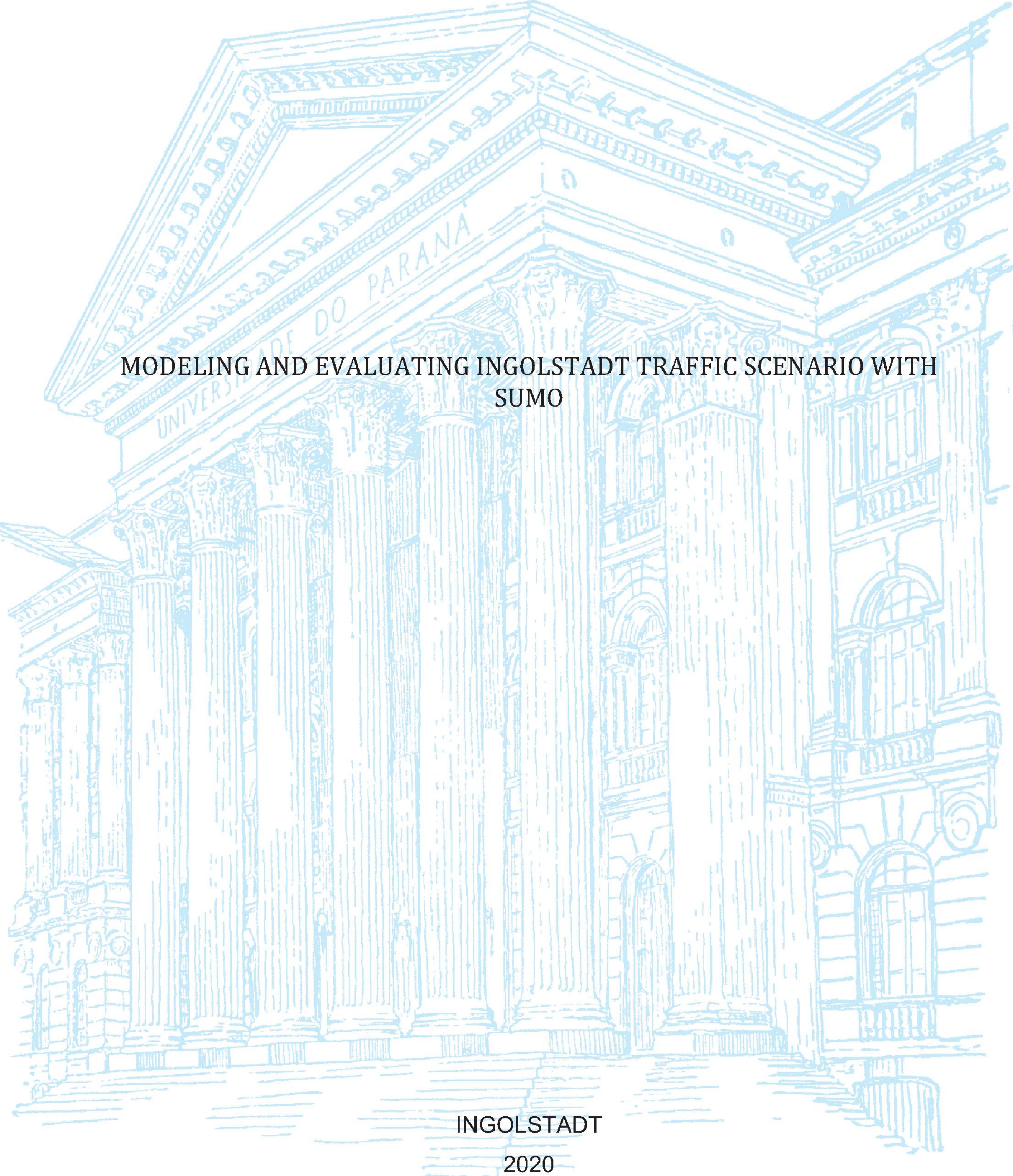
UNIVERSIDADE FEDERAL DO PARANÁ

SILAS CORREIA LOBO

# MODELING AND EVALUATING INGOLSTADT TRAFFIC SCENARIO WITH SUMO




INGOLSTADT

2020

SILAS CORREIA LOBO

# MODELING AND EVALUATING INGOLSTADT TRAFFIC SCENARIO WITH SUMO

Dissertação apresentada aos cursos de Pós-Graduação em Engenharia Elétrica, Setor de Ciências Exatas, Universidade Federal do Paraná e em Engenharia Automotiva Internacional, Faculdade de Engenharia Elétrica e Ciências da Computação, Technische Hochschule Ingolstadt como requisito parcial à obtenção dos títulos de Mestre em Engenharia Elétrica e Master em Engenharia Automotiva.
Orientador: Prof. Dr. Evelio Martín García Fernández
Coorientador: Prof. Dr. Christian Facchi

INGOLSTADT

2020

Catalogação na Fonte: Sistema de Bibliotecas, UFPR
Biblioteca de Ciência e Tecnologia

F949m Lobo, Silas Correia
Modeling and evaluating Ingolstadt traffic scenario with SUMO [recurso eletrônico] / Silas Correia Lobo. – Curitiba, 2020.

Dissertação - Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, 2020.

Orientador: Evelio Martín García Fernández.
Coorientador: Christian Facchi.

1. Sistemas de comunicação móvel. 2. Veículos. I. Universidade Federal do Paraná. II. Fernández, Evelio Martín García. III. Facchi, Christian. IV. Título.

CDD: 621.382

Bibliotecária: Vanusa Maciel CRB- 9/1928

# TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ENGENHARIA ELÉTRICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **SILAS CORREIA LOBO** intitulada: **Modeling and Evaluating Ingolstadt Traffic Scenario with SUMO**, sob orientação do Prof. Dr. EVELIO MARTÍN GARCÍA FERNÁNDEZ, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 16 de Julho de 2020.

Assinatura Eletrônica
21/07/2020 18:40:13.0
EVELIO MARTÍN GARCÍA FERNÁNDEZ
Presidente da Banca Examinadora

Assinatura Eletrônica
22/07/2020 11:42:26.0
LUIS HENRIQUE ASSUMPÇÃO LOLIS
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
23/07/2020 08:39:36.0
HORACIO TERTULIANO DOS SANTOS FILHO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
22/07/2020 18:06:13.0
KEIKO VERONICA ONO FONSECA
Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

*To my lovely wife and my supportive parents.*

# ACKNOWLEDGEMENTS

I would like to express all my gratitude to God, for His support during the period of challenges experienced while I have written this Thesis.

I thank Prof. Dr. Christian Facchi, who since our first contact showed me that I could go much further as I believed, encouraging and supporting me with his passion. I also thank him for his friendship and persistence.

I am thankful to Prof. Dr. Evelio Martín García Fernández, who despite the difficulties faced by the long-distance, was always prepared to guide my research and answer my doubts.

I would also like to thank Stefan Neumeier for his fully support and all my colleagues in Car2X Lab at Technische Hochschule Ingolstadt for their friendship and support whenever needed.

I can never forget my parents José Arly Pimenta Lobo and Elisabete Lobo, who always encourage me to take the next step and taught me that knowledge is ever the best investment we have in life.

I am also thankful for my lovely wife Alline Cristina Schüncke Lobo, who supported me with all strength during this great journey.

# RESUMO

Vehicular Ad-hoc Networks (VANETs), também conhecida como comunicação Car2X, tem surgido nos últimos anos com uma das áreas mais promissoras para tecnologias automobilísticas. Elas melhoram a segurança nas ruas, podem resolver problemas de congestionamento de tráfego e são esperada como uma das chaves principais para permitir veículos com o mais alto grau de autonomia. Devido a isso, muitos estudos estão focando no desenvolvimento de novas aplicações, incluindo *infotainment*, gerenciamento de tráfego e segurança. Observando a área de segurança, o desenvolvimento de novas aplicações demanda um grande número de testes. Os testes devem cobrir e analisar todas as situações possíveis relacionadas a tráfego veicular, para garantir que essas aplicações não irão impactar na segurança dos sistemas de transportes nem colocar vidas humanas em risco. Por esse motivo, ferramentas de simulação são altamente demandas e importantes para desenvolver os testes. Simulações de VANETS necessitam de dois *frameworks*: modelo de tráfego veicular e protocolo de simulação de comunicação sem fio. Quanto mais realístico o cenário de tráfego for, mais confiáveis e úteis serão os resultados. Portanto, essa pesquisa apresenta um cenário de tráfego realístico para a cidade de Ingolstadt, aplicando as metodologias do *Simulation of Urban Mobility* (SUMO). O mapa da cidade e todas as informações relacionadas, como posição dos pontos de ônibus, construções e estacionamentos foram importados do OpenStreetMap (OSM). As topologias das ruas foram fielmente ajustadas comparando as imagens de satélite do Google Maps. Esse ajuste corrigiu o número de faixas por ruas, adicionou faixas exclusivas para virar a esquerda, bem como faixas exclusivas para ônibus e táxis. O tráfego veicular foi modelado de acordo com o `activitygen`, que demanda detalhadas informações sócio-demográficas, e também foi considerado os trabalhadores e estacionamentos das grandes empresas da cidade. Além disso, para a demanda de tráfego foram inseridos todas as linhas de ônibus que atualmente trafegam por Ingolstadt. Um método para otimizar o fluxo de tráfego foi implementado, aplicando o *Dynamic User Assignment* (DUA) focando no *Dynamic User Equilibrium* (DUE), o qual foi atingido após 25 iterações. Como os parâmetros de simulação influenciam o cenário de tráfego do SUMO, esta tese implementou um algoritmo para calcular o melhor valor para *device.rerouting.probability*. Para definir o melhor valor para este parâmetro, um conjunto de dados fornecido pela cidade de Ingolstadt, contendo valores reais de tráfego para 24 pontos de medição de tráfego, foi implementado para comparar valores de simulação e valores reais. O resultado dessa comparação foi baseado no resultado fornecido pelos detectores colocados na simulação e no conjunto de dados. O conjunto de dados foi dividido em dois subconjuntos. Um subconjunto foi implementado para definir a melhor *device.rerouting.probability*, que considerou o número médio de veículos em trânsito nas 24 pontos de medição de tráfego durante outubro de 2019. O segundo subconjunto foi implementado para validar os resultados do cenário, considerando o valores de tráfego de novembro de 2019. O método de avaliação aplicado em ambas as etapas foi o Erro Médio Quadrático Normalizado (NRMSE). O menor NRMSE calculado nesta tese em relação à *device.rerouting.probability* é 0,438234. A validação também levou em consideração o valor de NRMSE individualmente para todas as 24 pontos de medição de tráfego e uma análise mais profunda para o melhor e o pior caso foi desenvolvida.

**Palavras-chave:** VANETs, Car2X, cenário de tráfego realístico, SUMO, Ingolstadt

## ABSTRACT

Vehicular Ad-hoc Networks (VANETs), also known as Vehicle2X communication, have emerged as one of the most promising branches in automotive technologies in the last years. The development of new applications for VANETs demands a large number of tests, which have to analyze all the situations regarded to vehicle traffic, to ensure that the novel application will not impact transport system safety either risk human lives. Thus, to perform these tests simulation tools are used. VANETs' simulations require two frameworks: vehicle traffic model and wireless communication protocol. The more realistic the traffic scenario is, the more reliable and useful the results are. Therefore, this thesis presents a realistic traffic scenario for Ingolstadt city, applying the methodologies presented in the Simulation of Urban Mobility (SUMO). The city map and all related data, as bus stop positions, buildings, and parking lots were imported from OpenStreetMap (OSM). The road topology was adjusted according to images from the Google Maps satellite view. The vehicle traffic has been modeled based on detailed socio-demographic data, considering also the largest companies in the city with their workers and placing the parking lots. Furthermore, traffic demand inserted all the bus lines that are currently running in Ingolstadt. Thirteen traffic light systems were simulated in this thesis based on the real program deployed on real traffic lights in Ingolstadt. A method to optimize the traffic flow was implemented, applying the Dynamic User Assignment (DUA) focusing on the Dynamic User Equilibrium (DUE), which was reached after 25 iterations. As simulation parameters influence SUMO's traffic scenario, this thesis implemented an algorithm to compute the best value for *device.rerouting.probability*. To define the best value for this parameter, a data-set provided by the City of Ingolstadt, containing real traffic values for 24 intersections was implemented to compare simulation values and real values. The result of this comparison was based on the output delivered by detectors placed in the simulation and the data-set. The data-set has been divided into two subsets. One subset has been implemented to define the best *device.rerouting.probability*, which considered the average number of vehicles transiting over the 24 intersections during October of 2019. The second subset has been implemented to validate the scenario results, considering the traffic values from November of 2019. The evaluation method applied in both steps was the Normalized Root Mean Square Error (NRMSE). The lowest NRMSE computed in this thesis regarding the *device.rerouting.probability* is 0.438234. The validation also took into consideration the individually NRMSE value for all 24 intersections and a deeper analysis for the best and worst-case was done.

**Keywords:** VANETs, Car2X, realistic traffic scenario, SUMO, Ingolstadt

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **CDF** | Cumulative Distribution Function |
| **DLR** | Deutsches Zentrum für Luft- und Raumfahrt |
| **DUA** | Dynamic User Assignment |
| **DUE** | Dynamic User Equilibrium |
| **INVG** | Ingolstädter Verkehrsgesellschaft |
| **ITS** | Intelligent Transport System |
| **JOSM** | Java OpenStreetMap |
| **LUST** | Luxembourg SUMO Traffic |
| **MOST** | Monaco SUMO Traffic Scenario |
| **ODM** | Origin and Destination Matrix |
| **OSM** | OpenStreetMap |
| **POI** | Point of Interests |
| **RMSE** | Root Mean Square Error |
| **RSU** | Road Side Unit |
| **RT** | Real Trace |
| **ST** | Simulation Trace |
| **SUMO** | Simulation of Urban Mobility |
| **TAPAS** | Travel Activity Pattern Simulation |
| **TAZ** | Traffic Assignment Zone |
| **TL** | Traffic Light |

| | |
|---|---|
| **TLS** | Traffic Light System |
| **VANET** | Vehicular Ad-hoc Network |
| **V2I** | Vehicle-to-Infrastructure |
| **V2V** | Vehicle-to-Vehicle |
| **V2X** | Vehicle-to-Everything |

# CONTENTS

# CHAPTER 1

# INTRODUCTION

Over the last years, the number of vehicles on the streets has grown continuously, and by the year of 2050 it is expected to reach the mark of 2.4 billion units, representing a 140% rise compared to the year 2015 (OECD-ITF, 2017). More vehicles on streets increase traffic issues, and to reduce this impact the Intelligent Transport System (ITS) is emerging with new features. ITS is a technological solution which manages information between on-road vehicles, intending to improve transport safety, mobility, and efficiency (MENEGUETTE et al., 2018). ITS' features have a wide usage from simple applications as traffic signal systems control, managing car navigation system, car plate recognition, speed cameras, to more complex applications as parking guidance system, road weather information, and collision avoidance system. Aiming to expand ITS usage, a significant wireless communication solution is necessary to provide opportunities suitable to attend demands performing real-time data, developing a new research area responsible to set communication between all users.

Vehicular Ad-hoc Networks (VANETs) have emerged as one of the most promising automotive technologies in the last years. In a near-future, it is expected that VANETs will be a key enabling technology for autonomous driving, improve road safety, notify drivers about a hazard situation, decrease traffic congestion issues, and many other applications (MIUCIC, 2019). VANETs are cooperative vehicular networks based on wireless communication among vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), also known as Road Side Unit (RSU), and vehicle-to-everything (V2X), where X represents the communication among vehicles and any other device. Among these benefits provided by VANETs, the capacity to share pertinent information on-road situations with other vehicles and RSU in real-time must be highlighted. This allows a better and safer driving experience.

## 1.1 Motivation

VANETs' applications are mostly concentrated in three branches: infotainment, traffic management, and safety (Sommer, 2014). Infotainment is related to drivers' and passengers' convenience, entertainment, and their relationship with the vehicle. In contrast, traffic management applications focus on the vehicle's behavior, as soon as it enters the street network. This also includes the impacts such a vehicle can bring to the environment, i.e. traffic jams, possible accidents, etc. According to the safety branch, VANETs are prepared to deal with different events, and then warn the driver about an incident, as adverse weather (ENGEL, 2019a), dangerous situation (BIEHLE; KRUMBIEGEL, 2019), impact reduction (BUCHHOLZ, 2019), a stationary vehicle (ENGEL, 2019b), and others.

Dealing with the development of novel VANETs applications, especially when focusing on the safety area, demands a huge amount of tests cases, e.g. for analyzing the impacts and validating. Furthermore, these tests have to consider all possible situations by the fact that any failure on these applications would cause severe impacts on transport system safety and could risk human lives (MIUCIC, 2019). Moreover, testing technology like VANETs in the real world is not only extremely expensive but it is also hardly possible to reproduce test cases. Thus, simulation tools are required and very important to make testing cheaper and more reproducible.

Intending to generate a reliable VANETs' simulation, it is required a complete simulation framework. This framework may contain at least a vehicle traffic model and a wireless communication protocol implementation (OBERMAIER et al., 2018). The more realistic a traffic scenario is, the more reliable the VANETs simulation results are. A realistic traffic scenario involves a real-world road topology including all public road categories, ranging from residential streets to highways, computes driver's behavior, and the final result should be evaluated with real traffic data. All these points are required to accurately model a mobility over a realistic traffic demand. Traffic scenarios have been developed for the transportation community for a while, but their scope is not mainly on the network community to evaluate a VANETs simulation, which demands a deeper view of the mobility patterns, analyzing vehicle's position and driver's behavior, known as a microscopic view. Furthermore, city structures as buildings, bridges, and passages have to be emphasized, as they have an impact on wireless communication (TCHOUANKEM et al., 2015).

Up to now, to the best of our knowledge, there are three freely available realistic traffic scenarios for Simulation of Urban Mobility (SUMO): TAPAS Cologne (UPPOOR; FIORE, 2011), Luxembourg SUMO Traffic (CODECA et al., 2015), and Monaco SUMO Traffic (CODECA; HÄRRI, 2018).None of the previously mentioned scenarios have modeled a city with characteristics presented in Ingolstadt. This is because the city has peculiarities as a large industry that concentrates approximately half of work positions and operates 24 hours a day in shifting model. The city also detains a high income per inhabitant and an extremely low unemployment rate (STADT INGOLSTADT, 2018). Thus, using Ingolstadt for a new traffic scenario, the research community can be benefited from a new and partly different type of map.

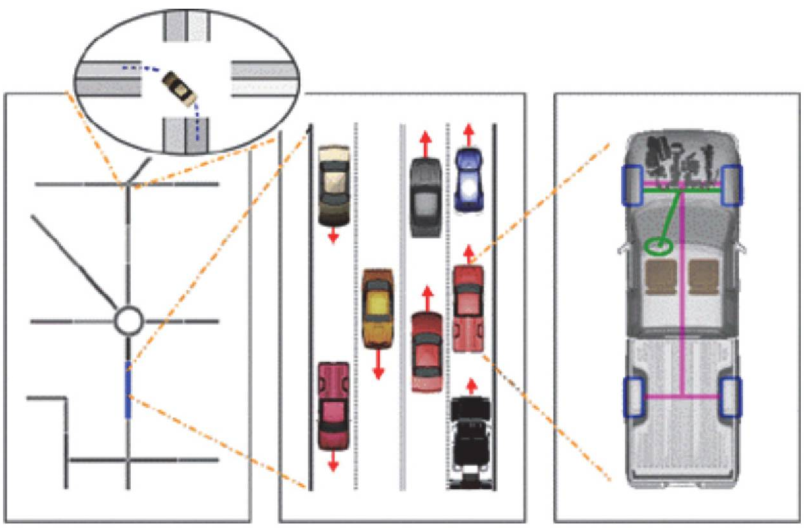
Ingolstadt is located in the state of Bavaria, southern Germany, with an area of 133.36 km$^2$ and a registered population of about 135,000 inhabitants counted in December 2018 (STADT INGOLSTADT, 2018). It is ranked the fifth biggest city in the state and has characteristics that extremely influence its traffic, e.g. it is the site of Audi AG, which employs approximately 44,000 workers, representing over than 43% of Ingolstadt's work positions (STADT INGOLSTADT, 2018). Another point is that the car usage rate compared to other cities in Germany is relatively high (AMT FÜR VERKEHRSMANAGEMENT UND GEOINFORMATION, 2018), and a high penetration rate of new cars among the inhabitants is noticed. Additionally, in the near-future the City will implement a Car2X communication system over the city (AUDI AG, 2019), which will put

Ingolstadt among the first cities in the World to be covered by a VANET.

## 1.2 Goals

This thesis develops a realistic traffic scenario enclosing Ingolstadt City and enable it especially for the use case of V2X simulations, implementing a realistic traffic flow and driver's behavior. Ingolstadt Traffic Scenario for SUMO (InTAS) will help to speed up the development of Car2X systems. On the other hand, such a simulation can be used in further approaches as advanced driving simulators like OpenROUTS3D (NEUMEIER et al., 2019). Moreover, this scenario will be developed using the Simulation of Urban Mobility (SUMO), which is a powerful open-source simulator that supports large road networks. It is suitable for both macroscopic and microscopic simulations and provides great interaction with network simulators, e.g. Omnet++ (LOPEZ et al., 2018). Figure 1.1 represents the mobility granularities, where from left to right the following mobility model types are presented: macroscopic (within the circle: mesoscopic), microscopic and sub-microscopic.

Figure 1.1: Mobility Models: (From left to right) Macroscopic (in the circle: mesoscopic), Microscopic, Sub-Microscopic



Source: Lopez et al. (2018)

Thus, the main research question is:

**Is it possible to model and evaluate a realistic traffic scenario for Ingolstadt city using SUMO's methodologies?**

This question can be divided in the following sub-parts:

1. How to develop a realistic traffic scenario for Ingolstadt considering the available data?
2. How accurate is the traffic scenario?
    - Can this scenario be validated?

## 1.3 Structure of this Thesis

This thesis is structured as follows: Chapter 2 provides an overview of SUMO as a realistic traffic generator, and a foundation for statistical fitness tests. Chapter 3 presents related works for SUMO scenarios that used real data to be developed or validated. Chapter 4 shows the methods implemented to developed the InTAS. Chapter 5 presents the conclusions and recommendations for future work.

# CHAPTER 2

# FOUNDATIONS

The process to develop a traffic scenario involves three stages: network topology definition, traffic demand modeling, and scenario simulation. In this thesis, the term network refers to the road topology, i.e. street representation.

1. **Network topology** is the basis of a traffic scenario, where the roads and all elements are developed. The roads' characteristics are represented by the number of lanes, maximum speed, exclusive lanes for buses, lanes priority, junction type, and more.

2. **Traffic demand modeling** computes the traffic flow, assigning moving vehicles to this flow. Moreover, traffic demand defines the origin and destination points of each vehicle, and the path they will drive between these points.

3. **Scenario simulation** introduces parameters and manages the simulation, e.g. simulation time and drivers' behavior.

All these previous steps are required to develop a traffic simulation in SUMO, However, simulations seeking a real-world representation should take into consideration additional features from a traffic scenario. The network topology representing the road shall correspond to a real area, considering the number of lanes, junctions, and exclusive lanes for buses. Traffic demand has to represent the real traffic, matching the number of vehicles transiting during the simulation time with real numbers, and also their origin and destination inside the city. Moreover, the traffic has to include public transport, due to their influence on traffic performance. Furthermore, a realistic traffic scenario has to be validated, measuring its accuracy according to real traffic performance.

An evaluation method is a procedure to compare the parameters of the simulation with real traffic data. This methodology might consider a great number of approaches, e.g. average speed and the number of vehicles. In SUMO all these approaches can be considered in the simulation step and generate an output file, to be further compared with real data. In this thesis, a method of applying a vehicle detector representation, known in SUMO as E1, has been used. The E1s represent induction loops detectors and compute the number of vehicles driving over it for a specific time-window. The generated output is a representation of a time-series that can be compared with the real number of vehicles driving in the same position at the same time. Intending to measure the realism brought from this study, the simulation value and real data will be compared based on the Quantile-Quantile Plot (QQPlot), Normalized Root Mean Square Error (NRMSE), and Absolute Error for the number of vehicles driving on defined spots during a given time-stamp.

## 2.1 OpenStreetMap

OpenStreetMap (OSM)(OPENSTREETMAP, 2019) is an open-source database, which provides detailed information regarded to a map including road topology, building shape and location, bus stops, and more. It is originated from an open project of an editable map, which provides Geo-spatial data of the World. It means that every point data, in OSM, also known as a *node*, is designated with its latitude and longitude information based on the World Geodetic System 84 (WGS 84) (KUMAR; ANAND, 2015). WGS 84 is a standard used by satellite navigation, geodesy, and cartography that defines the global coordinate system (FELL; TANENBAUM, 2001).

Information from OSM can be easily visualized on the website[1] or downloaded for different usages. Downloading the data from the website a file in `.osm` format is generated, which contains the elements: *nodes*, *ways*, and *relations*.

[1] https://www.openstreetmap.org

The *nodes* are points on the Earth's surface composed of its own *id* number and Geo-coordinate. On the `.osm` file, they could represent a stand-alone feature, e.g. park bench, or also be used as corners to define a shape or a path.

In Listing 2.1 a *node* expression is represented and it is possible to distinguish all important elements. Element *id* is unique for each *node* and can not be repeated. *Timestamp* is the date and time from the last update from this *node*. *Uid* and *user* are the user id and user name from who has implemented the last updated. *Visible* is the attribute responsible to allow the *node* to be visible. *Version* and *changeset* are internal attributes to track modifications. The elements *lat* and *lon* represents the latitude and longitude of that *node* (OPENSTREETMAP, 2019).

Listing 2.1: *Node* representation in OSM

```
<node id="139623" timestamp="2011-08-28T13:51:24Z" uid="162465"
user="ckol" visible="true" version="6" changeset="9147463"
lat="48.792093" lon="11.4185161"/>
```

A *way* is a polygonal chain that defines the aggregation of at least two *nodes* and can be characterized as an *open-way* and *closed-way*. An *open-way* is a line segment that represents linear features like roads and rivers. A *closed-way* is when the last *node* is attached to the first *node*, representing a loop, e.g. roundabouts, land areas, and buildings. Furthermore, to describe more complex areas, a different structure has been used, this structure is named *multipolygonon* (OPENSTREETMAP, 2019). Listing 2.2 represents a *way*, containing its unique *id*. The structures `<nd ref=' ' />` describe the *id* for the two *nodes* represented in this *way*. The structure `tag` provides a classification for this *way*. The first `tag` classifies the type of the *way* as a highway and the second `tag` assigns to the this *highway* the name Sachsstrasse.

Listing 2.2: *Way* representation in OSM

```
<way id="4396678" timestamp="2011-01-31T19:49:05Z" uid="85218"
user="schuencke" visible="true" version="9" changeset="71445">
```

```
    <nd ref="26423953" />
    <nd ref="434068567" />
    <tag k="highway" v="unclassified" />
    <tag k="name" v="Sachsstrasse" />
    </way>
```

*Relation* is a combination of elements, consisting of an ordered list of *nodes* and *ways* that have similar characteristics. It may contain a collection of roads, bridges, tunnels, and other representations.

This thesis used OSM as basis to import all map elements concerned to InTAS, i.e. road topology, buildings shape and location, bus stop locations, parking lot locations, and traffic light locations.

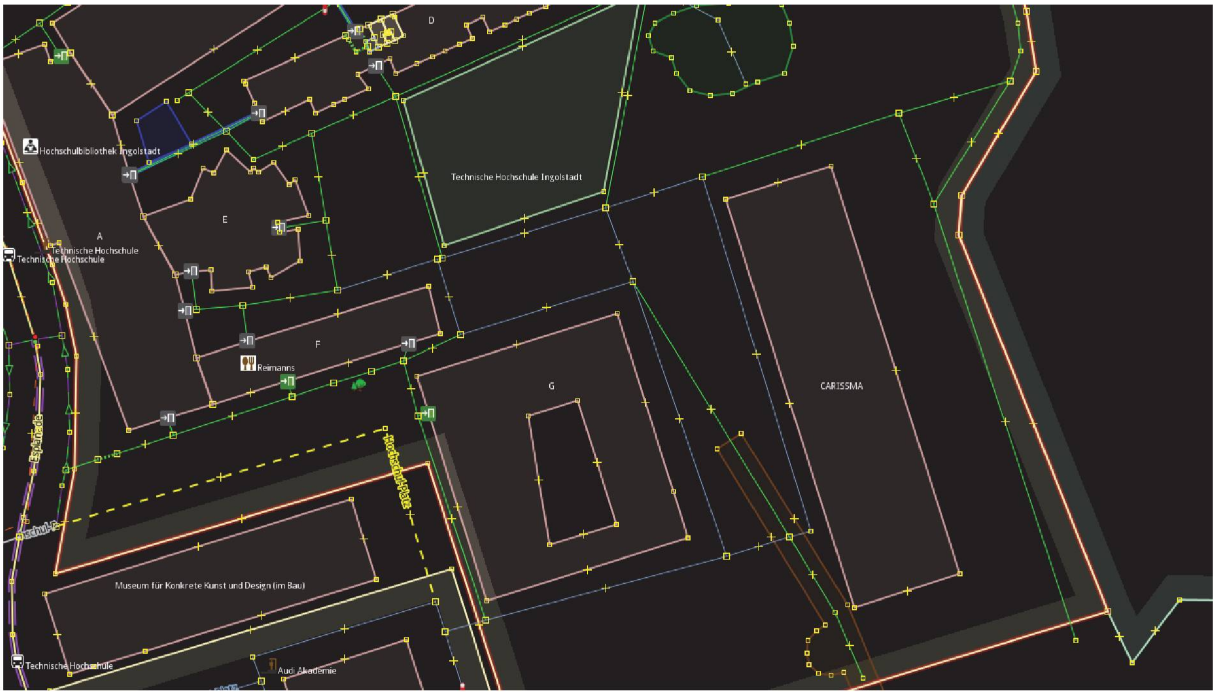
## 2.2 Java OpenStreetMap

Java Open Street Map (JOSM)[2] is an open-source application that load, edit, render, validate and upload `.osm` data displaying it on a Graphical User Interface (GUI). Figure 2.1 presents a `.osm` data reproduced in JOSM, where yellow squares and "x" represent the *nodes* and the lines connecting these nodes are the *ways*.

Many features can be deployed on JOSM, and one of the most interesting is that it allows users to edit the `.osm` file through the GUI application, where the user can directly see the parameter that is under change. During this editing process, it is possible to change characteristic parameters, e.g. highways classification, and building shapes (JAVA OPENSTREETMAP, 2019).

[2]https://josm.openstreetmap.de/

Figure 2.1: JOSM




Source: JAVA OPENSTREETMAP (2019)

## 2.3 SUMO as a Realistic Vehicle Traffic Generator

Simulation of Urban Mobility (SUMO)[3] is a free powerful microscopic traffic simulator developed by the German Aerospace Center, english for Deutsches Zentrum für Luft- und Raumfahrt (DLR). SUMO allows simulation of multi-modal and inter-modal traffic system, large-scale roads and public transport system. Moreover, it has no artificial limitation in road network size and number of simulated vehicles. For the microscopic model, SUMO also describes vehicles' behavior and their interactions with the road networking applying models for car-flowing, lane-changing, and intersection behavior. It is widely used by the scientific community as a traffic simulator, but beyond that, it is also a peculiar application to generate realistic traffics based on different types of information.

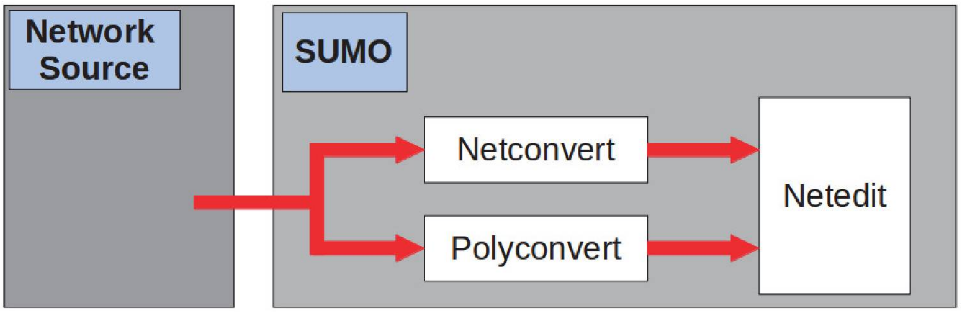
[3]https://www.eclipse.org/sumo/

### 2.3.1 Map Creation

The first step to build a traffic scenario using SUMO is to define the map characteristics. Will be the road network imported or drawn by hand? Will be considered buildings? How about public transport, will it be inserted or not? Answering all these questions will define the expanse to be followed to create the simulation's map. As this research focus on how to generate a realistic traffic scenario, only elements related to this will be explored.

Figure 2.2 represents the SUMO structure considering the tools implemented in this thesis, intending to create a realistic map. A source has to provide the road topology, building features, bus stops, and parking lots location. The `netconvert` tool extracts the road elements from the input file, and converts it into `.xml` file. Moreover, `Polyconvert` tool converts all polygons represented in the input file and also converts them into `.xml`. The `.xml` files from both tools are gathered in `netedit`. `Netedit` is a GUI environment where all input information can be visualized and edited.

Figure 2.2: SUMO tools




Source: Author

#### 2.3.1.1 Road Network

In SUMO, road networks has to be represented as a `.xml` file extension, and the contents are grouped by instances: *edges* and *junctions*. *Edges* comprise all the road segments and *junctions* that correspond to a connection between two or more *edges*, e.g. intersections, or dead-end streets. The combination of *edges* and *junctions* provides the road network characteristic. Listing 2.3 presents an edge structure with its singular *id*, the previous and next *edges*, the *priority* assigned to this *edge*, the *type* of the road, and the nodes it connects defining the *shape* of this *edge*.

Listing 2.3: *Edge* representation in OSM

```
<edge id="165520703#0" from="266813214" to="271664587" priority="5"
type="highway.unclassified" shape="214524.00,452205.10
    214533.69,452191.58 214545.66,452175.61 214561.48,452154.51"/>
```

A road topology can be manually defined, informing parameters for *edges* and *junctions*, or it can be imported from different sources, e.g. OpenStreetMap, VISUM, VISSIM, and MATsim. Using a road network downloaded from one of those database, it is possible to bring a realistic road network based on a real area, as a neighborhood, a district, or even an entire city. In this circumstance, two important tools, provided by SUMO, have an important role. One is `netconvert`, which transforms the `.osm` file based on WGS84 format into a `.xml` file format. The other is `netedit`, a GUI network editor that permits the user to do some adjustment, and small improvements on the imported network, e.g. the correct number of lanes, remove road segments, connect *junctions*, and assign road priority.

#### 2.3.1.2 Buildings, Parking Lots and Bus Stops

According to the scenario usage and the realistic accuracy expected of the traffic simulation, some points are crucial to achieving this objective, as the impact buildings generate on VANETs simulation. Fortunately, `.osm` is a resourceful database, and not only road topology is represented, but also a great number of features, e.g. amenities, buildings, land usage, public transport, leisure, etc. As the scope of this thesis is to develop a realistic traffic scenario focused on VANETs applications, buildings, parking lots, and bus stops will be gathered with the road network to create a detailed map of

Ingolstadt. SUMO provides `polyconvert`, a tool that helps to identify all polygons, which are *closed-way* on the `.osm` file, and convert them from `.osm` into `.xml`. Furthermore, it is possible to filter *relations* in the `.osm` file, and isolate the relevant entities related to buildings, parking lots, and bus stops.

### 2.3.2 Traffic Demand

The second step to elaborate a traffic scenario in SUMO is to insert moving vehicles on the previously developed map. This method, known as traffic demand, defines the number of vehicles in the simulation, their origin and destination edges, departing time, and how they drive through the map to reach their objective, i.e. which are the consecutive edges between the start and the final edge. SUMO distinguishes traffic demand in two views: *trips* and *routes*. *Trips* represent a general view from traffic demand, and it is a model containing only the edge of origin, the edge of destination, and departure time. A *route* is an expanded view, where not only the origin and destination *edges* are represented, but also all the *edges* that the vehicle transits through. Both representations are correlated to each other and are implemented consecutively. First, the *trips* are computed, and then a path is associated with them, expressing the *routes*.

#### 2.3.2.1 Trip

SUMO provides some approaches to assist a *trip* development. Among these tools, some generate a realistic traffic demand, and some provide an unrealistic traffic demand. As this work focus on creating a realistic traffic scenario, only the four methods that can attend this scope have been described: `od2trips`, `jtrrouter`, `dfrouter` and `activitygen`.

Origin and Destination Matrix (ODM) is often developed by traffic authorities and it is an approach to create *trips*. ODM contains the number of vehicles that flows from one area to a distinct area, representing an origin and destination *edges*, in a certain time period. A Traffic Assignment Zone (TAZ), that might be represented by a district or a region of the city, is defined based on the areas covered by ODM. Intending to compute the *trips* using ODM, `od2trips` is the tool that assists in this assignment.

Flow Definitions and Turning Ratios is a method to define *trips* using the traffic volume and junctions turning ratio applying the tool `jtrrouter`. This tool focus on junctions scenarios and do not take into consideration others variables for a *trip* development, as a public transport system.

Detector Data is an approach commonly used by traffic authorities, applying devices to measure the vehicle traffic, as induction loops, and this data may be used to generate the traffic demand in SUMO. `Dfrouter` is the application tool deployed to use inductions loop values and compute then into *trips*. This application is more relevant to be used when developing a traffic demand for one road, like highways, where induction loops measure the number of entering and leaving vehicles.

Population Statistics uses demographic information as the basis for a traffic demand generation. `Activitygen` is the key application responsible to generate a traffic demand based on population statistics, computing daily activities as work and school. It also allows to implement the number of vehicles driving through the city's entrance and provide all the information regarded to the public transport system. This application tool assigns areas that generate traffic and areas that demand traffics, and this information will create the *trip* for each vehicle.

#### 2.3.2.2 Route

After defining the *trip* for each vehicle, it is important to assign them to a route, representing the path they will drive through. `Duarouter` is an application that will create the routes for the vehicles, considering the optimum path between origin and destination of each one. The optimum path, considered on this tool, is based on travel time and road length, and is calculated considering that each vehicle drives alone in the network. Due to this consideration, vehicles do not compute other vehicles' influences and all of them could take the same path. It may cause traffic jams and bottlenecks during the simulation time. Intend to solve this issue, a rerouting algorithm will be implemented in the simulation, creating an iteration during the running time, and allowing vehicles to change their paths when a traffic jam is detected.

### 2.3.3 Simulation of Public Transport

The insertion of a public transport system obeys a different rule when compared to the *trip* and *route* generation. As the busses already have a defined route, no computation is needed. Among the methods to insert bus routes is the manually insertion directly in `netedit`, assigning routes to the `edges`. For some cities, the bus `.osm` database also provide bus routes, which have to be converted to `.xml` file to be inserted into the simulation. Independently from the chosen method, bus' schedules for each stop have to be defined, as well as the time a bus starts and ends a journey.

### 2.3.4 Traffic Light System

A Traffic Light System (TLS) plays a important roll for a city's traffic management, and due to the impact a TLS has to be deployed for a realistic traffic scenario. The `.osm` file contains the junctions controlled by a TLS, and in the conversion process from `.osm` to `.xml`, SUMO assigns a standard program to each traffic light considering a default cycle time of 90 seconds and *static* model, that is when the traffic light never changes its phases. In the `netedit` is also possible to set `actuated` model for traffic lights, which controls TLS features based on time gaps, prolonging a phase whenever a continuous traffic flow is detected or identifies a time loss. Moreover, SUMO allows to edit the default cycle of a traffic light, or even to program a TLS with new features and behaviors.

## 2.4 Statistical Tests

Two given discrete time-series may be compared to each other to understand if both have the same behavior, and measure the mismatch distance in each time stamp. These analysis are also known as adherence tests, and among all of them are Normalized Root Mean Square Error (NRMSE) and Quantile-Quantile Plot (QQPlot), which are well accepted by the scientific community (BOX et al., 2008).

### 2.4.1 Quantile-Quantile Plot

Quantile-Quantile Plot (QQPlot) is a visual graphic tool used to compare characteristic from two different populations. In this technique, two cumulative distributions $F(x)$ and $G(x)$ are associated to quantile functions $G^{-1}(x)$ and $F^{-1}(x)$ - the inverse function of CDF provides the quantile function. Assuming that the quantiles of one function is given by $q_1, q_2, ..., q_n$, the QQPlot is generated plotting the coordinates $(F(q_i), G(q_i), 1 \leq i \leq n)$ (PEDROSO, 2018).

On this method, the data-set is sorted in order of magnitude, with the values that divide the set into four equal parts called quartiles, into ten parts deciles and $N$ parts, which may correspond to the number of data in the set, the quantiles. On a QQPlot, each point represents the quantis of each sample, placed on $x$ and $y$ axes. If both samples are based on the same population, points should be around the diagonal line at 45°. Comparing the points on the graphic to this diagonal line, in case the points are located on a parallel line to this diagonal, both populations has similar distribution with one process in a higher level related to the other, but if the points are over the diagonal line, means that both populations have the same distribution (RICCI, 2005).

### 2.4.2 Normalized Root Mean Square Error

The Root Mean Square Error (RMSE) is a statistical method to measure the mismatch between two discrete time-series, comparing each point. When this methodology is applied, errors are represented in the same dimensions as the analyzed variables. The RMSE is given by:

$$RMSE = \sqrt{\frac{\sum_{n=1}^{N}(x_{r,n} - x_{s,n})^2}{N}} \tag{2.1}$$

where $x_{r,n}$ represents point-values for the first sample, $n$ and $x_{s,n}$ represents the values for the second sample at the same time-window. Moreover, $N$ represents the total number of samples. Related to this formula, it is possible to infer that the lower the value, the more correlated the series are (RICCI, 2005).

To compare data-sets with different scales the Normalized Root Mean Square Error (NRMSE) can be implemented. This approach provides a tool to compare different scenarios with different numbers of samples. The NRMSE is given by:

$$NRMSE = \frac{RMSE}{\bar{x}} \tag{2.2}$$

where $\bar{x}$ represents the mean value of the measured data.

## 2.5 Chapter Considerations

This Chapter has introduced the methods to develop a realistic traffic scenario for SUMO and important backgrounds for a better understanding of the related works presented in Chapter 3. Section 2.1 has presented the OSM structures and their importance to define a map and its elements. Afterward, the JOSM tool and its application were reviewed. The second part of this Chapter introduced SUMO as a realistic traffic generator. The first approach showed how to create a map structure, considering a road topology network, and inserting the elements buildings, parking lots and bus stops. Furthermore, methods to create a traffic demand using real data have been presented.Moreover, a method for public transport and traffic light consideration has been introduced. The last part of this Chapter introduced two statistical tests: QQPlot and NRMSE. Those tests have been applied as a validation method in Chapter 4, intending to evaluate the simulation result with real data, and to measure the mismatch between both.

# CHAPTER 3

# RELATED WORK

Simulation of realistic traffic scenario is fundamental when these simulations analyze the microscopic traffic model, exhibiting the behavior and position of each vehicle. In VANETs, vehicles are supposed as communication nodes and are continuously exchanging information with the neighborhood, so interpreting the driver's behavior is important to map a system data transmission. Furthermore, a realistic traffic scenario is essential to understand traffic models providing mobility traces with different densities and to provide a safe simulation environment for new application tests. A scenario with those characteristics has to consider real traffic information in the creation step or the validation. Therefore, this chapter focus on related works that introduced methodologies applying real traffic data.

## 3.1 Vila Real Case Study

Intending to develop a scenario for an electric vehicle simulator, a SUMO realistic traffic scenario for the Vila Real City (SOARES et al., 2014), in Portugal, was developed. The methodology implemented to create traffic demand consisted of evaluating census, and survey data, provided by governmental institutions, to estimate individuals activities.

At first, the road information was extracted from the OSM database and all the road segment corrections were elaborated using JOSM, and the final result was imported into SUMO. The information provided by the Portuguese census concatenate several city areas with its geo-spatial coordinates and statistics information. To match and convert the coordinate system from census into SUMO, an algorithm defined as a Synthesizer was developed. This algorithm identifies each region on census data and links them with the corresponding edge on SUMO, and the final values were written on file.

Traffic demand was computed by `activitygen`, which was inputted with data provided by the Synthesizer, afterward, the routes were generated by `duarouter` and integrated with the map.

This paper created a traffic scenario based on demographic data collected on the governmental census, but therefore, it is really small scenario restricted to the Portuguese census format, which cannot be applied to the data format provided by the City of Ingolstadt.

## 3.2 Downtown Ottawa

Downtown Ottawa scenario (MCKENNEY; WHITE, 2013) is a realistic traffic scenario based on a $9 \times 7$ block section of Ottawa to estimate the effectiveness of an intelligent traffic control system in a realistic scenario. The traffic model was built in SUMO as a microscopic traffic simulator and OSM to represent the road topology. Every intersection inside the selected area is measured by the City of Ottawa with vehicle count, providing the number of vehicles going straight or making turns on a time interval during the day. Unfortunately, the data did not provide continuous information on the time domain, thus, the authors filled the information gaps with linear interpolation for a smoother transition between the measurements. To generate the routes, a vehicle flow based on the processed data is needed, and to do so, it was determined a vehicle's ratios proceeding in each possible direction.

This study presented an approach and improvement results when an adaptive signal control system is implemented, increasing the simulation's average speed. Nevertheless, this scenario does not cover an entire city behavior, nor public transport network.

## 3.3 TAPAS Cologne

The Travel Activity Pattern Simulation (TAPAS) Cologne data-set (HERTKORN, 2004) is a test case applied to the City of Cologne to evaluate a novel microscopic vehicular traffic model, named Origin-Destination Matrix (ODM), which describes mobility wishes for a specific population area generated based on travel habits and the infrastructure of the area they live. This scenario was introduced and made available by the TAPAS-Cologne project, which was developed by the German Aerospace Center (DLR). The model presented on this project was developed based on travel demand and can predict the population trips for each person. This model considers a typical working day, using attributes as departure time and location, destination, and means of transport. It relies on an activity-based approach, whereby the activities are derived from the survey data of the Federal Statistical Office between the years 1991 and 1992. Time-use patterns were attributed to individuals of the population based on sociodemographic characteristics. Travel depends on the type of person, trip purpose, and distance. It was developed considering that every vehicle was used only for one trip at a certain point in time. Travel demand developed on this simulation were compared to empirical findings.

The City of Cologne was divided into 85 different zones, representing all the districts of the city. For each of them, sociodemographic data, e.g. the number of inhabitants, the number of cars, schools, and working positions were computed. That information fulfilled the ODM modeling the traffic flow, identifying traffic provider areas and traffic receiver areas, creating a traffic demand. The City map and information were generated using a closed source tool, but the traffic result was published as an open-source data-set.

Afterward, intending to develop a completely open-source database, was developed an urban vehicular mobility for networking research based on TAPAS Cologne data-set (UPPOOR; FIORE, 2011), creating a realistic traffic scenario for V2X application. Firstly, the street layout was obtained from OSM, encompassing an area of approximately 400 km$^2$ and almost 4,500 km of roads. The imported roads were not suitable for microscopic mobility, and Java Open Street Map (JOSM) was used to correct and adjust it to the needs.

SUMO was selected to simulate vehicle traffic and driver's behavior, using Krauss' car-following model (KRAUß, 1998) and Krajzewicz's (KRAJZEWICZ, 2009) lane changing model for acceleration and overtaking decisions. Traffic demand was implemented taking into consideration the TAPAS Cologne ODM data-set, previously introduced, and this data-set was gathered with the road topology. The scenario simulation represents the combination of TAPAS Cologne ODM as input for both OSM map and Gawron's algorithm, OSM map is input for SUMO simulation.

Due to the scenario's size and complexity, it demands high computation time and still needs additional improvements in some feature, e.g. junction corrections, correct lane numbers per street, adjust traffic lights position and insert public transport. At this scenario, traffic demand presents a realistic behavior, and at the same time road topology and public transport does not reflect the real-world equivalents, which makes it partly not realistic enough.

## 3.4 Luxembourg SUMO Traffic Scenario

Luxembourg Sumo Traffic Scenario (LUST) (CODECA et al., 2015) was developed to provide a realistic traffic scenario based on Luxembourg and to create a novel realistic scenario for VANETs simulations. The proposed work scenario covers an area of 156 km$^2$ and 930 km of road length and assembles more than 35 bus lines with 561 stops around the city.

The road topology of the city was collected from the OSM and were refined using JOSM since they did not represent the road topology faithfully. On this step, road segments, Points of Interest (POI), traffic lights location, and bus stop positions were manually corrected. To convert the final road topology into SUMO, `netconvert` were applied, and exported to netedit. In netedit, the number of lanes of each street and the junctions characteristics were corrected, comparing them with the images provided by Google Earth. Traffic light system was implemented using the actuated model with the standard value provided by SUMO.

The traffic demand was developed using real demographic data modeled according to activitygen. Bus routes were manually inserted using the city's public transport database together with bus stop positions set on JOSM.

Vehicle routes described traffic flow and were elaborated based on two different methods to evaluate the driver's behavior. These methods select the path to reach the destination, providing two different simulation data-set. The first method used was Dynamic User Assignment (DUA), which computes the optimal path between origin and destination, considering travel time and length over an empty network, that is, an idealistic scenario not taking into consideration traffic jams. This approach was implemented in SUMO using the tool duarouter. The other method implemented to the traffic flow was Dynamic User Equilibrium (DUE) (GAWRON, 1998), performing the Gawron's optimization, applying the duaiterate tool. SUMO allows to equip vehicles with different types of devices with different penetration rates, and the routing behavior was assigned to the vehicles using this possibility.

The scenario's final result was compared with real data collected from over 14,000 trips between 6 am and 10 pm. The data were harvested in a real traffic situation, and got information concerned to the speed in the geographical position of the vehicle. Based on this data, they compared simulation data to real data using QQPlot comparison and also applied the Kolmogorov-Smirnov (KS) test (BOX et al., 2008), which resulted in a distance $D$ of 0.1569 with a $p-value$ less than 0.05 during off-peak hours, what they assumed to be an approximate value during off-peak hours (CODECA et al., 2015).

Scenario evaluation using the KS test presented a small error value representing a precise scenario. However, the validation method took into consideration only a small portion of the city characterized by a high number of traffic jams and vehicles driving slowly. Other parts of the city were not evaluated, which does not bring an overall evaluation of the scenario. Thereby, more areas should be evaluated to present a wider view of the city.

## 3.5 Monaco SUMO Traffic Scenario

Monaco SUMO Traffic Scenario (MOST) (CODECA; HÄRRI, 2018) covers a 73 $Km^2$ area and was developed intending to create a multi-modal scenario considering the land elevation. The road network was obtained using the OSM dataset, and in parallel, with the information provided by Monaco's governmental institutions, land's altitudes were extracted.

The data was processed and reviewed in the `.osm` file, applying JOSM together with `netconvert`, intending to correct intersections and to eliminate unrealistic bottleneck characteristics. This scenario considers the pedestrian flows, and for this reason, sidewalks were inserted.

In the Principality of Monaco, most of the intersections are priority-based, and a very small number of traffic lights are placed. To reproduces this interaction, between pedestrians and vehicles, a great number of traffic lights were considered, because, with that, this behavior is better modeled.

To generate traffic patterns, the scenario was divided into various Traffic Assignment Zone (TAZ), according to the administrative boundaries. For each TAZ was considered the total number of Points of Interest (POIs), and the area referred to it. Unfortunately, this scenario covers only the morning peak hour and its traffic realism is not measured, because it was not compared to real traffic data.

## 3.6 Research Gap

Traffic scenarios have been developed using real traffic information and are freely available. Although, to the best of our knowledge, none of them simulate the traffic light system deployed by traffic authorities in the real traffic system, either have a robust validation method applying NRSME in different points of the city. Those programs will be implemented in this research intending to improve traffic realism and mimic the Ingolstadt's traffic. Moreover, none of them cover the characteristics presented in Ingolstadt city, as:

- An industrial city;
- One industry concentrating around 43% of all work positions in one spot;
- Some companies working 24 hours a day in a 3 shift operation model;
- A high rate of vehicles per inhabitants (STADT INGOLSTADT, 2018);
- A low unemployment rate (STADT INGOLSTADT, 2019a);
- A high income per inhabitant (STADT INGOLSTADT, 2019a);
- A higher rate regarding vehicle usage when compared to other cities (STADT INGOLSTADT, 2019a);
- Incoming traffic represents approximately 44% of the total traffic (STADT INGOLSTADT, 2019b);

Among the traffic modeling methodologies presented in SUMO is `activygen`, which generates traffic demand based on the description of the population presented in the road network. This method takes into consideration the probability of a person to own a car, the unemployment rate, the probability an adult prefers to use the car instead of other means of transportation, and incoming and outgoing traffic.

# CHAPTER 4

# INTAS

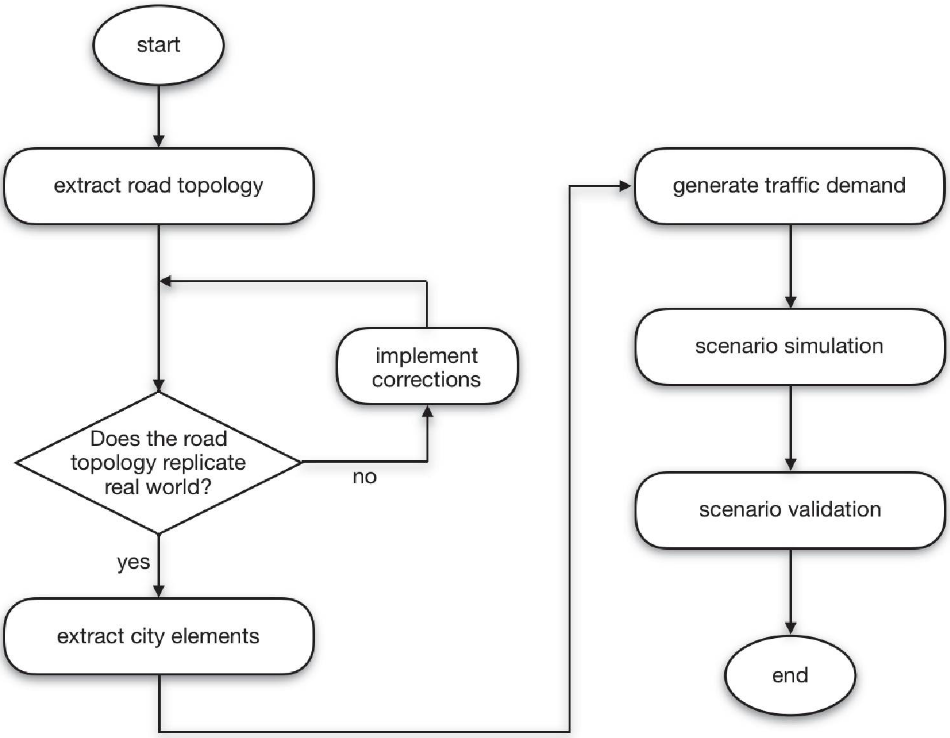
The **Ingolstadt Traffic Scenario for SUMO (InTAS)** development consisted of four procedures: definition of network topology, traffic demand modeling, scenario simulation, and scenario validation. The objective of the first step was to elaborate a city map containing all important information, e.g. road topology, buildings, parking lots, bus stops and traffic light positions. All the information related to the first step has been imported form `.osm` file. The second step consists of modeling all available real-world data using `activygen` method to create a realistic traffic demand. The third step involves setting the best simulation parameters and the data collection for the scenario's evaluation. Scenario validation step compares the simulation result with real traffic data applying the Normalized Root Means Square Error (NRMSE) as statistic method. Figure 4.1 presents a flowchart of the developed process of InTAS.

## 4.1 Map Creation

This stage is the foundation and consisted in the scenario area delimitation. The chosen area represents 87% of city's work positions, approximately 79% of the total inhabitants, and roughly 81% of the registered cars in Ingolstadt. However, this selection excluded the inside traffic pattern of surrounding villages. This might not be an issue as some villages are over 12 km away from the city center, and their inner traffic does not influence the main area of Ingolstadt. Thus, instead of modeling their internal traffic, this study took into consideration the traffic demand between the villages and

The major part of this chapter has been accepted as following paper "InTAS - The Ingolstadt Traffic Scenario for SUMO", for SUMO User Conference, October, 2020.

Figure 4.1: InTAS Flowchart




Source: Author

Ingolstadt.

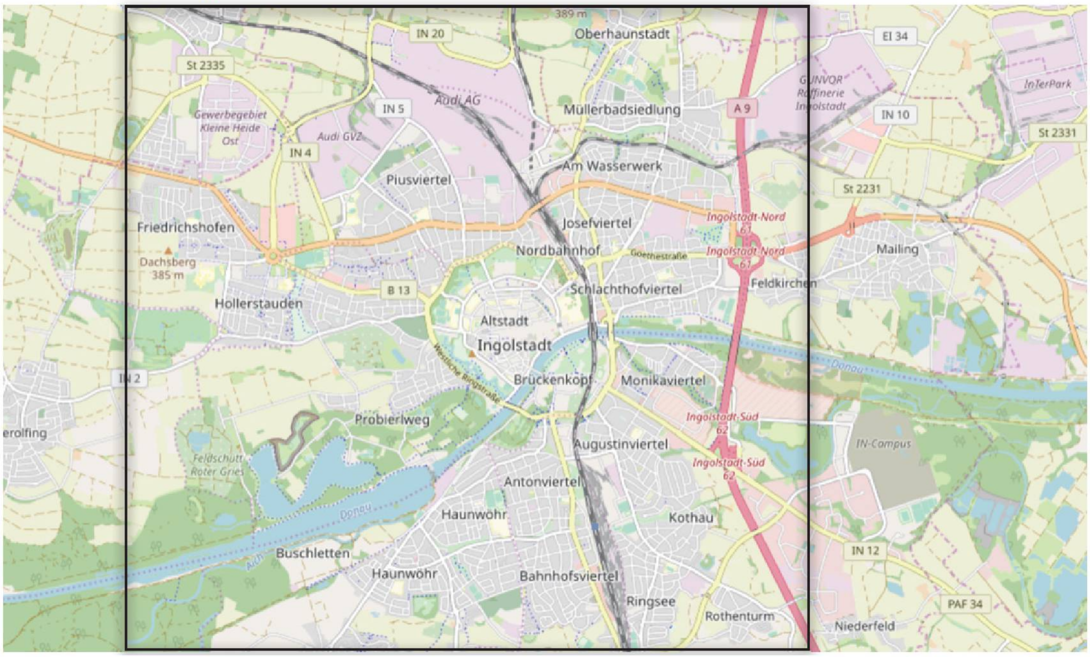
After defining the interesting area, this region was extracted from OpenStreetMap (OSM) (OPENSTREETMAP, 2019) containing all information enclosed to this area. Figure 4.2 shows the area selected for InTAS in OSM.

### 4.1.1 Road Network

In SUMO, road networks are represented as `.xml` file grouped by the instances: edges and junctions. Edges represent road segments and junctions either correspond to intersections or dead-end streets (LOPEZ et al., 2018). As the OSM source file is presented with the `.osm` extension, it was converted into `.xml` applying `netconvert`.

By examining the converted data, it was observed that a great number of streets were not representing the real-world topology, i.e. incorrect number of lanes, missing

Figure 4.2: InTAS border of the selected area

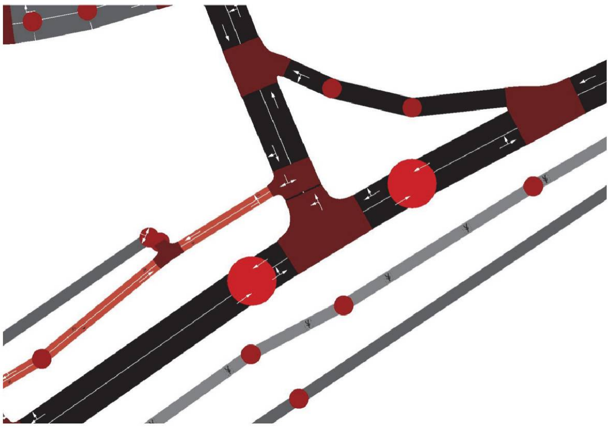



Source: OPENSTREETMAP (2019)

exclusive turn left lanes and exclusive bus lanes. This divergence might be caused by outdated information retrieved from OSM. Although information is frequently updated on OSM, as it is an open-source project working on the wiki-style process, some areas are not detailed enough, and, contain only the street segment but not the number of lanes or exclusive lanes. Furthermore, there were additional information in the map, which were not relevant for the current version of the scenario, e.g. sidewalks, bike lanes and also private roads, i.e. streets inner an industry, which are not accessible for all vehicles.

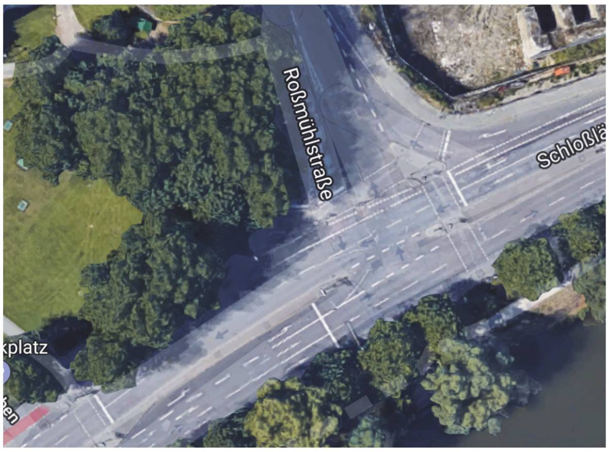
Intending to develop a reliable map, which accurately represents Ingolstadt, it was necessary to implement a method to correct all the issues. Thereby, a thorough process to compare each of the 7,966 edges and all of the 3,341 nodes with the satellite image, on-line accessible, on Google Street Maps (GOOGLE, 2019) was undertaken. This correction were applied with the `netedit` tool, where the entire map was manually inspected and validated. During the correction, all junctions were checked to reinforce lane connections. Moreover, all bicycle lanes, sidewalks, and private streets,

Figure 4.3: Conversion Result



Source: Author

Figure 4.4: In Google Maps

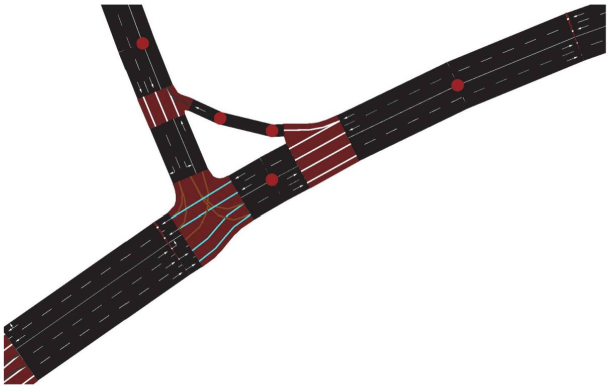



Source: Google (2019)

as commercial shopping facilities, residential and industrial condominiums, were removed. After the cleaning process, the total road length of the InTAS is 717.13 km. Figure 4.3 is the first conversion result, Figure 4.4 is the same road in Google Maps and Figure 4.5 is the final result for this area after editing it by hand.

Traffic light systems also play an important roll in the city traffic behavior (WU et al., 2019). Due to their importance, they are also considered for this scenario. As `.osm` file describes the traffic light position, `netconvert` assembled it together with the road network `.xml` file, assigning a hypothetical phase length to all traffic lights (TL). At this step, the objective was to check if all real TL were represented on the correct position

Figure 4.5: Manually Corrected



Source: Author

in the map. To confirm the TL's location, an up-to-date document provided by the City of Ingolstadt, with the position of all TL managed by them, was analyzed. Furthermore, all only pedestrians TL were removed from the map, keeping just the ones that control junctions with a minimum of two streets, resulting in a total of 98 traffic lights across the map.

The final InTAS' road network, after implementing all the corrections and adjustments, is presented in Figure 4.6. Table 4.1 shows the information concerning to road network developed at this stage of the development.

Table 4.1: Network numbers

| Parameter | Value |
|---|---|
| Total Area | 51.54 km$^2$ |
| Road Length | 717.23 km |
| Nodes | 3,342 |
| Edges | 7,968 |
| Traffic Lights | 98 |

Source: Author

### 4.1.2 Parking, Traffic Lights, Buildings and Bus Stops

It is extremely complex to model city traffic, due to the great number of variables that directly impact traffic behavior. This section presents the considerations for the scenario:

Figure 4.6: InTAS Road Topology

Source: Author

parking areas, traffic lights, buildings, and bus stops.

#### 4.1.2.1 Parking Areas

In the `.osm` data, 59 parking areas are represented, but not all of them were used in this work. InTAS has focused on public parking areas and companies' parking. The City of Ingolstadt manages a total of 13 parking areas with 5,568 parking lots. Those 13 parking areas were tracked during an average usage at business days between Tuesday to Thursday, from September to December of 2019. This measurements resulted in a daily average usage of 4,247 public parking lots in the city. This average value has been further considered to model the traffic demand presented in Section 4.2. The main objective is to define how many workers use these parking areas during the day considering them as a region that demands traffic, and finally allow vehicles to drive to them and park there. Table 4.2 lists those parking areas and compares their total capacity with the average number implemented in InTAS.

The parking areas currently closed for construction work "P2" and P8" were considered to be always full, which partly reflects their usage before getting closed.

Table 4.2: Comparison of real capacity parking areas with used in InTAS

| Parking Area Name | Total Capacity | Implemented Capacity |
|---|---|---|
| P1 Tiefgarage Theater West | 398 | 345 |
| P1 Tiefgarage Theater Ost | 388 | 282 |
| P2 Tiefgarage Schloß | 448 | 448 |
| P3 Tiefgarage Münster | 207 | 151 |
| P4 Tiefgarage Reduit Tilly | 281 | 130 |
| P5 Parkplatz Hallenbad | 612 | 507 |
| P6 Parkplatz Festplatz | 1,287 | 723 |
| P7 Parkplatz Südl. Ringstraße | 244 | 178 |
| P8 Parkhaus Hauptbahnhof West | 811 | 811 |
| P9 Parkhaus Hauptbahnhof Ost | 185 | 184 |
| P10 Parkhaus Nordbahnhof | 232 | 166 |
| P11 Congressgarage | 375 | 222 |
| P12 Tiefgarage Zeughaus | 100 | 100 |

Source: STADT INGOLSTADT (2019c)

Moreover, additional eight parking areas were took into consideration. Five serving AUDI AG, one attending Klinikum Ingolstadt and one serving Technische Hochschule Ingolstadt (THI). For THI, only the number of employees and not the students was considered. The total workers from these three employers represent 54.52% of the total employees in the scenario, where AUDI AG is the largest company in Ingolstadt, employing 44,526 workers, Klinikum Ingolstadt employs 3,630 and THI employs 650 workers. The parking areas were identified in the map and each edge, which represents the road segment related to these areas, has assigned with the number of employees for each region.

#### 4.1.2.2 Traffic Light System

Importing traffic lights using `netconvert`, a generic program automatically generated is assigned to each traffic light, defining the traffic light cycle time, each phase duration, states, and the traffic light logic type.

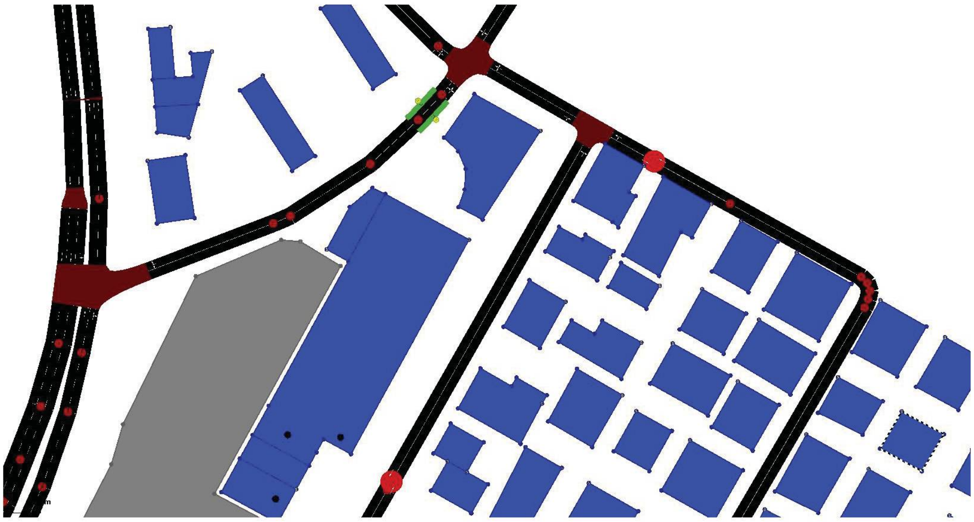
TL state is the definition linked for each lane under a TLS operation, i.e. assigning if the light is green, red or amber for this lane. TLS cycle is the total time necessary for a program to run all phases. Phase duration is the time a state will be activated. The parameter TL logic type may assume *static* or *actuated* values. A *static* parameter represents that a TL will always have the same behavior, never changing phase duration. Furthermore, TL logic type was set as *actuated* traffic control, extending a phase once continuous traffic is detected (LOPEZ et al., 2018).

As TLS is one of the highest influencing factors to traffic behavior (TIELERT et al., 2010), this work seeks to provide a realistic TLS, intending to approximate simulation traffic to real traffic. In Goethestraße, which is an arterial road and one of the most important gates to Ingolstadt, TL programs in all nine traffic lights were implemented according to the real program deployed in them. This section of Goethestraße has 2 km length, and over 21,000 vehicles drive through it per day. In Manchingerstraße, both traffic light systems for intersections between this street and the Autobahn A9 have been simulated based on the real TLS deployed. Manchingerstraße is one of the most important gates in Ingolstadt city, where approximately 34,500 vehicles drive through. The junction between the Römerstraße and the Autobahn A9 has been also implemented, according to the real TLS. This junction is the Ingolstadt gate for roughly 32,000 vehicles during a day. A total of 13 intersections had their TLS simulated in this thesis based on the real TLS deployed. The other TLs around the city were implemented with the automatically generated program with the *actuated* traffic logic.

#### 4.1.2.3 Bus Stops

InTAS also reproduces the public transport system, considering all bus lines inside the simulation area. Yet, to provide more realistic representation, all bus stops were imported from the `.osm` file. Those stops were compared with the on-line available infor-

Figure 4.7: Extract of InTAS with its city elements



Source: Author

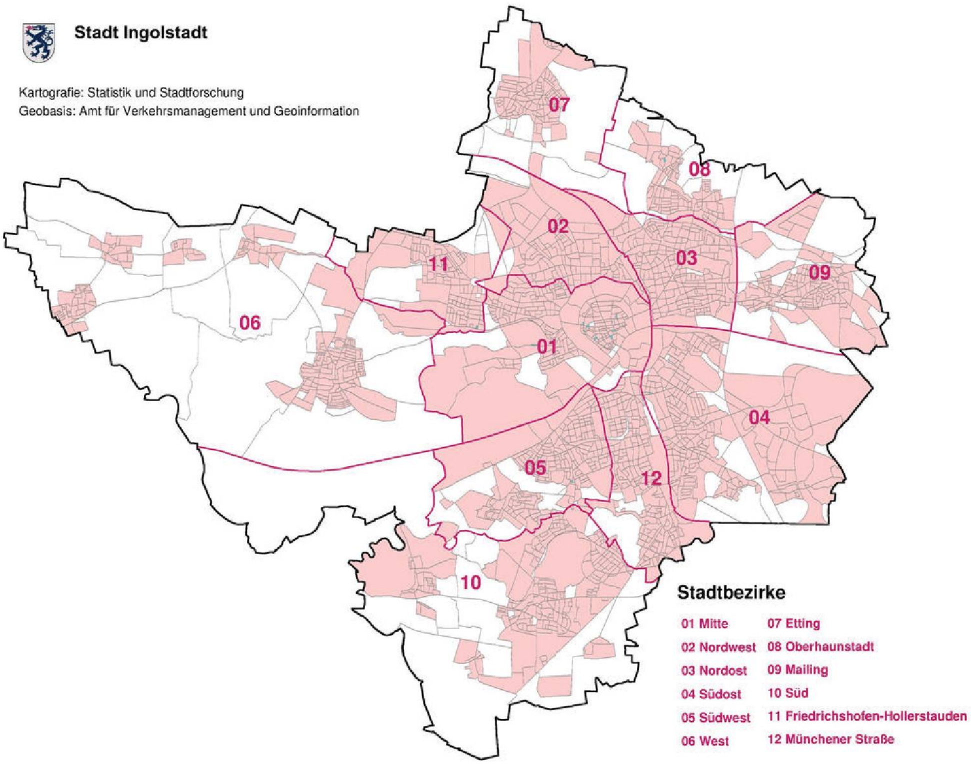
mation from local bus service company - Ingolstädter Verkehrsgesellschaft (INVG) (INVG, 2019). A total of 404 bus stops were inserted to the scenario.

#### 4.1.2.4 Buildings

InTAS was developed to be a complete simulation environment, where traffic behavior as well as VANET studies can be deployed. To encompass a higher simulation potential, this scenario implemented all buildings represented by the `.osm` file. Due to their influence in the performance for an inter-vehicular wireless communication environment, when operating in the standardized frequency, they are very important (TCHOUANKEM et al., 2015). As aforementioned, the `.osm` data is a resourceful database, and all information related to buildings was converted from the `.osm` to `.xml` applying the `polyconvert`. A total number of 21,756 buildings were incorporated into InTAS.

To put all theses points togheter, Figure 4.7 presents an extracted part of the Ingolstadt map, where the road network is shown together with all city elements. In blue color, buildings are represented with their dimensions and shapes. The gray color

Figure 4.8: Districts of Ingolstadt City




Source: STADT INGOLSTADT (2020)

indicates parking lots. Bus stops are marked in green aside the roads.

## 4.2 Traffic Demand

Ingolstadt is divided into 12 different larger areas, where each of these areas are sub-divided, creating a total of 62 sub-areas, as shown in Figure 4.8, where districts are numbered from 1 up to 12 and sub-districts are delimited by the gray line inside the district. For each of these sub-districts, based on data from the City of Ingolstadt, it is known: number of inhabitants, households, living workers, unemployed, and number of registered vehicles. These numbers are at a high level of detail, providing a reliable database to model Ingolstadt's traffic, and the more detailed the numbers to model the traffic is, the more realistic the scenario is.

Pondering traffic demand methods, previously described in Section 2.3.2, traffic

Table 4.3: Vehicle traffic numbers for InTAS

| Attribute | Value |
|---|---|
| Car rate | 0.9363 |
| Incoming traffic | 95,927 |
| Outgoing traffic | 21.423 |
| Car preference rate | 0.5890 |

Source: AMT FÜR VERKEHRSMANAGEMENT UND GEOINFORMATION (2018)

scenario expectations and, with a heavier weight, available data, it has been decided that the traffic demand in InTAS would be modeled by `activitygen` method. Table 4.3 summarizes the numbers and also presents the car rate, which describes the rate of adults that own a vehicle inside the scenario area (LOPEZ et al., 2018).

Amongst the scenario, 38 out of 62 sub-areas were considered laying within the InTAS borders. The inhabitants for these selected areas were divided into 13 age groups, ranging from 0 to above 85 years. Thereafter, the number of workers that live in each region, were determined according to the number of social numbers reported in each sub-area. Table 4.4 presents the difference between the total Ingolstadt demographic numbers and the demographic numbers modeled by InTAS using `activitygen`. This difference is based on the selected area showed in Figure 4.2, where some villages were not selected to have their inner traffic modeled. The values presented on the column labeled *Ingolstadt City* are according to the demographic numbers for the entire Ingolstadt (STADT INGOLSTADT, 2019a). The column labeled *InTAS* refers to the same demographic numbers regarded to the selected area, as introduced in Section 4.1 and Figure 4.2.In Table 4.4 attribute workers refers to the number of workers living inside the area. Based on those numbers, a difference of approximately 21% related to a number of inhabitants and workers between entire Ingolstadt and InTAS is observable. The difference observed for the number of vehicles and householders is smaller and is nearly at 18% for both. However, a tinier difference is noticed for work positions and unemployed, showing 13% and 7% respectively.

The difference observed between City's numbers and InTAS may influence the traffic behavior and the number of vehicles driving through the map. To solve this issue, the traffic demand generated outside the InTAS' border, concerning to Ingolstadt city, was considered as incoming traffic to the scenario. A difference between working positions and workers inside the scenario was also observed. Therefore, the same solution, considering these as incoming traffic, was applied.

Table 4.4: Comparison of demographic numbers between Ingolstadt and InTAS

| Attribute | Ingolstadt City | InTAS |
|---|---|---|
| Inhabitants | 138,180 | 109,090 |
| Workers | 61,670 | 49,020 |
| Work positions | 102,925 | 89,515 |
| Unemployed | 1,219 | 1,138 |
| Vehicles | 97,950 | 80,337 |
| House holders | 69,379 | 57,118 |

Source: STADT INGOLSTADT (2019a)

Not only incoming traffic is relevant for an inner city traffic, but also outgoing traffic. Intending to model this phenomenon, it was important to define all gates, through which the traffic comes and leaves the scenario area. Based on available traffic information, a total number of 15 points, where the traffic can income and outgo from the scenario, were defined (HUMBERG, 2017). Moreover, it was primordial to assign the number of vehicles incoming and outgoing through each one of these gates. With given data from the city of Ingolstadt, each gate was defined with its traffic flow, representing a total of 32,634 incoming people and 14,879 outgoing people to their work. Furthermore, based on the same database, the car preference rate of 0.589 was defined, representing the probability an inhabitant uses the vehicle instead of using other transportation means (LOPEZ et al., 2018).

Companies' opening and closing hours are also relevant for modelling the traffic. This information was assigned considering the proportion of workers that have to start and finish their jobs at that time. For companies that work 24 hours uninterruptedly the start and end of shift time were considered as opening and closing times. The data provided by the city is based on measurements of a normal business day. Thus, it were selected Tuesdays, Wednesdays or Thursdays. According to the traffic management office, these days are the busiest traffic days and were taken into consideration for traffic improvements.

Children also play an important role in traffic demand. Although most of them do not go to work, a multitude of them is driven to kindergarten or might be to school by their parents. To represent this behavior in the Ingolstadt scenario, each school was defined, containing their exact position on the map, the age range it covers, capacity and class hours. Thus, this step includes children from the kindergarten age to high school age. Moreover, to include students from both universities, Technische Hochschule Ingolstadt (THI) and Katholische Universität Eichstätt-Ingolstadt (KU), a similar implementation was designed, but at this point, parents do not drop them off. Instead, they drive their own vehicles. Likewise the workers, but with the universities as the final destination. InTAS considered 17 kindergartens, 36 schools, and 2 universities.

All structured demographic data were used to define the trips, i.e. identify where people leave, work and study. The number of trips reached by this study was 264,714, considering entire traffic for 24 hours to all vehicles and public transport. Although, this information was not sufficient to describe the path each driver will take to achieve his destination. For this reason, the `duarouter` tool was used, which is the application to assign an entire path between origin and destination points, computing the routes for each vehicle.

### 4.2.1 Simulation of Public Transport System

Bus lines are an important factor in real-world city traffic because they influence the traffic behavior of all participants. Especially on one lane driving roads, when they have to stop at a bus stop. Therefore, this work considers the bus lines as they drive through the city of Ingolstadt. It is challenging to represent busses' behavior in SUMO, where is possible to set drivers behavior, busses' routes, and simulate busses stopping

Table 4.5: Public transport numbers

| **Attribute** | **Values** |
|---|---|
| Number of lines | 56 |
| Total bus stops | 404 |
| Number of busses trips | 1,620 |
| Bus routes length | 880.6 km |

Source: INVG (2019)

at their stops. However, to simulate vehicles passing of stopped busses is not possible. Thus, the average time a bus spend in each stop has been set in 10 $seconds$ (CODECA et al., 2015), which approaches to the main objective.

In Section 4.1.2 it was presented how bus stops were extracted from a `.osm` file and how they were converted into a `.xml` file. To represent a realistic public transport system, all bus routes for this scenario were considered. Intending to typify bus routes, it was resorted again to INVG public data. Thus, it was possible to feed the real bus route information to the simulation. In total, there are 56 bus lines, where 28 are regular lines, 15 are night lines, 7 are shift lines (running only at specific period), and 6 are lines that attend surrounding cities but have their departure or arrival in Ingolstadt. The bus lines cover a total of 880.6 km of road and compute 1,620 bus trips during the 24 hours simulation. Parameters of public transportation are shown in Table 4.5.

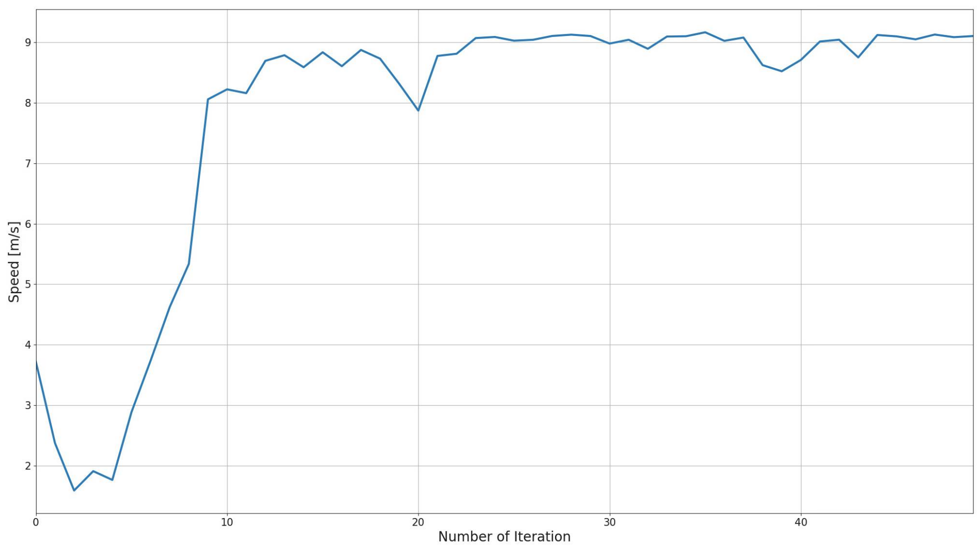
### 4.2.2 Traffic Flow Optimization

In Section 4.2, `duarouter` was applied to assign routes to trips. This step defines the in-between edges a vehicle will drive from an origin point and to a destination place. Implementing `duarouter`, the assignment performance computes the shortest path through the network using the Djikstra (DIJKSTRA, 1959) routing algorithm. At this point, `duarouter` defines the shortest route for each vehicle considering they are alone independent of other cars at the road network. After loading all the vehicles in the simulation, they will behave as if they are alone, and it will lead to traffic jams.

Seeking to mitigate the issue caused by `duarouter` and bring more realism to the traffic flow, an equilibrium state might be reached. For this reason, the Gawron's (GAWRON, 1998) method to optimize the traffic flow has been implemented. This method calculates the user equilibrium for each vehicle and implement a route optimization, running the simulation and re-computing the DUA over the network (CODECA, 2016).

SUMO provides the tool `duaIterate`, which iteratively tries to find the user equilibrium, i.e. to find a route for each vehicle without reducing the travel cost (SUMO, 2020a). As the number of iterations to reach the equilibrium may vary, it was analyzed average speed, time lost, and travel time for InTAS.

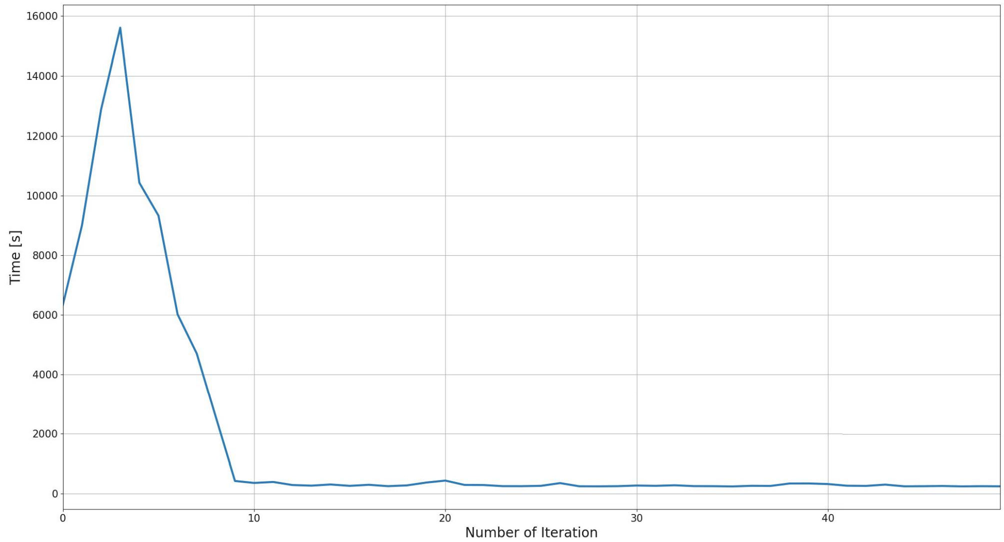
Figure 4.9: Iterations: InTAS Average Speed




Source: Author

Among the outputs provided by `duaIterate`, time loss, average speed, and average travel time have been evaluated. These parameters tend to converge to a stability when the equilibrium is reached. Figures 4.9, 4.10, and 4.11 show the result obtained after 50 iterations applying `duaIterate`, and it is observed that the equilibrium has been achieved for all parameters after 25 iterations. Figure 4.9 describes the average speed presented in the scenario for each iteration, demonstrating an unexpected behavior in the initial iterations. This behavior reduced InTAS' average speed to a minimum value of 1.60 $m/s$ on the 2nd iteration, and only from that iteration, the average speed grows up to the optimum value. Between 8th and 22nd iterations an oscillatory behavior is observed. Only on the iteration 25, the equilibrium has been reached with an average speed of 9.03 $m/s$

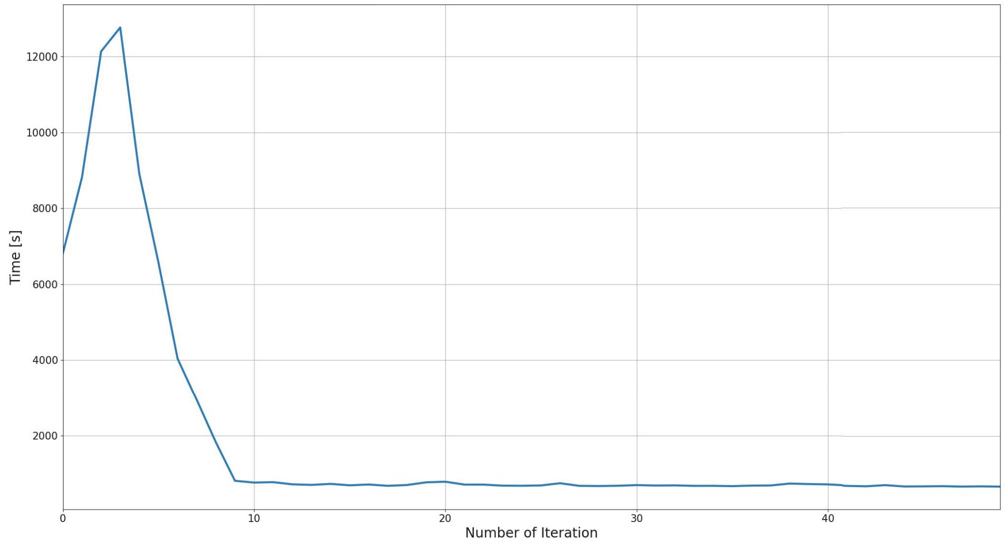
Figure 4.10: Iterations: InTAS Time Lost




Source: Author

Figure 4.10 exhibits the parameter time loss for each iteration. This parameter is provided by the SUMO simulation and calculates, for each vehicle, the difference between the actual trip duration and theoretical trip duration (SUMO, 2020b). For SUMO the theoretical trip duration is the minimum time required by the vehicle to drive from the origin to destination over an empty road network. On the 9th iteration is noticed that the time loss reached 380.8 *seconds*, and decreases slower, when compared to the initial iterations, up to 234.62 *seconds* in the iteration 25. Figure 4.11 shows the average travel time for each travel in each iteration, converging to 830 *seconds* on the 9th iteration. These three figures also present that in the first iterations the traffic scenario

Figure 4.11: Iterations: InTAS Average Travel Time




Source: Author

was denser and with a low mobility pattern. However, after iterations of `duaIterate`, the equilibrium state has been reached. Therefore, since all parameters converged in the 25$^{th}$ iteration, it has been assumed to proceed with the further development of InTAS.

## 4.3 InTAS Simulation

Simulation is the phase where scenario map file, moving vehicles represented in route file, and additional files as bus stations, buildings and detectors files are gathered. These files are regulated by parameters defined at this step.

Simulation parameter *begin* and *end* time were set to cover 24 hours of a day, with a *step-length* of 1 second. Usually, it may happen that a vehicle blocks an intersection, which leads to a huge traffic jam, causing an unrealistic pattern in the simulation. Avoiding such behavior, the parameter *ignore-junction-blocker* allows vehicles to ignore a junction after a specific time and continue their travel from there. A value of 15

Table 4.6: Simulation Parameters

| Parameter | Value |
|---|---|
| begin | 0 seconds |
| end | 86,400 seconds |
| step-length | 1 second |
| ignore-junction-blocker | 15 seconds |
| time-to-teleport | 300 seconds |
| default.carfollowmodel | Krauss |
| routing-algorithm | Dijkstra |
| device.rerouting.probability | 0.88 |
| device.rerouting.period | 300 seconds |

Source: Author

$seconds$ was implemented by this feature, intending to minimize the impacts. Another setting feature is *time-to-teleport*, which defines the maximum vehicle's waiting time in seconds on a traffic jam before it is teleported to a further position of its own route, intending to reduce impact created for huge traffic jams. For this, the parameter was set to the default value presented by SUMO, which is 300 $seconds$. In the simulation settings, it is also possible to define the routing algorithm and vehicle following model. As vehicle following model, the *Krauss* model (KRAUß, 1998) was defined, which models the reaction times and human behavior during the drive, introducing a stochastic component, e.g. the driver's behavior when changing lanes. Table 4.6 summarizes the simulation parameter used by InTAS.

Another feature presented in the simulation phase in SUMO is the parameter *device.rerouting.probability*, which allows vehicles to change their routes during the simulation. In real-world traffic, some drivers may change their path, due to the knowledge they have about the city traffic. To address this behavior, this parameter was applied to this scenario. To calculate the best rate for *device.rerouting.probability*, an algorithm, which is detailed in Section 4.3.1, has been developed to search this value.

### 4.3.1 Defining InTAS Best Rerouting Probability

SUMO has a variety of simulation parameters, which influences the traffic behavior. As each city has its characteristics, each parameter may change from city to city. Among these parameters, there is the *device.rerouting.probability* representing the probability of a vehicle to have a rerouting device. Vehicles equipped with this device may compute a new route as soon they come across an unexpected situation, like a traffic jam that can increase the time cost to reach the destination.

Changes to the extreme on this parameter will lead the traffic behavior to two distinguishes performances. Setting it to null represents that all vehicles will drive through the same roads - edges. The performance noticed is that the traffic jams will increase until SUMO crashes due to hardware limitations to deal with the high number of vehicles running in the simulation. Another issue faced at this point is that with more vehicles in the simulation they could not reach their destination, and that is the reason why the number of vehicles keeps growing. On the other hand, setting *device.rerouting.probability* to 1.00 will provide a large traffic capillarity. That will force vehicles to use roads that are not often used and will reduce the number of vehicles using the main roads, inducing an unrealistic pattern.

Intending to find the best value for *device.rerouting.probability* that fits for InTAs, an algorithm to iterate the rerouting probability from 0.00 to 1.00 has been developed. This algorithm was structured with three modules. The first writes the configuration file to be simulated in SUMO. The second module applies the Python TraCI library (LOPEZ et al., 2018) to call, to connect, and to run SUMO. The last module implements a statistic evaluation applying NRMSE to compare simulation output values with a real data-set. Algorithm 1 describes the steps implemented to reach the best *device.rerouting.probability* value.

**Algorithm 1:** Calculate the Best Rerouting Probability for InTAS

```
Result: Provide Simulation with the lowest NRMSE
initialize routingRate = 0.00;
initialize finalEvaluationList;
while routingRate <= 1.0 do
    open file InTAS.sumocfg;
    update device.rerouting.probability with routingRate;
    save file InTAS.sumocfg;
    start traci;
    initialize simulationTime = 0;
    while simulationTime < 86400 do
        print traci.simulationStep();
        simulationTime = simulationTime + 1
    end
    close traci;
    open file path/to/crossing/and/detectors as detectorFile;
    for intersection in detectorFile create dataSet;
    open file path/to/simulation/detectors/output;
    parse xml file;
    initialize sumVehSimulationList;
    initialize sumTotalSimulationList;
    initialize itrTimeSimulation = 900;
    if timeStamp.attrib['end'] = itrTimeSimulation then
        append vehNumbert to sumVehSimulationList;
    else
        append sum of sumVehSimulationList to sumTotalSimulationList;
        clear sumVehSimulationList;
        itrTimeSimulation = itrTimeSimulation + 900
    end
    calculate NRMSE between dataSet and sumTotalSimulationList;
    append NRMSE to finalEvaluationList;
    routingRate = routingRate + 0.01
end
bestRate = min(finalEvaluationList);
initialize countKey = 0;
for element in finalEvaluationList;
if element != bestRate then
    countKey = countKey + 1;
else
    print countKey;
end
```

#### 4.3.1.1 Data-Set

In interaction with the Ingolstadt Verkehrsmanagement und Geoinformation Office, which is a branch of the City of Ingolstadt, an SFTP server with information from 24 crossings have been structured. For each intersection, information between September 3$^{rd}$ 2019 and December 15$^{th}$ of the same year, has been taken into consideration. The available data describes the number of vehicles which daily drove through these crossings over 24 hours of the day, grouping the total number of vehicles in a 15 minutes time window.

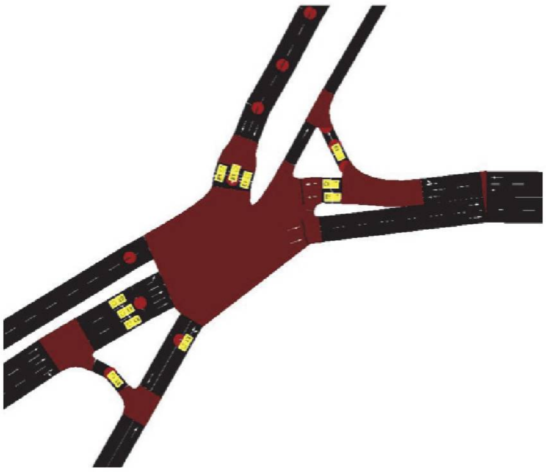
For each crossing, values for Tuesdays, Wednesdays, and Thursdays were selected. These values compute the heaviest traffic days according to the Ingolstadt Verkehrsmanagement und Geoinformation Office. Among the selected days, holidays and days before the holiday were excluded, because the traffic may change its characteristic in these days. Data from days that faced any issue have been also removed. In the end, each crossing remained with data from 27 days[1]. Thereafter, the average value for each detector in each junction has been calculated. Computing the average value avoids choosing a day with unusual behavior, e.g. working sites and snowy days.

A data-set was structure to provide information for modeling and validating, which has been split into two sub-sets. One sub-set consisted of October 2019 for modeling and evaluate *device.rerouting.probability*. The other sub-set has been used to the validation step, presented in Section 4.4, which took into consideration traffic values from November 2019.

SUMO allows to represent these detectors in the simulation with the same counting characteristics as presented in the data-set. Count detectors in SUMO are named E1 detectors, and they are elements that simulate the induction loop traffic detectors, computing the number of vehicles driving over it during a set time window. As an example, Figure 4.12 shows the intersection between Goethestraße and Friedrich-Ebertstraße

[1]average number of days for intersection

Figure 4.12: SUMO representation for detectors



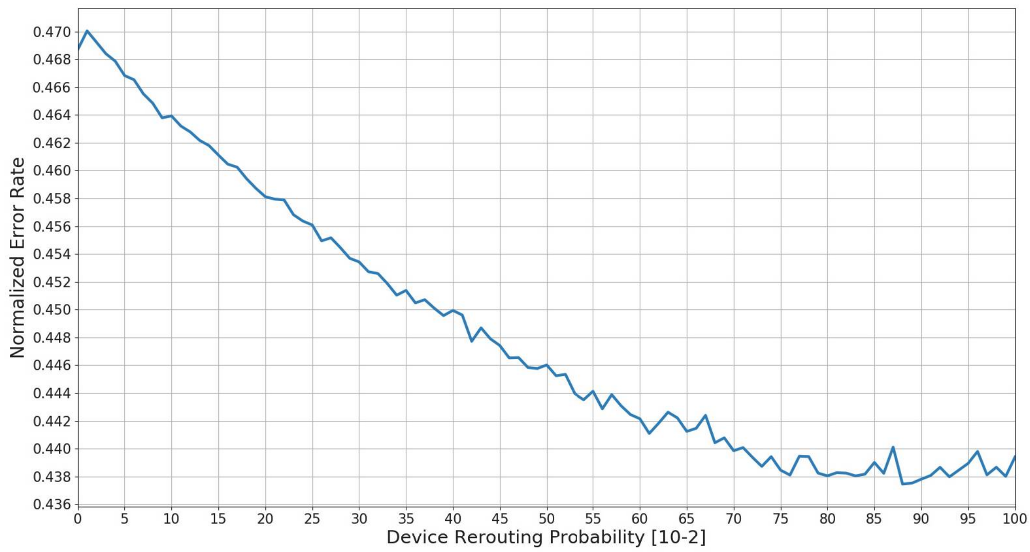
Source: Author

represented in SUMO and also with the yellow detectors placed on this junction. The detectors were placed in the simulation as close as their position in real are.

The total number of cars driving through each crossing in the data-set was compared with its respective value in the simulation applying the NRSME (RICCI, 2005). This statistic approach measures the error rate between two samples, represented in this study by the real-world data and simulation data. Intending to measure this difference has been applied the Root Mean Square Error, equation 2.1, and the Normalized Root Mean Square Error, equation 2.2.

Figure 4.13 shows the behavior of error rates obtained by the simulations, where the error is higher for low values, decreases over the iteration until reaches the lower error value, and raises again until the end of the iterations. The best value reached for *device.rerouting.probability*, which presents the lower error rate, is the simulation with 0.88 of probability with an NRMSE, i.e. an error rate of 0.438234. This value was considered for evaluation analysis presented in Section 4.4 and as the final value for this parameter in InTAS.

Figure 4.13: Device Rerouting Probability Evaluation



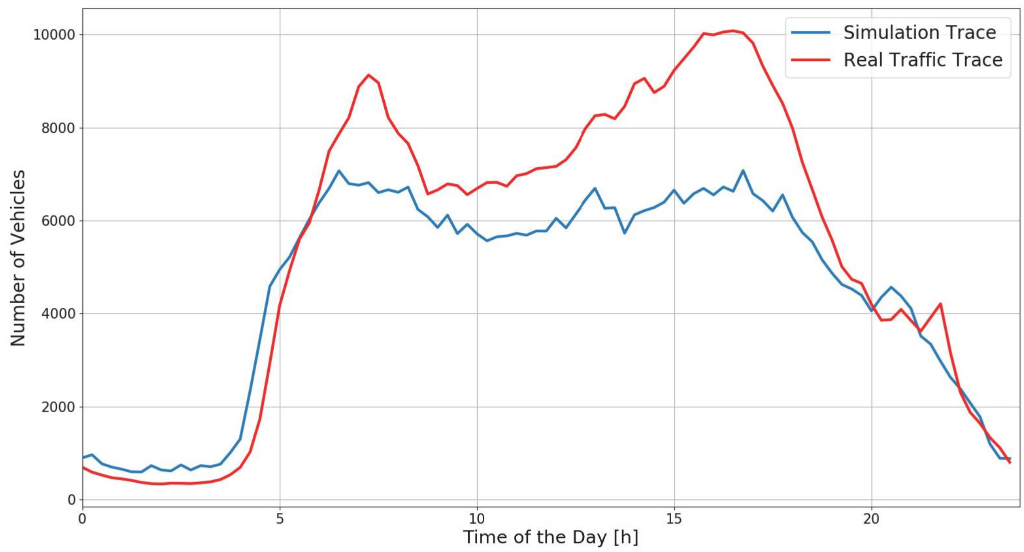

Source: Author

## 4.4 InTAS Validation

The validation of the InTAS scenario was done based on the data-set presented in Section 4.3.1.1, considering the detector values from November 2019. The comparison between all detectors is depicted in Figure 4.14, where the blue line represents the trace resulting from the simulation, and the red line shows the values from the data-set. The mismatch calculated applying NRMSE brings a rate of 0.438234 for the scenario. In the Simulation Trace (ST), a large number of vehicles are observed from 0:00 until 6:00 when compared with the Real Trace (RT). The RT trace exceeds ST from 6:00 to 20:06. After this time, a lower mismatch between the traces is observed. Around 17:00 the highest peak is detected, and thereafter the number of vehicles in both traces reduces. Starting at 22:12 both traces have a similar behavior until the end of the day.

Intending to enrich the analysis and to understand the traffic behavior, an absolute error has been calculated. This error considers the absolute difference for each of the time samples, comparing simulation and real values. Figure 4.15 shows the abso-

Figure 4.14: Traces Comparison



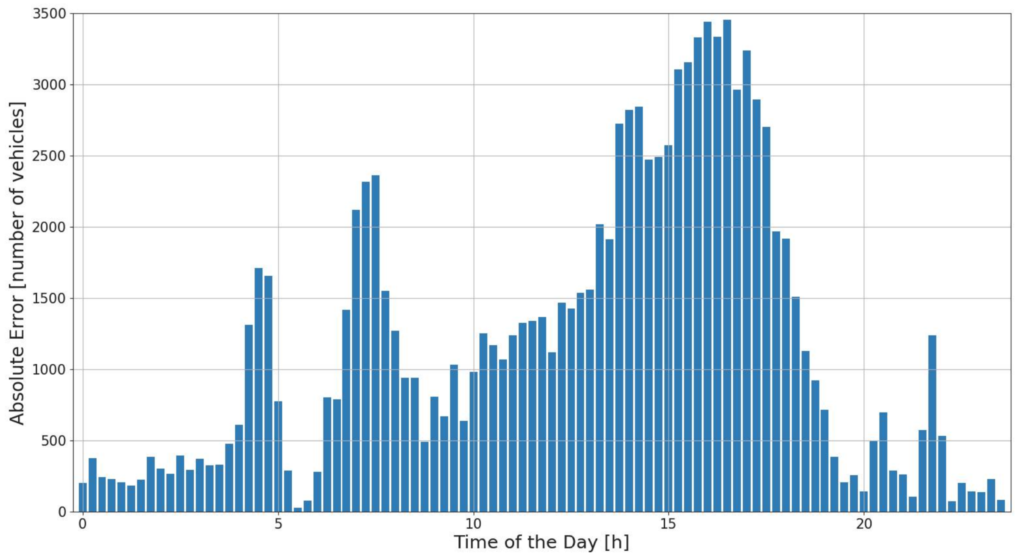

Source: Author

lute error behavior during the day, and it is possible to imply that the highest error is computed between 15:18 and 17:30 in the afternoon. At this time, a higher number of vehicles is computed, and the simulation did not follow the same behavior presented on the data-set. The lowest errors occur at the time when traces cross to each other, and during the period from 0:00 to 3:45, between 8:45 and 9:45, and after 19:30.

Although Figure 4.15 shows that the highest absolute error occurs between 15:18 and 17:30 in the afternoon, it is necessary to measure the influence caused by the absolute error values. Therefore, an analysis comparing the NRMSE for each time window is depicted in Figure 4.16. As observed, even that the higher absolute error is between 15:18 and 17:30, with an NRMSE around 0.54346, which is 25% higher than the scenario's NRMSE. On the other hand, the error presented between 1:30 and 4:30 in the morning has a larger impact, even that it has a smaller absolute error, when compared with other periods of the day.

Intending to analyze if traces ST and RT have a common distribution a QQPlot test was implemented. When plotting the QQPlot, a linear parameter is expected when the

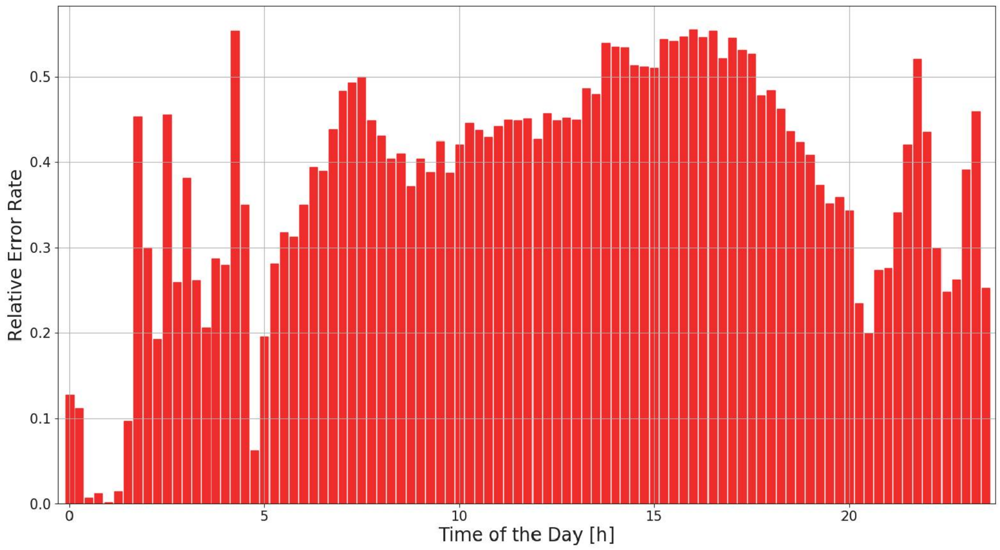
Figure 4.15: Absolute Error per Time Window




Source: Author

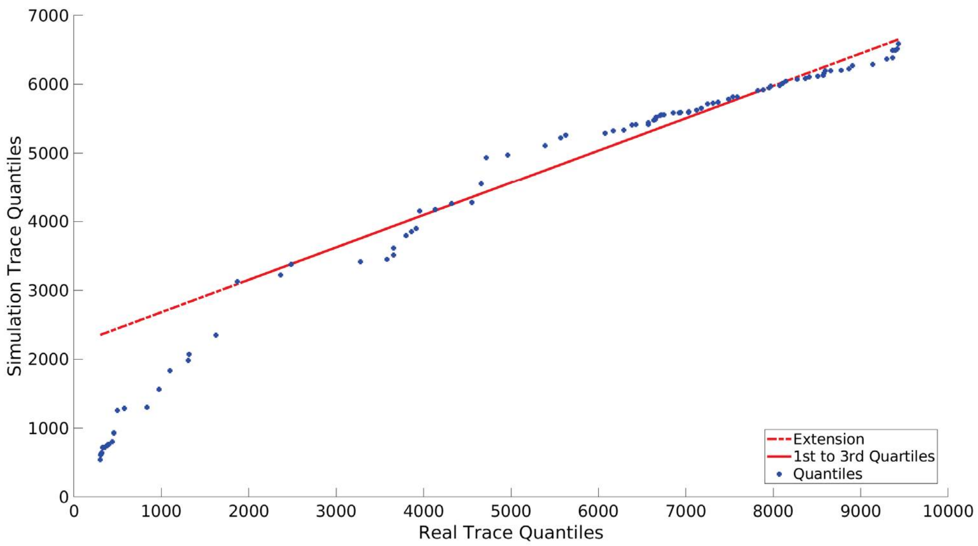
traces have the same distribution. The more linear the QQPlot is, the better the fit of both series (RICCI, 2005). Figure 4.17 shows the result of a QQPlot for RT and ST. The red line presented in this figure, represents the common distribution between both traces, i.e. if the traces have the same distribution, the quantiles would be plotted over this line. The blue marks are the result when plotting the simulation values against the expected value, which is the real data. The solid red line connects the first and the third quartiles, and the dashed extended the solid date until the end of the data. In this figure, the behavior for the initial quantiles does not match between the traces. At the same time, for the highest quantiles, a behavior closer to the red line is observed as the blue dots approach the red line. Even thought the traces present different values, the approximate linear behavior noticed, suggest that ST and RT may have the same distribution.

Figure 4.16: NRMSE per Time Window




Source: Author

Figure 4.17: QQPlot among RT and ST




Source: Author

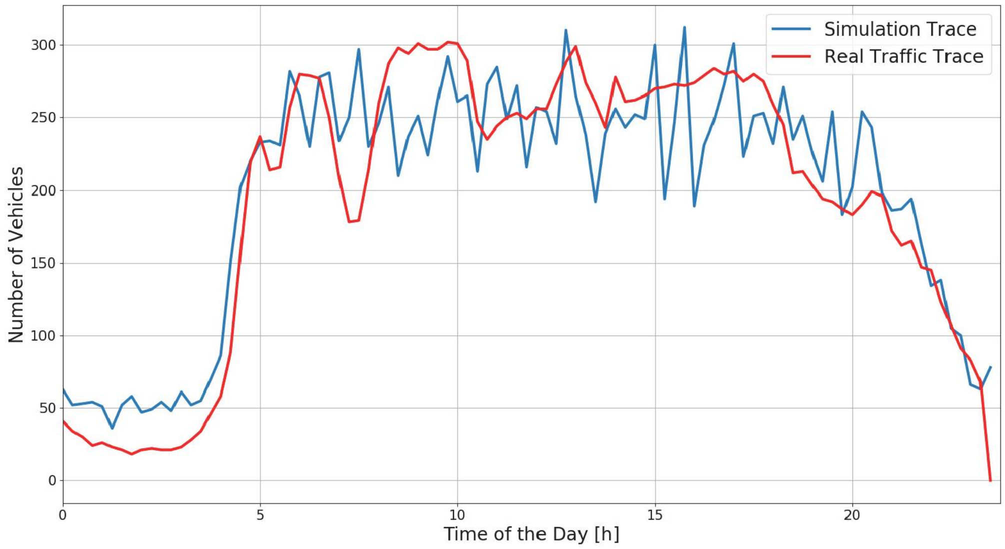
### 4.4.1 Crossing Evaluation

A total number of 24 junctions were compared, and no pattern to describe the error has been observed. Due to this, the intersections with the highest error and the one with the lowest error were deeper analyzed. Table 4.7 details the total number of vehicles per crossing considering real traffic and simulation, and error rate for each intersection. Based on these values, crossing ID-04 has the lowest NRMSE and crossing ID-24 has the highest NRMSE.

The crossing represented by the ID-04 is the junction between the arterial road Westliche Ringstrasse and the way Probierlweg. This junction is a three-way intersection where approximately 18,414 vehicles daily drive. In this junction were implemented four vehicle detectors, one for each road lane. All four detectors are placed on Westliche Ringstrasse, three in the north direction, and one detector in the south direction. Figure 4.18 shows the different behavior from the ST and RT, where it is observed that both traces are close with few mismatches periods. Figure 4.19 shows the crossing ID-04 represented in SUMO with the detectors placed on the simulation. Figure 4.20 depicts the image from the same crossing in Google Maps.

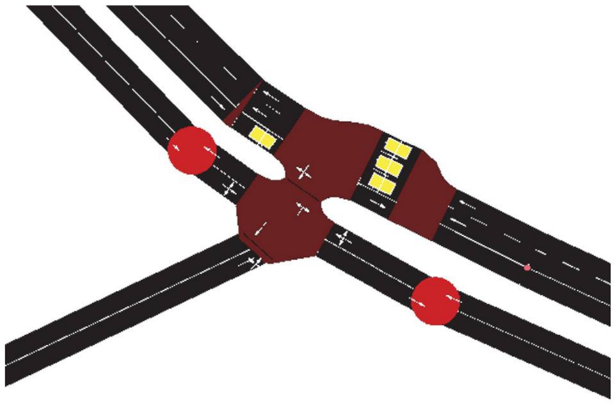
On the other hand, crossing ID-24 presented the highest NRMSE. This intersection is between the arterial roads Südliche Ringstrasse and Münchener Strasse. This junction is among the most important crossings in Ingolstadt, registering over 55,000 vehicles a day. There are placed sixteen detectors, four on the Münchener Strasse in the south direction, three on the same street but in the north direction, four on the Südliche Ringstrasse in the east direction, and five on Südliche Ringstrasse in the west direction. Figure 4.21 shows the behavior of this intersection, where it is possible to evaluate that the mismatch of both traces is relevant. The RT shows that in this crossing a large number of vehicles drive through during the day. Analyzing ST, it is observed that in the simulation this junction is not well used by the drivers as expected. Figure 4.22 presents all the 16 detectors placed on the simulation. Figure 4.23 the same intersection, as it is represented in Google Maps.

Figure 4.18: Crossing ID-04



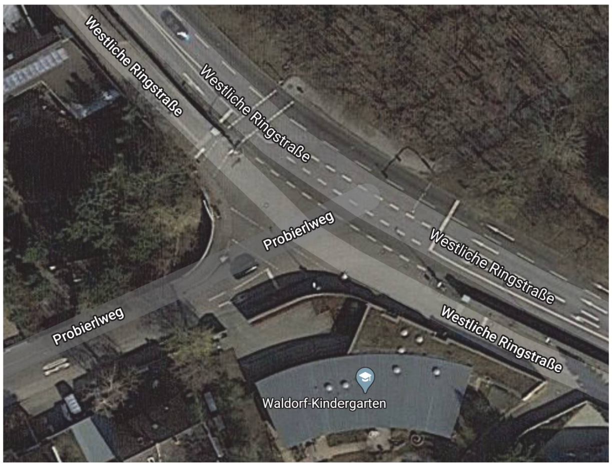

Source: Author

Figure 4.19: ID-04 in InTAS



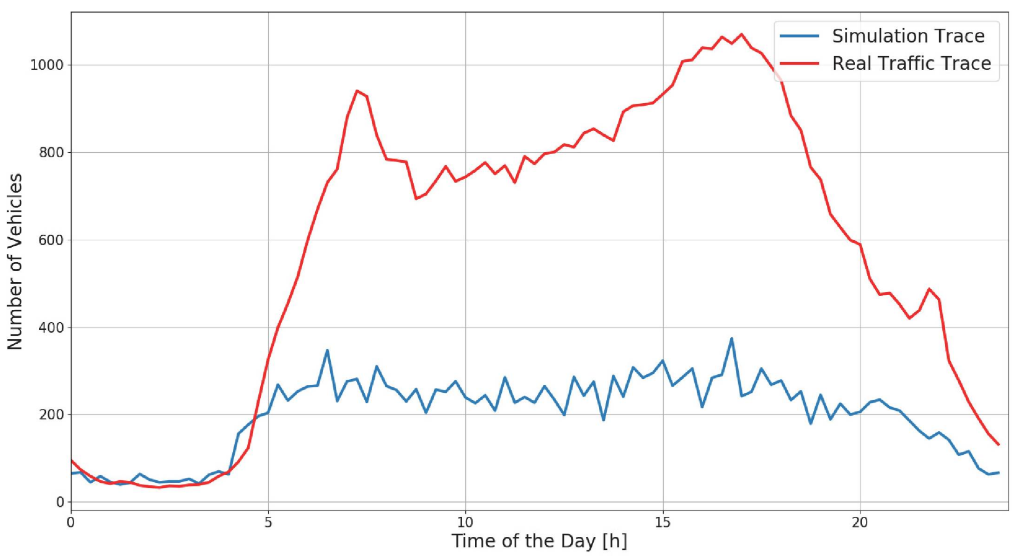
Source: Author

Figure 4.20: ID-04 in Google Maps



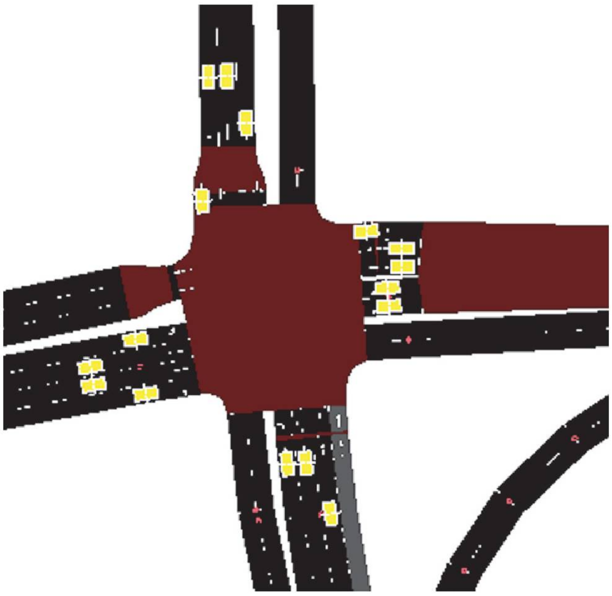

Source: (GOOGLE, 2019)

Figure 4.21: Crossing ID-24 Analysis




Source: Author

Figure 4.22: ID-24 in InTAS



Source: Author

Figure 4.23: ID-24 in Google Maps


Source: (GOOGLE, 2019)

Table 4.7: Junctions Analysis

| Crossing ID | Real Trace | Simulated Trace | NRMSE |
|---|---|---|---|
| 01 | 14,326 | 5,788 | 0.60310 |
| 02 | 14,913 | 17,455 | 0.31824 |
| 03 | 17,549 | 17,599 | 0.25859 |
| 04 | 18,414 | 18,760 | 0.15390 |
| 05 | 19,294 | 8,860 | 0.55074 |
| 06 | 23,516 | 15,642 | 0.36205 |
| 07 | 25,670 | 15,308 | 0.44207 |
| 08 | 26,139 | 18,510 | 0.31191 |
| 09 | 27,659 | 13,009 | 0.53313 |
| 10 | 29,318 | 21,314 | 0.31230 |
| 11 | 30,980 | 22,922 | 0.32731 |
| 12 | 31,410 | 15,391 | 0.52178 |
| 13 | 32,282 | 13,842 | 0.57122 |
| 14 | 32,430 | 18,957 | 0.42661 |
| 15 | 35,364 | 18,644 | 0.47811 |
| 16 | 35,711 | 16,418 | 0.56103 |
| 17 | 36,179 | 14,558 | 0.59866 |
| 18 | 37,075 | 19,447 | 0.51641 |
| 19 | 38,612 | 20,836 | 0.50466 |
| 20 | 38,972 | 19,082 | 0.53669 |
| 21 | 39,548 | 25,277 | 0.40980 |
| 22 | 43,097 | 19,241 | 0.55544 |
| 23 | 44,651 | 18,383 | 0.61611 |
| 24 | 55,505 | 19,134 | 0.66435 |

Source: Author

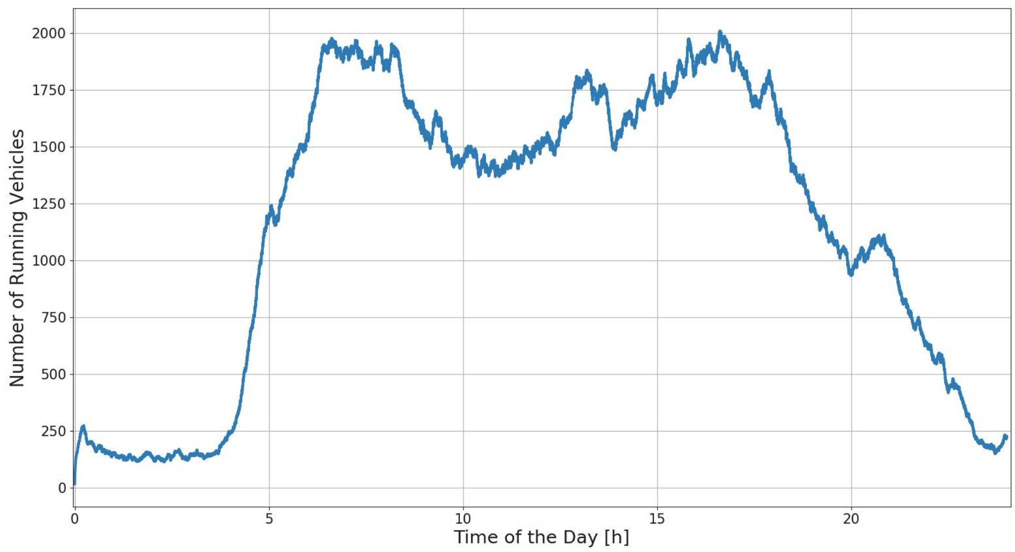
Based on all these results, it is possible to infer that the InTAS provides similarities when compared with the real traffic. Further analysis of this scenario is shown in Figure 4.24, presenting InTAS' traffic behavior. The first peak starts right before 4:00, representing the flow for the beginning of the first shift gathered with the end of the night shift. Traffic is still increasing until around 6:25 and remains in the morning peak up to 8:20 with approximately 1,950 vehicles on the streets. The morning peak time is a bit before when most offices start their activities and is also correlated with students. After reaching the morning peak, the traffic behavior starts decreasing until 10:25, when the lowest number of vehicles driving around the scenario after the morning peak is observed. This valley computes 1,372 vehicles on the simulation. After this time the number of vehicles grows until 12:20, where a peak in this growth behavior is notice. This noon peak lasts from 12.20 up to 13:12, which can represent people going for lunch and the finish of morning classes. After lunchtime peak the number of vehicles still growing, representing the end of the first shift and the beginning of the second shift. The phenomenon slightly increases the number of running vehicles until the afternoon peak around 16:40, which presents the highest number of running vehicles for InTAS, with 2,002 in total. Afterward, the number of vehicles decreases until the end of the day, with a slight peak from 20:00 to 21:09, representing the end of the second shift and the beginning of the third shift.

## 4.5 Final Considerations

In this Chapter, the InTAS, a novel traffic scenario for SUMO based on Ingolstadt city, was presented and validated. It has described the four methodologies steps applied for InTAS development.

The first step introduced the map creation, where the selected are and its analysis have been discussed. This stage presented the road network, which has been imported from OSM, and the correction comparing the OSM data with the Google Maps

Figure 4.24: Running Vehicles for InTAS




Source: Author

satellite view. Furthermore, it was considered parking areas, traffic lights systems, bus stops, and buildings. A total of thirteen traffic lights had their programs simulated according to the real program deployed on these intersections.

Traffic demand is the second step approached and it modeled the traffic based on socio-demographic data according to the method `activitygen`. The demographic data is online available and has a high level of detail, assisting to define each parameter from `activitygen`. The selected area for InTAS did not take into consideration all Ingolstadt, however, the traffic impact from villages not represented in the scenario was taken into consideration. The public transport system has also been simulated based on the online available data provided by the INVG. Furthermore, a traffic flow optimization has been implemented, seeking to reach an equilibrium according to Gawron's method. This equilibrium was reached on the 25$^{th}$ iteration applying the `duaIterate` tool.

The third stage presented in this Chapter was the InTAS simulation. This step gathered the map and the traffic demand, and it managed the scenario according to the simulation's parameters. The parameters were set to simulate traffic over 24 hours of a business day, and are presented in Table 4.6. The simulation's parameter that most influences the traffic simulation is the *device.rerouting.probability*, and for this, an algorithm to find the best value for InTAS has been developed. This algorithm compared the simulation output with a data-set, where real traffic data is represented. The comparison implemented the NRMSE for each probability regarding real-data from October 2019. The best fit for *device.rerouting.probability* was the probability 0.88 with an NRMSE of 0.438234.

The last step in InTAS development was the scenario's validation. This stage compared the simulation output with real traffic numbers from November 2019, comparing the simulation data with the real data. The first analysis compared vehicle numbers from 24 intersections during the day creating the simulation trace and the real trace, as depicted in Figure 4.14. This figure shows a higher absolute error during peak hours. However, it also shows that the behavior from 0:00 and 6:00, and from 22:12 until the end of the simulation, are similar in the simulation and real data. An absolute error for each time-window has been plotted, and it is represented in Figure 4.15, proving that the highest absolute error occurred during peak times. Intending to measure the impact of each absolute error, the NRMSE per time window has been performed and showed in Figure 4.16, where is observed that even with a higher absolute error during peak times, the impact on the traffic is close to the average value for the scenario. Furthermore, a QQPlot test to evaluate the behavior of both traces has been implemented, and it has been observed that the initial values for the simulation do not match with real values, but for highest quantiles, a behavior close to the real trace is depicted, suggesting that the traces may have the same distribution.

The NRMSE for each of 24 intersections is presented in Table 4.7 and the best and worst intersections have been analyzed comparing simulation trace and real trace. The best junction presented an NRMSE of 0.15390 and represents a three-way intersection between an arterial road and a way. On the other hand, the worst-case presented an

NRMSE of 0.66435 and is one of the most important crossings of Ingolstadt, where over 55,000 vehicles daily drive.

In the end, InTAS' traffic behavior has been introduced in Figure 4.24, considering all running vehicles in the simulation. Three higher peaks are observed, which represent the morning, noon, and afternoon peaks. The morning and afternoon peaks are directly related to the offices' work-time. The noon peak is regarding lunchtime and end of morning school time. Two smaller peaks are observed, one around 5:00, and the other around 20:45. Both are directly related to the work shift time, where the number of vehicles increases between the first and second shifts, and the second and third shifts, consecutively.

# CHAPTER 5

# CONCLUSION

This thesis introduced InTAS, the Ingolstadt traffic scenario for SUMO. This traffic scenario is the first SUMO-based scenario using programs close to the deployed in the real traffic light system, and not only standards programs provided by SUMO, as implemented in LuST and MoST. Traffic light programs' lengths and phases for thirteen crossings were provided by the City of Ingolstadt. Nine of them cover the 2 km length of the Goethestraße, where around 21,000 vehicles daily drive. Three exists of Autobahn A9 have their TLS simulated on this thesis. Two are between the A9 and the Manchingerstraße, where 34,500 vehicles drive over a day, and the other is between A9 and Römerstraße, which is a gate entrance for around 32,000 vehicles per day. InTAS faithfully represents the road network of Ingolstadt, due to the thoroughly work to correct all the streets based on the information on-line available on satellite view from Google Maps. Traffic modeling took into consideration where people live and where they spend their daily activities, like work, school, and free time. Also, the nine traffic light systems junctions here represented have improved the green waves, traffic light cycle time, and phases description. All these features establish an environment for simulations, seeking a real-world representation, and can cooperate with all kinds of vehicular simulations, e.g. C-ITS and VANETs. Furthermore, soon, this scenario will represent one of the first cities on Germany where the Car2X communication system is available (AUDI AG, 2019). However, InTAS is the first realistic traffic scenario for SUMO analyzing a City with this feature.

InTAS has been developed considering detailed demographic data and a solid road topology, which represent the real roads of Ingolstadt. This scenario covers an area of 51.54 Km$^2$, with a total road length of 771.23 Km. It also represents a trustworthy public transport network, simulating 56 bus lines running over 1,620 daily trips and covering 880.6 km of routes length. A total of 21,756 buildings were inserted aiming to create an environment for network simulations, allowing effects such as signal-fading and shadow areas. Traffic demand was generated based on demographic data, taking into consideration the impact of all public parking areas, which represented 13 different places with 4,247 parking lots. All traffic lights were implemented using the `actuated` method, which optimizes the traffic flow. Based on the fact that traffic light systems play an important roll in traffic simulations.

Ingolstadt Scenario has been modeled and validated using real traffic information from 24 intersections. A data-set was elaborated based on the information from each junction. This data-set computed the average number of vehicles driving through the crossing for an entire day. Detectors to count the vehicle's number are placed on all lanes from each intersection. This research only considered data from Tuesdays, Wednesdays, and Thursdays, due to the heaviest traffic faced on these days and its regularity. An algorithm was implemented to reach the best *device.rerouting.probability* value, comparing the simulation output and the vehicle's number from October of 2019. To evaluate the simulation output, virtual detectors were placed on the simulation as close as they are in the real world. The detectors' output was compared with the real data-set, based on November of 2019, creating traffic traces to be analyzed. The first analysis compared the total number of vehicles for all detectors, presenting an NRMSE of 0.438234. Thereafter, the NRMSE for each intersection was evaluated, and the best and worst-case were deeper discussed and improved. To evaluate the traffic over the detectors a QQPlot test was also implemented, and it showed that for the initial quantiles the traces do not have a similar distribution. On the other hand, for the remaining quantiles, both Traces presented a closer behavior.

The algorithm implemented to reach the best *device.rerouting.probability* value sought for optimization only in this parameter, due to the impact in VANETs branch other parameters might cause. As an example, *time-to-teleport* is a parameter that may decrease the NRMSE and increases traffic realism. However, changing this parameter will lead to a great number of vehicles teleporting, which will directly impact the Cooperative Awareness Message (CAM) (Festag, 2014) that is broadcasted based on the actual position of the car and Collective Perception Message (CPM) (Günther et al., 2016) that shares relevant information about obstacles surrounding the vehicle according to their position. In SUMO, when a vehicle is teleported, it is removed from the simulation and placed in a further position on its route.

Based on the values reached when comparing simulation trace with real trace, this thesis has presented the best fit for a traffic scenario for Ingolstadt, when implementing the available data, tools, and methodologies.

Finally, it can be concluded that it is feasible to develop a realistic traffic scenario for SUMO based on the city of Ingolstadt, applying the demographic data freely available. It has been also proved that it is possible to measure traffic accuracy by comparing the simulation's output and real data. However, this poses the next research question: How to mitigate the mismatch between the real trace and simulation trace, i.e. how to increase scenario's realism? This question leads to future works, which are the improvements that can be made to the InTAS.

In the end, it is known that simulation's values will never fit real-life values, due to the complexity behind vehicle traffic, estimations, and simplifications on human behavior modeling. However, a realistic traffic scenario must be as realistic as possible, because it will be the basis for further improvements. These simulations are not only concentrated on the VANETs area, but VANETs requires more reliable data. Hence, to raise trust in new applications developed for this area, InTAS has been presented.

## 5.1 Future Work

Tools should evolve, and for this reason, improvement points are presented in this Section. Currently, only thirteen traffic lights reproduce the programs deployed in the real traffic lights. The other simulated traffic lights in InTAS should follow the same perspective, which means a wider partnership with the City of Ingolstadt to have access to this information. Traffic demand modeling took into consideration the average values for the parameters `population` and `workPositions` for each Ingolstadt's sub-area were considered. When the average value is implemented, it brings particular errors. Hence, it is possible to model demographic data with more details, considering smaller regions inside the sub-areas, and as this information is not on-line available, once again is necessary to strengthen the relationship with the City of Ingolstadt to have this data. At present, no street parking is modeled in IntAS, and an interesting improvement would be to implement this feature, allowing vehicles to park along the road, as it happens in reality. For the city center area, information as the number of park slots per region is provided by the City (INGOLSTADT, 2020), which can be implemented in the simulation, but more information is demanded to cover all the city.

The other point faced in this thesis is the time-base between demographic data and traffic data. Demographic data was published by the authorities considering the year 2018. On the other hand, real traffic used to validate InTAS were collected between August and December 2019. Even though the time between the demographic data and traffic is only eight months, it might change the traffic numbers and add errors.

Furthermore, a fitness function for weighting SUMO parameters, the relevant features for VANETs, and traffic realism could be implemented. This function can evaluate the impact on traffic realism considering all SUMO's parameters and lead to a better fit, decreasing the error value. Among the solutions approaches to define the optimum value is Artificial Intelligence (SWARNKAR, 2020), where Genetic Algorithm, Swarm Optimization, and Ant Colony Optimization are feasible optimization solutions to find the best fit.

# REFERENCES


AMT FÜR VERKEHRSMANAGEMENT UND GEOINFORMATION. **Verkehrsentwicklungsplan 2025**. 2018. [online]. https://www.ingolstadt.de/media/custom/2789_1089_1.PDF?1540885166, last viewed December 2019.

AUDI AG. **Audi vernetzt sich mit Ampeln in Deutschland**. 2019. [online]. https://www.audi-mediacenter.com/de/pressemitteilungen/audi-vernetzt-sich-mit-ampeln-in-deutschland-11649, last viewed January 2020.

BIEHLE, T.; KRUMBIEGEL, K. **Triggering Conditions and Data Quality Dangerous Situation**. September 2019.

BOX, G.; JENKINS, G.; REINSEL, G. **Time series analysis : forecasting and control**. Wiley, 2008. (Wiley series in probability and statistics). ISBN 978-0-470-27284-8.

BUCHHOLZ, J. **Triggering Conditions and Data Quality Exchange of IRCs**. September 2019.

CODECA, L. **Dynamic Vehicular Routing in Urban Environments**. 77-82 p. Thesis (Ph.D.) — Université du Luxembourg, 12 2016.

CODECA, L.; FRANK, R.; ENGEL, T. Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research. In: **2015 IEEE Vehicular Networking Conference (VNC)**. 2015. p. 1–8. ISSN 2157-9865.

CODECA, L.; HÄRRI, J. Monaco SUMO traffic (MoST) scenario: A 3d mobility scenario for cooperative its. In: **SUMO 2018- Simulating Autonomous and Intermodal Transport Systems**. 2018. v. 2, p. 43–55.

DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische mathematik**, Springer, v. 1, n. 1, p. 269–271, 1959.

ENGEL, S. **Triggering Conditions and Data Quality Adverse Weather Conditions**. September 2019.

ENGEL, S. **Triggering Conditions and Data Quality Stationary Vehicle Warning**. September 2019.

FELL, F.; TANENBAUM, M. Preliminary comparisons of the wgs84(egm 96) geoid with national vertical datums. In: **MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No.01CH37295)**. 2001. v. 1, p. 571–574 vol.1. ISSN null.

Festag, A. Cooperative intelligent transport systems standards in europe. **IEEE Communications Magazine**, v. 52, n. 12, p. 166–172, 2014.

GAWRON, C. An Iterative Algorithm to Determine the Dynamic User Equilibrium in a Traffic Simulation Model. In: **Proceedings of the 3rd Industrial Simulation Conference 2005, EUROSIS-ETI**. 1998. p. 285–290. ISSN 90-77381-18-X.

GOOGLE. **Google Maps**. 2019. [online]. `https://www.google.com/maps/@48.7621617,11.4349463,6820m/data=!3m1!1e3`, last viewed June 2019.

Günther, H. et al. Collective perception and decentralized congestion control in vehicular ad-hoc networks. In: **2016 IEEE Vehicular Networking Conference (VNC)**. 2016. p. 1–8.

HERTKORN, G. **Mikroskopische Modellierung von zeitabhängiger Verkehrsnachfrage und von Verkehrsflußmustern**. 99-116 p. Thesis (Ph.D.) — Universität zu Köln, July 2004.

HUMBERG, G. **INTEGRIERTES VERKEHRSMODELL DATENERFASSUNGSKONZEPTE INGOLSTADT**. 2017. [online]. `https://www.gevas-ingenieure.de/pdf/REF_VP_Verkehrsmodell_Ingolstadt_IN-VM002_neu.pdf`, last viewed November 2019.

INGOLSTADT, S. **Parken in der Altstadt**. 2020. [online]. `https://www.ingolstadt.de/Rathaus/Verkehr/Parken/index.php?&object=tx,2789.2&ModID=10&FID=2789.111.1&call=suche&kat=&kuo=1&ort=0&sfwort=0&k_sub=0&La=1`, last viewed July 2020.

INVG. **Ingolstädter Verkehrsgesellschaft**. 2019. [online]. `https://www.invg.de/`, last viewed November 2019.

JAVA OPENSTREETMAP. **JOSM**. 2019. [online]. `https://josm.openstreetmap.de/`, last viewed July 2019.

KRAJZEWICZ, D. Kombination von taktischen und strategischen einflüssen in einer mikroskopischen verkehrsflusssimulation. In: **Fahrermodellierung in Wissenschaft und Wirtschaft, 2. Berliner Fachtagung für Fahrermodellierung**. 2009. p. 104–115. ISSN 1439-958X.

KRAUß, S. **Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics**. 115 p. Thesis (Ph.D.) — DLR-Forschungsbericht, August 1998.

KUMAR, V.; ANAND, V. Development of web map service for openstreetmaps (osm) data. In: **Proceedings of International Conference on ICT for Sustainable Development**. 2015. v. 1, p. 309–318. ISSN 2194-5365.

LOPEZ, P. A. et al. Microscopic Traffic Simulation using SUMO. In: **The 21st IEEE International Conference on Intelligent Transportation Systems**. IEEE, 2018. Disponível em: <<https://elib.dlr.de/124092/>.

MCKENNEY, D.; WHITE, T. Distributed and Adaptive Traffic Signal Control within a Realistic Traffic Simulation. In: **Engineering Applications of Artificial Intelligence**. 2013. v. 26, p. 574–583.

MENEGUETTE, R. I.; GRANDE, R. E. D.; LOUREIRO, A. A. F. **Intelligent Transport System in Smart Cities**. Springer, 2018. (Urban Computing). ISBN 9783-319-93332-0.

MIUCIC, R. **Connected Vehicles**. Springer, 2019. (Wireless Networks).

NEUMEIER, S.; HÖPP, M.; FACCHI, C. Yet another driving simulator openrouts3d: The driving simulator for teleoperated driving. In: **2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)**. Graz, AUT, 2019. p. 1–6. ISSN 2378-1289.

OBERMAIER, C.; RIEBL, R.; FACCHI, C. Fully reactive hardware-in-the-loop simulation for vanet devices. In: **2018 21st International Conference on Intelligent Transportation Systems (ITSC)**. 2018. p. 3755–3760. ISSN 2153-0009.

OECD-ITF. **ITF Transport Outlook 2017**. 2017. [online]. https://www.oecd-ilibrary.org/docserver/e979b24d-en.pdf?expires=1580749674&id=id&accname=guest&checksum=06E88E987008C4098E6B1A4865C1B285, last viewed June 2019.

OPENSTREETMAP. **OpenStreetMap**. 2019. [online]. https://openstreetmap.org, last viewed June 2019.

PEDROSO, C. M. **Distribuições de Probabilidade**. 2018. http://www.eletrica.ufpr.br/pedroso/2017/TE816/Distribuicoes.pdf, last viewed January 2020.

RICCI, V. **Fitting Distributions with R**. 2005. [online]. https://cran.r-project.org/doc/contrib/Ricci-distributions-en.pdf, last viewed January 2020.

SOARES, J. et al. Realistic traffic scenarios using a census methodology: Vila real case study. In: **2014 IEEE PES General Meeting | Conference Exposition**. 2014. p. 1–5. ISSN 1932-5517.

Sommer, F. D. C. **Vehicular Networking**. Cambridge University Press, 2014.

STADT INGOLSTADT. **Ingolstadt in Zahlen 2018/2019**. 2018. [online]. https://ingolstadt.de/media/custom/465_1995_1.PDF?1534841095, last viewed September 2019.

STADT INGOLSTADT. **Kleinräumige Statistiken zum 31.12.2018**. 2019. [online]. https://www.ingolstadt.de/media/custom/3052_1647_1.PDF?1563171021, last viewed October 2019.

STADT INGOLSTADT. **Verkehrspolitische Ziele der Stadt Ingolstadt**. 2019. [online]. https://www.ingolstadt.de/Rathaus/Verkehr/Verkehrsmanagement/Verkehrsplanung/index.php?La=1&object=tx,465.4793.1&kat=&kuo=2&sub=0, last viewed December 2019.

STADT INGOLSTADT. **Öffentliche Parkhäuser, -plätze und Tiefgaragen**. 2019. [online]. https://www.ingolstadt.de/parken, last viewed December 2019.

STADT INGOLSTADT. **Kartografie: Statistik und Stadtforschung**. 2020. [online]. https://www.ingolstadt.de/media/custom/2789_150_1_g.JPG?1501493618, last viewed January 2020.

SUMO. **Demand/Dynamic User Assignment**. 2020. [online]. https://sumo.dlr.de/docs/Demand/Dynamic_User_Assignment.html, last viewed June 2020.

SUMO. **Simulation/Output/TripInfo**. 2020. [online]. https://sumo.dlr.de/docs/Simulation/Output/TripInfo.html, last viewed June 2020.

SWARNKAR, A. Artificial intelligence based optimization techniques: A review. In: KALAM, A. et al. (Ed.). **Intelligent Computing Techniques for Smart Energy Systems**. Singapore: Springer Singapore, 2020. p. 95–103. ISBN 978-981-15-0214-9.

TCHOUANKEM, H.; ZINCHENKO, T.; SCHUMACHER, H. Impact of buildings on vehicle-to-vehicle communication at urban intersections. In: **2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)**. 2015. p. 206–212. ISSN 2331-9860.

TIELERT, T. et al. The impact of traffic-light-to-vehicle communication on fuel consumption and emissions. In: **2010 Internet of Things (IOT)**. 2010. p. 1–8. ISSN null.

UPPOOR, S.; FIORE, M. Large-scale urban vehicular mobility for networking research. In: **2011 IEEE Vehicular Networking Conference (VNC)**. 2011. p. 62–69. ISSN 2157-9857.

WU, N.; LI, D.; XI, Y. Distributed weighted balanced control of traffic signals for urban traffic congestion. **IEEE Transactions on Intelligent Transportation Systems**, v. 20, n. 10, p. 3710–3720, Oct 2019. ISSN 1558-0016.