530	8.0%	237	3.6%	121	1.8%
161	5.56	3.93	386	5.8%	189	2.9%	98	1.5%
214	5.75	4.15	404	6.1%	195	2.9%	97	1.5%
321	5.71	4.08	403	6.1%	184	2.8%	91	1.4%

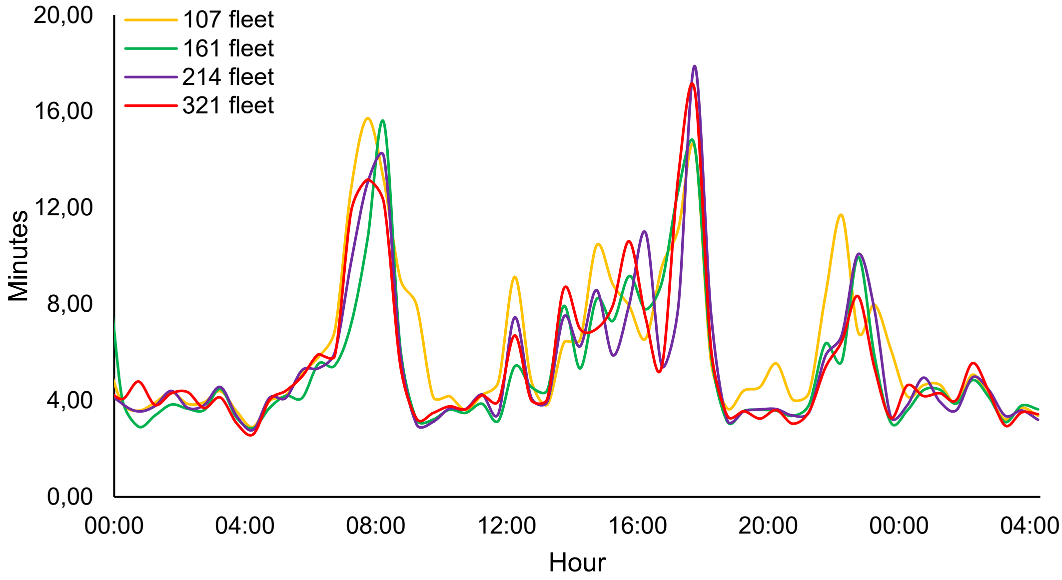

Figure 4.5 Evolution of the average of the waiting time calculated every half of hour

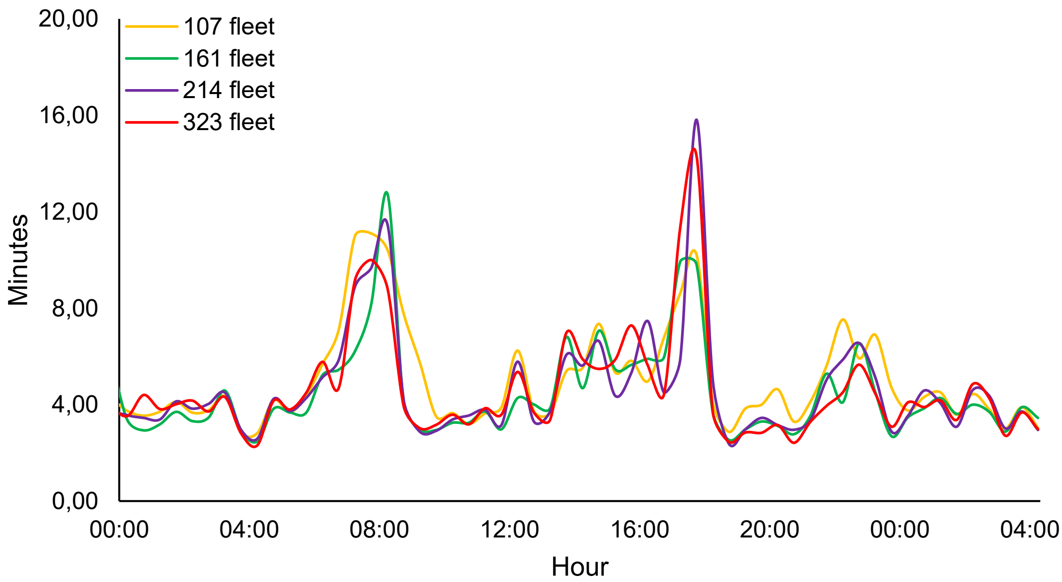

Figure 4.6 Evolution of the median of the waiting time calculated every half of hour

Last investigation performed in the scenario, concerns statistical data about each car, Table 4.7 summarizes the information concerning the total mileage of each car, the number of trips assigned and the hours of activity. Two different cumulated times are reported:

- Hours of Activity: refers at the total amount of hours in the period considered in which the vehicle is busy or anavailable for new assignments, i.e. the sum of the total time spent either reaching the customer or bringing it to destination
- Hours Moving: refer to the total amount of hours in which the vehicle is not waiting at the stand, it is the time in which the vehicle is busy plus the time spent going back to the station.

Maxima and minima reported concern only the active portion of the fleet while average, median and percentiles have been calculated considering the whole fleet, taking into consideration also the vehicles that stayed at the stand for the whole time.

It is possible to notice that, as the fleet size increases, so does the number of cars have not been called to activity in the 28 hours of the scenario. Considering the number in relation to the fleet size, it grows disproportionally, reaching even 31.7% in the case of the 321 fleet. This number confirms the fact that such a fleet is already oversized for the trip demand considered. It is also interesting to notice that minima in terms of kilometers covered and total hours of activity, are reached with the 161 fleet instead that with the bigger ones. Moreover, no particular differences in figures can be noticed between the 214 and the 321 fleet cases, the bigger number of vehicles decreases the values of averages and percentiles, but maxima and aggregated data show minimal differences.

Table 4.7 Statistic about km covered and hours of activity about each car in the different fleet cases

Type of data		**Unit**	**107 fleet**	**161 fleet**	**214 fleet**	**321 fleet**
Cars not moved			0	23	45	102
Trips assigned	Max	trips per vehicle	96	85	89	88
	Min	trips per vehicle	17	4	1	1
	Average	trips per vehicle	58.89	41.66	31.10	20.70
	25th Percentile	trips per vehicle	50.25	28.00	7.00	0.00
	Median	trips per vehicle	59.00	46.00	30.00	15.00
	75th Percentile	trips per vehicle	68.75	60.50	53.00	36.25
Km covered	Total of the fleet	km	77440	74539	75426	75165
	Max	km	949.99	894.85	893.85	899.77
	Min	km	340.42	76.93	47.85	43.60
	Average	km	730.56	468.80	354.11	234.89
	25th Percentile	km	664.54	361.30	98.81	0.00
	Median	km	728.81	531.98	361.14	181.73
	75th Percentile	km	815.35	642.77	590.63	392.35
Hours of Activity	Total of the fleet	hrs	1653	1561	1583	1578
	Max hours active	hrs	21.56	18.21	19.10	19.06
	Min	hrs	6.70	1.14	0.82	0.55
	Average	hrs	15.60	9.82	7.43	4.93
	25th Percentile	hrs	14.06	7.20	1.96	0.00
	Median	hrs	15.79	11.26	7.30	3.71
	75th Percentile	hrs	17.22	13.47	13.48	8.19
Hours Moving	Total of the fleet	hrs	1812	1739	1759	1754
	Max	hrs	23.55	20.33	21.38	21.39
	Min	hrs	7.44	1.33	0.86	0.59
	Average	hrs	17.09	10.94	8.26	5.48
	25th Percentile	hrs	15.23	7.93	2.15	0.00
	Median	hrs	17.27	12.45	8.18	4.11
	75th Percentile	hrs	19.23	15.11	13.84	9.09

4.3 Wednesday with limited stand capacity

Figure 4.7 reports the position of the taxis in the case of 214 fleet at 4:00 of Wednesday without considering any limit in the stand capacity (case exposed in section 4.2). After 28 hours of simulation, at this time of the day, 99% of the fleet is waiting at the stand due to the low demand. In the picture, the dimension of the dots is proportional to the quantity of taxis present. It is possible to notice that not all the available stands register the presence of a vehicle and that the large majority of the taxis (110 vehicles) are concentrated at the airport. A simulation that took into account the stand capacity has been performed and this showed to have a high impact on the waiting times reduction. To improve its effect, total capacity of the stands at the airport has been reduced from 90 to 35 places available. Appendix B reports the 100% stacked area chart describing the activity of the fleets in this case.

Figure 4.7 Distribution of taxis among the available stands at 4:00 of Wednesday if no limit in the stand capacity is considered, airport represented in upper right quadrant

With the use of QGIS, the position of the requested starting point have been investigated. Table 4.8 reports the quantity of origins and destinations in the area of the airport divided by time of the day (with shorter intervals at night because they are characterized by a higher demand). It can be noticed that a higer number of vehicles reaches the airport than the quantity leaving it. This leads to a gradual cumulation of vehicles shown in Figure 4.9: the number decreases suddently when they are called back to the city due to an insufficient number there that would satisfy the demand. After sudden decreases, gradual cumulation of vehicles at the airport starts again. Due to the low request from the airport, it has been decided to reduce the stand capacity in this area at 5% of the fleet, namely 5 places for the 107 fleet case, 7 for the 161 case and 10 for the 214 fleet case. No simulation has been performed with bigger fleets since they showed a low utilization.

Figure 4.8 shows the distribution of the 214 fleet in the aforementioned case (10 available places at the airport). The effects of the redistribution are evident, with an increase of the presence of taxis in the city: most of the taxis that in the case described in section 4.2 cumulated at the airport redistributed in the central stands, guaranteeing a higher coverage for the inner city demand.

Figure 4.8 Distribution of taxis among the available stands at 4:00 of Wednesday if limit in the stand capacity is considered and stands at the airport are reduced to 10, airport represented in upper right quadrant

Table 4.8 Origins and destinations from and to airport and surroundings on Wednesday

Time interval	**Starts from airport and surroundings**	**Avg per hour**	**Percentage of the trips**	**Destination to airport and surroundings**	**Avg per hour**	**Percentage of the trips**
00:00-08:00	21	2.62	1.36%	16	2.00	1.04%
08:00-16:00	62	7.75	2.81%	157	19.60	7.12%
16:00-21:00	16	3.20	1.23%	102	20.40	7.85%
21:00-01:00	9	2.25	0.68%	15	3.85	1.13%

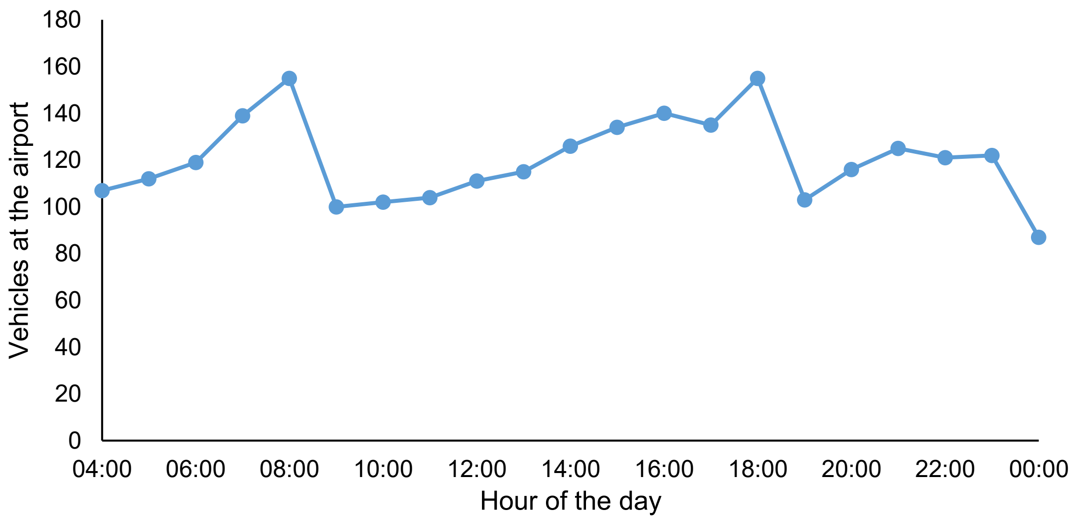

Figure 4.9 Evolution of the number of vehicles standing at the airport during the day for the 214 fleet case with unlimited stand capacity

Data comparison with the input showed very similar results to those presented in Table 4.4, table referring to comparisons considering the activity with customer on board is consultable in Appendix B. It is important to highlight that faulty trips, i.e. the trips that failed to be effectively recorded have decreased with respect to the first case analyzed and they don't overcome the 2.4% (107 fleet) with a minimum of 1.6% for the 214 fleet.

Looking Figure 4.10, Figure 4.11 and Figure 4.12 it is possible to notice an important reduction of the green portion with respect to figures Figure 4.2, Figure 4.3 and Figure 4.4. From Figure 4.10 it is possible to notice that the fleet counting 107 taxis reaches saturation just once at 23:30, instead of three times as reported in the first case considered, an important improvement in the ability to satisfy the demand is then recorded. The morning and afternoon utilization peaks have diappeared for 161 and 214 fleets with a maximum utilization of 60% for the 161 fleet and 50% for the 214 fleet reached only once at 23:30. In the Appendix B, the graph showing the demand curve and the trip execution curves for the three fleets analysed is reported while Figure 4.15 in section 4.3.1 shows the quantity of trips postponed during the day by the 107 vehicle fleet considering or not considering the stand capacity.

Figure 4.10 100% stacked area chart describing the activity of the 107 taxi fleet with limit in stand capacity

Figure 4.11 100% stacked area chart describing the activity of the 161 taxi fleet with limit in stand capacity

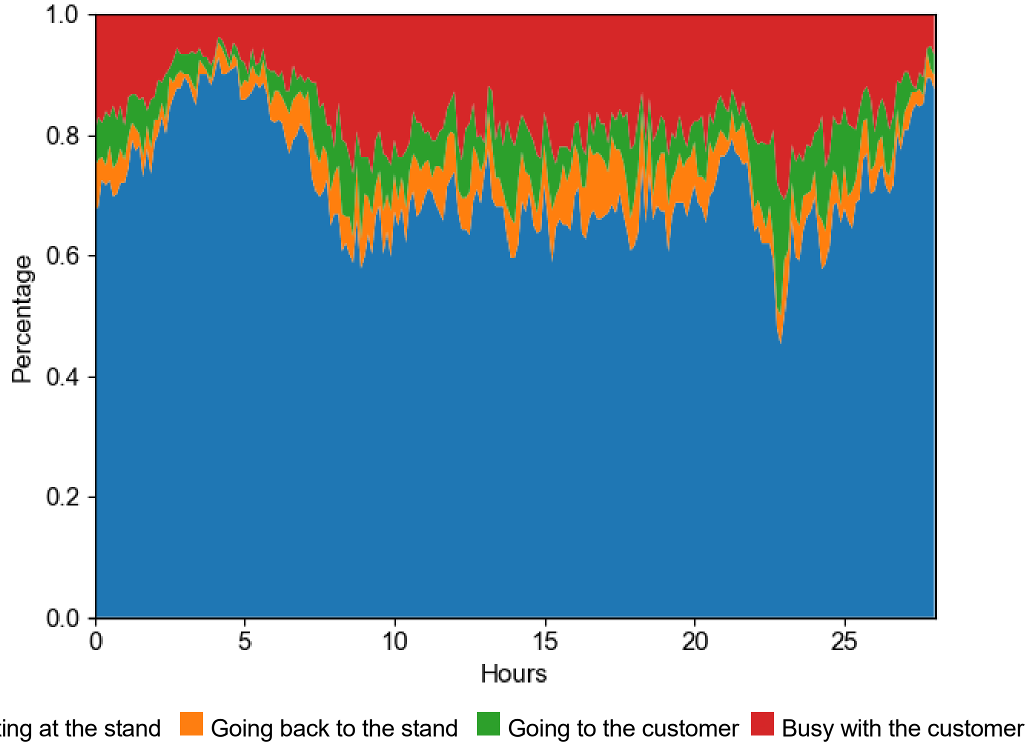

Figure 4.12 100% stacked area chart describing the activity of the 214 taxi fleet with limit in stand capacity

An overall reduction of waiting times characterizes this scenario with respect to the one with no limitation of stand capacity. Table 4.9 reports the aggregated waiting times data. Considering Figure 4.13 it is possible to notice how peaks of waiting time that were present around 8:00, 18:00 and 23:00 have strongly reduced and even disappeared for 161 and 214 taxi fleets. This is more visible in figure Figure 4.14 in section 4.3.1

Table 4.9 Aggregated data concerning the waiting time considering station limit

Fleet size	**Average in minutes**	**Median in minutes**	**Trips Above 15 Minutes**		**Trips Above 25 Minutes**	
107	4.67	3.98	90.00	1.3%	11.00	0.2%
161	3.90	3.50	7.00	0.1%	0.00	0.0%

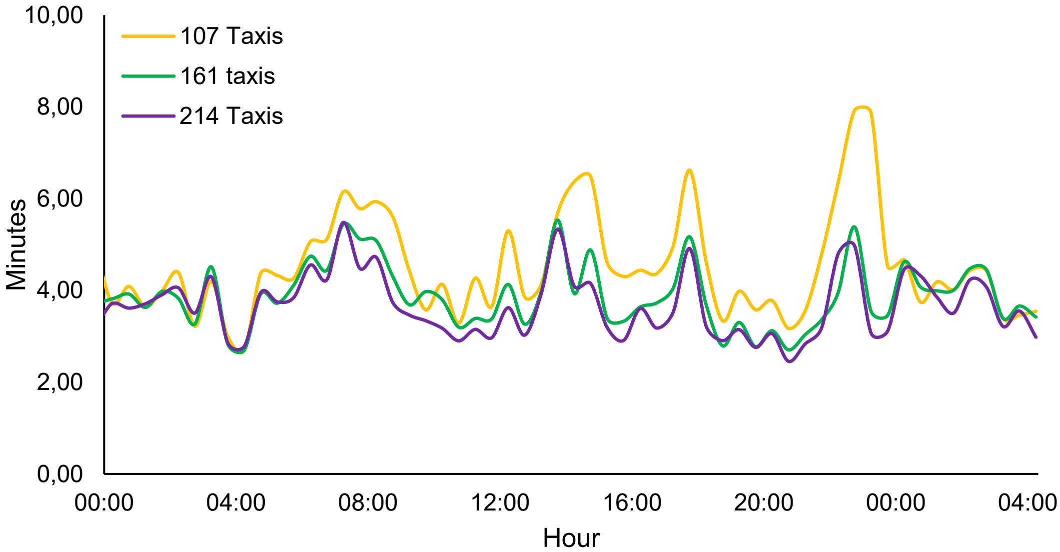

Figure 4.13 Evolution of the average of the waiting time calculated every half of hour

Finally, last data to observe concern the cars statistics reported in Table 4.10. It is worth to notice that a higher fleet utilization is characterizing this scenario with respect to the previous one. Distributions of trips are more even, as a consequence of a lower cumulation of cars at the airport that are less likely to be reassigned to new trips. Moreover, the fleet characterized by the minimum amount of total kilometers covered and amount of hours of activity the 214 one, differently to what seen in Table 4.7 where the 161 fleet was the most effective one from the activity point of view. In general, due to the better fleet distribution, it is possible to notice that amounts of total kilometers covered and hours of activity have decreased with respect to the previous case.

Table 4.10 Statistic about km covered and hours of activity about each car in the different fleet cases with limited stand capacity

Type of data		Unit	107 fleet	161 fleet	214 fleet
Cars not moved			0	5	11
Trips assigned	Max	trips per vehicle	97	79	80
	Min	trips per vehicle	28	3	1
	Average	trips per vehicle	64.04	41.67	31.10
	25th Percentile	trips per vehicle	57.00	28.00	17.00
	Median	trips per vehicle	66.00	43.00	33.00
	75th Percentile	trips per vehicle	72.75	57.00	44.00
Km covered	Total of the fleet	km	75078	71829	71320
	Max	km	1028.61	896.63	844.94
	Min	km	261.21	53.50	7.26
	Average	km	708.28	451.75	334.84
	25th Percentile	km	626.27	313.71	189.01
	Median	km	738.54	471.35	343.79
	75th Percentile	km	793.26	613.93	478.62
Hours of Activity	Total of the fleet	hrs	1510	1395	1376
	Max hours active	hrs	18.98	16.40	15.46
	Min	hrs	5.71	1.06	0.19
	Average	hrs	14.24	8.77	6.46
	25th Percentile	hrs	13.18	6.16	3.60
	Median	hrs	14.74	9.09	6.91
	75th Percentile	hrs	15.86	12.38	8.99
Hours Moving	Total of the fleet	hrs	1769	1680	1674
	Max	hrs	22.54	19.47	18.56
	Min	hrs	6.67	1.20	0.22
	Average	hrs	16.68	10.57	7.86
	25th Percentile	hrs	15.62	7.39	4.44
	Median	hrs	17.30	10.94	8.43
	75th Percentile	hrs	18.76	14.89	10.95

4.3.1 Conclusions on taxi stand size

Limit in stand capacity prevents the vehicles cumulation at the airport thus fostering a better distribution in the city. This in turn, leads to important improvements of the performance, reducing waiting times as it can be seen in Figure 4.14. The figure shows the improvements for 107 and 214 fleets. Peaks in waiting times have disappeared for the case with limit in stand capacity, only at 23:30 for the 107 fleet, waiting time increases to comparable levels with respect to the simulation unlimited stand capacity. Saturation is actually reached at this time but despite that, the peak is still lower than the one registered for the 214 fleet with unlimited stand capacity. As seen in Figure 4.10, Figure 4.11 and Figure 4.12, fleet utilization benefits from that reduction with a dramatic reduction in trip postponements for the 107 fleet case (Figure 4.15).

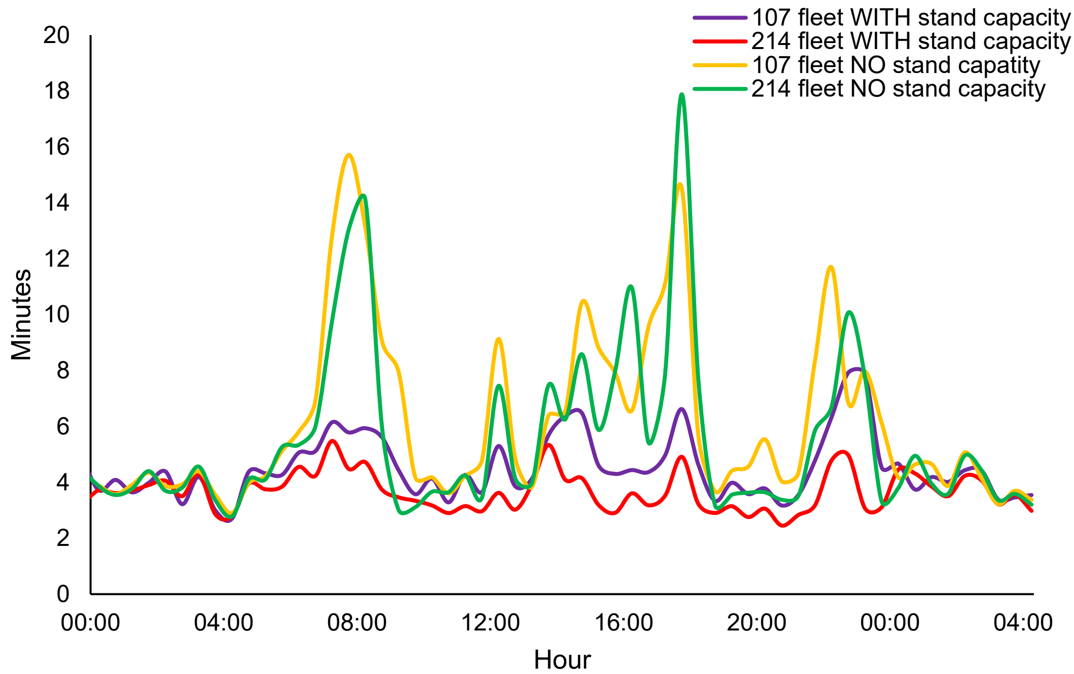

Figure 4.14 Comparison of evolution of waiting times with and without limits in stand capacity for 107 and 214 taxi fleets

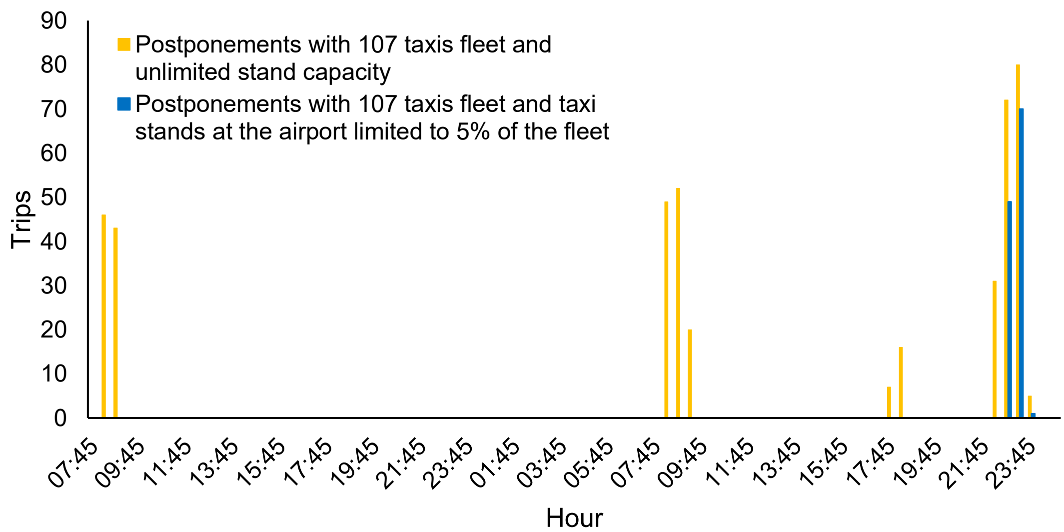

Figure 4.15 Postponements during the day for the 107 taxi fleet

Table 4.11 reports the aggregated data concerning the trips to reach the customer, last column offers the comparison with the scenario analized in section 4.2, highlighting an important reduction of total hours and kilometres necessary to perform such activity.

Table 4.11 Aggregated comparison of time and distances characterizing the trips necessary for the taxis to reach customer position considering stand capacity

Fleet case	**Data**	**Unit**	**Value**	**Comparison with 214 fleet case**	**Reduction compared to case of section 4.2**
107 taxi fleet	Total Time	hrs	521.4	26.8%	-26.8%
	Total Distance	km	18,317	33.1%	-40.5%
161 taxi fleet	Total Time	hrs	436.0	6.1%	-29.6%
	Total Distance	km	14,805	7.6%	-43.8%
214 taxi fleet	Total Time	hrs	411.1	0.0%	-35.6%
	Total Distance	km	12,761	0.0%	-49.6%

4.4 Saturday

As already mentioned, during the weekend the highest demand peaks are reached. Figure 4.16 shows the minimum presence of vehicles necessary to satisfy the demand during the 31 hours examined: from 00:00 of Saturday until 7:00 of the following Sunday. The figure represents the quantity of vehicles simultaneously present in the simulation if these are generated at the origin of each trip and leave the simulation at the destination. From the graph it is possible to notice that in the period of maximum demand, between 23:00 and 4:00 of the night between Saturday and Sunday the number of vehicles present approaches the threshold of 100 and even overcomes it reaching a peak of 114 vehicles present. It is evident that, due to the fact that it is very unlikely that vehicles are always exactly where needed at the beginning of a trip that is assigned, a phase in which the vehicle approaches the customer is usually necessary (it corresponds to the waiting time) and so, the minimum number of vehicles necessary can be only approached. In view of the above, the simulation with 107 taxi fleet has been discarded. Results of Wednesday also showed that fleets counting more than 214 vehicles where oversized for the task. Fleets selected then to be tested in the more demanding weekend situation are the 161 and the 214 taxi fleets.

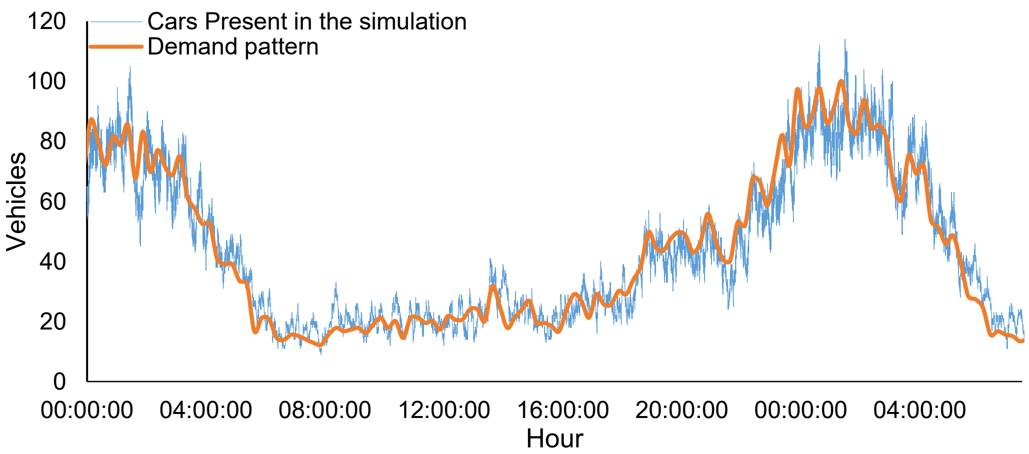

Figure 4.16 Minimum number of vehicles necessary to satisfy the demand in each time of Saturday

Table 4.12 shows that there is a very low demand of trips from the airport, for this reason it has been decided to further reduce the quantity of places available at the stands in this area to a number of 3, equal for both the fleet tested. The results of a simulation with 35 stands at the airport showed important criticalities and important saturations of the fleets, some charts can be consulted in Appendix C but will not be further discussed.

Table 4.12 Origins and destinations from and to airport and surroundings on Saturday

Time interval	Starts from airport and surroundings	Avg per hour	Percent-age of the trips	Destination to airport and surroundings	Avg per hour	Percentage of the trips
00:00-08:00	3	0.375	0.1%	57	7.125	2.0%
08:00-16:00	10	1.25	0.9%	79	9.875	6.8%
16:00-20:00	2	0.5	0.2%	21	5.25	2.1%
20:00-23:00	0	0	0.0%	10	3.33	0.7%
23:00-3:00	0	0	0.0%	3	0.75	0.1%

Appendix D reports the table with the comparisons of the kilometres and times required to bring the customers to destination in the two fleet cases and in the input situation. Faulty trip in this simulation are 3.9% of the demand in both the cases.

Figure 4.17 represents the trip demand and the trips execution every 15 minutes for the two fleets examined. The two curves can be perfectly superimposed, this means that both the fleets manage to handle the demand. The loss due to some residual defects in the map that make trip execution curves not to pass through all the black dots is of the 1.44%. For the case of the 161 taxi fleet, Figure 4.18 shows that some of the trips have to be postponed due to a lack of available taxis but postponements have reached a maximum of seven minutes, not causing any deviation of the curve in Figure 4.17.

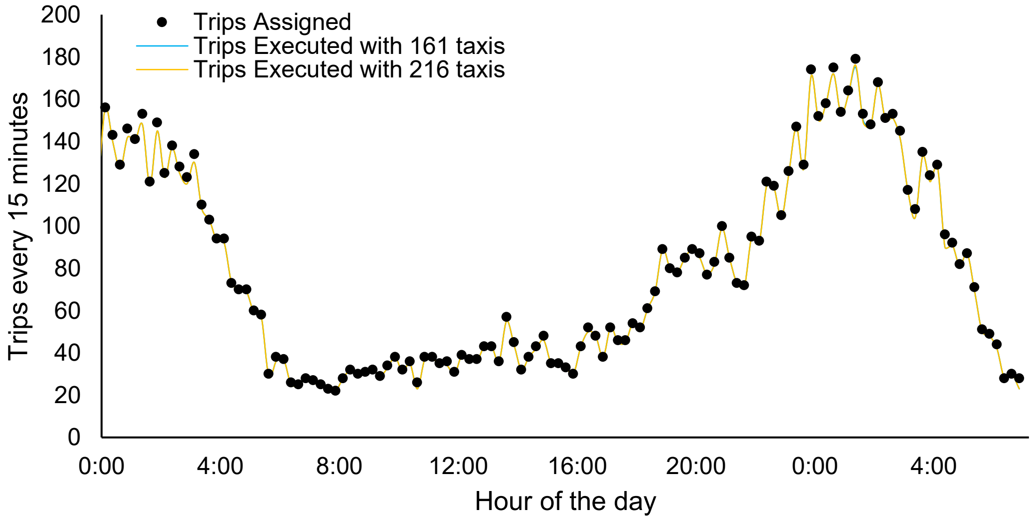

Figure 4.17 Evolution of the demand on Saturday and the ability of the different fleets to satisfy it

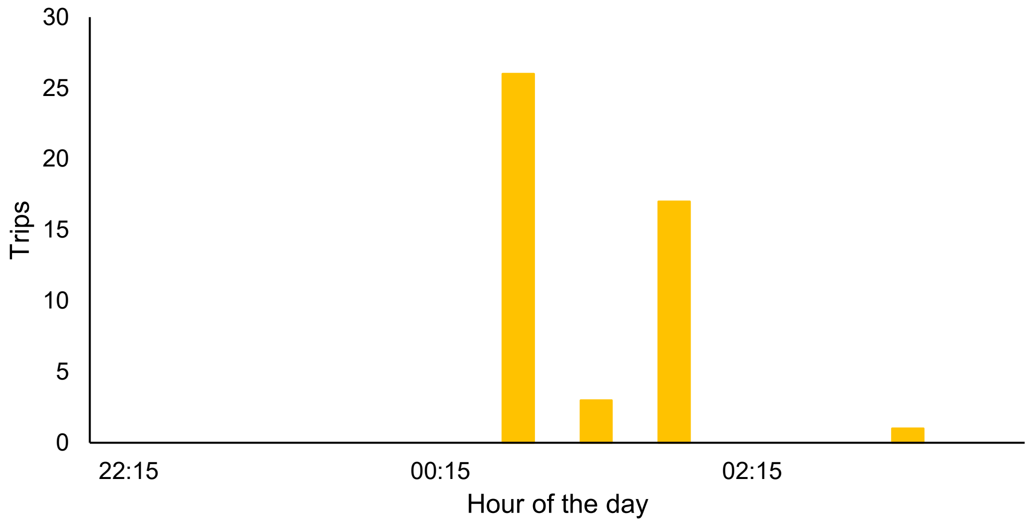

Figure 4.18 Distribution of the postponements during Saturday night for the 161 taxi fleet case

Figure 4.19 and Figure 4.20 show the activity distribution of the two fleets during the day. A high utilization is reached during the peak demand in the hours between Saturday and Sunday, reaching saturation at certain times in the smaller fleet case. This translates in an increase of the waiting times but from Figure 4.21, it can be seen that such increase remain always with an average below 6 minutes.

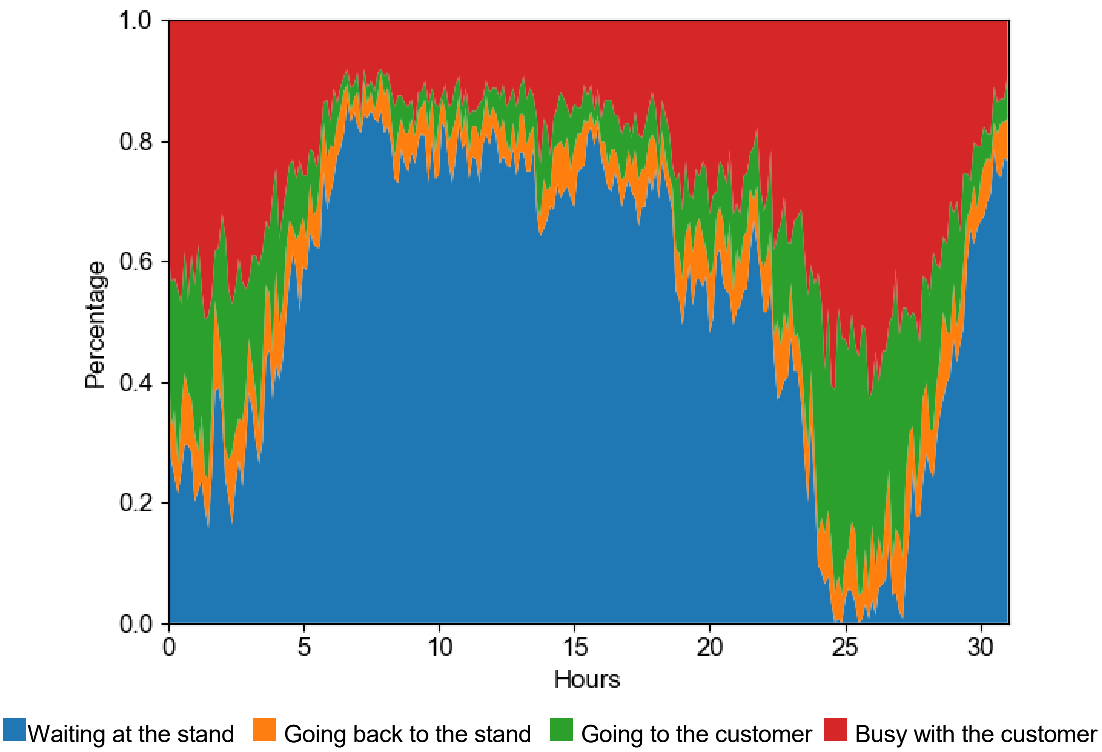

Figure 4.19 100% stacked area chart describing the activity of the 161 taxi fleet on Saturday

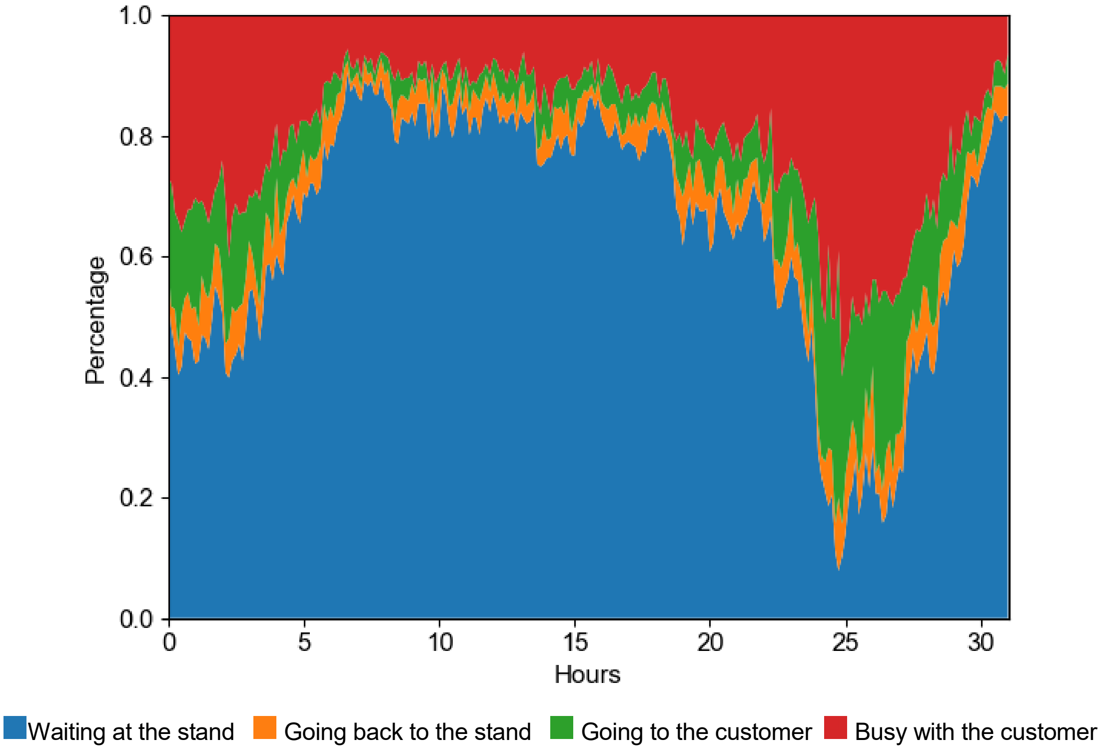

Figure 4.20 100% stacked area chart describing the activity of the 214 taxi fleet on Saturday

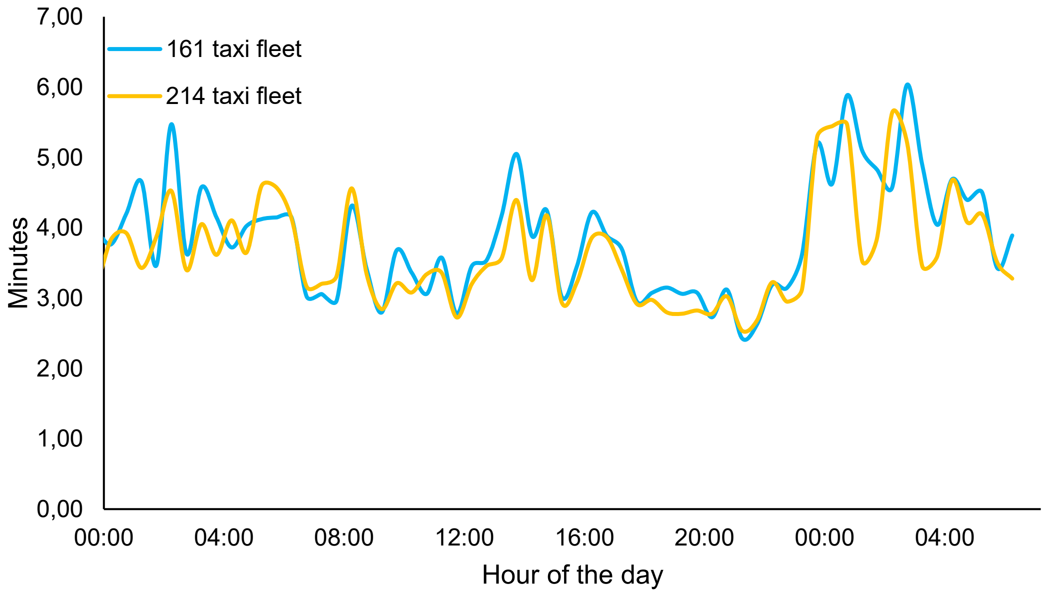

Figure 4.21 Evolution of the average of the waiting time calculated every half of hour on Saturday

Table 4.13 reports the statistics of the kilometers covered, trips assigned and working hours of each vehicle. As already noticed in the cases analyzed in the previous sections, a higher activity by car is recorded for the smaller fleet.

Table 4.13 Statistic about km covered and hours of activity about each car in the different fleet cases on Saturday

Type of data		Unit	161 fleet	214 fleet
Cars not moved			0	2
Trips assigned	Max	trips per vehicle	115	113
	Min	trips per vehicle	18	2
	Average	trips per vehicle	60.75	45.78
	25th Percentile	trips per vehicle	46.00	30.00
	Median	trips per vehicle	60.00	46.00
	75th Percentile	trips per vehicle	76.50	60.50
Km covered	Total of the fleet	km	86682	84702
	Max	km	939.28	856.53
	Min	km	178.02	15.65
	Average	km	545.17	401.43
	25th Percentile	km	426.19	269.48
	Median	km	537.92	399.75
	75th Percentile	km	668.19	529.58
Hours of Activity	Total of the fleet	hrs	1962	1967
	Max hours active	hrs	21.12	19.23
	Min	hrs	4.69	0.38
	Average	hrs	12.34	9.32
	25th Percentile	hrs	9.61	6.44
	Median	hrs	12.18	9.40
	75th Percentile	hrs	14.97	12.19
Hours Moving	Total of the fleet	hrs	2274	2294
	Max	hrs	24.00	22.61
	Min	hrs	5.12	0.46
	Average	hrs	14.30	10.87
	25th Percentile	hrs	11.10	7.30
	Median	hrs	13.86	11.07
	75th Percentile	hrs	17.45	14.15

5 Analysis of the results

In this chapter, results presented in chapter 4 are discussed and analyzed. In section 3.6.2, a brief summary is dedicated to the codes comparison. Comments on the differences between the possible alternatives have already been developed in those sections. It is important in this chapter, to discuss the significance of the results obtained, starting from the validation of the simulation and successively focusing on the impact of the possible factors on fleet performance.

5.1 Validation

To evaluate the quality of the results, it is necessary to compare input data with those obtained from the simulation. The most important factor to be considered in order to interpret the results is the absence of traffic: this is the most contributing cause to the differences between results. The comparison is reported in Table 4.2 and it involves the data concerning the trips executed with the customer on board since the trips executed to reach the customer are highly dependent on fleet size and distribution and this information was not available in the input.

From the data it is possible to notice a strong reduction of time needed to bring the customer to destination, much more relevant than the reduction of distance covered for the same purpose. This is compatible with the expectation that traffic influences the time needed for a trip more than the distance covered. Marta Gonzales and Antonio Lima highlighted that drivers have some usual route preferences and these are influenced by many factors but only in half of the cases they correspond to the optimal one [63]. Another study by Khattak, Schofer and Koppelman concluded that quality of traffic information influence the impact on route choice but has mostly the effect of reducing drivers' anxiety even though many times it does not result in route changes. The effect that is most expectable is to shift the departure [64]. Due to an average difference of less that 4% in the distance covered, the results can be considered compatible. It is not possible to estimate which portion of difference is given by the traffic influence and which one by differences in routing decisions between reality and SUMO model.

To further understand the results, it is interesting to consider the weekend situation. Inner city traffic does not present any rush hour during this period and lower traffic levels are usually recorded [65]. Furthermore, as already mentioned in section 2.7 there is a lower tendency in the usage of private car during the night hours when taxi demand peaks happen. In these conditions average reduction in distance covered with respect to the input is of 4.0% while the reduction in time needed is of 28.4%. Impact of traffic on operating time is thus very high, determining important differences due to congestion. Since a similar difference in distance covered has been measured in the weekday and in and in the weekend day, it can be stated that the influence of traffic is minimal on this dimension. Not knowing precisely and extensively the difference in private traffic between these two periods, no further conclusion can be advanced.

5.2 Fleet distribution

It can be noticed that bigger fleets not only ensure a higher ability in satisfying the demand but also they guarantee a higher capillarity. This is explained by the overall decreasing trend shown in Table 4.5 for what concerns aggregated kilometers and hours spent in total to reach the customers.

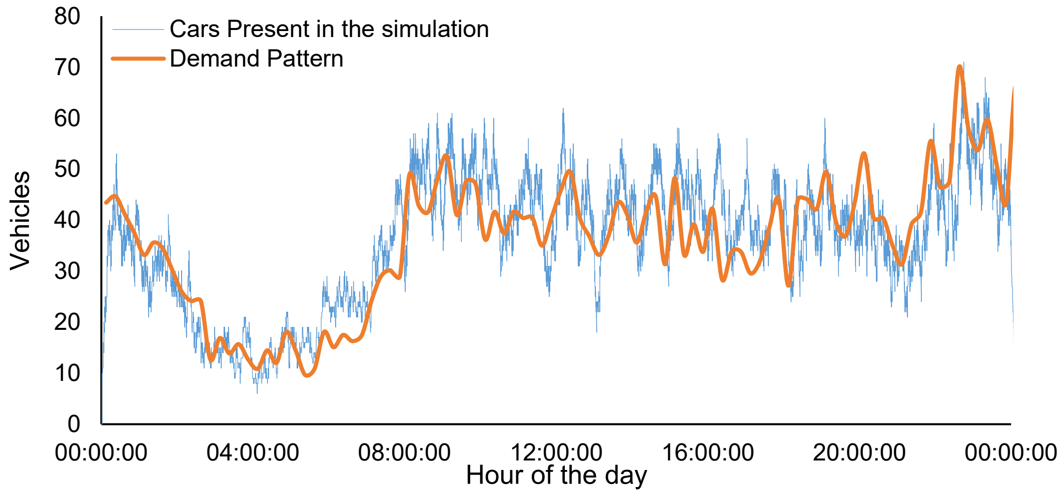

Figure 5.1 Minimum number of vehicles necessary to satisfy the demand in each time of Wednesday

Only the smallest fleet shows problems in satisfying the demand while fleets bigger than 214 taxis show quite low values of utilization. This can be explained by data reported in Figure 5.1 that shows the quantity of vehicles present simultaneously in the simulation if these were generated at the origin of each trip and leave the simulation once the destination is reached, similarly to what already seen in Figure 4.16. Peaks of vehicles present are reached at evening hours and they never exceed 71. From Figure 4.2, Figure 4.3 and Figure 4.4 it is possible to notice that the green portion, i.e. the quantity of vehicles reaching the customer increases considerably when the demand rises, determining a higher utilization than necessary of the fleet and thus reducing dramatically the quantity of vehicles free to be assigned. This is caused by a not optimized distribution: taxis distant from origin of the request require more time to reach the customer, thus staying busy for longer time, this in turn forces the taxi headquarter to assign the trips to ever farther vehicles, further increasing waiting times and occupation. This has been proved by the results illustrated in section 4.3. A tendency of cumulation at the airport has been noticed due to a demand towards the airport higher than a demand from it as shown in Figure 4.9.

Such a cumulation also explains the peaks of waiting times recorded for all the fleet cases at 8:00, 18:00 and 23:00, despite a low utilization for the bigger fleets: the quantity of taxis cumulated at the airport is so remarkable that the ones that remained in the city centre are not enough to cope with the demand. For this reason, even though the demand can be considered as almost constant after 8:00, the waiting times decrease: a number of taxis high enough has been recalled in the inner city and the condition of good distribution is temporarily restored. Restarting the gradual cumulation at the airport due to the higher number of arrivals with respect to the returns as shown in Table 4.8.

From Table 4.7 and Table 4.10 can be noticed that a better distribution also ensures a more even utilization of the vehicles: very few vehicles have not been used for the whole day even for the fleets with 214 and 321 taxis. A higher availability in the inner city allows to choose among a higher number of vehicles while the ones that cumulates at the airport are less likely to be used.

The effect of a controlled distribution can be noticed in Figure 4.13 where peaks have been nearly halved compared to the previous situation. Moreover 107 taxi fleet proved to be already big enough reaching saturation only in the evening but still able to satisfy the demand without and important amount of postponements and keeping waiting time at acceptable levels considered the researches illustrated in section 4.3.

Due to a lower utilization of the fleet as this is increased in number, cumulation is more consistent in presence of a large fleet. As a consequence the effects become more evident with a higher number of taxis: Table 4.11 shows that improvements on waiting times are more relevant as the fleet grows.

5.3 Fleet dimension

Taking the fleet into consideration, it is evident from the first simulations (section 4.2) that fleets bigger than 214 taxis show too low utilizations during Wednesday. These have been then discarded for further analysis.

107 taxi fleet deserves some particular comments: the simulation that takes into account the stand capacity shows that the fleet is able enough to satisfy the demand. Even accepting a certain quantity of postponements and thus a decrease in the quality of the service, the waiting times are anyway compatible with the numbers found for the public transport in Munich. Although, if this number is sufficient for a weekday, it is not for Saturday: in section 4.3.1, it has already been explained that the fleet is too small for the demand and some partial results of simulations proved this hypothesis. Moreover, it must be taken into consideration that traffic is here absent and its impact can be important as seen in section 4.1. A closeness to saturation for such a quantity of times and for such a duration is strongly unstable and the consequences due to any factor that can negatively affect the performance of the fleet are amplified by this closeness to saturation. An example of that can be found in the results of the simulation illustrated in section 4.2: a non optimized distribution of taxis decreased very importantly the performances with 421 postponements (6.40% of the total with an average of 7.44 minutes per postponement and a maximum of 41 minutes) against the 120 (1.82% of the total with an average of 3.63 minutes per postponement and a maximum of 19 minutes).

As the fleet increase the utilization decreases reducing the possibility to justify an initial investment. Although, it is important to consider that Table 4.11 shows how a bigger fleet implies higher savings in kilometers in reaching the customers due to a better distribution. This does not translate in particular savings in overall kilometers travelled by the whole fleet because the kilometers covered to return at the stand partially balance the overall saving, what is characterized by an important reduction though, are the average kilometers covered by a single vehicle. Table 5.1 shows the reductions obtained passing from the 161 to the 214 fleets.

Table 5.1 Kilometers reduction passing from 161 taxi to 214

Day	Reduction in overall covered km	Reduction in average km covered by a single vehicle
Wednesday	0.7%	25.9%
Saturday	2.3%	26.4%

It is to be commented the fact that 161 taxi fleet also approaches the saturation at night hours. The main difference with the saturation case on Wednesday for the 107 fleet to be discussed though, is the fact that these are actually the most demanding conditions and a decrease in the service can be accepted during weekend as discussed in section 2.7. Trips postponed in this case are 47, i.e. 0.48% of the total with an average postponement of 2.06 minutes and a maximum of 7, thus they have virtually no influence on waiting times. Also in this case it has to be considered that results have been obtained without taking into account traffic

Cost evaluation

The choice of the solution to be adopted is strongly dependent on the costs that they imply: a bigger fleet involves a higher initial investment but ongoing costs are to be examined. Moreover the duration of the investment is longer for the bigger fleet due to a lower amount of kilometers covered daily by each car. A simplified cost model have been implemented in order to obtain a better idea of the advantages of the two solutions.

Since no AV is currently on the market only estimates can be done about it. Current costs of production due to the use of new technologies, and to the absence of a mass production, make the building of such a solution very expensive: GM's estimated costs to build an AV ranges around $200,000 [66]. Other estimates made by Austin Russell, CEO of Silicon Valley startup Luminar, expect an autonomous vehicle to cost between $300,000 and $400,000 a piece [67]. These costs are expected to drop in the next years thanks to technology developments and volume: Delphi CEO Kevin Clark expects a decrease of 90% by 2025 [68]. Combining these figures the cost of production of a vehicle ranges between $20,000 and $40,000.

Due to the high uncertainty around these figures, a different approach is to be looked for: Patrick M. Bösch, Felix Becker, Henrik Becker, Kay W. Axhausen, in their work "Cost-based analysis of autonomous mobility services" [69] started from the cost of actual cars and based on HIS Automotive report estimated the cost increase due to the automation in 20% as average. In the work different estimates have been made concerning the cost variations due to automation: fuel costs are expected to decrease of a 10% thanks to a more regular driving but this decrease would be balanced by a higher maintenance cost for sensors periodic maintenance; insurance rates are expected to decrease at least of some 50%. Another significant cost driver is the battery, that due to losses in performances is expected to be changed every 150,000km.

The car selected as reference for the cost estimation is a C-segment electric: Volkswagen E-golf. The characteristics and costs related to the car are reported in Table 5.2. To the purchase price, an increase due to automation has to be added minus a 30% of discount due to the large number bought by the fleet operator [69]. Energy costs are based on car's energy consumption [70] and energy prices in Germany [71].

Table 5.2 Characteristics and costs referring to a VW E-Golf 2018

Item	Unit	Measure
Cost per vehicle [72]	€	30,150
Energy consumption every 100 km	kWh	15.94
Battery capacity [70]	kWh	35.8
Cost of energy	€/kWh	0.20
Cost of battery [73]	€/kWh	170
Insurance per car per year [74]	€	650
Maintenance per km per car [75]	€	0.03

From the data expressed in the table initial investment and total on going costs per day have been calculated for the two fleet cases, these are reported in

. Initial investment for the smaller fleet is of course lower but it carries higher operating costs due to higher quantity of kilometers covered in total that translates in more energy consumption and by car that translates higher battery change frequency. Costs of operation linked to the usage are balanced by higher cost of insurance due to the bigger fleet. The result is that ongoing costs are very similar, slightly higher for the 214 taxi fleet.

Table 5.3 Fixed and Variable costs concerning the two fleets examined

Item	Unit	161 taxi fleet	214 taxi fleet
Initial investment	k€	4,854.15	6,453.1
Energy costs per day	k€	1.97	1.95
Insurance costs per day	k€	0.29	0.38
Battery costs per day	k€	2.53	2.51
Maintenance costs per day	k€	2.24	2.22
Total costs per day	k€	7.03	7.06

Daily costs have been considered as every weekday apart Saturday was characterized by the same average hourly demand of the Wednesday while Saturday has been just normalized to 24 h. The quantity of trip demand obtained multiplying by 6 the one of Wednesday and adding the day of Saturday is only 1.1% higher with respect to the total quantity of trips requested during the whole week. Moreover, from Figure 5.2 it can be seen that the demand pattern obtained summing the whole week demand by hour of the day and the modelled day are very similar, only during the first hours of the day the modelled day demand pattern is lower than the whole week pattern, starting from a difference around 15% until a complete convergence is reached at 7:00.

If the investment is to be judged only on a short time basis without takin into account mileage, it is evident that the 161 taxi fleet results more advantageous. Due to a higher mileage per car for the 161 fleet solution though, lifecycle of the vehicles is expected to be lower thus requirig a higher frequency of renovation of the invesment.

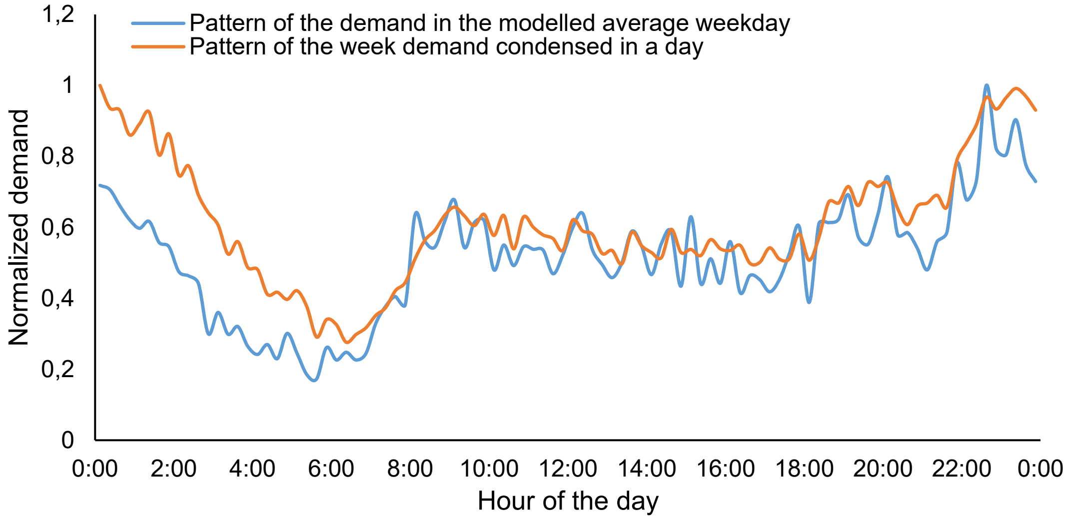

Figure 5.2 Demand pattern of the day model considered for the costs and the whole week model

Due to an intensive usage, a useful life of 300,000 km per car can be hypothesized [69]. Being the vehilce autonomous, the driving style is supposed to be more regular, reducing stresses on mechanical parts, an even longer life can thus be possible. To reach the mentioned kilometer limit for each car, the 161 taxi fleet requires 775 days, i.e. 2.12 years while the 214 taxi fleet requires 1,039 days, i.e. 2.84 years. Corresponding the duration of the investment to the aforementioned times, the evolution of the expenses on a 10 years perspective can thus be obtained and it is represented in Figure 5.3. The two amounts of expenses are very similar and depending on the time horizon considered one can be more advantageous than the other. Due to a higher graduality of the investment the smaller fleet is to be preferred but there is no other clear economical advantage as the gap between the ongoing costs carried by each of the two solutions is too narrow and figures provided are just indicative and not reliable enough.

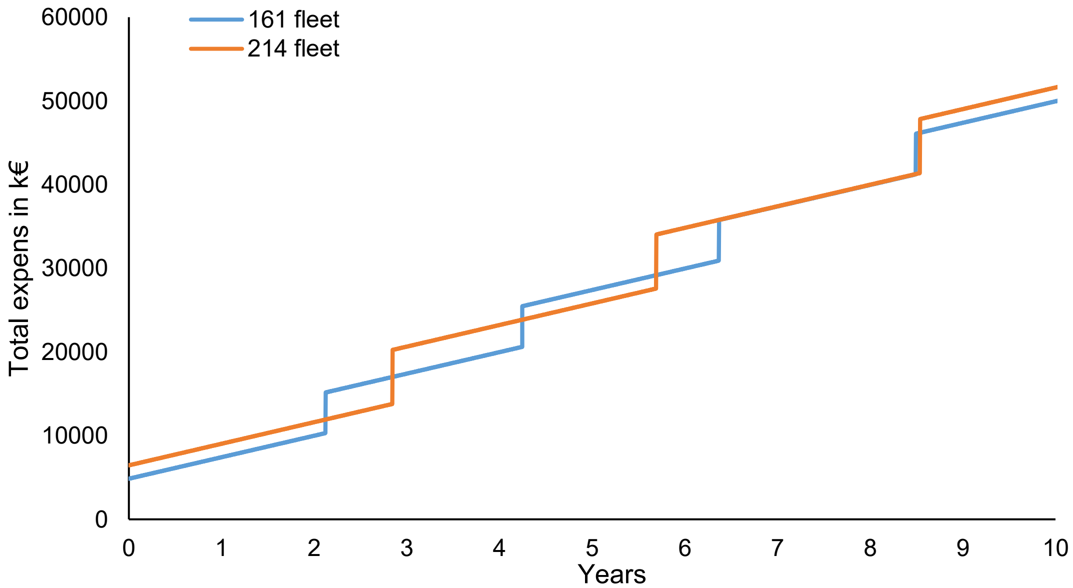

Figure 5.3 Expense evolution during the year for the two fleets considered

Comments

Combining results concerning the performance of the fleet and the outcome of the cost analysis, both fleet solutions seem to be adequate for the required task and also carry a very similar amount of expenses. As already mentioned, 161 vehicle fleet has the advantage of implying a smaller initial investment and lower renovation costs (although with a higer frequency). It has to be considered though that in the simulations no traffic has been introduced and as discussed in section 5.1, its impact can be important. In section 4.4 possible criticalities have been highlighted with a saturation of the 161 fleet reached Saturday night. Due to an extention of moving times, and considering the levels of utilizations of the two fleets, performance of 161 fleet are expeted to suffer a more severe decay with respect to 214 fleet.

In view of these assumptions, 214 vehicle fleet guaranteeing a higher performance resilience to external factors such as variation of the demand and traffic introduction and carrying no cost burdens with respect to the smaller fleet, is the fleet to be preferred.

6 Conclusions of the study

A study of the behavior of a taxi fleet in the city of Munich has been performed with the use of SUMO open source program. The goal of the study was double: to explore possibilities offered by SUMO software and to simulate the taxi traffic in the city of Munich in order to investigate its performances. By tuning scenario settings such as fleet size and distribution and by testing the results with a different demand pattern, performances of the fleet have been investigated paying particular attention to waiting times and fleet utilization. From this analysis of performances a final ideal fleet size had to be identified.

In a first phase of simulation generation and validation, main factors of influence on performance have been identified and investigated. The highest impacts have been found in fleet size, vehicle distribution and demand pattern.

After a phase of analysis and tuning concerning the distribution factors based on an average weekday, a validation of the advanced hypothesis and a test of performance has been performed on a more demanding scenario: Saturday. Results have driven to the selection of two fleets: 161 vehicle and 214 vehicle one. These fleets manage to fulfill the demand in both the scenarios without an excessive waste of resource, i.e. avoiding too large numbers of vehicles. A smaller fleet showed criticalities from the saturation point of view and was underdimensioned for the weekend demand (see Figure 4.16). Bigger fleets instead showed too low utilizations and poor or no improvements in the performance.

In order to support the identifcation of the ideal solution for the case study, an indicative cost model has been developed. It has been supposed that the fleets, instead of counting on traditional human operated taxis, are based on Autonomous Vehicles. This kind of vehicle offers a 24 h availability, posing no problems for what concerns the shifts management, a factor that can be cause of a fleet increase. To calculate the costs carried by such a solution, it has been decided to start from the selection of an electric car available on the market and to apply expected cost corrections due to its automation. The model showed that in terms of total amount of expenses no solution was better than the other since a smaller fleet would imply a lower initial investment but also a higher frequency in the fleet renovation.

Taking into consideration limitation encountered in the scenario generation, particularly related to the absence of traffic in the city (see section 3.6.4) and the fact that results obtained from the cost model do not favor any particular solution instead of the other, a reasoning more based on performance variations has been adopted for the fleet choice. It is expectable that the introduction of traffic could extend moving times, thus increasing the saturation period during the weekend for what concerns the 161 AV fleet. Because of this reason, the choice of a 214 vehicle fleet is safer from a service level point of view.

It is important to mention that, besides traffic introduction that would decrease performance, a further fleet distribution optimization could at least partially compensate. It is reasonable not to expect dramatic improvements as the ones seen with the reduction of the stands at the airport (see section 4.3.1) since all the vehicles are now much closer to the main demand actractors, mostly located in the city centre. Instead of a static stand distribution, a dynamic one, changing during the day based on demand forecasts, is to be implemented to achieve more effective results. This can be a subject that can be developed in further studies.

In conclusion, in order to satisfy the trip demand registered for what concerns the taxis, the future application of electric AVs is a viable solution with a number ranging between 160 and 215 vehicles. A bigger fleet guarantees a higher resilience of the service level with respect to the variation of external factors such as demand and traffic intensity, thus, a 215 fleet is to be considered the best choice.

6.1 Future steps

The simulation developed within this study offers the possibility for further studies on mobility in Munich and its surrounding area but it can also be adapted to any different area. Future developments of the model can take two alternative directions.

One possibility is to maintain the scenario and the simulation developed, generating and introducing the private traffic. This solution would aim at the reproduction of a situation close to the current transport situation in Munich, substituting the traditional taxi service with an optimized AVs one in order to investigate cost and service improvements. Solutions found with these studies are to be expected of easier and closer in time implementation. Dinamic distribution of the fleet instead of a fixed stand based one can also be developed.

The second study direction aims at the complete rethinking of the mobility concept in the area, extending the service, from an exclusive taxi demand to a complete mobility demand. This scenario would aim at the substitution of the private traffic with a fleet of shared AVs and would estimate benefits achievable from points of view concerning congestion, moving times and cost. Such an transportation evolution would not only reduce traffic but would also free a high portion of spaces today used for high capacity roads parking lots, opening the way for a larger rethinking of the urban environment.

List of Figures

List of Tables

List of Codes

References

[1] United Nations - Department of Economics and Social Affairs, *68% of the world population projected to live in urban areas by 2050, says UN.* [Online] Available: www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. Accessed on: Aug. 10 2018.

[2] Christina Flitsch, Sarah Pfoser, Matthias Neubauer, Oliver Schauer, “Investigating the application of an alternative fuel technology – A scenario-based simulation approach to analyze environmental impacts,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 29–38, 2017.

[3] Christos Tsotskas, Didac Busquets, Robin North, Raymond Francis, Patrizia Franco, Fabio Galatioto, Carl Goves, Roland Guichard, Eifion Jenkins, Ecaterina McCormick, Abraham Narh, Graham Ravenscroft, Richard Tusting, Vittoria Parisi, Nicolas Le Glatin, Graham Fletcher, “The design and development of an HLA-based open platform to model urban environments in SUMO,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 30, no. 1866-721X, pp. 73–84, 2016.

[4] Andrew Keen, *The future of travel: How driverless cars could change everything.* [Online] Available: www.cnn.com/2013/05/14/business/bussiness-traveller-futurecast-driverless-car. Accessed on: Aug. 10 2018.

[5] Andrea Rossi, *A settembre i test per la guida autonoma: a Torino un circuito di 35 chilometri.* [Online] Available: www.lastampa.it/2018/07/25/cronaca/a-settembre-i-test-per-la-guida-autonoma-a-torino-un-circuito-di-chilometri-Hwu5EQjyEUDo6e4NamqpNN/pagina.html. Accessed on: Aug. 10 2018.

[6] Wharton School - University of Pennsylvania, *Demographic Shifts: Shaping the Future of Car Ownership.* [Online] Available: http://knowledge.wharton.upenn.edu/article/demographic-shifts-shaping-future-car-ownership/. Accessed on: Aug. 10 2018.

[7] François Vaudrin, Jakob Erdmann, Laurence Capus, “Impact of Autonomous Vehicles in an Urban Environment Controlled by Static Traffic Lights System,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 81–92, 2017.

[8] Michele Segata, “Platooning in SUMO: An Open Source Implementation,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 51–62, 2017.

[9] *Das DLR im Überblick.* [Online] Available: https://www.dlr.de/dlr/desktopdefault.aspx/tabid-10443/637_read-251/#/gallery/8570. Accessed on: April 2018.

[10] Christian Portilla, Andrés Acosta, Jairo Espinosa, “SciTraffic: A Macroscopic Simulator for Estimation and Control of Urban Traffic,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 71–80, 2017.

[11] Michael Wittmann, “Nachfrageorientierte Agentensimulation zur Bestimmung des Elektrifizierungspotentials von Fahrzeugflotten,” Master Thesis, Lehrstuhl für Fahrzeugtechnik, Technische Universität München, München, 2015.

[12] Yali Yang, Hao Chen, Lihua Chen, “Evaluation of public transportation system in Shanghai, China,” (eng), *2010 International Conference On*, vol. 2, pp. 197–199, 2010.

[13] *SUMO – Simulation of Urban MObility.* [Online] Available: www.dlr.de/ts/desktopdefault.aspx/tabid-9883/16931_read-41000/.

[14] Serge P. Hoogendoorn, Piet H.L. Bovy, “State-of-the-art of Vehicular Traffic Flow Modelling: Special Issue on Road Traffic Modelling and Control of the Journal of Systems and Control Engineering.,” 2001.

[15] Wilco Burghout, “Hybrid microscopic-mesoscopic traffic simulation,” PhD Thesis, Department of Infrastructure - Division of Transportation & Logistics, Royal Institute of Technology, Stockolm, 2004.

[16] *Sumo at a Glance.* [Online] Available: http://sumo.dlr.de/wiki/Sumo_at_a_Glance. Accessed on: April 2018.

[17] *SUMO User Documentation.* [Online] Available: http://sumo.dlr.de/wiki/SUMO_User_Documentation. Accessed on: April 2018.

[18] José Capela Dias, Pedro Henriques Abreu, Daniel Castro Silva, Gustavo Fernades, Penousal Machado, António Leitão, “Preparing Data for Urban Traffic Simulation using SUMO,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 21, no. 1866-721X, pp. 25–34, 2013.

[19] *FAQ: How to simulate an accident.* [Online] Available: http://sumo.dlr.de/wiki/FAQ#How_to_simulate_an_accident. Accessed on: April 2018.

[20] Daniel Krajzewicz, “Summary on Publications citing SUMO, 2002-2012,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 21, no. ISSN 1866-721X, pp. 11–24, 2013.

[21] Robbin Blokpoel, “Network Conversion for SUMO Integration,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 24, no. 1866-721X, pp. 35–44, 2014.

[22] *MAVEN: Managing Automated Vehicles Enhances Network.* [Online] Available: http://www.maven-its.eu/.

[23] P. P. Karl-Heinz Kastner, “TOMS – Traffic Online Monitoring System for ITS Austria Wests,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 24, no. 1866-721X, pp. 157–166, 2014.

[24] Iván Y. Hernández-Paniagua, Juan C. Zavala-Reyes, Pablo López-Ramírez, Ulises Diego-Ayala, Irma Rosas, Arón Jazcilevich, “Use of SUMO to assess human exposure and intake to PM2.5 near motorways: Preliminary Results,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 63–70, 2017.

[25] *VABENE++.* [Online] Available: https://verkehrsforschung.dlr.de/en/projects/vabene. Accessed on: Aug. 31 2018.

[26] *TraCI.* [Online] Available: http://sumo.dlr.de/wiki/TraCI. Accessed on: Aug. 31 2018.

[27] *TraCI/Interfacing TraCI from Python.* [Online] Available: http://sumo.dlr.de/wiki/TraCI/Interfacing_TraCI_from_Python. Accessed on: Aug. 31 2018.

[28] X. M. Junchen Jin, “Stochastic Optimization of Signal Plan and Coordination using Parallelized Traffic Simulation,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 24, no. 1866-721X, pp. 115–126, 2014.

[29] Laura Bieker, Daniel Krajzewicz, Antonio Pio Morra, Carlo Michelacci, Fabio Cartolano, “Traffic simulation for all: a real world traffic scenario from the city of Bologna,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 24, no. 1866-721X, pp. 19–26, 2014.

[30] Gerhard Hermanns, Thomas Zaksek, Michael Schreckenberg, “A Use Case for SUMO: Simulating Traffic around the Port of Duisburg,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 21, no. 1866-721X, pp. 35–42, 2013.

[31] TeRon V. Nguyen, Daniel Krajzewicz, Matthew Fullerton, Eric Nicolay, “DFROUTER—Estimation of Vehicle Routes from Cross-Section Measurements,” pp. 3–23, 2015.

[32] Lara Codeca, Raphael Frank, Thomas Engel, “LuST: a 24-hour Scenario of Luxembourg City for SUMO Traffic simulations,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 28, no. 1866-721X, pp. 11–22, 2015.

[33] Andreas Richter, Hartmut Friedl, Michael Scholz, “Beyond OSM – Alternative Data Sources and Approaches Enhancing Generation of Road Networks for Traffic and Driving Simulations,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 30, no. 1866-721X, pp. 23–32, 2016.

[34] *OpenStreetMapWiki.* [Online] Available: https://wiki.openstreetmap.org/wiki/Main_Page. Accessed on: Aug. 25 2018.

[35] Mordechai Haklay, “How good is volunteered geographical information?: A comparative study of OpenStreetMap and Ordnance Survey datasets,” *Environment and Planning B: Planning and Design*, vol. 37, pp. 682–703, 2010.

[36] Rainer Gasper, Stephan Beutelschieß, Mario Krumnow, Levente Simon, Zoltan Baksa, Jochen Schwarzer, “Simulation of Autonomous RoboShuttles in Shared Space,” 2018.

[37] Sabina Alazzawi, Mathias Hummel, Pascal Kordt, “Simulating the Impact of Shared, Autonomous Vehicles on Urban Mobility - A Case Study of Milan,” 2018.

[38] M. M. Joschka Bischoff, “Simulation of City-wide Replacement of Private Cars with Autonomous Taxis in Berlin,” *Procedia Computer Science*, vol. 83, pp. 237–244, 2016.

[39] Fares Maximilian Mrad Agua, “Study of a Shared Autonomous Vehicles Based Mobility Solution for Munich: **Investigation of Operational and Economic Feasibility**,” München, Chair of Automotive Technology, Technische Universität München, 2016.

[40] *MVV Our Concept: 1 network. 1 timetable. 1 ticket.* [Online] Available: https://www.mvv-muenchen.de/en/the-mvv/der-verbund/our-concept/index.html. Accessed on: Sep. 03 2018.

[41] “MVG in figures,” https://web.archive.org/web/20131004215908/http://www.mvg-mobil.de/ueberuns/images/mvg_in_figures_s.pdf, 2013.

[42] *moovit: the #1 Transit App Worldwide.* [Online] Available: https://www.company.moovit.com/?_ga=2.219190937.1235153007.1535970862-1557699811.1535970862&utm_medium=Organic&utm_source=web_app. Accessed on: Sep. 03 2018.

[43] *GTT: con gtt e moovit tutte le informazioni a portata di smartphone.* [Online] Available: http://www.gtt.to.it/cms/cms/notizie-eventi-e-informazioni/2777-con-gtt-e-moovit-tutte-le-informazioni-a-portata-di-smartphone. Accessed on: Sep. 03 2018.

[44] *moovit: Moovit's Mobility as a Service (MaaS) Platform.* [Online] Available: https://www.solutions.moovitapp.com/. Accessed on: Sep. 03 2018.

[45] *moovit INSIGHTS: Facts and usage statistics about public transit in Munich, Germany.* [Online] Available: https://moovitapp.com/insights/en/Moovit_Insights_Public_Transit_Index_Germany_Munchen-3144#. Accessed on: Sep. 03 2018.

[46] Laura Lyne, *Dubliners facing long wait times for taxis due to rising insurance.* [Online] Available: https://www.dublinlive.ie/news/dublin-news/dubliners-facing-long-wait-times-13858217. Accessed on: Sep. 03 2018.

[47] japan-guide.com, *Taxi.* [Online] Available: https://www.japan-guide.com/e/e2021.html. Accessed on: Sep. 03 2018.

[48] Eric Jaffe, *People in a Hurry Choose Uber Over Traditional Cabs.* [Online] Available: https://www.citylab.com/life/2014/08/uber-has-an-enormous-wait-time-advantage-over-regular-taxis/379358/. Accessed on: Sep. 03 2018.

[49] L. Rayle, D. Dai, N. Chan, R. Cervero, and S. Shaheen, “Just a better taxi? A survey-based comparison of taxis, transit, and ridesourcing services in San Francisco,” *Transport Policy*, vol. 45, pp. 168–178, 2016.

[50] *Google Maps: Marienplatz.* [Online] Available: https://www.google.it/maps/place/Marienplatz/@48.1394016,11.5706855,16.77z/data=!4m5!3m4!1s0x47a84e248d37632d:0xdead51b35f0e0bb3!8m2!3d48.1373932!4d11.5754485?hl=it. Accessed on: Sep. 01 2018.

[51] Landeshauptstadt München Direktorium, “Munich Facts and Figures,” 2017.

[52] “Munich Metropolitan Region,” 2012.

[53] *Stadtwerke Muenchen: SWM Renewable Energies expansion campaign.* [Online] Available: https://www.swm.de/english/company/energy-generation/renewable-energies.html. Accessed on: Sep. 14 2018.

[54] *Developer/Main.* [Online] Available: http://sumo.dlr.de/wiki/Developer/Main. Accessed on: Sep. 01 2018.

[55] José Luis Ferrás Pereira, “An Integrated Architecture for Autonomous Vehicles Simulation,” Master Thesis, Faculdade de Engenharia, Universidade Do Porto, Porto, 2011.

[56] *Planet.osm.* [Online] Available: https://wiki.openstreetmap.org/wiki/Planet.osm. Accessed on: Sep. 01 2018.

[57] *Definition of Vehicles, Vehicle Types, and Routes: Vehicle Types.* [Online] Available: http://sumo.dlr.de/wiki/Definition_of_Vehicles,_Vehicle_Types,_and_Routes#Vehicle_Types. Accessed on: Sep. 01 2018.

[58] François Vaudrin, Jakob Erdmann, Laurence Capus, “Impact of Autonomous Vehicles in an Urban Environment Controlled by Static Traffic Lights System,” *Berichte aus dem DLR-Institut für Verkehrssystemtechnik*, vol. 31, no. 1866-721X, pp. 81–92, 2017.

[59] *ACTIVITYGEN.* [Online] Available: http://sumo.dlr.de/wiki/ACTIVITYGEN. Accessed on: Sep. 01 2018.

[60] *Demand/Activity-based Demand Generation.* [Online] Available: http://sumo.dlr.de/wiki/Demand/Activity-based_Demand_Generation. Accessed on: Sep. 01 2018.

[61] *DUAROUTER.* [Online] Available: http://sumo.dlr.de/wiki/DUAROUTER. Accessed on: Sep. 01 2018.

[62] *Demand/Dynamic User Assignment: Iterative Assignment (Dynamic User Equilibrium).* [Online] Available: http://www.sumo.dlr.de/wiki/Demand/Dynamic_User_Assignment. Accessed on: Sep. 01 2018.

[63] A. L. Marta Gonzalez, *Recalculating! By not driving the optimal route, you're causing traffic jams.* [Online] Available: http://theconversation.com/recalculating-by-not-driving-the-optimal-route-youre-causing-traffic-jams-56135. Accessed on: Sep. 05 2018.

[64] A. J. Khattak, J. L. Schofer, and F. S. Koppelman, "Effect of traffic information on commuters' propensity to change route and departure time," *ATR*, vol. 29, no. 2, pp. 193–212, 1995.

[65] *TOMTOM city: Munich.* [Online] Available: https://www.tomtom.com/en_gb/traffic-news/munich-traffic/traffic-flow. Accessed on: Sep. 11 2018.

[66] E. B. David Welch, "Who's Winning the Self-Driving Car Race?: A scorecard breaking down everyone from Alphabet's Waymo to Zoox.," *Bloomberg*, 07 May., 2018, https://www.bloomberg.com/news/features/2018-05-07/who-s-winning-the-self-driving-car-race.

[67] Stephen Edelstein, *Fully-Autonomous Cars Will Cost 'Hundreds of Thousands' of Dollars, Silicon Valley Exec Says.* [Online] Available: http://www.thedrive.com/tech/12856/fully-autonomous-cars-will-cost-hundreds-of-thousands-of-dollars-silicon-valley-exec-says. Accessed on: Sep. 12 2018.

[68] Paul Lienert, *Self-driving costs could drop 90 percent by 2025, Delphi CEO says.* [Online] Available: https://www.reuters.com/article/us-autos-delphi/self-driving-costs-could-drop-90-percent-by-2025-delphi-ceo-says-idUSKBN1DY2AC. Accessed on: Sep. 12 2017.

[69] P. M. Bösch, F. Becker, H. Becker, and K. W. Axhausen, "Cost-based analysis of autonomous mobility services," *Transport Policy*, vol. 64, pp. 76–91, 2018.

[70] *Volkswagen e-Golf.* [Online] Available: https://ev-database.uk/car/1087/Volkswagen-e-Golf. Accessed on: Sep. 12 2018.

[71] Eurostat, *Electricity prices for non-household consumers - bi-annual data (from 2007 onwards).* [Online] Available: http://appsso.eurostat.ec.europa.eu/nui/submitViewTableAction.do. Accessed on: Sep. 12 2018.

[72] Volkswagen, *Der e-golf.* [Online] Available: https://www.volkswagen.de/de/models/e-golf.html. Accessed on: Sep. 12 2018.

[73] Union of Concerned Scientists, *Electric Vehicle Battery: Materials, Cost, Lifespan.* [Online] Available: https://www.ucsusa.org/clean-vehicles/electric-vehicles/electric-cars-battery-life-materials-cost#.W5kIvPZuKIU. Accessed on: Sep. 12 2018.

[74] *Autoversicherung-Vergleich.* [Online] Available: https://www.autoversicherung-vergleich.info/vergleich. Accessed on: Sep. 12 2018.

[75] “ADAC Autokosten,” https://www.adac.de/_mmm/pdf/autokostenuebersicht_47085.pdf.

Appendix A

Data related to Wednesday, with no limitation in stand capacity.

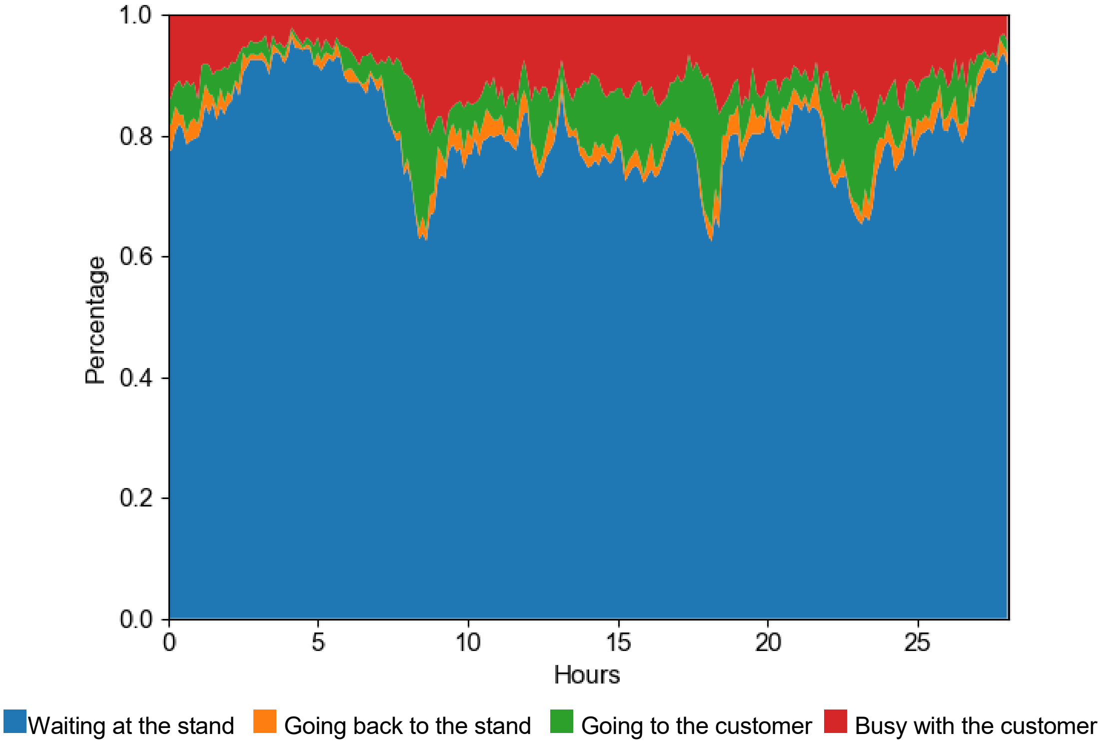

Figure A - 1 100% stacked area chart describing the activity of the 321 taxi fleet

Figure A - 2 100% stacked area chart describing the activity of the 428 taxi fleet

Figure A - 3 100% stacked area chart describing the activity of the 535 taxi fleet

Appendix B

Data related to Wednesday, taking into account stand capacity and an overall capacity of 35 vehicles at the airport.

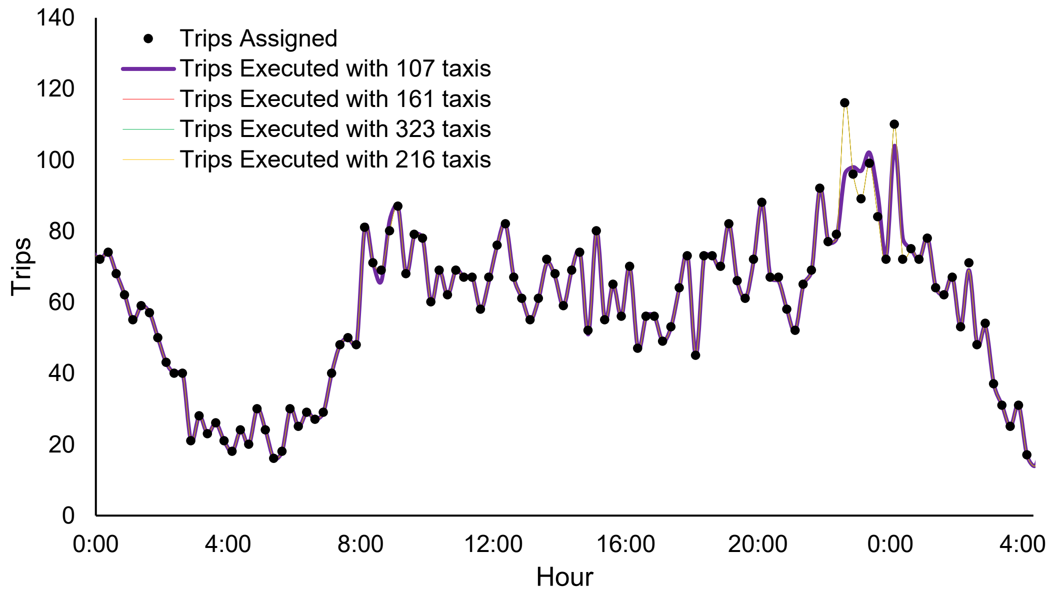

Figure B - 1 Evolution of the demand on Wednesday and the ability of the different fleets to satisfy it

Table B - 1 Average differences in times and distances concerning trips with customer on board

Comparison	Data compared	Average difference
Input – 214 fleet	Time	-35.1%
	Distance	-3.9%
214 fleet – 107 fleet	Time	0.8%
	Distance	-0.5%
214 fleet – 161 fleet	Time	0.0%
	Distance	-0.3%
214 fleet – 321 fleet	Time	-4.5%
	Distance	-0.4%

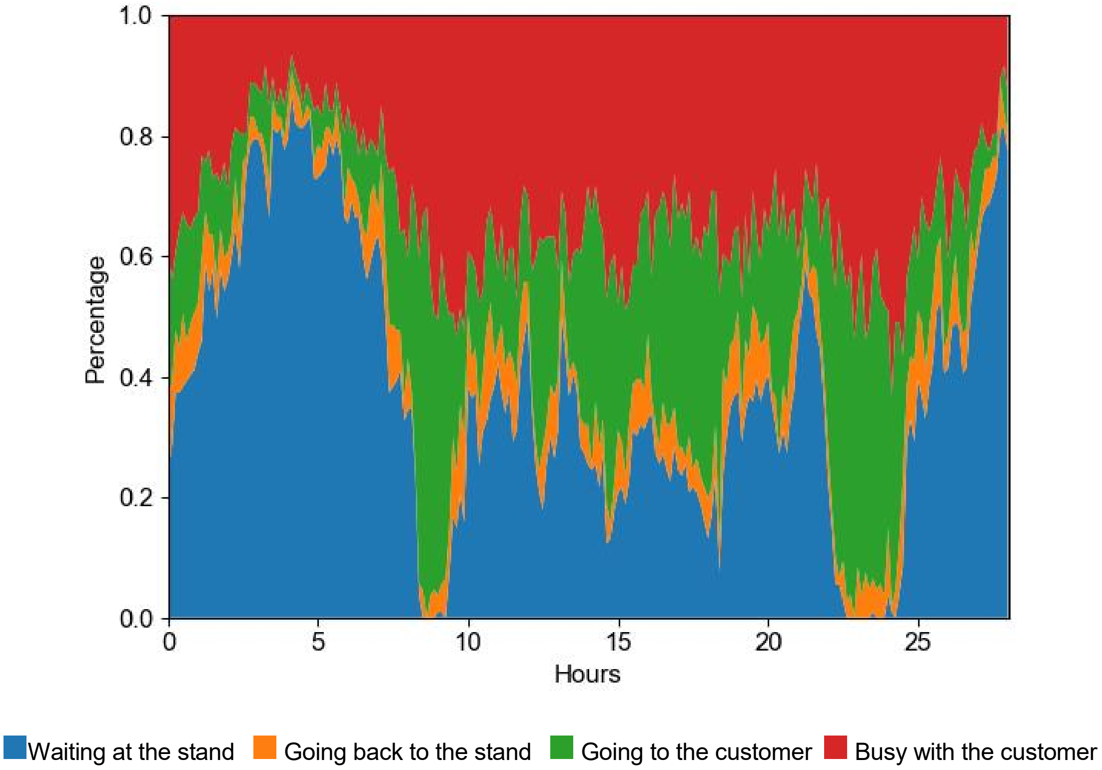

Figure B - 2 100% stacked area chart describing the activity of the 107 taxi fleet

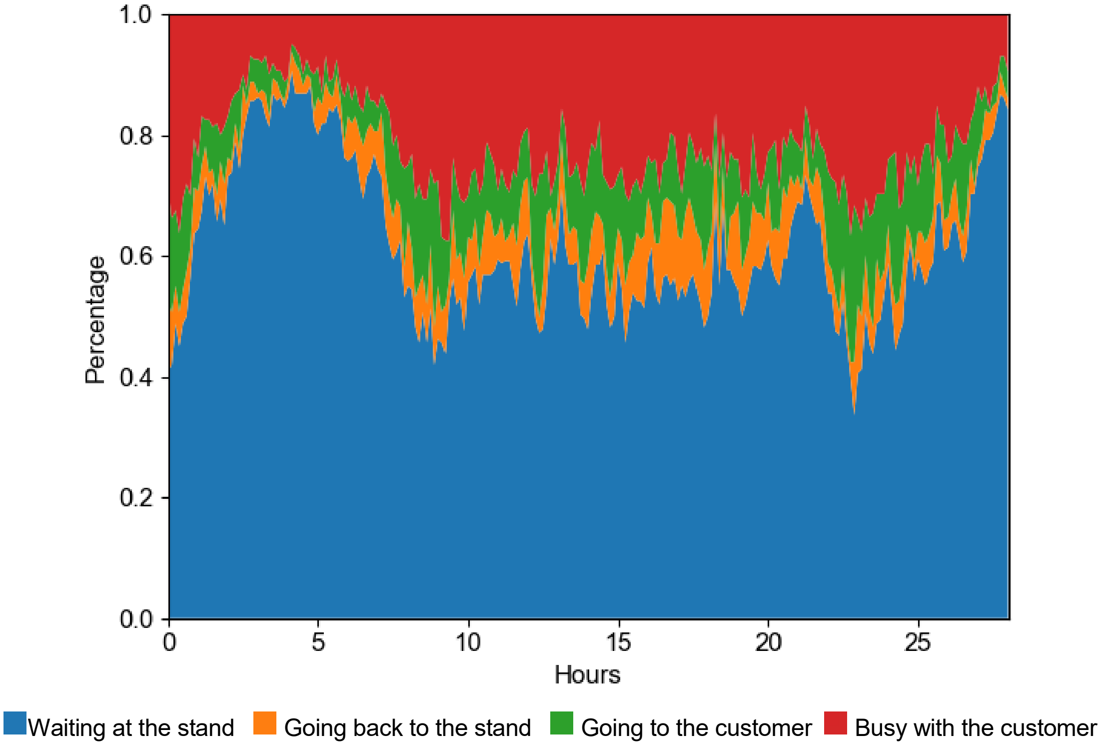

Figure B - 3 100% stacked area chart describing the activity of the 161 taxi fleet

Figure B - 4 100% stacked area chart describing the activity of the 321 taxi fleet

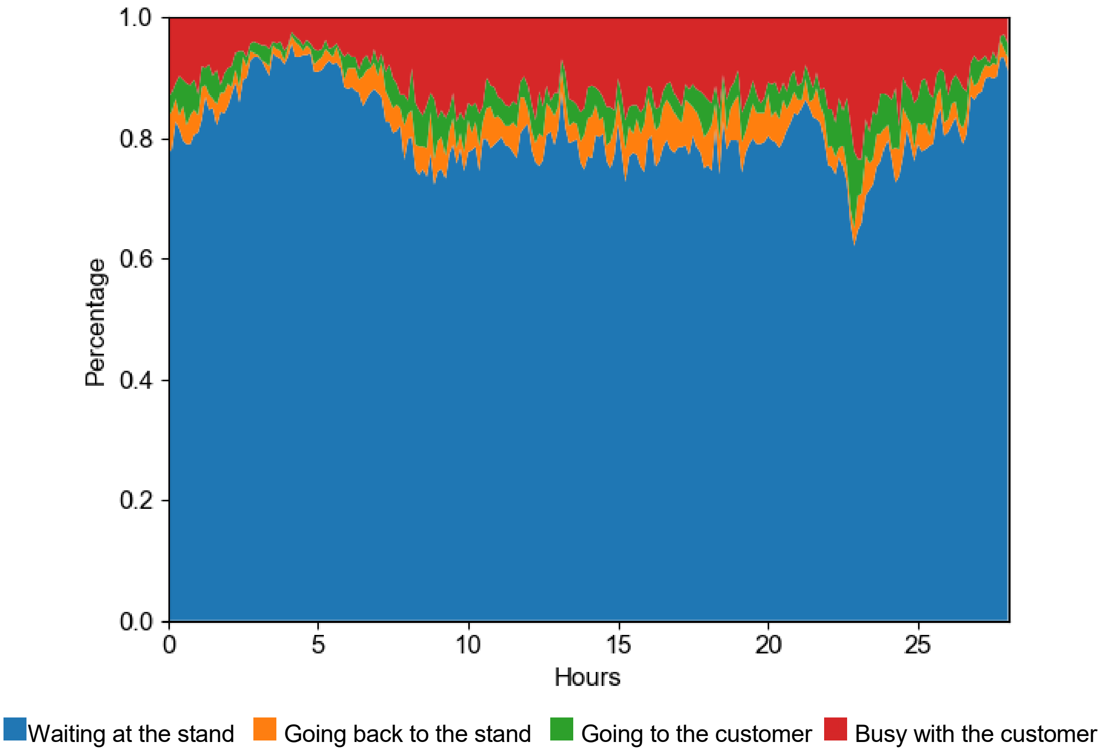

Figure B - 5 100% stacked area chart describing the activity of the 321 taxi fleet

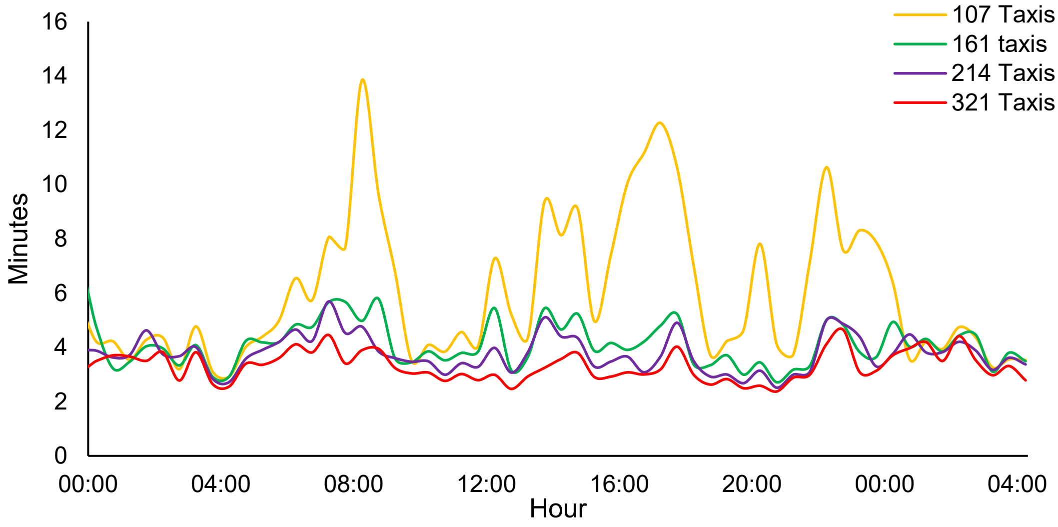

Figure B - 6 Evolution of the average of the waiting time calculated every half of hour

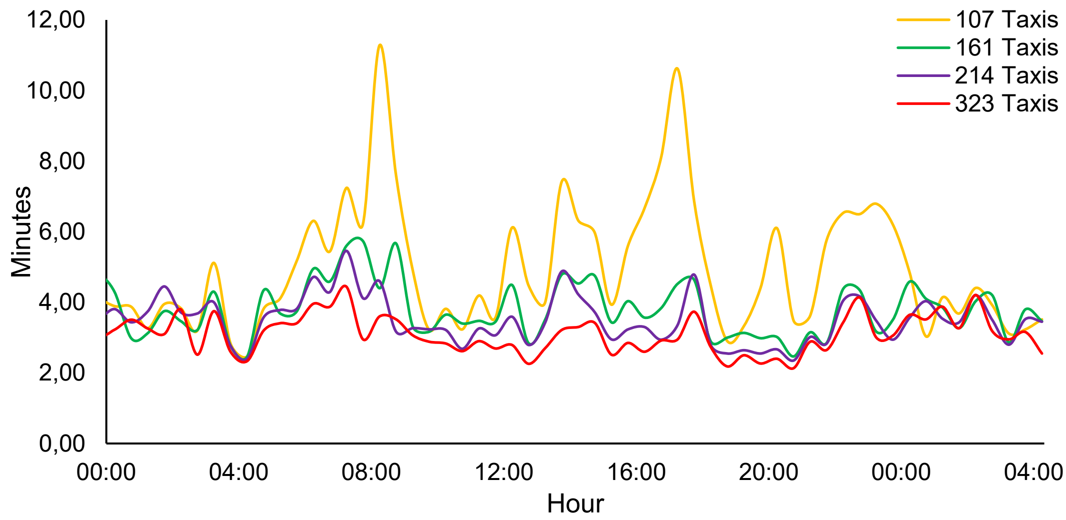

Figure B - 7 Evolution of the median of the waiting time calculated every half of hour

Appendix C

Data referring to Saturday with an overall stand capacity of 35 vehicles at the airport

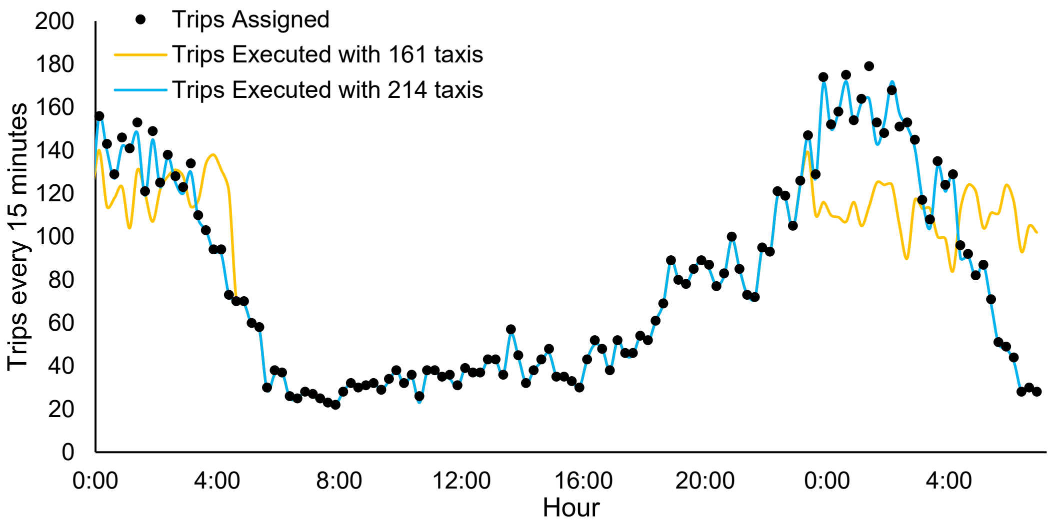

Figure C - 1 Evolution of the demand on Saturday and the ability of the different fleets to satisfy it

Table C - 1 Average differences in times and distances concerning trips with customer on board

Comparison	**Data compared**	**Average difference**
Input – 214 fleet	Time	-23.1%
	Distance	-4.3%
214 fleet – 161 fleet	Time	1.7%
	Distance	-4.1%

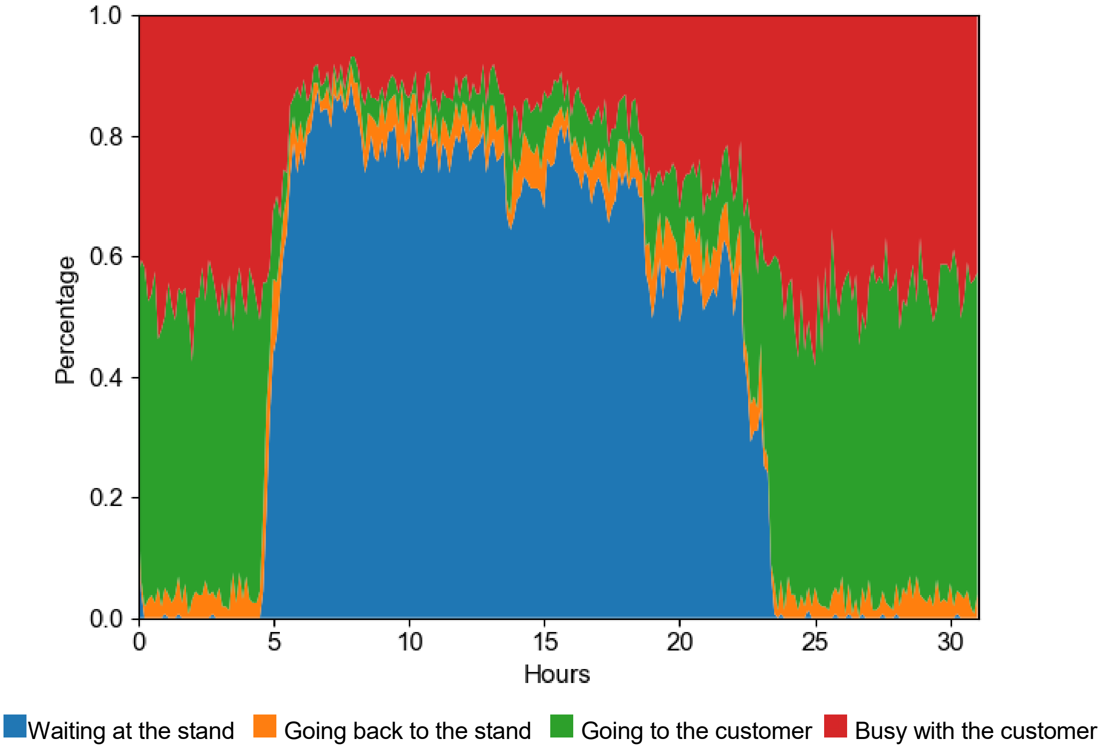

Figure C - 2 100% stacked area chart describing the activity of the 161 taxi fleet

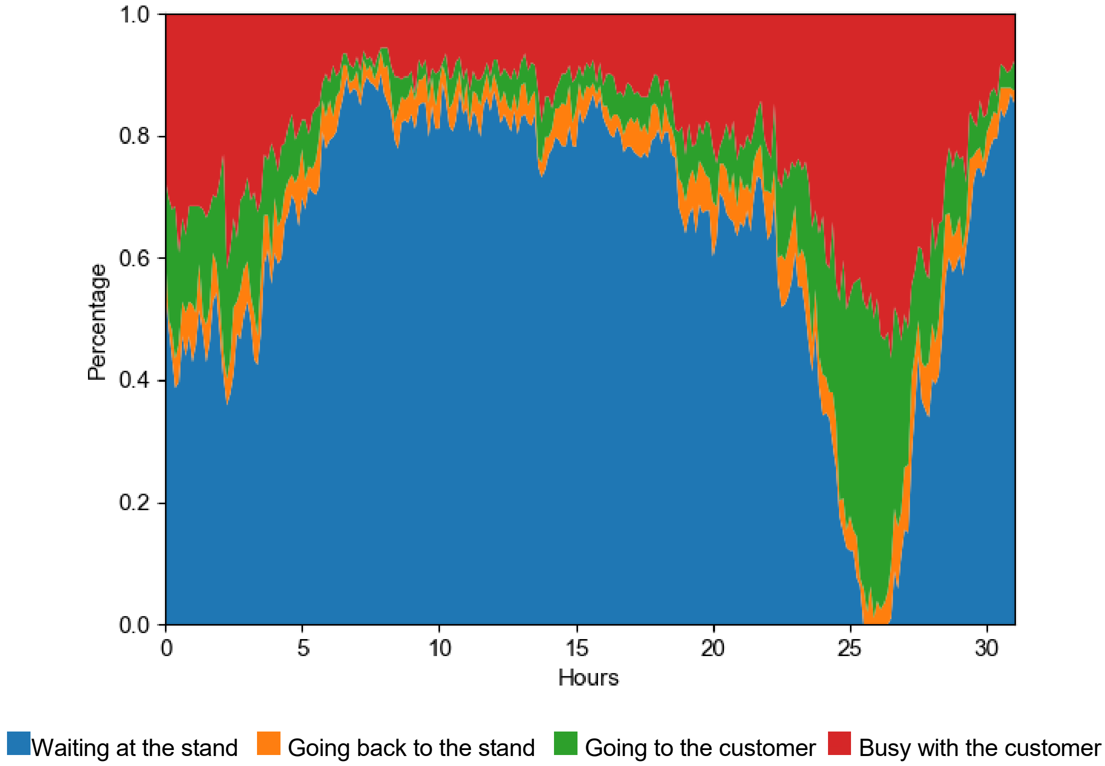

Figure C - 3 100% stacked area chart describing the activity of the 214 taxi fleet

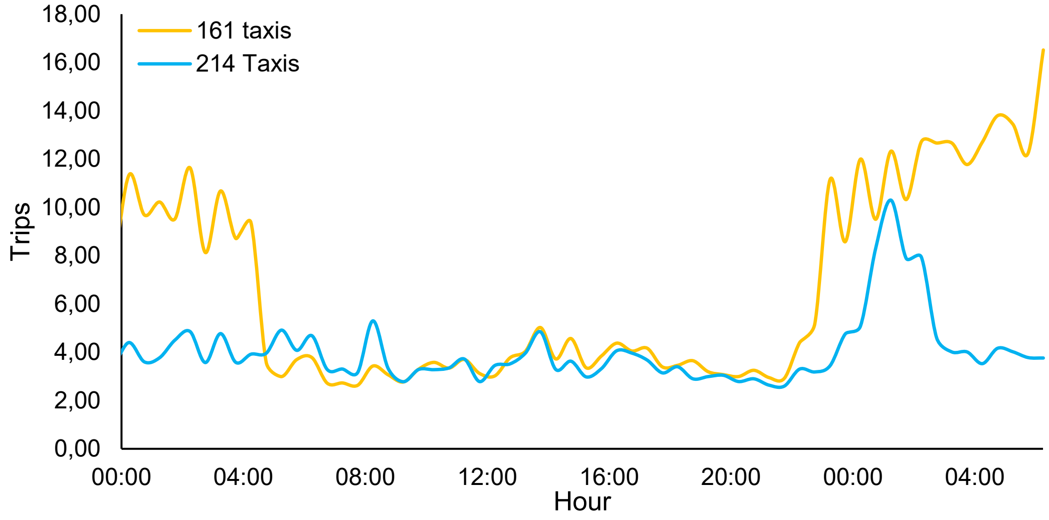

Figure C - 4 Evolution of the average of the waiting time calculated every half of hour

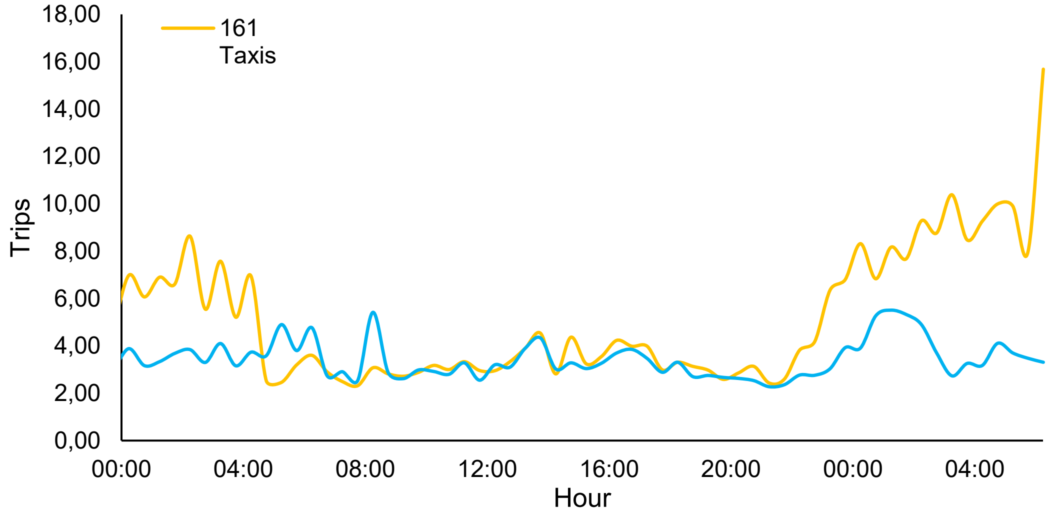

Figure C - 5 Evolution of the median of the waiting time calculated every half of hour

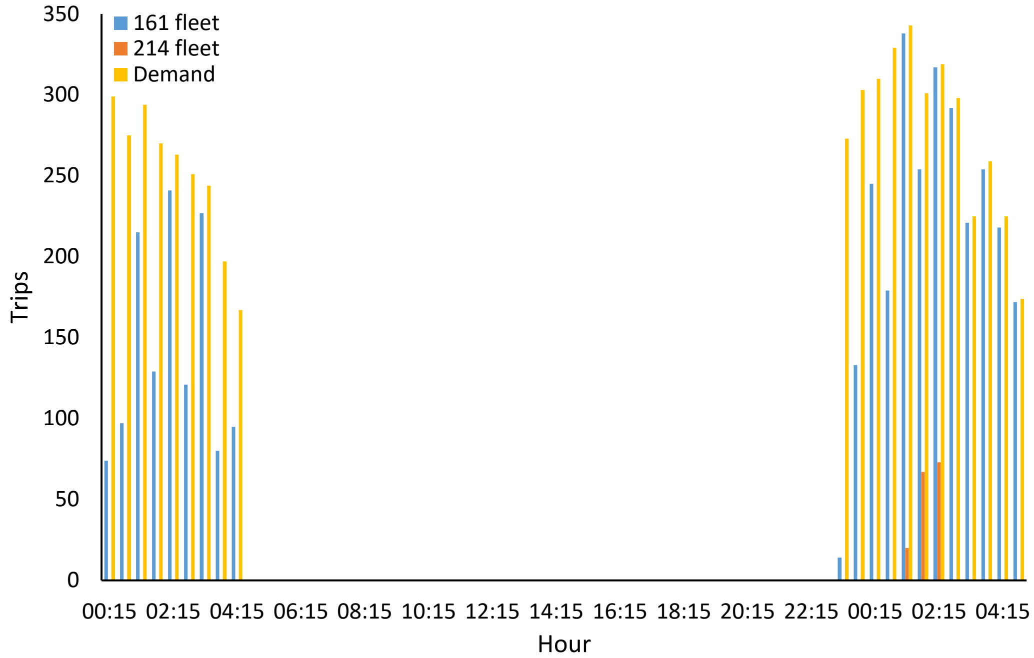

Figure C - 6 Comparison of trips postponed with the two fleets and the demand in the same time intervals

Appendix D

Data referring to Saturday with an overall stand capacity of 3 vehicles at the airport

Table D - 1 Average differences in times and distances concerning trips with customer on board

Comparison	**Data compared**	**Average difference**
Input – 214 fleet	Time	-28.35%
	Distance	-4.0%
214 fleet – 161 fleet	Time	-0.6%
	Distance	-0.1%