POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Co-simulation environment to assess human perception of Cooperative Connected and Automated Vehicles: calibration of the microscopic traffic simulation

Tesi di Laurea Magistrale in
Mechanical Engineering - Ingegneria Meccanica

Author: **Álvaro Varela Daniel**

Student ID: 986140
Advisor: Prof. Gianpiero Mastinu
Academic Year: 2022-23

Abstract

Cooperative Connected and Automated Vehicles have been proposed as a solution to address congestion and safety issues in urban environments. However, it is difficult to test in real environments how a human driver of a non-automated vehicle or a passenger of an automated vehicle perceives these improvements. To overcome this, driving simulators are seen as a great tool when a sufficient level of realism is provided both in the visual scene and the traffic behavior.

On the scope of this thesis, we designed a traffic scenario calibrated based on real traffic data in an urban roundabout environment. We used the software SUMO, where we individuated the main parameters affecting the output of the simulation using ANOVA tests and then calibrated them to represent reality. This scenario will be used for the training of an automated vehicle policy and as the traffic scenario used in the co-simulation that allows to introduce the human in the loop.

After this, we aligned the traffic simulator scenario with the driving simulator scenario and performed some tests in which non-professional drivers were exposed to different traffic mixes of non-automated and automated vehicles. As the preliminary results show, the policy was perceived as safe and more fluent.

Keywords: microscopic simulation, calibration, co-simulation

Abstract in lingua italiana

Per risolvere i problemi legati alla congestione e alla sicurezza stradale in ambienti urbani, Cooperative Connected and Automated Vehicles sono stati proposti come una possibile soluzione. La loro introduzione aumenta la sicurezza e la scorrevolezza del traffico d'accordo alla letteratura disponibile. Tuttavia, ci sono delle difficoltà per testare nella realtà come un guidatore di una macchina non automatizzata interagisce e percepisce il comportamento di queste vetture automatizzate, e quanto sono comode per i passeggeri. Per superare queste difficoltà, i simulatori di guida sono un ottimo strumento quando si usano con uno scenario di traffico realistico.

Nello scopo di questa tesi, abbiamo disegnato uno scenario di traffico calibrato sulla base di una rotonda urbana reale. Per la simulazione di traffico abbiamo usato il software SUMO, individuando i parametri più significativi per l'output della simulazione e calibrandoli per rappresentare il traffico reale. Questo scenario sarà poi utilizzato per allenare una policy in grado di gestire il comportamento delle vetture automatizzate nell'approccio d'una rotonda urbana, e sarà anche usato per la co-simulazione che permette introdurre un guidatore umano nel ciclo.

Dopo di questa calibrazione, abbiamo allineato lo scenario di traffico con quello del simulatore di guida per fare delle prove in cui guidatore non professionali hanno sperimentato diverse combinazioni di traffico di veicoli automatizzati e non automatizzati. I primi risultati ottenuti mostrano una preferenza sia per la scorrevolezza del traffico sia per la sicurezza per lo scenario con un maggiore numero di veicoli automatizzati

Parole chiave: simulazione di traffico, calibrazione, co-simulazione

Contents

Introduction

During the last decades, to address safety and fluency problems related to the increasing number of vehicles on the road, different solutions have been implemented. Among these solutions, we can find the introduction of dedicated infrastructure to address conflicts in intersections, such as roundabouts; and, more recently, the introduction of Cooperative Connected and Automated Vehicles (CCAV) which offer promising results [1]. However, the introduction of this new technology faces challenges such as safety in complex urban environments, like roundabouts, and lack of acceptance by other road users. To address this, it is important to test the solutions prior to introducing them into the market. However, testing of innovative technologies is expensive and potentially dangerous. To overcome these issues, simulation is a good alternative, specially at early stages of development. However, testing CCAV related technologies includes different challenges, among which the communication between all the required components is key. To overcome this difficulty, the AI@EDGE architecture is used.

This Master's thesis project is framed inside the European project AI@EDGE that looks into industry-relevant applications of Artificial Intelligence (AI) and 5G technologies through 4 different use cases [2]. The first use case, *Virtual validation of vehicle co-operative perception*, is the one relevant for this thesis.

The objective of this use case is to test AI-based automated vehicles using V2N2V (Vehicle-to-Network-to-Vehicle), specifically in a roundabout scenario. To do this, a real road network and traffic will be reproduced in a traffic micro simulation. This will then be used to train an AI policy to govern automated vehicles looking at various KPIs related to traffic conditions. The co-simulation of traffic simulation, in conjunction with the AI policy, and the driving simulator allows the assessment of impacts on drivers' perception. The simulation will also take into account real-world latency, ensuring accurate testing conditions. The main components of the required co-simulation are:

- Microscopic traffic simulator
- Dynamic driving simulator

- AI-controlled policy for the automated vehicles
- Software and Hardware to reproduce the communications between vehicle and network

This thesis main focus is the first of these requirements: the microscopic traffic simulation. This traffic simulation comprises the definition of the road network and traffic conditions. Regarding the software used in the scope of this project, although different alternatives are available, Simulation of Urban MObiltiy (SUMO) [3] is chosen since its open-source nature allows for a higher degree of flexibility and interoperability with other required software, while maintaining a satisfactory level of realism of the simulations.

In previous stages of the project, preliminary tests were performed to analyze the overall workflow and identify any potential issues. For these tests, a roundabout scenario was designed from scratch and the vehicles were controlled using Car Following Models (CFM) parameters coming from available literature, in this case the Intelligent Driver Model (IDM) was used since it can faithfully reproduce the behaviour of human driver with good performance. This scenario was used to train an AI policy which was later tested in the driving simulator to assess the impact of automated vehicles on perceived safety and fluency of the traffic. It is important to note that this scenario was not based on real network but represented a general three-legged roundabout.

To improve the validity of the conclusions drawn from the preliminary tests, it was deemed necessary to perform new tests using a traffic scenario reproducing a real environment. This led to the scope of this thesis, which focuses on designing and calibrating the traffic simulation and aligning it with the driving simulator scene.

To perform the design of the scenario different aspects need to be considered and are dealt with in this thesis. The first of them is the selection of a real roundabout and the acquisition of real data that is later used as reference for creating the traffic simulation scenario. The second is the calibration of specific parameters available in the software to better reproduce the real conditions of traffic. It is important to consider that the scenario and parameters are later used to train a policy for automated vehicles that is finally tested in the driving simulator. Hence, this step is of great relevance since the AI output can only be as good and realistic as the input it has been trained after.

This thesis is structured as follows:

- Chapter 1 covers the relevant information on roundabout infrastructure, Cooperative Connected and Automated Vehicles, traffic simulation and microsimulation, and the calibration process.

- Chapter 2 provides a detailed explanation of the AI@EDGE project and its relevance.
- Chapter 3 presents the process of defining, calibrating and modelling the real scenario.
- Chapter 4 gives an overview of the process of aligning the traffic scenario and the scenario presented in the driving simulator. It also includes test results using the new scenario.
- Chapter 5 offers the main conclusions of this work as well as a review of the undertaken process with suggestions for possible further improvement.

1 | State of the art

In this chapter a brief recollection of the state of the art is presented. The analysis is divided in five different sections even though some of the concepts are transversal. This sections are: (1) Introduction to roundabouts (2) Connected and Automated Vehicles (3) Traffic simulation and microsimulation (4) Simulation of Urban MObility (SUMO [3]) software (5) Calibration and validation of traffic simulations.

1.1. Roundabouts

The increase in welfare during the last century has led to greater mobility, increasing the number of vehicles on the roads This increased vehicle number and usage in all kinds of situations has led to congestion and safety concerns specially in junctions. To improve the performance and safety of road networks, the design of junctions plays a crucial role and roundabouts have been found to be a good solution.

Roundabouts are a form of intersection whose key features are circular shape, yielding control of entering traffic and a specific set of right-of-way rules [4]. The roundabouts aim to accomplish speed reduction of the incoming traffic as well as allowing different typologies of vehicles to circulate on them. Splitter islands contribute to separate incoming and outgoing traffic on a leg and flare on the entries allows to increase the capacity.

The main features as presented in figure 1.1 are:

- Central island: central area of a roundabout around which the traffic flows. Usually this area is raised to avoid crossing it.
- Splitter island: raised or painted area at the approach of a roundabout to separate incoming and outgoing traffic as well as induce speed reduction of the incoming traffic.
- Apron: mountable portion of the central island to allow traffic of bigger vehicles.
- Yield line: pavement marking to indicate the entry of the roundabout.

- Landscaping buffer: elements separating road traffic from pedestrian traffic.

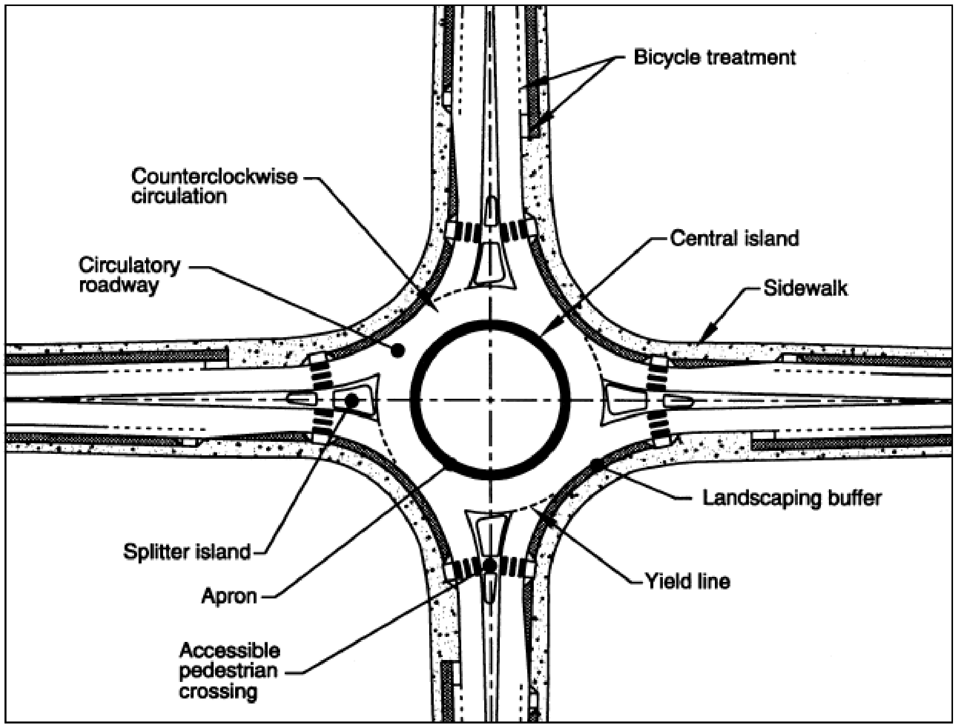


Figure 1.1: Main features of a roundabout [4].

The geometrical characteristics of a roundabout shown in figure 1.2 are:

- Inscribed circle diameter: basic parameter to define a roundabout. Measured between outer edges.
- Circulatory roadway width: measured between outer edge and the central island, excluding any apron.
- Departure width: width of the lane downstream the roundabout.
- Approach width: width of the lane upstream the roundabout.
- Exit width: width of the exit lane where it meets the inscribed circle measured perpendicularly from the right edge of the lane.
- Entry width: width of the entry lane where it meets the inscribed circle measured perpendicularly from the right edge of the lane.
- Entry radius: radius of curvature of the outside curb at the entry.
- Exit radius: radius of curvature of the outside curb at the exit.

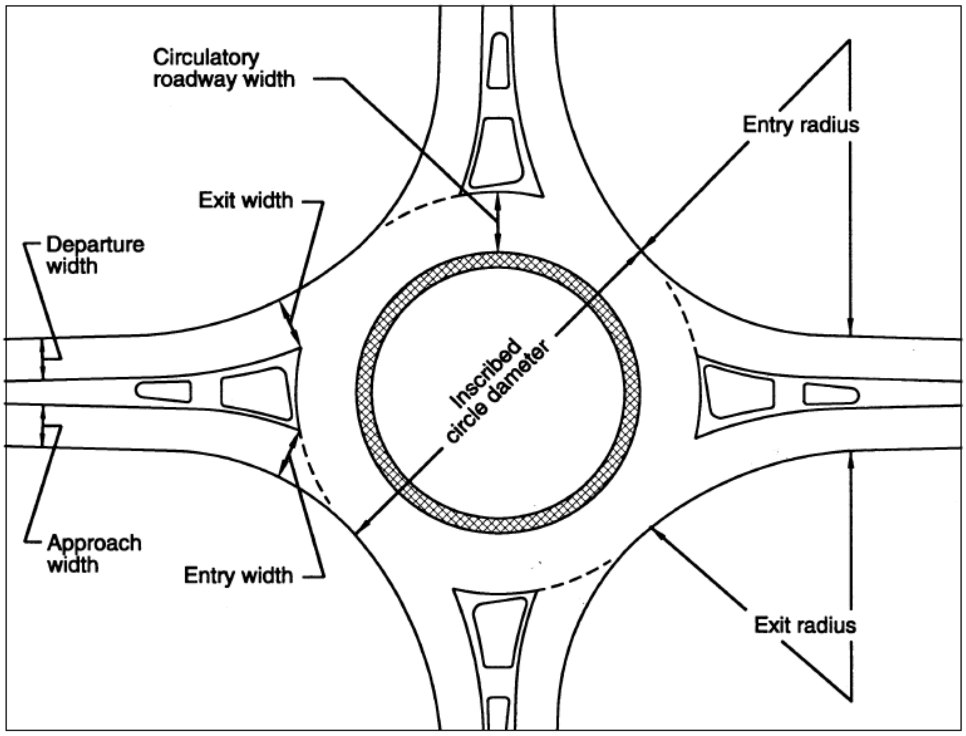


Figure 1.2: Main geometrical characteristics of a roundabout [4].

Different types of roundabouts are conceived depending on the capacity requirements of the road on which they are placed. The main differences between these types of roundabouts are the number of lanes, the size and desirable entry speed. The characteristics according to the National Cooperative Highway Research Program [4] are reported in table 1.1.

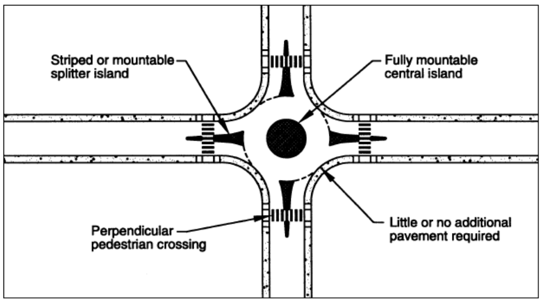
Main roundabout types

Design Element	Mini-Roundabout	Single-Lane Roundabout	Multilane Roundabout
Maximum desired speed	25-30 km/h	30-40 km/h	40-50 km/h
Maximum entering lanes per approach	1	1	2+
Central island	Trasversable	Raised	Raised

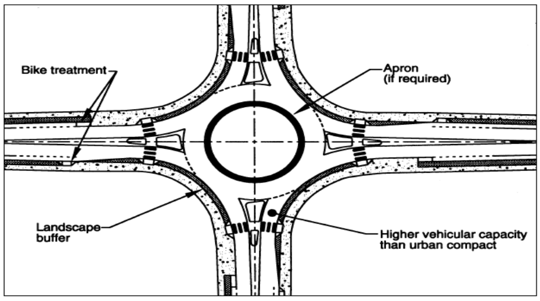
Table 1.1: Roundabout characteristics according to the National Cooperative Highway Research Program (USA) [4].

These characteristics are similar to the ones seen in Germany [5] and Italy [6] where the classification regards the inscribed circle diameter (mini-roundabouts: 14-25 m; compact roundabouts 25-40 m; and conventional roundabouts: 40-50 m).

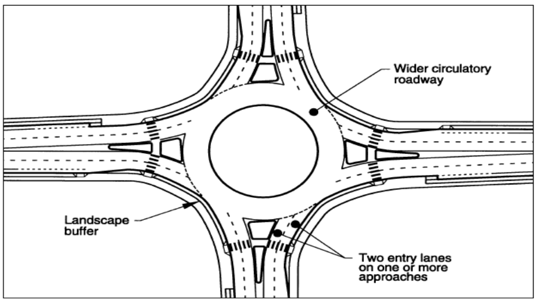
Another possible layout for roundabouts are Turbo roundabouts. It has been shown in the literature that they can improve safety compared to two-lane roundabouts and yet improve capacity when compared to smaller, single-lane, roundabouts [7]. To provide a better understanding of the types of roundabouts in figure 1.3 the basic urban roundabouts and turbo roundabouts schemes are showed.



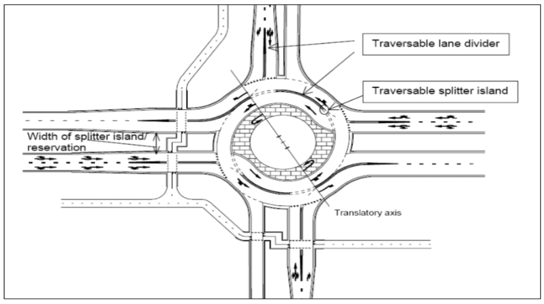
(a) Urban mini-roundabout [4].



(b) Urban single lane roundabout [4].



(c) Urban double lane roundabout [4].



(d) Turbo roundabout [8].

Figure 1.3: Schemes of different types of urban roundabouts

The main safety advantage of roundabouts when compared to normal intersections is that roundabouts reduce the amounts of possible conflicts, especially those that involve crossing trajectories and are the most dangerous. Multiple lane roundabouts do not eliminate this type of conflict, but they reduce the number of crossing conflicts from 16 to 8. This concept is clear when looking at figure 1.4.

However, analyzing the safety of intersections is not only a matter of studying conflicts between vehicles, it is also required to consider other users of the road such as pedestrians or cyclists [9]. The safety of pedestrian is known to increase, when looking at the reduction of number of injuries when comparing regulated intersections and roundabouts with unsignalized intersections. This decline is steeper in the case of roundabouts, specially in areas where low enforcement of traffic regulations is reported [10], this could be explained by the induced speed reduction due to the geometry of the roundabout [11]. It also has been shown analysing available accidents data that roundabouts significantly reduce serious injury accidents [12]. Analyzing traffic conflicts using surrogate measures of safety has shown the importance of geometrical design of roundabouts, mainly outer diameter

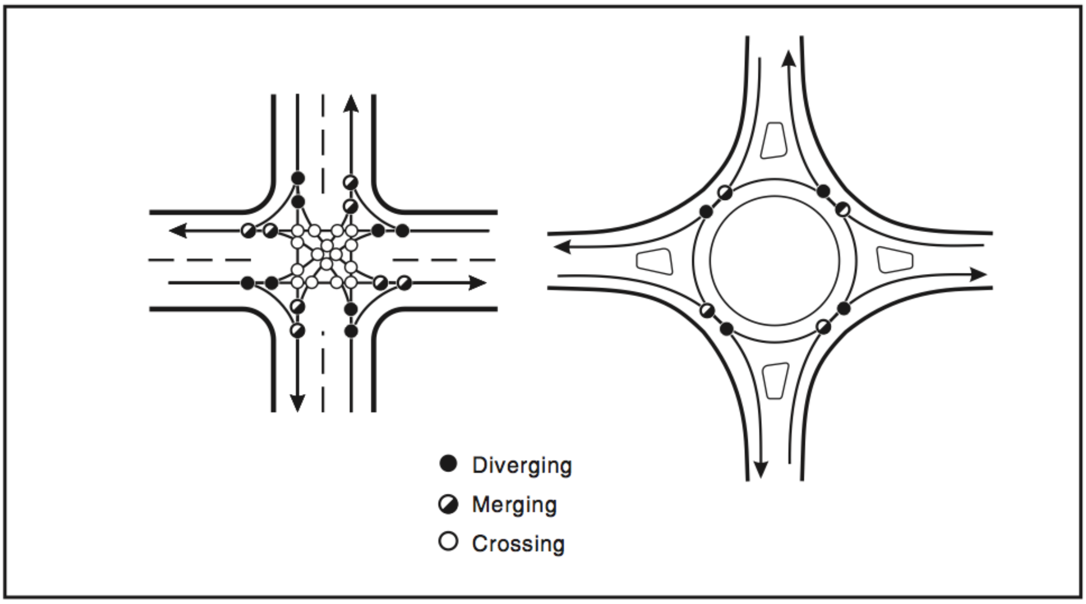


Figure 1.4: Vehicular conflicts for normal intersection and roundabouts [4].

and number of lanes, and flow distribution around the different legs, showing that it is important to analyze safety for all the possible conditions of flow [13].

The main disadvantage of roundabouts comes in high flow situations, in this circumstances the signalised intersections show advantages over roundabouts, allowing a higher throughput [14]. Another situation in which traffic is better controlled by traffic lights than by roundabouts in terms of performance are unbalanced flows coming from the different legs. A possible solution is to merge both alternatives and control the entrance of the roundabout with traffic lights. In this case, the flow can be further improved by applying the correct control strategy of the traffic lights [15], the main drawback of this approach is that the origin/destination flows must be known a priori. Another solution could be the already mentioned turbo roundabouts [16].

To evaluate an intersection not only capacity and safety performance need to be considered, but also the pollutant and noise emissions that are generated by cars running on it. In a suburban environment different layouts have proved to be source of different outputs regarding emissions [17]. These differences can be linked to different driving behaviors depending on the type of roundabout.

Despite the existence of clear guidelines on how to design and build roundabouts, the actual implementation can suffer variations from the theoretical design due to other requirements, such as space limitations, higher flows or the need to place parking slots inside, that need to be met [18]. These considerations lead to the necessity of analyzing every roundabout on its specific context to reach significant conclusions. On top of that, the different mix of vehicles (cars, trucks, motorcycles) that use the roundabouts leads to different impacts of roundabouts [19].

To fully evaluate the impacts and advantages of roundabouts, it is crucial to understand how drivers approach them and interact with each other within the intersections. Several studies have demonstrated that drivers' behavior is influenced by the geometrical characteristics and lane demarcation in two-lane roundabouts [20]. In addition, it is relevant to analyze the speed and acceleration upstream and downstream of the roundabouts [21]. These studies have shown that roundabouts promote speed stabilization. However, in the case of two-lane roundabouts, the homogenization of driver behavior in terms of trajectory and speed has been questioned due to the greater freedom of movement that drivers have [22].

Another important consideration is the discomfort experienced by passengers due to lateral acceleration, particularly at the entry and exit of roundabouts, which depends on the required trajectory by the driver [22]. Understanding the impact of lateral acceleration on passenger discomfort is crucial in designing safer and more comfortable roundabouts for all users as well as to design the behavior of automated vehicles when approaching this kind of intersections.

In conclusion, roundabouts are a type of intersection that can offer various benefits over traditional signalized intersections, such as increased safety and reduced congestion. However, the specific design and layout of a roundabout can greatly impact its performance, and factors such as pedestrian and cyclist safety, emissions, and driver behavior must also be considered. Different types of roundabouts have their own characteristics and requirements depending on the capacity of the road. Ultimately, the implementation of roundabouts must be tailored to the specific context and needs of each intersection. By carefully considering all factors and using appropriate design principles, roundabouts can continue to be a valuable tool in improving the efficiency and safety of urban transportation systems.

1.2. Cooperative Connected and Automated Vehicles

One of the main factors of road accidents is the human factor, meaning that having a human driver who is subjected to distractions and does not respond perfectly to all the situations increases the injuries and fatalities that occur. Hence, removing this factor from the driving equation could theoretically lead to more predictable driving scenarios, improving the safety and efficiency of the road network. In the last years, some advances have been made on the scope of automated and self-driving vehicles, although the technical and regulatory complexity of these systems render the date of full adoption of the technology still distant in time. In this section both connected and automated vehicles are considered and the focus is placed on the impact of this technologies in urban areas and specially at intersections.

Connected vehicles are considered those able to connect to external networks, through inbuilt technologies or extra devices added to the vehicle. This connectivity can be between vehicles, Vehicle to Vehicle (V2V), or between the vehicles and the infrastructure (V2I). These technologies among others take part in Inteligent Transportation Systems (ITS) with predicted impacts on traffic signal control, traffic and incident management, vehicle control technologies and shared-use mobility among others [23]. This is expected to reduce energy consumption while improving both safety and efficiency.

According to the taxonomy provided by the Society of Automotive Engineer (SAE) [24], 6 discrete and mutually exclusive levels of car automation can be distinguished, as shown in figure 1.5.

According to this taxonomy one of the important parameters to define a CAV is knowing its Operation Design Domain, meaning the conditions in which an automated vehicle can safely operate, to asses its level of automation and to guarantee safety operation of the vehicle, the results have been seen to vary according to the definition of safety boundaries [25].

One of the possible uses of connected vehicles is the automated control of intersections. Using V2I (vehicle to Infrastructure) communication allows to get rid of traffic signaling at intersection (if all the vehicles are automated). In this case the vehicle waits for the commands of a centralized system that determines the best order to enter the intersection providing space-time path reservations to each vehicle through time buffer when entering conflict point. A sensitivity analysis revealed the importance of this time buffer in the overall performance of the system [26]. One of the negative aspects of intersections is an increase of fuel consumption and emissions due to vehicle idling and excessive accelerations. However, this reduction of fuel consumption and emissions could be conflicting with the minimization of travel time. This conflicting objectives have been addressed in the literature using V2I and V2V communication to implement eco-driving strategies [27].

Regarding the behavior of automated vehicles one of the main areas of study is the trajectory planning, some theoretical approaches have been proposed and simulated. For

SAE level	Name	Narrative Definition	Execution of Steering and Acceleration/ Deceleration	*Monitoring* of Driving Environment	Fallback Performance of *Dynamic Driving Task*	System Capability *(Driving Modes)*
***Human driver* monitors the driving environment**						
0	No Automation	the full-time performance by the *human driver* of all aspects of the *dynamic driving task*, even when enhanced by warning or intervention systems	Human driver	Human driver	Human driver	n/a
1	Driver Assistance	the *driving mode*-specific execution by a driver assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	Human driver and system	Human driver	Human driver	Some driving modes
2	Partial Automation	the *driving mode*-specific execution by one or more driver assistance systems of both steering and acceleration/ deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	**System**	Human driver	Human driver	Some driving modes
***Automated driving system* ("system") monitors the driving environment**						
3	Conditional Automation	the *driving mode*-specific performance by an *automated driving system* of all aspects of the dynamic driving task with the expectation that the *human driver* will respond appropriately to a *request to intervene*	System	**System**	Human driver	Some driving modes
4	High Automation	the *driving mode*-specific performance by an automated driving system of all aspects of the *dynamic driving task*, even if a *human driver* does not respond appropriately to a *request to intervene*	System	System	**System**	Some driving modes
5	Full Automation	the full-time performance by an *automated driving system* of all aspects of the *dynamic driving task* under all roadway and environmental conditions that can be managed by a *human driver*	System	System	System	**All driving modes**

Figure 1.5: Automated vehicles taxonomy according to SAE [24]

instance, in the work done by Solea and Nunes [28] the planning of the trajectory is constrained by the desired destination and the acceleration and jerk limits proposed in the ISO 2631-1 standards. Some other works are more specifically designed for specific situations such as roundabouts [29]. In this case, the proposed algorithm differentiates between entry, circulatory roadway, and exit. However, this algorithm neglects the presence of other vehicles. On the plus side, the algorithm was tested through a real implementation on a vehicle. The decision making of CAVs could also be governed by game-theoretic approaches and in this case user preference and global or individual optimization focus plays a significant role on the performance [30].

The possible impacts of the introduction of CAVs in the existing networks have been analyzed in the literature looking at possible different outcomes and through different methods. The general consensus is that a significant penetration rate of CAVs would improve safety and efficiency [31]. However, the required penetration threshold for this improvements to materialize is disputed and ranges from 20% to 70%.

According to the English Department for Transport [1] the different levels of automation and user choice aspects need to be considered to understand the impact on traffic flow, network performance and road capacity of the introduction of CAVs. The user choices influence the behavior of CAVs both when the user is in the automated vehicle and when the user is driving in front of one of them, due to the desired rear gap.

The same study also highlights that different environments lead to different effects of the introduction of CAVs. Urban environment could see a greater improvement in delay and journey time even for penetration rates as low as 25%, even for low-tech driver assistance technologies, meanwhile this threshold is higher for highway scenarios. The analyzed outputs in this case are time delay and journey time.

According to the research, CAVs are said to affect different aspects of driving: free driving, vehicle following (time headway, acceleration and deceleration rates), lane changing, merging and joining (behavior at junction and gap acceptance) and planning and decision making. On top of this, a balance between comfort, safety and capacity needs to be found since these objectives are often not aligned. Also it is important to distinguish between the user-optimal and the network-optimal solutions, because they could also be conflicting.

Simulating the introduction of CAVs in a theoretical and real network [32] allowed the authors to indicate that the introduction of CAVs would lead to a quasi-linear increase of maximum flow. However, the maximum flow increase is said to be lower than that theoretically expected. Analyzing the fundamental diagram for a roundabout scenario [33] increasing the presence of CAVs in 20% steps led to a increase of critical density and hence maximum flow for different formulations of the fundamental diagram. However, in this case the improvement shows a saturation after 80% of CAVs presence.

In conclusion, CAVs are expected to have a positive impact in both fluency and safety of traffic, although it is not well established the minimum percentage of these vehicle to have a sensible impact. It is also discussed how this technology will manage complex situations, such as roundabouts, and how the perception of the passengers would be. Also important is to consider the perception of the surrounding non-automated drivers. To assess this impacts different solutions are proposed and implemented in the literature, being traffic simulation one of the preferred alternatives due to its cost-effectiveness.

1.3. Traffic simulation and microsimulation

The democratization of computational power during the last decades has given access to a significant number of tools that allow performing complex calculations in short times allowing to develop representations of reality with ever-increasing detail. This can also be seen in the traffic sector, where simulation tools help assess different planning scenarios, study complex behaviors or the potential impact of a given technology in a cost-effective way. Traffic simulations could be categorized according to its focus [34] in:

- Macroscopic: average values of traffic variables are analyzed.
- Microscopic: each vehicle is modeled individually.
- Mesoscopic: a combination of the previous approaches.
- Submicroscopic: specific functions of the vehicle are explicitly simulated.

The two basic simulation variables are time and space. The models are usually time-discrete for computational reasons, meanwhile both space-discrete and space-continuous models can be found. There are both commercial and open-source solutions available. Space-discrete models are usually simpler and more suited to analysis on performance of the network rather than individual behaviors, for example comparing phase diagrams of signalized intersections with those of roundabouts using a cellular automaton model [35]. Microsimulation tools, however, tend to use space-continuous models.

In the scope of this project, the most relevant simulations are microscopic simulations since it is required to analyze each vehicle individually. Regarding microsimulations the results strongly depend on how the behavior of drivers is modelled, mainly the behavior related to car following and lane changing. It is also important to consider limitations to reproduce incidents, errors and accidents on the network

Car Following Models (CFM) have been studied for many years and different approaches have been explored. The most used car following models and their working principles are explained in the next section. After, an overview of the main simulation software and the implementation of the CFM is presented.

1.3.1. Car Following Models (CFM)

Optimal Velocity Model

The basic explored idea behind this car following model is that drivers desired acceleration is a function of its own speed, the speed differential and the gap between them corrected by a calibration parameter as can bee seen in its formulation (1.1).

$$\begin{aligned} \ddot{x}_n(t) =& aF(..., \Delta x_{n+1}, \Delta x_n, \Delta x_{n-1}, ... \\ & ..., \dot{x}_{n+1}, \dot{x}_n, \dot{x}_{n-1}, ... \\ & ..., x_{n+1}, x_n, x_{n-1}, ...) \end{aligned} \tag{1.1}$$

Where x_n is the position of the ego vehicle and $n+1$ and $n-1$ represent the rest of the vehicles in front and behind the ego influencing its behaviour. The coefficient a is the calibration parameter.

Taking on this idea, considering that the driver reacts to the position, velocity and headway of the surrounding vehicles with a given sensitivity [36] and considering that the driver sets its desired speed by looking at the headway with respect to the leader the equation of motion of the Optimal Velicity Model can be expressed as follows.

$$\ddot{x}_n = a[V(\Delta x_n) - \dot{x}_n] \tag{1.2}$$

$$\Delta x_n = \dot{x}_{n+1} - \dot{x}_n \tag{1.3}$$

The function V needs to be monotonically increasing and has an upper bound that could be understood as the maximum speed [36]. The model is said to account for the effect of time lag through the second order differential equations based on the equation of motion [37].

A similar reasoning could be made to introduce a model in which headway and speed interchange its functions, this model is known as optimal headway model. Following this approach, the function that defines the desired headway uses the speed difference between leader and follower as the independent variable.

Gipps Model

The goal of this car following model was to mimic the behaviour of real traffic providing a model in which the parameter corresponded to obvious driver characteristics [38]. The model considers that the driver can either drive freely, limited only by its desired speed, acceleration and braking and only influenced by its immediate predecessor. The basic formulation of the model is shown below.

$$\begin{aligned} v_n(t+\tau) = & min\Bigg[v_n(t) + 2.5a_n\tau\Big(1 - \frac{v_n(t)}{V_n}\Big)\sqrt{0.025 + \frac{v_n(t)}{V_n}}, \\ & b_n\tau + \sqrt{b_n^2\tau^2 - b_n\Big[2[x_{n-1}(t) - s_{n-1} - x_n(t)] - v_n(t)\tau - \frac{v_{n-1}(t)^2}{\hat{b}}\Big]}\Bigg] \end{aligned} \tag{1.4}$$

where

- a_n: maximum desired acceleration.
- b_n: maximum desired deceleration.
- s_n: size of the vehicle composed by the actual length plus a gap.
- V_n: desired speed.
- $x_n(t)$: position of the front bumper at time t.
- $v_n(t)$: speed of the vehicle at time t.
- τ: reaction time.
- $\hat{b}$: follower's estimation of leader's maximum deceleration.

Using two expressions the model provides two driving situations, unconstrained when the leader tries to travel at its desired speed and following mode where the constrained is imposed by safety.

The first part of equation (1.4) represents that the driver will not exceed its desired speed and it will moderate the acceleration when getting closer to the desired speed. The second part represents the braking constraint, if the leader (vehicle n-1) breaks with a given deceleration the follower reacts with a delay due to the reaction time getting closer to the first vehicle, the safety constraint of avoiding collisions leads to the conclusion that the speed at the time in which the deceleration begins needs to be constrained.

The free driving equation of the original model can be expressed in a more general way providing the possibility of calibrating new parameters (alfa, beta, gamma) that define the relationship between speed and acceleration during free driving, equation (1.5). This has been proven to enhance the performance of the model when calibrating the model based on trajectory data [39].

$$v_n(t+\tau) = v_n(t) + \alpha a_n\tau\Big(1 - \frac{v_n(t)}{V_n}\Big)\Big(\beta + \frac{v_n(t)}{V_n}\Big)^\gamma \tag{1.5}$$

Krauss Model

This car following model [34] also considers two states of driving: free motion and interaction with another vehicle. In free motion the main characteristic is that the speed is limited by a maximum velocity. In the second state the model objective is to guarantee that no collisions occur, hence the speed is limited by the maximum speed at which this can happen. Another assumption of the model is that the maximum acceleration and deceleration are bounded. All these constraints can be expressed as follows.

$$v \leq v_{max} v \leq v_{safe} \tag{1.6}$$

$$-b \leq \dot{v} \leq a; \quad where \quad a, b > 0 \tag{1.7}$$

The implementation of this model relies on the calculation of the gap between the leader and the follower, the desired gap and the desired relaxation time.

$$v_f(t + \Delta t) \leq v_l(t) + \frac{g(t) - g_{des}(t)}{\tau_{des}(t)} \tag{1.8}$$

$$where: \quad g_{des} = v_l \tau; \quad \tau_{des} = \tau_b + \tau; \quad \tau_b = \bar{v}/b(\bar{v})$$

To sum up, the model for discrete time steps can be represented by the following equations.

$$v_{safe}(t) = v_l + \frac{g(t) - g_{des}(t)}{\tau_b + \tau} \tag{1.9}$$

$$v_{es}(t) = min[v_{max}, v(t) + a(v)\Delta t, v_{safe}(t)] \tag{1.10}$$

$$v(t + \Delta t) = max[0, v_{des}(t) - \eta] \tag{1.11}$$

$$x(t + \Delta t) = x(t) + v\Delta t \tag{1.12}$$

In equation (1.11) η represents a random perturbation. According to Krauss [34] a sufficiently small time step is required to guarantee that it is smaller than τ and the desired gap is smaller than the speed multiplied by the time step.

Wiedemann's Model

Wiedemann proposed a psycho-physical model [40] in which four driving regimes - free-flowing, approaching, following and emergency – are defined and the driver enters in one of the regimes when certain thresholds regarding speed difference and gap with respect to its predecessor are reached. These areas can be represented as in Figure 1.6.

The different thresholds represented in figure 1 are:

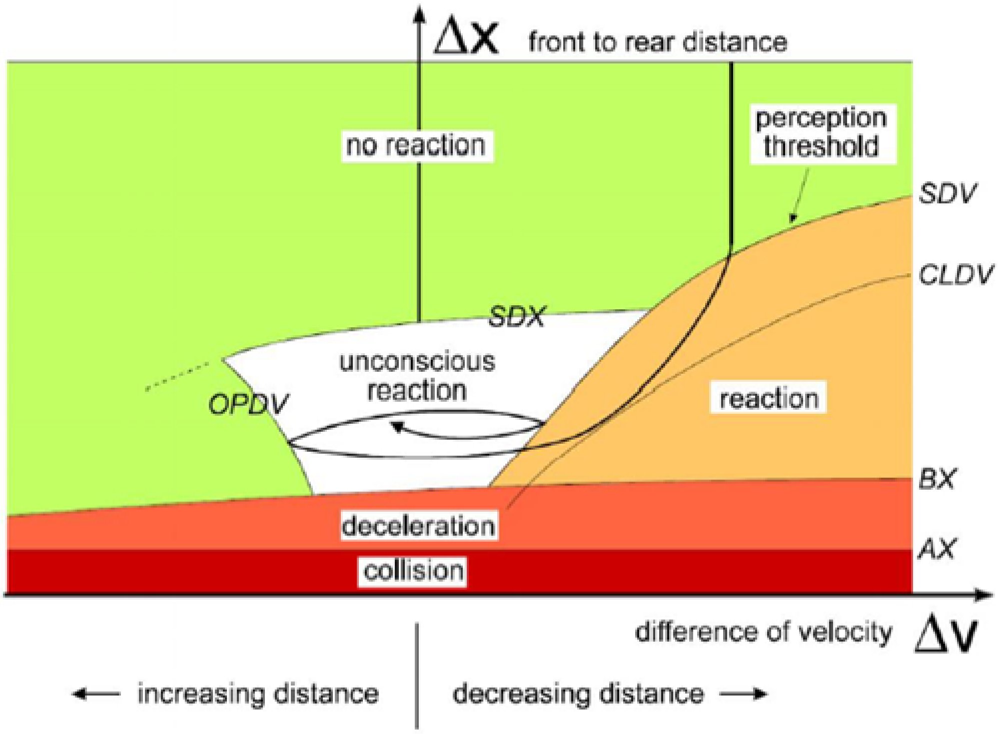


Figure 1.6: Wiedemann's car following model [40].

- AX: minimum standing distance
- BX: minimum following distance
- CLDV: points where drivers perceive their speed is higher than the leader speed
- SDV: points where drivers perceive they are approaching a slower vehicle
- OPDV: points where drivers perceive their speed is higher than the leader speed
- SDX: limit of the car-following model

The original model of Wiedemann was published in 1974 (W74) and it was latter reviewed in 1999 (W99), thus two models of the Wiedemann approach are available and, although they share the same concept, they have different parameters. The original model (W74) is formulated as follows .

$$v_n(t+\Delta t) = min\left[3.6\left(\frac{s_n(t)-s_j}{BX}\right)^2; u_f \right.$$
$$\left. 3.6\left(\frac{s_n(t)-s_j}{BX \cdot EX}\right)^2; u_f\right] \tag{1.13}$$

where BX and EX are computed as

$$BX = BX_{add} + BX_{mult} \cdot RND1_n \tag{1.14}$$

$$EX = EX_{add} + EX_{mult} \cdot (NRND - RND2_n) \tag{1.15}$$

Where BX_{add}, BX_{mult}, EX_{add}, EX_{mult} are the calibration parameters.

The expected threshold values are calculated as follows [41]:

$$E(AX) = s_j + 0.5 \approx s_j \tag{1.16}$$

$$E(ABX) = E(AX) + E(BX)\sqrt{u} = s_j + E(BX)\sqrt{u}, \quad u \leq u_{desired} \tag{1.17}$$

$$E(SDX) = s_j + E(BX) + E(EX)\sqrt{u}, \quad u \leq u_{desired} \tag{1.18}$$

The model W99 is governed by the following equations [41] .

$$v_n(t+\Delta t) = min\left[v_n(t) + 3.6 \cdot \left(CC8 + \frac{CC8 - CC9}{80} v_n(t)\right)\Delta t; \quad v_f \right. \\ \left. 3.6 \cdot \frac{s_n(t) - CC0 - L_{n-1}}{v_n(t)}; \quad v_f\right] \tag{1.19}$$

$$AX = L + CC0 \tag{1.20}$$

$$BX = AX + CC1 \cdot v \tag{1.21}$$

$$SDX = BX + CC2 \tag{1.22}$$

$$(SDV)_i = -\frac{\Delta x - (SDX)_i}{CC3} - CC4 \tag{1.23}$$

$$CLDV = \frac{CC6}{17000}(\Delta x - L)^2 - CC4 \tag{1.24}$$

$$OPDV = -\frac{CC6}{17000}(\Delta x - L)^2 - \delta \cdot CC5 \tag{1.25}$$

Where v is the speed of the subject vehicle if it is slower than leader or the leader speed with some random errors. Δx is the headway measured between front bumpers. δ is a dummy variable equal to 1 when the ego vehicle's speed is greater than CC5.

The parameters of W99 are usually defined as follows [40]:

- CC0: standstill distance
- CC1: headway time

- CC2: following variation
- CC3: entering following threshold
- CC4: negative following threshold
- CC5: positive following threshold
- CC6: oscillation speed dependency
- CC7: acceleration oscillation
- CC8: standstill acceleration
- CC9: acceleration at 80 km/h

Some of this parameter can be easily linked to actual measurable parameters and related to other CFM, meanwhile others are more difficult to interpret.

Intelligent Driver Model (IDM)

The IDM is a model conceived by Martin Treibet et al. [42] that tries to explain the acceleration as a sum of two components, one depending on the relationship between the actual speed and the desired speed, and the second one depending on the actual gap between follower and leader and the desired gap. This can be mathematically expressed in formulas (1.26) and (1.27).

$$\dot{v}_\alpha = a^{(\alpha)}\left[1 - \left(\frac{v_\alpha}{v_0^{(\alpha)}}\right)^\delta - \left(\frac{s^*(v_\alpha, \Delta v_\alpha)}{s_\alpha}\right)^2\right] \tag{1.26}$$

$$s^*(v, \Delta v) = s_0^{(\alpha)} + T^{(\alpha)}v + \frac{v\Delta v}{2\sqrt{a^{(\alpha)}b^{(\alpha)}}} \tag{1.27}$$

Where S^* is the desired gap by the follower vehicle and every vehicle can have a different set of parameters. The basic notation is explained in figure 1.7.

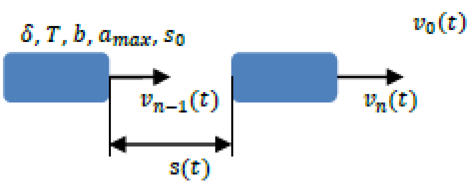


Figure 1.7: IDM notation [43]

Equations (1.26) and (1.27) are described in a continuous way. However, for simulation purposes the time discrete approach is used. One of the advantages of this model is that all the parameters are intuitive and easily measurable [39]. The meaning of the parameters is explain as follows:

- δ: eagerness to reach the desired speed
- T: time headway
- s_0: minimum standing gap
- a: maximum comfortable acceleration
- b: maximum desired deceleration
- v_0: desired speed, could be interpreted as the maximum legal speed

This model has been shown to perform well to reproduce traffic flow characteristics such as the ones obtained via the fundamental diagram [42]. However, the model could be further improved and some modifications and additions have been proposed over the years.

The Improved Intelligent Driver Model (IIDM) [44] allows to obtain more realistic gaps in homogeneous traffic conditions, this is obtained calculating a new acceleration used as a reference when the gap is above the desired gap, thus avoiding large gaps induced by the term of the gap in the original acceleration formulation.

The model can be further improved including finite reaction times, estimation errors, temporal and spatial anticipation. These characteristics are included in what is known as Human Driver Model (HDM) [45] and it is shown that can be paired with the IDM [43].

In the IDM the time-headway could be interpreted as the reaction time, however, this two could be different since one depends on driving style and the other is a psychological parameter [45]. The real value should be also different from the simulation step time. To represent the estimation errors of the driver a Wiener process to introduce stochasticity is proposed. The driver is also able to anticipate the velocities, own velocity, and leader velocity, as well as the gap between the two vehicles. Introducing this reduces the probability of accidents and instability produced by the addition of the reaction time as time action points [43].

The IDM was originally conceived as single lane mode, which led to high deceleration when applied to a multiple lane scenario, the Enhanced Intelligent Driver Model (EIDM) tries to tackle this problem [46]. This model provides a more relaxed and realistic change in the acceleration specially when a lane change has been performed [43]. The formulation of the model is shown below.

$$a_{cah}(t) = \begin{cases} \frac{v_{n-1}^2 \tilde{a}_n}{v_{n-1}^2 - 2s(t)\tilde{a}_n} & v_n(v_{n-1} - v_n) \leq -2s(t)\tilde{a}_n \\ \tilde{a}_n - \frac{(v_{n-1}-v_n)^2 \Theta}{2s(t)} & \text{otherwise.} \end{cases} \tag{1.28}$$

$$\Theta = \begin{cases} 0 & (v_{n-1} - v_n) < 0 \\ 1 & (v_{n-1} - v_n) \geq 0 \end{cases} \tag{1.29}$$

$$\tilde{a}_n = min(a_n(t), a_{max} \tag{1.30}$$

This model is not independent from the IDM model and needs the prior calculation of IDM to work, to be coupled with the IDM the parameter c_{ACC} is added to weigh the impact of the IDM and this modification in the actual acceleration.

$$a_{ACC} = \begin{cases} a_{IDM} & a_{IDM} \geq a_{ACC} \\ (1 - c_{acc})a_{IDM} + c_{ACC}\big[a_{CAH} + b \cdot tanh(\frac{a_{IDM} - a_{ACC}}{b})\big] & otherwise \end{cases} \tag{1.31}$$

Further improvements to the original IDM that are implemented in SUMO under the name of Extended IDM (EIDM) [43] include:

- Providing smooth deceleration to 0 when estimation and perception errors are considered.
- Modifying the desired speed with anticipation to the speed limit change.
- Instant reaction of the driver when the required change of the acceleration is greater than a certain limit to simulate
- Limit jerk in drive off scenarios to obtain more realistic results modifying the acceleration after stopping and proposing a hyperbolic tangent function.
- Limit jerk in scenarios with sudden changes of desired gap and actual gap limiting the ratio of change to a specific parameter.

  Although these modifications to the IDM improve the realism of the outcomes, the original IDM is still widely used due to its simplicity and straight forward formulation.

Other models

Other models can be found in the literature. Some rely on different approaches such as fuzzy logic instead of deterministic calculation of the acceleration to better represent the estimation capabilities and randomness of the drivers [47]. Probabilistic approaches can also be found [48]. Others are more specific and try to find innovative approaches to model how drivers interact, for example using microeconomic utility-based models [49] or based on game theory. Most of the models are designed to represent the conditions of traffic in developed countries, however some specific models try to represent the conditions of developing countries, for instance, heterogeneous traffic [19]. This traffic condition is characterized by numerous interactions among vehicles with different trajectories and very different characteristics in lane-less traffic, basically large presence of small vehicles, two-wheelers and no lane discipline [50].

To model the behavior when implementing autonomous vehicles (AV) or connected autonomous vehicles (CAV) the IDM could be used with the adequate parameter configuration [46]. However, specific models for these situations can be found in the literature such as the work done by Milanés et al. [51].

Although, the main CFM explained are generally accepted, some criticism can be found in the literature regarding the lack of realism to represent traffic flow breakdown at highway bottlenecks [52].

1.3.2. Lane Change Models

As it has been shown, the longitudinal behavior of a vehicle driven by a human can be represented through mathematical models. However, to completely represent the movement of cars in traffic scenarios, the lateral behavior, mainly lane changing, must also be represented to obtain realistic results. This is because lane changing can affect performance outputs of the network such as capacity due to shock-waves in heavy traffic conditions [53].

To reproduce the decision making process of human beings while driving, the literature divides these decisions into strategic, tactical and operational. Lane changing falls in the last two categories. According to Moridpour et al. [53], models based on search algorithms can be found. These models try to make the decision considering the final position of the driver's vehicle and other surrounding vehicles. The other option is to base the decision on current traffic characteristics, this second approach is the most common and different strategies have been implemented in the literature.

The first defined strategy is that of having a deterministic model. The model tries to reproduce a clear relationship between the independent variables and the output of whether to perform or not the lane change. It is crucial to define the explanatory variables to obtain a good performance of the lane changing model. The main advantage is that it is simple to model and has a low number of variables [53].

An example of this strategy is the Gipps' model [54]. According to Gipps, stimulus response CFM are easier because any obstacle can be analyzed as a leader vehicle, while Lane Changing models are more problematic because they depend on different and sometimes conflicting objectives. The three basic parameters influencing the decision of making the lane change are possibility, necessity, and desirability. These parameters are said to be affected by the driver assessment of the situation where the parameters to take into account are risk, required turns and proximity, if traffic is likely to represent a limit to the desired speed, the urgency of the lane changing as well as legal requirements.

The hierarchy between conflicting objectives or reasons needs to be clear and might vary from driver to driver or from region to region [54]. This hierarchy is explained in [53] in a simplified way: if the driver is far away from the desired exit, maintaining speed is the main objective; closer to the exit, the advantage of changing lane is ignored if moving away from desired exit; and if exit is immediate speed desire is irrelevant. The physical parameters of the defined mathematical model are the same explained for the Gipp's CFM: safe speed and brake, front gap and estimation of leader's braking. Hidas developed a similar model for a specific simulation model SITRAS [55]. This model includes a feasibility constrained linked to the acceleration change of the subject vehicle and the new follower vehicle. It also introduces the concept of courtesy which could be important to simulate merging, considering it a special case of lane changing.

The second strategy is the probabilistic approach. The model gives a probability to performing the lane change according to the entry variables that are fed into it. According to Moridpour et al. [53], this approach leads to a higher computational cost when using simulation since all the probabilities need to be calculated in every time step.

Toledo et al. [48] developed a model of integrated driving behavior, in which the analysis of lane change to either side or no lane change and then decision to accelerate once the target lane is set are considered. The variables that this model accounts for are speeds and spacing of surrounding vehicles, not only those in the current lane, trip plan variables, network knowledge and experience, as well as driving style and capabilities. All these parameters are used to calculate the utility of each lane and a logit model is used to obtain the probabilities of selecting each lane.

The third strategy is that of using Fuzzy logics [56]. Although random terms can be included in all the previously defined models to try to simulate the variability and randomness of drivers, this representation might not be good enough [53]. To better represent the driver decision making process fuzzy logic has been proposed in the literature because it allows to define specific uncertainty in the model. The better performance and capability to represent humans perception comes at the cost of difficult and complex to abstract fuzzy rules and a more cumbersome validation process [53].

Wu et al. developed a fuzzy logic simulation model (FLOWSIM) to reproduce driving in motorways [47]. Regarding Lane changing two different models are described by Wu et al.: changing lane offside or nearside. Changing nearside relies on pressure from rear (time headway of the following vehicle) and gap satisfaction (period of time in which is possible to stay in the lane without reducing speed) as basic parameters. Changing offside relies on overtaking benefit (speed gain) and opportunity (safety and comfort of the lane change time headway to the nearest approaching vehicle to the rear).

Some of the remarks found in the review performed by Moridpour et al. [53] about limitations of the lane changing models are: the lack of models for heavy vehicle lane changing decision, the use of macroscopic traffic measures for the validation of the models and the assumption that the lane change is an instantaneous maneuver.

1.3.3. Integration of CFM in microsimulation platforms

Different attempts to build microsimulation software with research purposes have been undertaken in the last decades to test specific models, such as the Intelligent Driver Model (IDM) [57], or to test vehicle features like Adaptative Cruise Control (ACC) in the case of PELOPS [58], a traffic microsimulation program created by Institut für Kraftfarhwesen Aachen and BMW to simulate traffic flow on motorways. The most relevant microsimulation platforms according to the literature [31] are shown in Table 1.2 indicating its developer, the included Car Following Models, Lane Change Models, the original release date and a stable release version.

PLATFORM	DEVELOPER	CFM	LCM	RELEASE
PARAMICS	Quadstone (University of Edinburgh)	Fritzsche	Gap-acceptance	1990 Paramics Discovery (2020)
VISSIM	PTV (Karlsruhe University)	Wiedemann 74, Wiedemann 99	Sparmann model	1992 PTV VISSIM 2021(2020)
CORSIM	McTrans Center, University of Florida	Pitt	Intralink LC	1998 TSIS-CORSIM 6.3 (2012)
AIMSUM	Siemens Mobility (TSS – Transport Simulation)	Gipps	Rule-based Gipps	1997 AIMSUM Next 20 (2020)
SUMO	German Aerospace Center	Krauss, Daniel, IDM, IDMM, EIDM, Wiedemann, ACC, CACC	DK2008, LC2013, SL2015	2001 SUMO 1.9.2 (2021)

Table 1.2: Most used microsimulation platforms [31]

VISSIM, Paramics, CORSIM, SUMO and AIMSUM are, according to publication share in the top 25 publishers, the most used microsimulation platforms. From this list, SUMO is the only open-source software.

Specific studies have been done trying to analyze the performance of different software, for instance, comparing transims, SUMO and VISSIM [59]. Transims is based on cellular automata, where a given cell (each of the areas in which the network is divided) can be empty or occupied by a vehicle and its state affects the decision making of nearby vehicles. Maciejewski tried to reproduce a real urban network an analyze the performance. It is stated that VISSIM allows a more realistic representation of the network, specially of intersections, and possible maneuvers and the highest realism level of vehicle dynamics. Even though some different results are obtained in the quantitative analyses, qualitatively all gave the same results regarding problematic areas of the network [59]. However, it is important to note that this study was performed considering older versions of the software, and these are in constant development. Giuffrè et al. [13] used VISSIM and AIMSUM to obtain trajectories to assess safety in roundabouts analyzing conflict points. The results pointed that it is possible to obtain comparable results regarding safety through the use of appropriate filters.

Other studies have used simulation to compare the performance of different CFMs. The most used models are Wiedemann and IDM. Goncu et al. using SUMO showed that W99 and IDM, when set for the same driver behaviour, give different results regarding travel time and total throughput, this difference is said to be higher than the difference obtained modifying the parameter for a given model [60]. Also, it is stated that IDM reproduces better the behaviour of real drivers for specific situations, this is due to its different approach to driver modelling [25, 60].

Sun et al. [56], added Gipps to the comparison and tried to reproduce different driver behaviors according to the aggressiveness of the driver, they found that Gipps model was the least performing regarding error with real data. Meanwhile, both Wiedemann and IDM were able to reproduce the two driver styles, but they found that some anomalies regarding the value of acceleration for W99. IDM has also been proven to be better to reproduce heterogeneous behaviors [61]. This last study done by Zang et al. introduces also the General Motors CFM, that was found to be under-performing when compared to W99 and IDM, and divides the driver behaviour in neutral, aggressive and timid. They also included an additional term to represent heterogeneity among same type of drivers, they found that for their analyzed data the best approach was to estimate this new parameter through and $\alpha - stable$ distribution.

The importance of the ability of a given CFM to reproduce different driving styles and heterogeneity among driver is further explored in the literature. Focusing on the IDM the possibility of using a distribution instead of a fixed parameter has been proposed as a way to represent driver variability [27, 62], the addition of noise to the acceleration signal has also been analyzed given different results [27].

Many of the studies carried on in the last years regard the implementation of Automated Vehicles (AV) in the current network. Some of the studies to assess the impact of different CAVs penetration rates focus on modelling them setting different CFM parameters. For example, using SUMO and Krauss parameters [32] or IDM [33], or using VISSIM and W99 [1, 63].

In some cases, modifying the parameters of the inbuilt models might not be sufficient to represent the behavior of AV, hence a different approach is needed and cosimulation has been used in the literature as a possible solution. Applying this strategy means that some vehicles are controlled by the inbuilt model meanwhile others are externally controlled, for instance through a different software or a driving simulator. In this case the preferred software is SUMO due to its open-source nature [31]. This also gives the possibility to simulate Connected Automated Vehicles (CAVs) simulating the communication via external software or hardware.

Simulation is also useful to test specific algorithms, for instance a game-theory approach to decision making of CAVs when they approach a roundabout [30]. Hang et al. proposed differentiated driving strategies based on giving different weights to the payoff function parameters representing safety, comfort, and efficiency to introduce the impact of different user preferences. They also considered the impacts of respecting personal preference on the overall system efficiency including some remarks on the relevant role of policy makers and OEMs.

Analyzing the impact of CAVs specifically in roundabouts has also been studied through the use of microsimulations [1, 33, 63]. The impacts have been studied on different outcomes, such as maximum flow or average speed or more roundabout-specific parameters, like queue length and time stopped. Analyzing safety looking at possible conflict points using Surrogate Safety Assessment Model software leads to the conclusion that rear end crashes could increase for certain levels of CAVs penetration [63]. That study concludes that a positive impact on queue length, average speed and time stopped can be seen, although the magnitude of the impact depends on the specific layout of the roundabout and it is greater for higher CAVs penetration rates. It is also important to recognize that when simulating roundabouts different parameters need to be taken care of such us lateral movement and lateral gap acceptance, specially for multiple lane roundabouts [1].

1.4. Simulation of Urban Mobility (SUMO)

As it has been discussed, different simulation software are available to perform analysis on traffic networks. However, the open-source nature of SUMO [3] renders it as the best alternative for the scope of this project. SUMO allows to represent road traffic (both private and public), bicycle traffic and pedestrian traffic. In this section, the software is described explaining the main characteristics that are useful for the project.

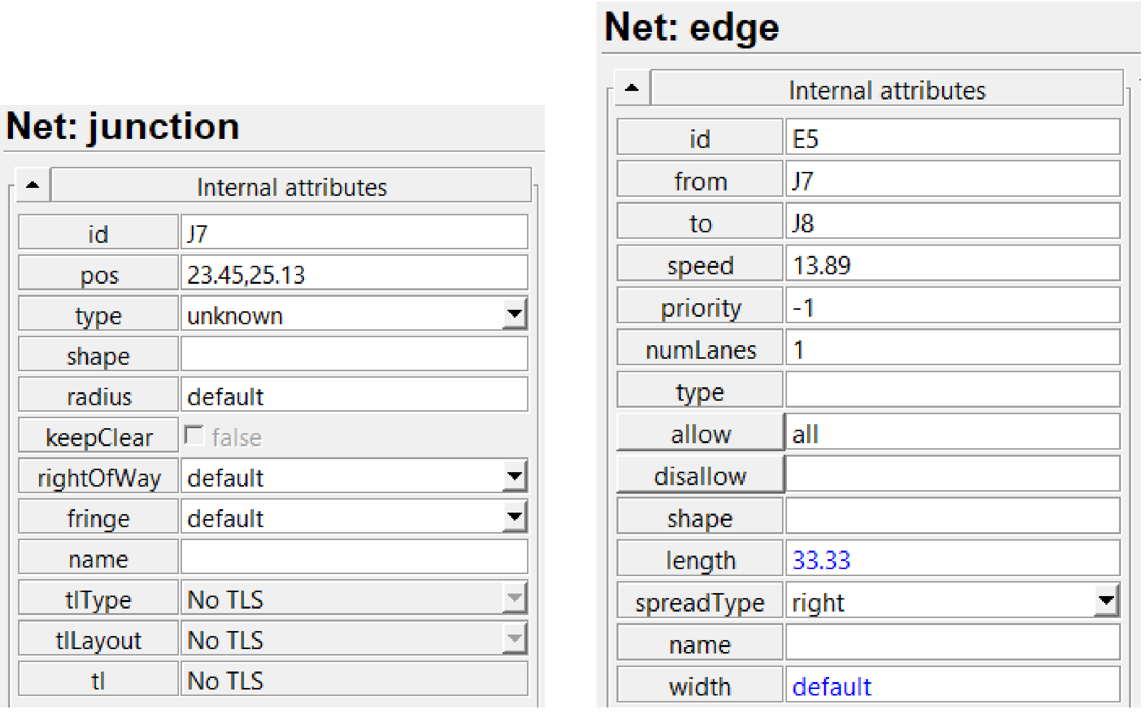
SUMO can be run through a graphical interface or using the command window. It can also be commanded through Python or Matlab codes. To reproduce a scenario in SUMO it is necessary to create at least three different XML files: a network file (.net.xml), a route file (.rou.xml) and configuration file (.sumocfg). All these files allow to set the different options available in the software.

The Network consists on the definition of nodes and edges connecting the nodes. The nodes represent the different junctions in the network in which different roads converge or suffer a change (e.g. reduction of the number of lanes). Edges represent the road or the street and can be formed by one or more lanes with different characteristics (e.g. 4 lanes: road, bike path and two sidewalk lanes).

To generate a network in SUMO it is possible to use different options. NETEDIT is the option provided by SUMO to generate the network and the demand using a visual interface. This is one of the easiest ways to create a network from scratch, using the visual interface the nodes and edges are created selecting their position on the screen and then different configuration options are presented as can be seen in figure 1.8. On top of this, SUMO provides different tools to be run using the command window that allow to generate abstract networks (*netgenerate*), create a network starting from two files defining the edges and nodes (*netconvert*) or import the network from different sources such as OpenStreetMap, VISSIM and VISUM among others.

Once the network is generated using whichever of the mentioned methods an XML file is generated. This file includes the definitions of nodes and edges previously explained as well as the connections among the different lanes and some internal lanes. SUMO generates these lanes inside the junctions to increase the realism of the simulation and it is explained more in detail later.

NETEDIT also allows to define the vehicles and other users of the network through a graphical interface. This is done in the demand module of the software. There are



(a) NETEDIT node parameters definition.

(b) NETEDIT node parameters definition.

Figure 1.8: NETEDIT Graphic interface to configure nodes and edges

different ways to define the desired movement of the vehicles. The first option is to define the initial junction and the final junction, the second possibility is to do the same with edges instead of junctions and the last one is to define a route, selecting the different edges that compose it. In the first two cases SUMO uses Dijkstra's algorithm (a shortest path tree is generated from the source node to all other nodes and then the shortest path to the desired node is chosen) as default to calculate the route because of its simplicity. After defining the desired path it is possible to decide if we want it to be covered by a single vehicle or by a flow of vehicles.

The demand module also allows to define different vehicle types. These vehicle types are then assigned to the defined routes or trips. The vehicle type includes general attributes of the vehicle (e.g. dimensions, capacity, emission class), Lane Change Model, Junction Model and Car Following Model attributes. Figure 1.9 shows the configuration menu of the vehicle type in NETEDIT. The specific attributes are discussed over the next paragraphs.

Also for the demand an XML file is generated containing the information related to the vehicle types and the different defined trips or routes, this allows to modify the settings of the demand without using the graphic interface.

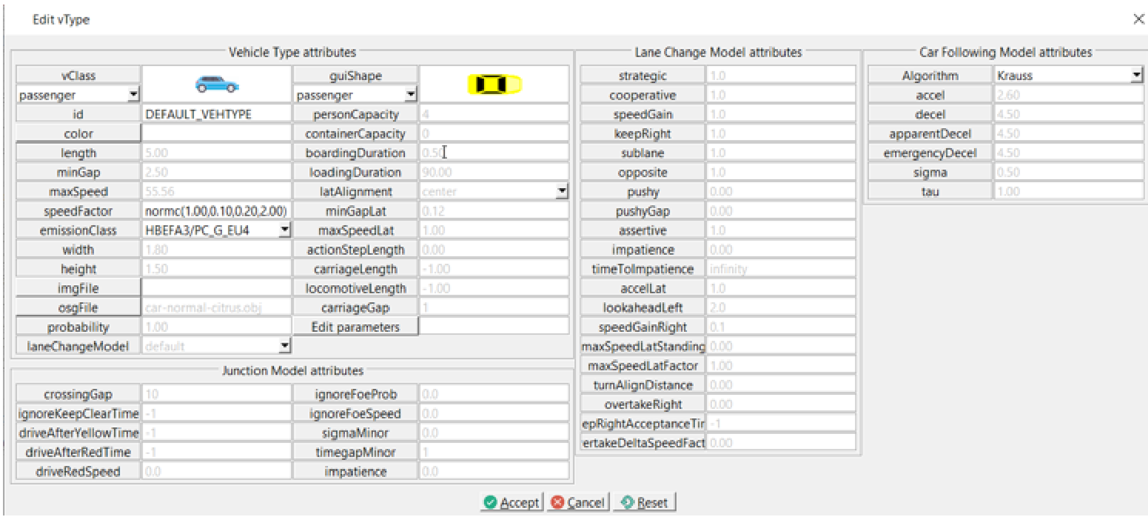


Figure 1.9: NETEDIT Vehicle type configuration menu

After having defined the network and the demand on the network the setup of the simulation is done. The configuration of the simulation is divided into input files, processing parameters and desired outputs. The input files are the netwok and route files and it is possible to include also additional files to represent different elements of the network (e.g. traffic lights, bus stops, loop elements to measure different traffic parameters...).

The processing parameters of the simulation have a significant impact on the results of the simulations. Some of the parameters are dedicated to define what is considered a collision and what the action should be if the simulation detected one. It is also at this point where it is necessary to state if the sublane model is going to be used, the sublane model divides the lane in smaller portions and a vehicle occupies more than one of these sublanes. This model also allows to modify the alignment of the vehicle on the lane and introduces more option for the Lane Change Model that are explained later as well as the possibility to consider the lane change event not instantaneous. The main two parameters are the definition of the integration method and the step length.

The integration methods available in SUMO are the Euler method and the Ballistic method. Treiber and Kanagaraj [64] analyzed these two methods among others and found that the ballistic method offers a superior performance. However, it is important to note that the implementation of some CFM in SUMO has not been tested using the ballistic method and might lead to unexpected results.

The step length is the most significant parameter of the configuration. This parameter impacts significantly the computational cost of the simulation increasing the resolution of the results. It is also important to note that this parameter is used by some CFM as the reaction time of the driver, changing the results of the simulation even when the other CFM parameters are the same.

The configuration process also includes defining which are the desired outputs of the simulation. By default, no output is generated and saved by the simulation other than the time of the simulation, the real time consumed, the number of vehicle and possible warning related to the vehicles (e.g. emergency braking, collision or teleport). However, SUMO allows to generate and save information related to the flow car data (position, speed, acceleration), tip info both aggregated and for specific vehicles (depart/arrival time, waiting time, time loss, emissions), emission using different available classes (CO_2, CO, HC, PM_x, NO_x, fuel consumption) and queue information.

It has already been mentioned that SUMO can be commanded externally using Python, this is thanks to the Traffic Control Interface (TraCI). TraCI gives access to a running simulation allowing to modify the behavior of objects of the simulation and to retrieve information at each step of the simulation.

The next sections describe the Car Following Models, Lane Change Models and Junction Models currently available in SUMO.

1.4.1. Car Following Models

The main concepts of Car Following Models (CFM) have already been explained in section 1.3.1. Although SUMO does not include all the mentioned CFMs, it includes the most commonly used as well as some specific models that are implemented thanks to the open-source nature of SUMO. The most relevant CFM are shown in table 1.3

Main Car Following Models available in SUMO

CFM in SUMO	Description
Krauss	Original Krauss model (see 1.3.1) with some modifications (default SUMO model)
IDM	Intelligent Driver Model as described by Treiber [42]
EIDM	Extended Intelligent Driver Model [43]
Wiedemann	2-Parameters model. Similar to W74 (see 1.3.1)
W99	Wiedemann model with 10 parameters (see 1.3.1)
ACC & CACC	Model developed by Milanés and Shladover [51]

Table 1.3: Main Car Following Models implemented in SUMO [65].

On top of the described models, three other versions of the Krauss model are available in SUMO, another IDM modification and two different CFM as well as one model for rail vehicles. From the models shown in the table the most relevant are: Krauss, which is the default used in the simulations, whose parameters are shown in table 1.4; IDM, with parameters displayed in table 1.5, note that in the SUMO implementation two extra parameters are added to the model, emergencyDecel, which allows to avoid collisions exceeding the comfortable deceleration, and the stepping parameter that is relevant for simulations using a large time step; and W99, with parameters shown in table 1.6.

Krauss Model parameters in SUMO

Attribute	Range (Default Value)	Description
minGap	≥ 0 (2.5 m)	Minimum standing distance (from front bumper to rear bumper)
accel	≥ 0 (2.6 m/s^2)	Maximum desired acceleration
decel	≥ 0 (4.5 m/s^2)	Maximum desired deceleration
emergencyDecel	≥ 0 (9 m/s^2)	Maximum ability to break
sigma	[0,1] (0.5)	Driver imperfection
tau	≥ 0 (1 s	Time headway

Table 1.4: Parameters of the Krauss model in SUMO.

IDM parameters in SUMO

Attribute	Range (Default Value)	Description
minGap (s_0)	≥ 0 (2.5 m	Minimum standing distance (from front bumper to rear bumper)
accel (a)	≥ 0 (2.6 m/s^2)	Maximum desired acceleration
decel (b)	≥ 0 (4.5 m/s^2)	Maximum desired deceleration
emergencyDecel	≥ 0 (9 m/s^2)	Maximum ability to break
delta (δ)	≥ 0 (4)	Acceleration exponent, eagerness to reach desired speed
tau (T)	≥ 0 (1 s	Time headway
Stepping	≥ 0 (0.25 s	Internal step length to compute follow speed

Table 1.5: Parameters of the IDM model in SUMO.

W99 parameters in SUMO

Attribute	Range (Default Value)	Description
minGap (CC0)	≥ 0 (2.5 m	Minimum standing distance (from front bumper to rear bumper)
CC1	≥ 0 (1 s	Spacing time
CC2	≥ 0 (8 m	Following variation
CC3	≤ 0 (-12 m	Threshold for following
CC4	≤ 0 (-0.25 m/s)	Positive following threshold
CC5	≥ 0 (0.35 m/s)	Negative following threshold
CC6	≥ 0 (6 10^4rad/s)	Oscillation of speed dependency
CC7	≥ 0 (0.25 m/s^2)	Oscillation acceleration
CC8	≥ 0 (2 m/s^2)	Standstill acceleration
CC9	≥ 0 (1.5 m/s^2)	Acceleration at 80 km/h

Table 1.6: Parameters of the W99 model in SUMO.

One particular aspect of drivers' behavior that is not implemented into this models is the reaction time. In SUMO, by default, the reaction time is equal to the step length of the simulation. When using very low time steps this might lead to unrealistically low reaction times, in order to decouple these two aspects SUMO has a configurable parameter called *actionStepLength* that allows to modify when the logic of the driver is computed. However, this is not a real reaction time because, even though the driver reacts after the time set in the *actionStepLength* parameter, it reacts to the conditions seen in the previous time step.

1.4.2. Lane Change Models

As discussed in section 1.3.2, to faithfully represent the behavior of a vehicle, the lane change movement needs to be considered. The LCM that can be found in SUMO are those considered deterministic, in which a given set of variables gives the output of whether or not the lane change needs to be performed. Erdmann [66] explains the JE2013 Lane Changing Model that is implemented in SUMO. The model distinguishes four different motivations to perform a lane change:

- Strategic: change of lane to continue with the planned route.
- Cooperative: to allow other vehicle to perform a lane change or merge in a lane. Includes also the possibility of slowing down.
- Tactical: change lane to reach the desired speed.

- Regulatory: to comply with the regulations stating that circulation needs to be done on the rightmost lane.

The lane change is only feasible if there is enough forward and rear space on the target lane. However, some conclusions coming from the literature [66] point out that drivers may accept lower rear and front gaps during lane changing due to the expected behavior of the vehicle performing the lane change. This is said to be a reason to modify the CFM parameters when evaluating the safety of these situations. Each of the lanes is numbered from left to right starting from 0 and are ranked according to their convenience.

The strategic lane changing is characterized by the urgency of the movement. This urgency depends on the distance to the dead end, the speed and the occupation of the lanes that need to be crossed. The lane changing model is affected by the existence of a blocking leader or a blocking follower. In this case, it is required to control the speed of the ego vehicle depending on the speed of the leader vehicle and the remaining distance and time. To prevent deadlock in the simulation, if a vehicle needs to perform more than one lane change to reach each target lane an additional space of 20 or 40 m is saved for right and left changes respectively.

The cooperative behavior assumes that a vehicle knows when it has become a blocking follower and changes lane to allow the blocked vehicle to perform its desired lane change. In SUMO, multi-lane roundabouts can be considered a special case for cooperative behavior. Meanwhile, tactical lane change needs to balance the speed gain and the effort required to perform the lane change.

The hierarchy of lane changing motivation in SUMO is also described by Erdmann. The following lists shows the motives ordered as by default, but the relative importance can be tuned modifying the parameters of the LCM.

- Urgent strategic change needed: strategic change
- Changing creates an urgent situation: strategic stay
- Vehicle is a blocking follower for strategic change: cooperative change
- Speed gain outweighs lane change discomfort: tactical change
- Necessary to stay to the right by regulations: regulatory change
- non urgent strategic change

SUMO has three different lane-change models implemented (DK2008, LC2013, and SL2015). The first one is the original model, that was later substituted by a more performing model that is currently the default model (LC2013). This model includes the possibility of tuning different parameters related to lane changing. These parameters affect the 4 lane changing motives already explained. The last model (SL2015) is a lane changing model designed to work when the sublane model is active. This model allows to configure new parameters such as minimum lateral gap and willingness to laterally encroach another vehicle. On top of this, it also includes an impatience factor. The parameters of the LCM in SUMO can be found in the documentation [65] and the main ones are shown in table 1.7.

Main LCM parameters in SUMO

Attribute	Description
lcStrategic	Eagerness to perform strategic lane changing. Higher values lead to earlier lane change. [0, inf]
lcCooperative	Willingness to perform cooperative lane changing. [0, 1]
lcSpeedGain	Eagerness to perform tactical lane changing. [0, inf)
lcKeepRight	Eagerness to perform regulatory lane change. [0, inf)
lcAssertive	Willingness to accept lower gaps on the target lane.
minGapLat	Minimum lateral gap. (SL2015)
lcPushy	Willingness to encroach laterally other drivers. [0, 1] (SL2015)
lcPushyGap	Minimum lateral gap when encroaching other drivers. (SL2015)
lcImpatience	Factor modifying the assertiveness and the willingnes to encroach other driver. [-1, 1] (SL2015)

Table 1.7: SUMO Lane Change Models main parameters [65].

Other parameters are available to represent situations such as overtaking on the right lane, or using the opposite lane, or the speed difference to perform the lane change, or to modify parameters like the cooperative for roundabouts or to allows other vehicles to gain speed. On top of this, the lane change behavior can also be externally controlled using TraCI.

1.4.3. Junction Models

The original SUMO junction model acted as black box, distributing vehicles from the entry lane to the exit lane, however this was not realistic, especially for larger intersections. To allow for a more realistic behavior, internal lanes are used inside the intersections, these lanes connect the entry lane of the intersection to the exit lane of the intersection. On top of this, internal intersections exist to represent specific traffic situations. To represent the priority of vehicles when approaching an intersection a right of way matrix is computed during the generation of the network, this approach in which an static matrix is composed did not work well for sub second simulations according to Erdman and

Kraijzweicz [67], leading vehicles to stop at the entrance of the intersection even if no foe vehicle is approaching it.

Erdmann and Kraijzweicz [67] point out that the requirements for a realistic intersection simulation are no deadlocks, no collisions, efficient use of the intersections, realistic gap acceptance, approach unprioritized links without stopping and dynamics independent of the simulation step length.

The junction model in SUMO needs to have information about the expected time and speed of the entrance in the intersection. The vehicle computes how long it will take to cross the intersection and checks the approaching vehicles on the prioritized lanes. If there is a sufficient time gap to enter the intersection the vehicle keeps going. To determine if the situation is safe when two vehicles have the same target lane the formula $v_L^2/d_L > v_F^2/d_F$ guarantees that the follower vehicle can stop before the leader vehicle stops, avoiding collision (v: speed, d: maximum deceleration).

The approaching speed to the intersection is set by multiplying a given time by the maximum deceleration of the vehicle. According to the SUMO proceedings [67], this speed could be seen as a speed that allows to safely check if the intersection can be crossed according to the speeds and position of vehicle in the foe lanes. However, the authors of the model highlight that the deceleration and acceleration profiles might not be completely realistic.

To model the dynamics of the vehicle once it is inside the intersection the CFM is used which requires to know which is the leader vehicle - the first one that enters the intersection - and the distance between follower and leader vehicles, using the start of the internal lanes as a reference. The choice of CFM influences the estimation of entry and exit times due to possible dawdling of the model used [67].

Another parameter that was introduced in SUMO specifically for the intersection model is the impatience. This parameter accounts for a vehicle that could enter the intersection aggressively forcing other vehicles to brake hard to keep the safety conditions.

The parameters included in the Junction Model currently available are described in table 1.8 according to the SUMO documentation [65]:

Main Junction Model Parameters

Parameter	Description
jmIgnoreKeepClearTime	Accumulated waiting time for a vehicle to enter a junction even if this might cause jamming.
jmIgnoreFoeProb	Probability to ignore a vehicle with right of way and enter the intersection.
jmIgnoreFoeSpeed	Ignore only vehicles with a speed below the set speed. Works in conjunction with jmIgnoreFoeProb.
jmIgnoreJunctionFoeProb	Probability to ignore foe vehicles that have already entered a junction.
jmTimegapMinor	Minimum time gap when passing ahead of a prioritized vehicle.
jmCrossingGap	Distance to a pedestrians willing to cross under which the vehicle stops.
impatience	Willingness of drivers to impede vehicles with higher priority.

Table 1.8: Main parameters of the Junction Mode implemented in SUMO [65].

1.5. Calibration and validation

The concept of calibration in simulation regards studying the output of the simulation and comparing it with real data with the objective of minimizing the error. The calibration and validation procedure is a key part of the simulation process. Since almost any output could be generated with the adequate set of parameters [68], the significance and acceptability of the results strongly depends on the transparency and the goodness of the calibration process.

Calibration is usually seen as an optimization process where objective function is the error between observed measurements and simulated measurements. However, in this process the minimum error is limited and depends on the type of data and parameters used for calibration, for instance velocity is easier to reproduce correctly than gap between vehicles [69].

It is also important to realize that calibration needs to be site specific and situation specific, since driver behavior differs from region to region and between environments as has been shown in the literature [68, 70–72].

To calibrate a model the first requirement is to have a sufficient amount of relevant data. This data needs to be feasible to obtain both from the real environment and as an output of the simulation process.

Car following experiments can be useful to obtain data for the calibration process. In these experiments, two cars are considered in a given driving situation in which one car follows the other. The velocities and gaps between the vehicles are recorded as function of time. For the simulation, the speed function of the leading vehicle is prescribed and the follower speed profile over time tries to be replicated. According to Krauss, one drawback of this approach is that it looks at microscopic aspects of the model, regardless of its performance to simulate accurately the macroscopic behavior [34]. However, to perform the calibration and validation of a model is more common to use real world data that can be obtained in different forms looking at different calibration objectives.

A different experimental approach is that of demanding a driver to perform certain manoeuvres in a real environment, the site of interest, while recording the necessary data and complementing this data with questions to the driver to better understand their behavior. This approach is presented by Wu et al. when calibrating a fuzzy logic CFM [47].

Some attempts to obtain real world information about driver behavior at a large scale can be found. For example, the Shangai Naturalistic Driving Study [73], where 60 anonymous drivers where analyzed for 3 years using sensorized vehicles and cameras; the Naturalistic Truck Driving Study [74] carried out in the US, where the main focus were drivers of commercial vehicles; the 100-car Naturalistic Study [75], also in the US but looking at cars instead of trucks. The main scope of these projects was not to retrieve data to perform calibration of car following models, however, it has been proven to be useful for this [56, 62, 71]. Other studies in the United States obtain the data from aerial views of selected sites [76, 77].

These two strategies can be found in the literature applied to the specific study area or for a reduced number of vehicles. For instance, using Global Navigation Satellite Systems to define the different speeds when crossing a roundabout [63]. The recording strategy has also been implemented successfully in the literature to obtain data from specific intersections [15, 78, 79].

Other than these two approaches, to calibrate traffic simulation parameters the measurement of traffic parameters by direct observation can be done, for instance looking at flows and queue lengths at a given intersection during peak hour [80].

The Guidelines for Applying Traffic Microsimulation Modeling Software published by the Federal Highway Administration (FHWA) [81] provide a general framework for calibration. The proposed order of calibration is capacity of the network, fine tuning of links' capacity, calibration of the route choice (if required) and, finally, system performance (travel times and queues). The recommendation is to keep the number of modifiable parameters as low as possible to reduce the calibration effort and to use observed field data to reflect local conditions when possible. Regarding the objective function, their recommendation is to use the Mean Square Error, although in the literature Root Mean Square Error is widely used.

The choice of the objectives used to calibrate depends on the CFM that is selected for the simulation. It has been shown in the literature that IDM and Gipps are easier to calibrate using trajectories due to their easier mathematical formulation. On the other hand, Wiedemann models are more difficult to calibrate trying to replicate a trajectory, although it is possible [82, 83].

In the literature, different methodologies to improve the calibration process have been proposed. Maheshwary et al. [70] defined a methodology to calibrate heterogeneous traffic. After defining the adequate flows and geometry of the scenario, they proposed to modify the parameters $\pm 10\%$ and select the ones that induce a more significant change in the output. To define the range in which modify the parameters, they found the values of each parameter that led to an output of 2 times the original output and half of the original output. Then, using Latin Hypercube Sampling, test different sets of parameters, running the simulation multiple times to account for stochasticity. They suggested to further analyze the significance of these parameters using one-way ANOVA tests. Then use linear regression considering this significant parameters to determine the overall fitness of the model and the relative contribution of each factor, considering that these equations are scenario specific. These equations are later used to perform the optimization and they proposed to use genetic algorithms.

The importance of performing a correct sensitivity analysis is highlighted by Punzo et al. [68] and helps to select relevant parameters. It has been already mentioned the use of ANOVA to identify which parameter where more significant for the calibration result. Ge and Menendez [84] proposed to use variance based sensitivity indices assuming dependence among the parameters, this index accounts for the main variance, the variance caused by interactions and the variance caused by dependence. The approach is based on generating random samples and estimating the independent and dependent sensitivity indices using Gaussian copula with known marginal distributions, extracted from literature or reasonable assumptions. If the obtained sensitivity index is low, that parameter can be fixed for the calibration process. The main concern of the authors is that an incorrect modelling of the dependence could lead to biased results [84].

Regarding the optimization process, the use of different optimization algorithms has been widely studied in the literature over the last years. Zu et al. [73] calibrated different car following models using inter-vehicle spacing obtained from naturalistic data running a genetic algorithm multiple times and then selecting the combination of values that gave the best solution. Aghabayk et al. [85] proposed the use of a Particle Swarm Optimization algorithm in which the position of a particle is updated towards its optimal position. According to the authors this method allows to reduce time running the process on separated threads. They distinguished between parallel tasks (find personal best for a given particle) and joint tasks (find global best fitness and checking stopping criteria).

Other approach to calibrate CFM is that described by Rakha and Gao [86], where a steady state calibration is proposed. They obtained the CFM parameters from macroscopic traffic stream parameters: capacity, jam density and free-flow speed. The objective function in this case is minimizing the error between field observations and functional relationship. They tested the approach for different CFM using macroscopic loop detector data and recommended that some parameters should be link specific, revealing again the difficulty to have a general model that suits different scenarios.

Studies using maximum likelihood as method to calibrate the model can also be found. This approach makes possible to jointly estimate parameters for multiple vehicles, including a priori knowledge of their distribution acquired from other date sources or literature [87]. According to the same study, the high correlation of the parameters explains the difficulty of the calibration process. Taking on this approach and using the Bayes' rules to transform prior probabilities into posterior probabilities, van Hinsbergen et al. [88] developed a strategy to quantitatively analyze inter-driver differences. This last approach has been proposed to used selecting the parameters found relevant after analyzing the variance sensitivity indeces [84].

The limitation of the validity of the CFM parametrization is also shown by Asamer [72], who calibrated a CFM for specific weather conditions. He focused on the most affected parameters for slippery road conditions that are acceleration, deceleration, desired speed, and clearance distance when considering saturation flow and start up delay. His paper also shows the high impact of the modification of one parameter in the influence of others and suggests the design of a feasibility region of parameter combinations to perform the calibration. However, the proposed approach for calibration is that of running simulation for all of the parameter combinations that are in the feasibility region. Continuing this line of work, Pótári et al. [89] implemented this methodology to check again the effect of weather conditions looking at trajectory data. They performed the calibration using different softwares (VISSIM and SUMO) and concluded that SUMO was slower due to the difficulty of post-processing the outputs. However, in both cases the brute-force calibration was found to take a significant amount of time.

Although the configuration of the network in the simulation outputs has a non-negligible effect, the main effect comes from the correct definition of the parameters of the CFM. The next section gives some examples of how different models have been calibrated according to the literature, focusing on Wiedemann models and IDM, since they are the most commonly used.

1.5.1. Application on Car Following Models

The Intelligent Driver Model (IDM) has already been explained in previous sections (1.3.1). One of the mentioned advantages of the model is the straight forward interpretation of its parameters and the possibility of measuring them.

Regarding the possibility of measuring the parameters from real data, it can be seen in the literature that even these parameters need to be calibrated. Zu et al. [73] calibrated the IDM looking at inter-vehicle spacing and saw that the observed parameters are generally correlated with and distributed similarly to the calibrated, but the calibrated values tend to be lower. This is in line with the conclusions of Punzo et al. [68].

The literature also shows, that not all parameters are equally important. Punzo et al. [62] show that when calibrating the model trying to reproduce the speed and the spacing the most influential parameters are time headway, maximum acceleration and delta when looking at contribution to variance of the error. According to Ge and Menendez [84], the parameters that could be fixed when calibrating the IDM looking at position error depend on the specific trajectory that needs to be reproduced, but the desired speed appears to be the least relevant parameter. These two studies suggest that, when performing calibration based on trajectory data, the calibration of the model needs to be done for each trajectory individually and then aggregate the calibration results instead of aggregating the trajectory data and then performing the calibration.

Another study that addresses the calibration of IDM through trajectory data is the work presented by Hoogendoorn et al. [87], they used the maximum likelihood approach to jointly estimate the parameters for multiple vehicles and concluded that the correlation among the parameters is usually very high, hence the complexity of the calibration process. They also pointed out that the least relevant parameters where deceleration and desired speed. Sun et al. [56] distinguished two different models for aggressive and non-aggressive drivers using spacing gap as the Measure of Performance, they show that it is possible to perform the calibration without looking beforehand at the impact of different parameters, just knowing the range in which this parameters are.

It is important to highlight, that these studies were looking at data taken from highway scenarios in different regions, hence significantly different values of the parameters are reported.

The use of Wiedemann models is also widely spread in traffic simulation studies due to its implementation in the popular VISSIM software. The two Wiedemann models that exist (W74 and W99) have already been described in section 1.3.1. Since there are differences among the two models, they will be discussed independently.

In the aforementioned study by Ge and Menendez [84], they tested their proposal for variance-based sensitivity indices also using the W74 model. They concluded that out of the 12 parameters that were defined in the model, only 8 were relevant. Praticò et al. [79] performed the calibration of the model for a specific roundabout, they used the speed at different stretches of the roundabout area to perform the calibration. Their results concluded that the most relevant parameter to be set of the CFM were those related to gap acceptance. However, this study looks at roundabouts focusing only on the through movement.

Some studies focus on analyzing the parameters of the W74 under different conditions. Asamer et al. [72] looked at saturation flow and start up delays to calibrate a CFM representing slippery conditions and found that the most affected parameters where those related to acceleration, deceleration, desired speed, and clearance distance. Higgs et al. [71] used naturalistic data to understand the effect of the actual speed of the vehicle on the parameters of the W74 founding that the effect was significant and that different models could be necessary to represent the real behavior of drivers at different speeds.

Regarding the W99, different attempts of calibrating the model can be found in the literature. Derrani et al. [83] looked at the action points and perception thresholds of drivers using the trajectory data coming from FHWA data-set [76]. They estimated some of the parameters from direct observation (CC1 and CC2) and others through regression looking at the values of the action points extracted from the trajectory data (CC4, CC5 and CC6) using the equations of the Wiedemann model. Sun et al. [56] also used trajectory data to calibrate the W99 model, however, they used an optimization approach trying to reduce the error in the spacing gap between follower and leader. Even though this two studies look at similar driving conditions, the values of the calibrated parameter are significantly different, highlighting again the necessity for a specific calibration and validation to obtain relevant results for a given scenario.

Mahsehwary et al. [70] calibrated the W99 model for urban conditions looking at travel time as the measure of performance, setting limits to the individual travel time error as constraints for the optimization process. They found CC0, CC1 and CC3 as the most relevant parameters of the W99 model. They also distinguished two situations in the urban scenario to consider the different behavior at intersections. Another study that looks at intersections, specifically at roundabouts, is the one carried out by Fang and Castaneda [78]. They extracted speed, gaps, headway, travel time and queue length from video recordings. They looked at the calibration of the queue length and travel time and concluded that for queue length the most relevant factors are minimum gap, headway distance and reduced speed on the approach. On the other hand, they exposed the difficulty to calibrate travel time due to the impossibility to assign in their model different circulating speeds for different turning movements. It is important to highlight that in this last study not only CFM parameters are modified to obtain the required results. These two last studies look at urban environments and intersections and this affects the value of the parameters, having significant differences from the studies referred to highways, especially those related to time headways.

When looking at the complexity of the calibration process, it is clear that more parameters make the process more cumbersome. Intuitively, it could be assumed that this higher complexity is compensated by better results in terms of accuracy. However, the literature shows that the IDM, even with less parameters tends to outperform the Wiedemann models [56, 60, 61, 73].

2 | AI@EDGE

In this chapter the AI@EDGE project is further explained. The consortium encompasses a group of diverse organizations coming from a variety of fields. The main goal of the consortium is to introduce a system architecture and explore industry-relevant applications of Artificial Intelligence (AI), edge computing and 5G technologies in a variety of cases. Edge computing refers to the paradigm in which the computation and storage of the data is close to the source of the data itself, which helps to provide low latency applications. The scope of the European project is to design an architecture able to take advantage of these technologies while maintaining high standards of privacy and security of the data. The general architecture of the AI@EDGE project is displayed in figure 2.1.

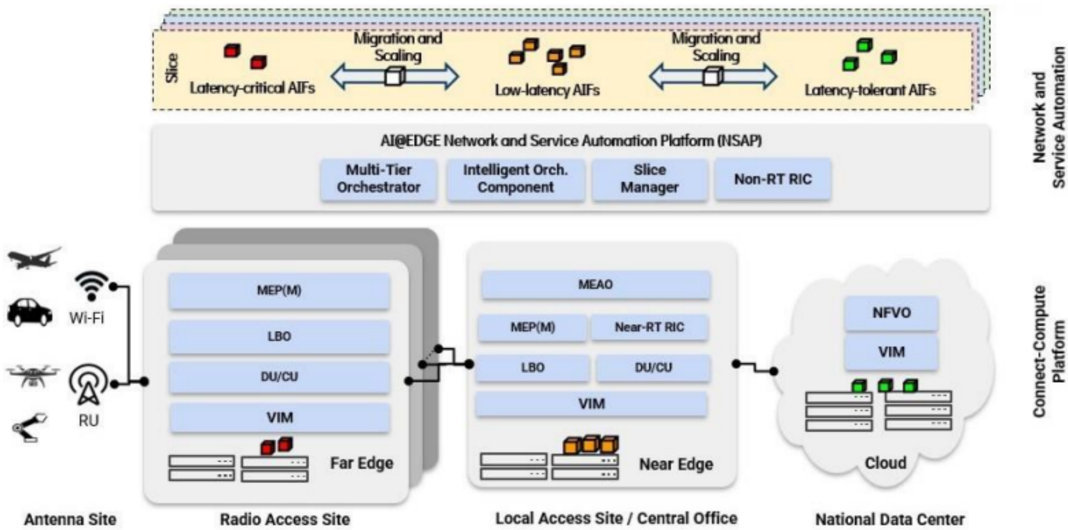


Figure 2.1: General AI@EDGE architecture [90]

It can be seen that the proposed architecture is a complex system, including different technologies and its complete definition is out of the scope of this thesis, but it can be found in the documentation of the project [90]. This architecture is adapted to be validated under different use cases (UC):

- UC1: Virtual validation of vehicle cooperative perception

- UC2: Secure and resilient orchestration of large (I)IoT networks
- UC3: Edge AI assisted monitoring of linear infrastructures using drones in BVLOS operation
- UC4: Smart content and data curation for in-flight entertainment services

It is noticeable the wide range of applications to which this architecture is intended to be useful. However, this thesis refers only to the first of the defined use cases.

2.1. UC 1: Virtual validation of vehicle cooperative perception

This case regards the development of 5G and edge computing in a road environment to implement cooperative connected automated vehicles. The basic goals of this use case are summarized as [2]:

- Showing that the architecture can be used to support industry 4.0 digital twinning;
- Design and implement the digital twinning of a mix of real and emulated vehicles;
- Recreate the data exchange required to build the cooperative perception system between emulated vehicles and the human-driven vehicle.

The general components and architecture of this use case are shown in figure 2.2, where the green boxes represent case-specific components and the blue ones represent general architecture components [91].

The case specific components of the architecture shown in figure 2.2 are the traffic simulation, the driving simulator and Worldsim synchronization and the cooperative perception algorithm. The traffic simulation includes the definition of a network and the traffic conditions in a way that they represent a real environment. The driving simulator refers to the driving simulator facilities at Politecnico di Milano (DriSMi). The cooperative perception algorithm uses artificial intelligence and Reinforcement Learning, an unsupervised machine learning technique in which the agent tries to maximize an output through a process of trial and error, to make decisions based on the conditions of the traffic simulation (position of the vehicles at each time). The goals for the training are represented by predefined KPIs referring to safety, emissions and time to cross the roundabout and are later explained. Figure 2.3 shows the scheme of the testing architecture in greater detail.

In figure 2.2 we can also see the different nodes that compose the validation process. The

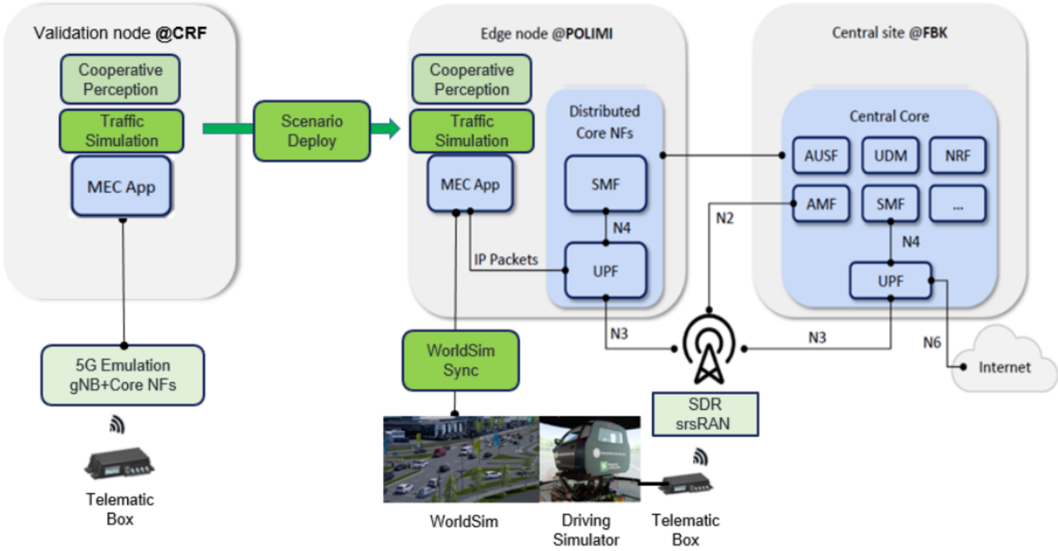


Figure 2.2: AI@EDGE Use Case 1 5G architecture [91]

CRF validation node represents the real driving environment in which the communication takes place between the vehicles and the infrastructure through the use of telematic boxes. This information is then sent to the cooperative perception algorithm deployed on the edge node, in the scope of the project using a 5G emulator.

The most relevant node for this thesis is the one located at the Polimi site. This node includes the deployment of a digital twin of a real roundabout scenario with information coming from different sources. This digital twin needs to be replicated both for the traffic simulation, which acts emulating the individual behaviour of a mix of non-automated and automated vehicles, acting as a fully virtual scenario, and for the drving simulator scenario. The connection between the real scenario and the virtual scenario occurs using the driving simulator, that allows to introduce the human in the loop and the hardware in the loop through the connection of a telematic box and the use of a 5G emulator. The driving simulator acts as a Virtual-Real scenario in which the driver input is real as well as the hardware behaviour, although the surrounding scenario and vehicles are not. This provides a simulated environment to validate the Artificial Intelligence algorithms and the radio components of the telematic box as well as to analyze the perception of real drivers on Automated Vehicles. In figure 2.3 the specific components of the test bed at the polimi site are presented.

The overall working flow of the system is as follows. The RealTime database (Rtdb) is responsible of managing the data of the simulation that is constantly changing. This data is fed to the VI-GRADE software, *VI-CarRealTime*, that interacts with the driver

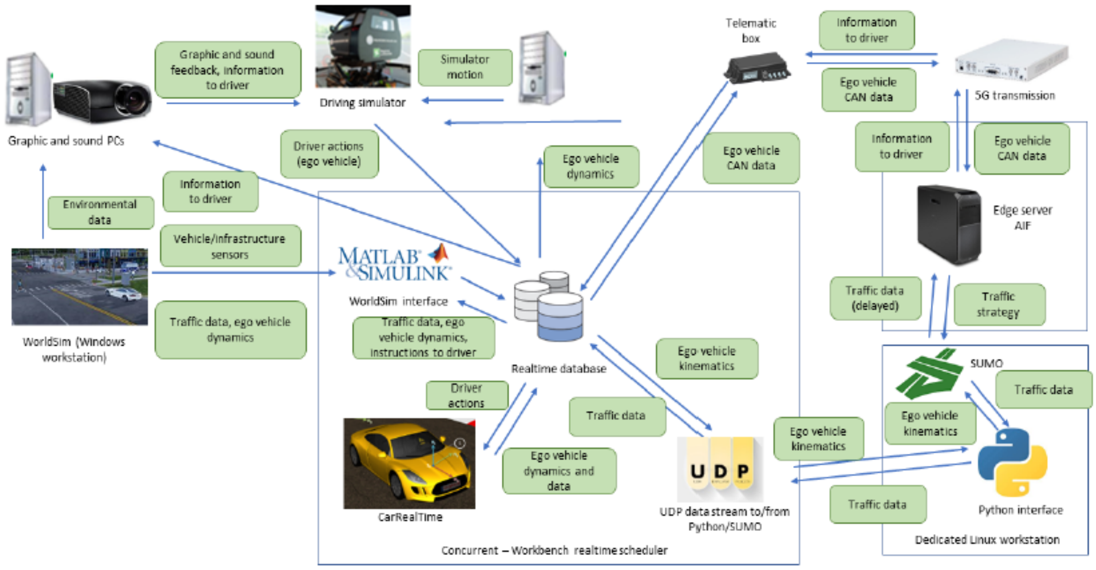


Figure 2.3: Detailed testbed architecture

actions and computes through its solver the dynamic responses, these can be feed to the cueing algorithm to provide the required movements and forces to the cockpit of the driving simulator. The graphical interface is displayed using the *VI-WorldSim* software that allows to represent different scenarios to give the sense of realism to the driver inside the cockpit. The behaviour of the other vehicles displayed in the scenario comes from the traffic simulation. This simulation is run on a different computer and uses SUMO and FLOW software. FLOW is a deep reinforcement learning framework for mixed autonomy traffic that is used in this project to train and control the automated vehicles. This simulation is interfaced using python to provide the information to the graphic interface of where the vehicles are and to retrieve the information about the position of the vehicle driven by a human in the driving simulator. This allows the automated and non-automated vehicles of the traffic simulation to interact with the human driver and respond to its behavior. The traffic simulation needs to be able to replicate the delay when providing the data to the AI policy to better represent the reality of the architecture. Then a telematic box and 5G emulator are used to reproduce the latency of the communication due to the V2N-N2V scheme used in this project, in which the vehicles are transmiting the information to the network that is able to process it and distribute it back to the vehicles. However, at the moment of developing this thesis this is not yet implement, so the communication is performed in real time.

The presented scheme of the testbed shows the complexity of the system and the wide number of elements that need to be consider, as well as the difficulties that can arise when connecting different software. In this last point, is when using open-source software (SUMO) for the traffic simulation is useful, specially due to its Traffic Control Interface (TraCI) feature that allows to control and retrieve data from the simulation while the simulation is going. The communication between software in the scope of the project represents a significant contribution, however, it is a topic on its own and is out of the scope of this thesis. Other breakthroughs of the project are the definition of a co-simulation environment able to introduce the Human-in-The-Loop feature that allows to test the impact of automated vehicles on human drivers and not only on the overall performance of the network as has been done in the literature.

2.2. Previous tests

During the preliminary stages of the project, some tests were conducted using a three-legged roundabout. Participants were asked to enter the roundabout and leave via the second exit, with tests repeated twice per leg using two different percentages of automated vehicles. During this tests, it was found that the trajectories of the non-human driven vehicles lacked realism and resulted in aggressive maneuvers when exiting the roundabout. To address this issue, an extra step in the communication process was included, which involved interpolating the position of the simulated vehicles to make their behavior more realistic on the driving simulator. Additionally, a new version of the software used in the driving simulator allows to display the turning signal and brake lights to help drivers understand the behavior of the simulated traffic.

Regarding the perception of the human drivers in this scenario in terms of safety and fluency of the traffic, when comparing a scenario with 20% and 80% of automated vehicles, no significant differences were found but these results could be influenced by the small number of participants in the tests (12 people) as well as the already mentioned lack of realism of some characteristics of the simulation.

For the preliminary campaign the policy governing the automated vehicles was trained using different scenarios varying the flow level for each entry, the used Car Following Model was the Intelligent Driver Model (IDM) mainly because of the advantages already explained in the literature review (see 1.5.1). However, the value of the parameters were taken from available literature [27] which can lead to some problems related to the validity of this data. Since this is later used to train the policy that guides the automated vehicles, not having a reliable model would lead to not having a reliable output either. This is why it is deemed necessary to generate a traffic scenario that is based on a real scenario.

2.3. Next steps

The main goal of this thesis is to provide a simulation scenario based on a real network with a calibrated CFM. The next chapters explain how this was done, however, this is just an intermediate step in the project. Once the simulation scenario is created and the calibration results are satisfactory, this information is passed to *Fondazione Bruno Kessler* (FBK) so they can work on developing the AI policy that guides the automated vehicles on the simulation. The policy is based on Reinforcement Learning considering different KPIs. The two main KPIs are the reduction of the crossing time inside the roundabout and the reduction of emissions, considering a control zone, starting meters before the entry of the roundabout and extending until the exit. The constraints of the optimization are maintaining safety conditions of traffic and guaranteeing certain levels of comfort to the passengers, assessing the lateral acceleration of the vehicle. Once the policy is trained it can be used in conjunction with the driving simulator at DriSMi where the human driver is introduced in the loop. To do so, the scenario displayed in the driving and traffic simulators need to be aligned and this alignment process is also discussed in this thesis. After this, different tests can be performed.

On top of testing the validity of the architecture to simulate automated vehicles connected to a 5G network, the biggest breakthrough of this use case is to test how human drivers react when sharing the network with automated vehicles. To assess this, similar tests to the ones performed during the preliminary campaign will be done, adding the possibility of checking the human impressions not only through a questionnaire but also monitoring physical response of the driver. It will also be possible to asses these reactions when the driver is a passenger of an automated vehicle, to check if the policy is perceived as comfortable and safe for the users. This could later lead to a revaluation of the policy considering all the feedback coming from real users. On top of this it could be possible to assess if the driver is able to perceive significant differences in comfort and safety when the policy is designed to follow a cooperative goal rather than an individual objective.

The next chapters of this thesis focus the attention on the calibration process of the real roundabout and the alignment of the scenario between the different software involved.

3 | Simulation and calibration of the real roundabout

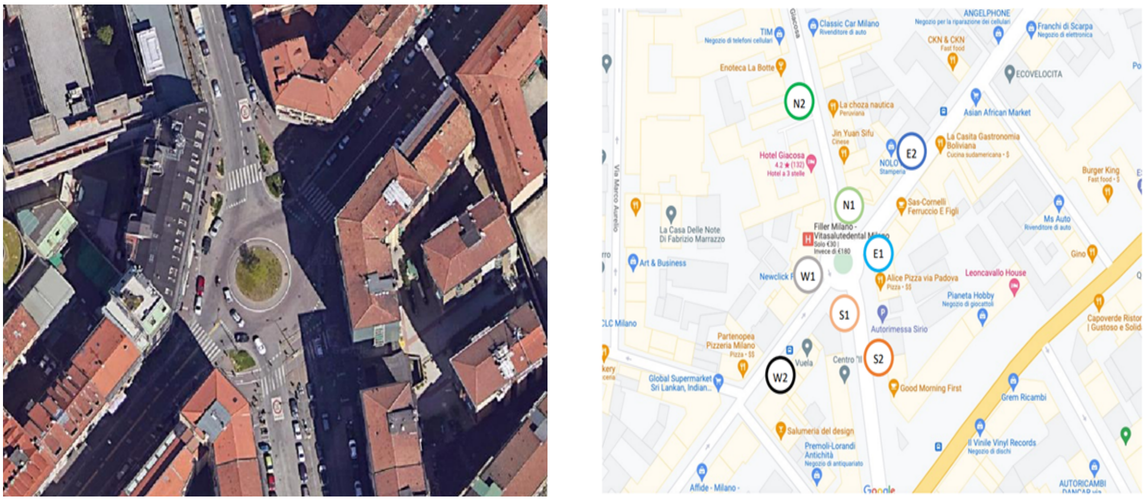
This chapter describes the design process of the scenario, starting from the selection of the location, data acquisition, scenario generation in SUMO, calibration of the parameters and discussion of the results. It is important to highlight that this includes not only the Car Following Model and Junction Model calibration but also the tuning of some parameters related to the network and demand definition trying to better reproduce reality.

3.1. Scenario description and data acquisition

In previous stages of the project a preliminary study was done using a theoretical scenario with a three-legged roundabout to evaluate the influence of different penetration rates of Connected and Automated Vehicles (CAV) on the perceived fluency and safety of traffic. In that case, the Car Following Model (CFM) parameters used to train the policy and to guide the behavior of non-automated vehicles in the simulation were taken from available literature [27]. However, as it has been shown in the literature review the parameters of the CFM strongly depend on the location and type of intersection. These limitations to use parameters coming from different studies motivates the necessity to select a location to have as reference to perform our own calibration of the CFM. The desired calibration scenario was a 4-legged roundabout with a flow level leading to a slight congestion on the approaches (queues) and with a medium size since this type is the most common across Europe, to guarantee the relevance of the findings.

Attending to these characteristics the selected roundabout to perform the calibration of the CFM was a roundabout located in the intersection between Via Padova, Via Giuseppe Giacosa and Via Francesco Predabissi in Milan shown in figure 3.1a.

The collected data to perform the calibration were flow data and queue length. The flow data is required to obtain an Origin/Destination (O/D) matrix and the queue length will be used as the objective for the calibration. The data collection process was performed



(a) Satellite image of the roundabout. (b) Entries and exits key.

Figure 3.1: Selected roundabout to perform the calibration

manually writing down the license plate numbers of the cars entering and exiting the roundabout. This required the participation of eight people, two per each leg. The people in charge of the exit lanes wrote down also the number of pedestrians and bicycles crossing the lane, meanwhile, those on the entry lanes recorded the maximum queue length and the average queue. On top of this, it is required to identify the type of vehicle, thus an extra comment is made when a heavy vehicle, bicycle, or motorbike is detected. In appendix A the templates used for the collection of the data are shown.

The data collection was conducted on 14-12-2022 (Wednesday) from 08:30 to 09:30. This collection time was divided in six periods of 10 minutes. With the collected data, it was possible to define different O/D matrices for the different periods of time and the different vehicle types. Figure 3.1b shows the names given to the exits and entries of the roundabout and serves as a key to understanding the O/D matrices.

To build the O/D matrices a match between the entering and exiting license plate numbers is expected. However, the data collection process led to difficulties matching all the detected vehicles (i.e: missing vehicles, especially two-wheelers because of the lack of lane discipline). From the total of 3783 vehicles noted (summing both entrying and exiting vehicles) it was possible to obtain a match for 3065 of them (81%). Nevertheless, it was possible to obtain an O/D matrix for each period of 10 minutes. To build the final O/D matrices, the percentages of the intermediate O/D matrices were used to infer the final O/D matrix, knowing that the exiting vehicles through each leg and considering this information to be more accurate than the entry rate, especially for 2-wheel vehicles.

An example of the O/D matrices for cars, heavy vehicles, motorbikes and bicycles is shown in table 3.1.

O/D matrices for the first 10-minute period

Cars					
O\D	E	N	S	W	TOTAL
E	0	7	24	36	67
N	21	1	56	7	85
S	20	50	4	8	82
W	31	0	13	0	44
TOTAL	72	58	97	51	278

Heavy vehicles					
O\D	E	N	S	W	TOTAL
E	0	0	0	2	2
N	3	0	1	0	4
S	3	1	0	0	4
W	5	0	0	0	5
TOTAL	11	1	1	2	15

Motorbikes					
O\D	E	N	S	W	TOTAL
E	0	0	2	24	26
N	2	0	5	4	11
S	2	4	0	5	11
W	2	0	1	0	3
TOTAL	6	4	8	33	51

Bicycles					
O\D	E	N	S	W	TOTAL
E	0	1	2	12	15
N	0	0	3	2	5
S	0	5	0	3	8
W	0	0	1	0	1
TOTAL	0	6	6	17	29

Table 3.1: O/D matrices for the first data acquisition period

It can be seen that the flows are not balanced along the 4 entries of the roundabout, in this first period the north and south legs present a significant higher flow, especially when compared with the south lane. This uneven distribution can be seen along the six different periods and it is also present in the queue length. The rest of the matrices can be checked in appendix B. In figure 3.2 the car flow coming through all four entries for the different time periods is displayed. It can be seen that the highest flow comes from the north entry and the lowest from the west entry. However, when looking at the total number of vehicles, including motorbikes, bicycles and heavy vehicles the most used entry is the east entry as shown in figure **??**.

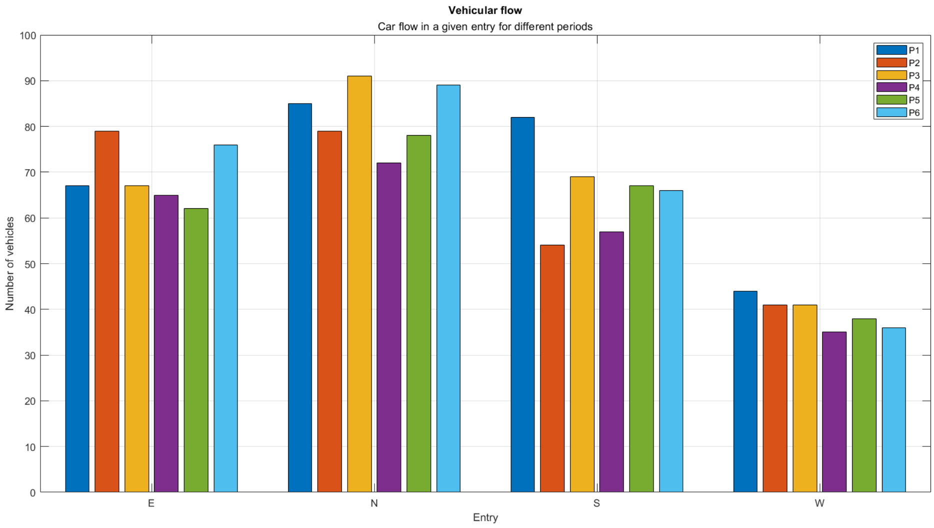


Figure 3.2: Car flows in different entries.

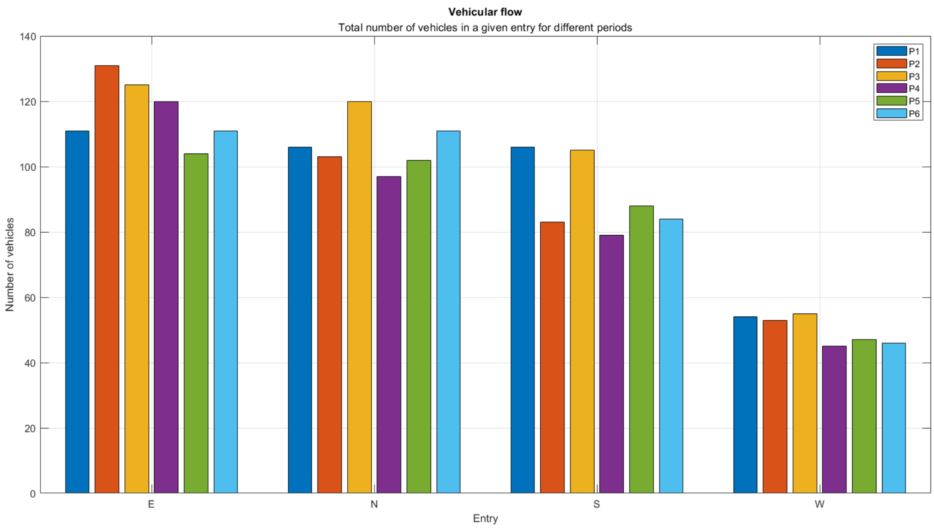


Figure 3.3: Vehicular flows in different entries.

Looking at the differences between both graphs we can see that the number of vehicles other than cars is significant and cannot be neglected when analyzing the traffic in the roundabout. However, it is important to note that not all the vehicles have the same influence on queue formation and traffic dynamics since their behavior can be widely different. This is why, especially under the scope of traffic simulation, to avoid having to represent these different behaviors that add complexity to the model, it is possible to represent a given vehicle as a fraction of other (e.g: one motorbike equals one half of a car). In the scope of this project, the factors used come from available guidelines [92], in which the factors are 0.2 for bicycles and 0.4 for motorbikes. After performing this transformation the number of equivalent vehicles is shown in figure 3.4.

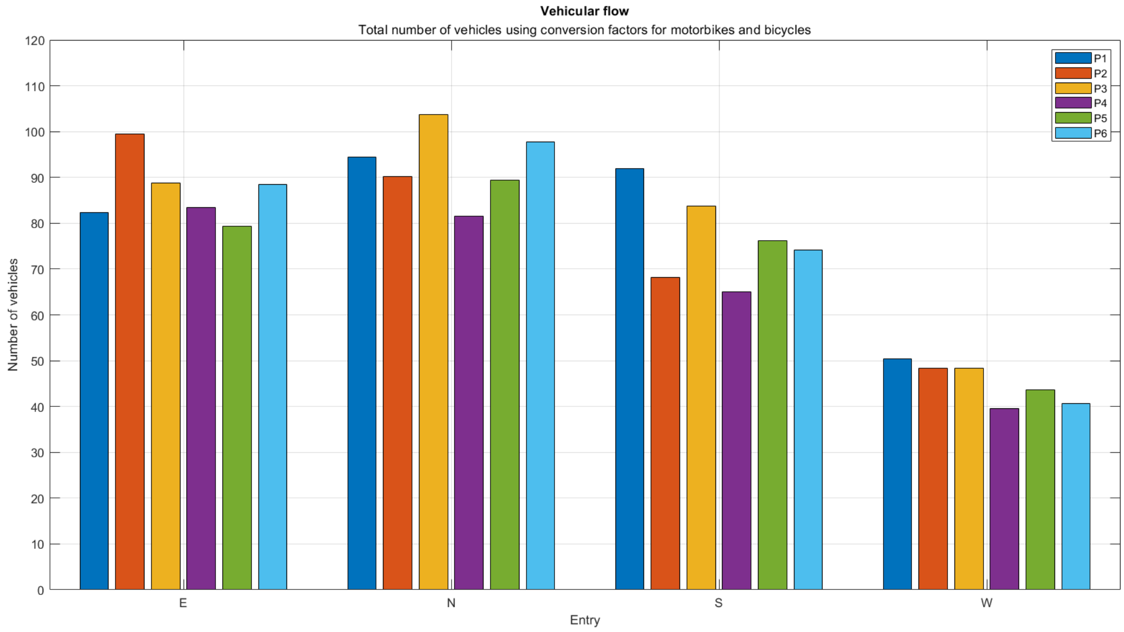


Figure 3.4: Entry flow at different periods considering conversion factors between bikes, motorbikes and cars

Regarding the length of the queues, both average and maximum queues were registered for every 10-minute period. When performing the data acquisition it was clear that checking the maximum queue was easier than assessing the average queue, since it strongly depends on perception, this is why it is considered a better objective to perform the calibration, however it is still subjected to error since the measurement was done based on physical references on the street. In figure 3.5 the length of the queues are shown for every entry and period.

Analyzing the queues, it can be seen that the north entry is the most congested one, followed by the east entry, especially in the first and last periods. It is also noticeable that the first period in the north entry represents a clear outlier, this could be caused by the time at which the data acquisition was performed and the presence of specific services on that street (i. e: there is a school on that street and the data acquisition started at 08:30). It is also important to note that not only the flow of the specific leg affects its queue, this can be seen when looking at the difference in flow between the south and west legs, in which the south entry has a significantly higher level of flow, while having a lower maximum queue. In the case of the west lane the queue could be formed because of the presence of a bus stop close to the roundabout, that concentrates the

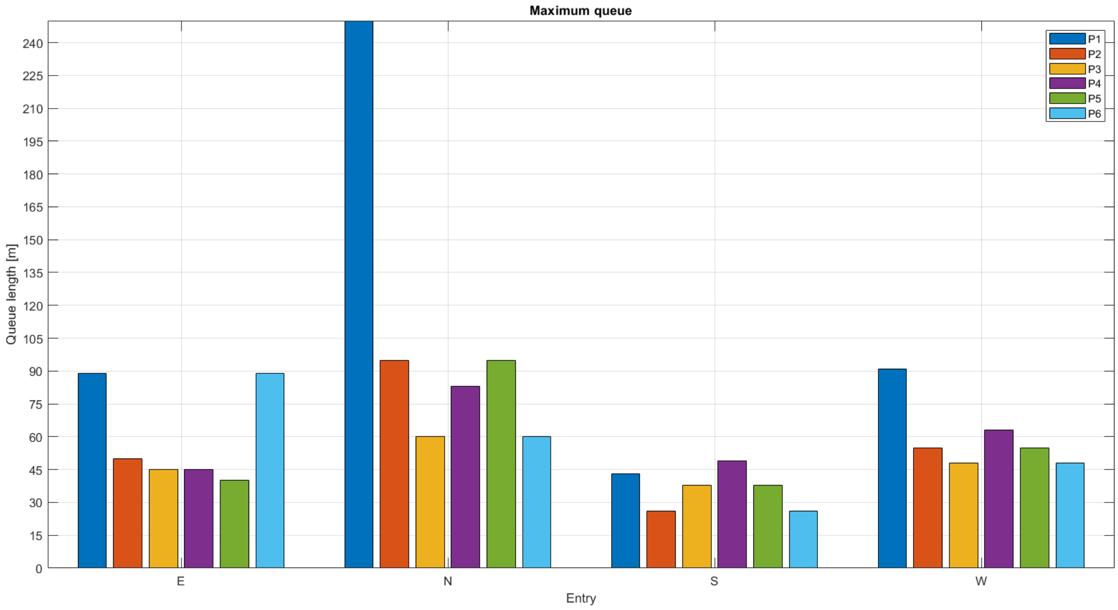


Figure 3.5: Queue length for every entry at different periods

flow of vehicles before entering the roundabout, leading to a higher punctual flow. The other possible cause of disruption of traffic and queue formation in the roundabout are pedestrians crossing the road. This is why that information was also recorded, although with a lover level of detail, only the number of pedestrians crossing a given leg are noted. This can be seen in figure 3.6

Looking at the number of pedestrians crossing the legs of the roundabout the maximum flow is seen that the most crossed leg is the north entry followed by the south legs, which means pedestrians going from west to east and vice-versa. It is important to highlight that this is the total number of pedestrians or cyclists crossing the legs of the roundabouts, which does not necessarily equal to the number of times in which the vehicular flow is interrupted by the pedestrian flow since more than one pedestrian can cross simultaneously. All this collected information modelled and introduced in the simulation as explained in the following section.

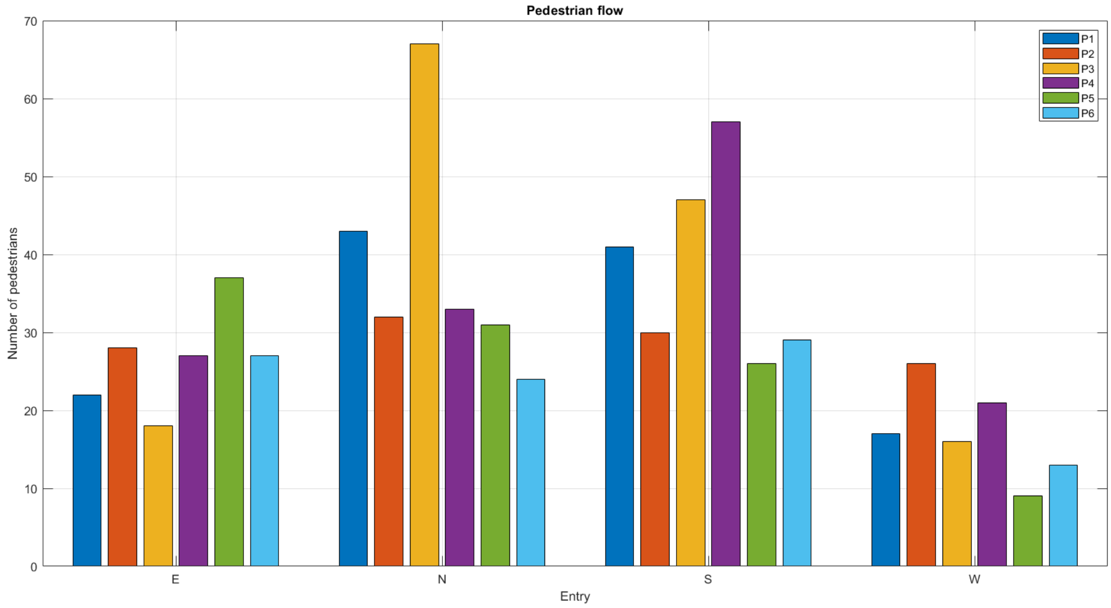
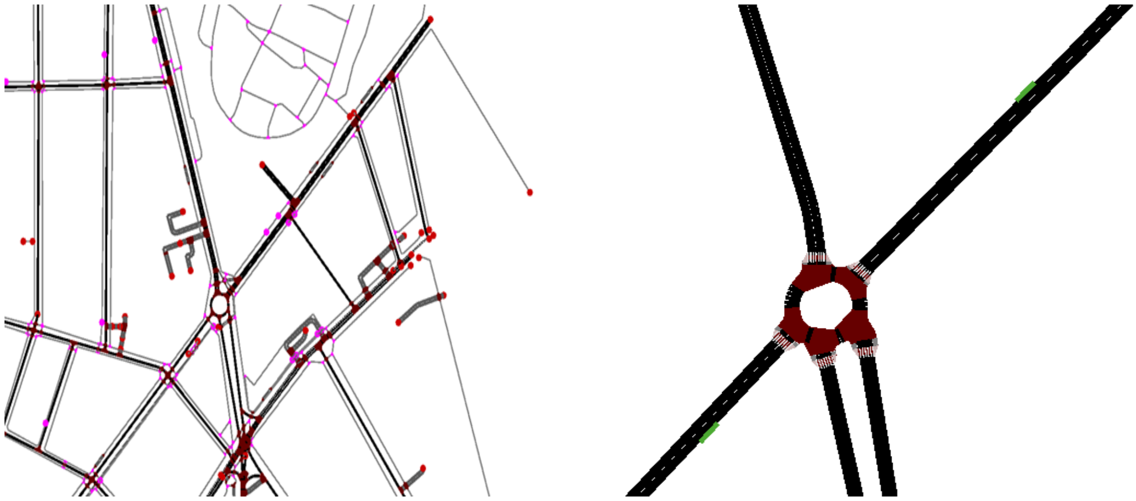


Figure 3.6: Number of pedestrians crossing a given leg of the roundabout

3.2. Network and demand definition

To calibrate the CFM, it is required to have a scenario as realistic as possible, to rule out other factors that could affect the CFM parameters, trying to obtain a model in which the CFM does not need to compensate for other defects of the scenario definition. To obtain a realistic scenario of the roundabout location the network is originally imported from OpenStreetMaps into SUMO using the *netconvert* functionalities already provided by the software. This first import includes not only roadways but also sidewalks and cycling lanes, so it needs to be refined and modified to represent only the infrastructure that affects our simulation. The original network obtained from OSM is shown in figure 3.7a where it can be seen the excess of edges and junctions, on top of this, some lanes do not have assigned the correct width and/or maximum speed.

On top of this, other modifications needed to be done. The west approach is missing a lane on the exiting direction. The width of the lanes is automatically set to the SUMO default (3.2 m) but, the north, west and east approaches have a width of 3.5 meters and the south lanes and internal lanes of the roundabout of 6 meters. This is relevant because it affects the junctions' shape and is more realistic if the sublane model needs to be used. On top of this, the size of the roundabout is not correct, having an internal diameter in the OSM scenario of 22 meters, when in reality it is 16 meters. This size affects the capacity of the roundabout, modifying the queues on the entry lanes, so it is reduced to



(a) OpenStreetMap import.

(b) Modified network.

Figure 3.7: Roundabout SUMO network

match reality. On top of this, the crossing roads are deleted, assuming that all vehicles come from the same street even if the crossing roads might be adding flow, because that information is not recorded. To introduce the collected pedestrian flow in the simulation, sidewalks and pedestrian crossings are added as shown in figure 3.8a. Finally, the legs are elongated to provide space between the entering point of vehicles in the simulation and the queue and the bus stops are located placed in the network where they are in reality, in lanes west and east, and can be seen in figure 3.8b. All of these modifications lead to the network shown in figure 3.7b.

(a) Pedestrian cross detail in SUMO.

(b) SUMO bus stop representation.

Figure 3.8: Details of the SUMO network

To represent the flow that was observed a flow distribution is extracted for each element of the OD matrix. Among the different options offered by SUMO to represent the flow, the exponential distribution is chosen since it represents a Poisson process with an expected number of vehicle insertions per second. This process is repeated over the 6 different time periods and for all vehicles types (heavy vehicles, motorbikes and bicycles). However, after some preliminary tests it is seen that the impact of motorbikes and bicycles on the simulation behaviour is different from the one observed in reality. For instance, bicycles and motorbikes tend to not respect lane discipline and this leads to lower impact on the queues that are form at the entry, however the impact on the circulatory roadway of the roundabout is not negligible. After these considerations it was decided to use a factor of correspondence between the observed number of bicycles and motorbikes an equivalent number of vehicles. Since it was not possible to calibrate this data for the specific location it was decided to use a factor coming from the literature, in this case available traffic modelling guidelines [92]. According to the aforementioned guidelines the equivalent factor for bicycles is 0.2 and for motorbikes is 0.4 and the impact on the total flows can be seen in figure 3.3 and 3.4. The flow configuration in SUMO can be seen in appendix B.

The demand was further modified by adding a warm-up phase. This warm-up phase became necessary because of the mismatch between the queue outputs observed during the first period of the simulation. To decide on the parameters of the warm-up phase the maximum and average queues of the first period were considered. The average observed queue plus 75% of the difference between maximum and average queue was considered to generate a flow of vehicles lasting 2 minutes that introduces vehicles in the simulation before the first period of flow based on real data is generated. On top of this, seeing that the queue in some of the legs were due to a bus stop it was decided to introduce also a flow of buses stopping for 12 seconds in the assigned stop, which is in line with what can be found in the literature [93, 94] for the low number of passengers that was observed, although not precisely measured, which could introduce some error into the model. The bus stops are located on the east and west legs of the roundabout and the path of the bus is from west to east and vice-versa. The data of vehicles per hour was calculated based on the available information of the specific route. Knowing that it is the route number 56 it was possible to obtain the programmed frequency of the bus service, which is 7 minutes. This is in line with the number of buses observed during the field data acquisition.

3.3. Calibration process

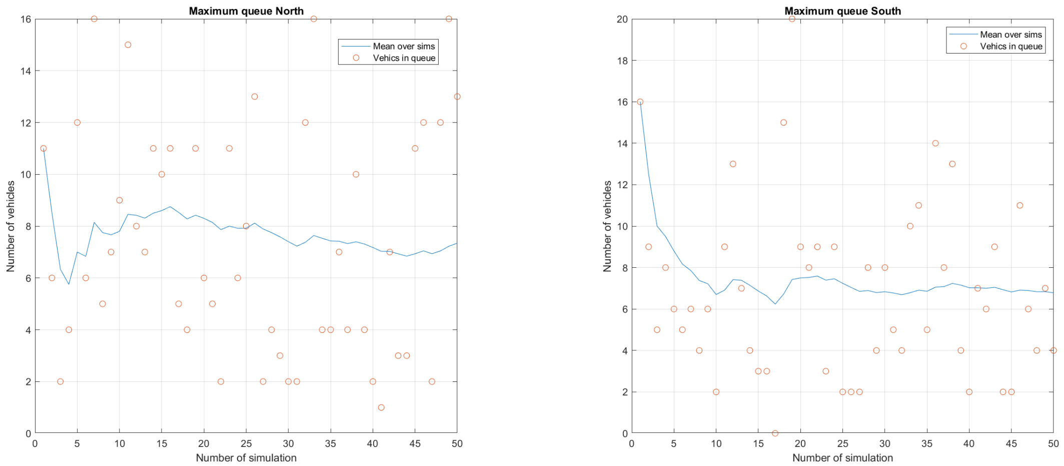
After defining the basic network and the demand on SUMO, it is necessary to perform the calibration of all the different models that affect the output of the simulation. In our case, these models are the Car Following Model (CFM) and the Junction Model (JM) implemented in SUMO that have been previously discussed in 1.4.1 and 1.4.3.

The first step of the calibration process is defining which are the relevant outputs of the simulation to be taken as the measure of performance. In this case, the output that is going to be considered is maximum and average queue length, but mainly focusing on maximum queue length, since it is more reliable. Deciding how to extract the output from the simulation is not trivial, since it has implications on the computational cost and speed of the process. Once the output is decided it is required to define how to deal with stochasticity. Afterwards, it is necessary to study which of parameters have the greatest influence in the queue output to select the parameters to modify through a sensitivity analysis. When the set of parameters to modify is clear, it is possible to begin the calibration process. In this section all these steps are explained, leading to the definition of the final parameters that are able to better reproduce the observations coming from reality.

To extract the queue information from SUMO, there are two possible options: using the queue output functionality already implemented in SUMO, that generates a file containing the queue length and queuing time at each edge for every time step; or using the TraCI interface in SUMO to obtain the number of halting vehicles (vehicles with a velocity under 0.1 m/s) in every edge each time step. Due to the requirement of a low time step, the first option leads to dealing with heavy files that slow down the simulation process, hence the second option was selected.

It is important to address the stochasticity of the simulation since it is of great relevance when dealing with traffic simulations. Randomness affects the simulation in SUMO mainly changing the specific desired speed of the vehicles and the traffic flow distributions. Both elements could be neglected defining a desired speed for all the vehicles with no variability and not using a probabilistic distribution for the vehicle flow, stating a constant time space between vehicle insertions, or specific times for each vehicle. However, enough information is not available to establish these parameters accurately. Thus, to account for stochasticity it is necessary to repeat the simulations with different seeds. The seed commands the initialization of the random processes in the software, hence repeating the simulation with the same seed would yield the same outputs. To understand the number of repetitions that are needed a convergence analysis was carried out.

The convergence study was done using the flows coming from just one period of time, thus simulating 10 minutes of real traffic instead of the complete hour to speed up the process. Regarding the parameters of the CFM, the default SUMO values are used. The analysed outputs were maximum and average queue for each of the legs of the roundabout. These outputs were then averaged over the number of simulations. The objective of this is trying to find the minimum number of simulations that are representative enough of the variability of the output due to the stochasticity of the simulation process. In figure 3.9 we can see the maximum queue on the south and north entries of the roundabout.



(a) Maximum queue on the north lane for different simulations seeds

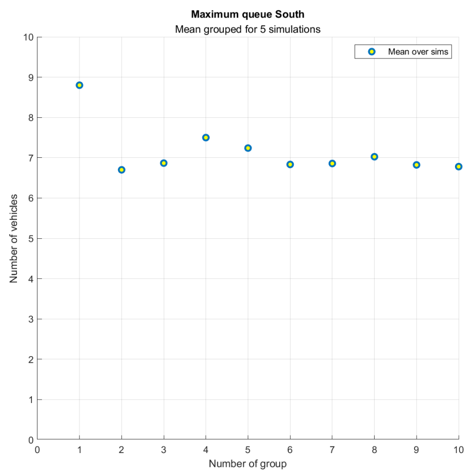
(b) Maximum queue on the south lane for different simulations seeds

Figure 3.9: Average simulation output for different seeds

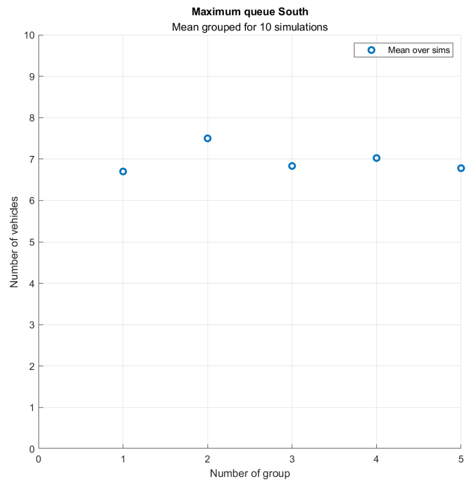
Analyzing this last graph we can see that the convergence tends to happen around 6 cars in queue, however the difference of outputs between seeds is significant, having a minimum value of 0 and a maximum value of 19 vehicles in queue. We can also see how the variation of the mean smooths out after 10-15 simulations. Ideally we would perform 15 simulations with 15 different seeds and then use the mean to analyze the outputs. However, the computational cost of running 15 simulations for every modification of parameters needs to be taken into account. Thus, it was decided to search for a minimum number of simulations. To further analyze the data, the outputs were grouped in 5 and 10 simulations to try to understand what is the variability that we would not be cathcing deciding to reduce the number of simulations. In figure 3.10 the two clusters for 10 and 5 simulations are displayed for the south lane.

It can be seen that averaging 5 simulations already improves the variability when comparing it with using a single value. This is further improved when using 10 simulations instead of 5, reducing the weigh of the outliers. However, the computational cost savings need to be taken into account. In conclusion, it was decided that 5 simulations could be sufficient for the scope of the project.

After deciding the minimum number of simulations required it was necessary to analyze which are the main parameters that should be modified to obtain the desired behaviour. To do so a preliminary sensitivity analysis is conducted. In this preliminary part the



(a) Maximum queue on the south lane averaged over 5 simulations



(b) Maximum queue on the south lane averaged over 10 simulations

Figure 3.10: Simulation outputs for different seeds averaged for 5 and 10 simulations

parameters are modified one by one and compared against the default parameters implemented in SUMO. The first parameters that are modified are parameters that according to available literature and expertise of people taking part in the calibration process are known to be significantly different from the SUMO default. The parameters are related to interaction with pedestrians in terms of priority distance and interaction between vehicles both in time and distance, as well as impatience. On top of this, parameters related to the CFM, such as acceleration and deceleration, the speed distribution and the action step length are modified. These parameters were explained during the literature discussion when dealing with SUMO. To select which of the initial parameters need to be modified an ANOVA test is performed and only the parameters that produce a significant difference when compared against the default parameters in SUMO are selected. This test is carried on with a 10 minute simulation to speed up the process.

After this preliminary analysis, a range in which the selected parameters can be modified is identified, this is done looking at the physical meaning of each of the parameters and deciding reasonable boundaries. These initial boundaries are presented in table 3.2.

It is important to consider that the *actionStepLength* is further constrained by the value of *tau*. Since *tau* can be interpreted as a following time headway and the *actionStepLength* acts as a proxy of the reaction time, having an *actionStepLength* greater than tau would lead to collisions.

To analyze the impact of the different parameters and evaluate a preliminary calibration

Maximum and minimum values for the selected parameters

Parameter	Minimum value	Maximum value
tau	0.5 s	3 s
jmCrossingGap	1 m	4 m
jmTimeGapMinor	0.3 s	2 s
impatience	0	1
acceleration	1.3 m/s^2	2.6 m/s^2
deceleration	2 m/s^2	5 m/s^2
actionStepLength	0.5 s	1.5 s

Table 3.2: Parameters range for further the evaluation of their effect.

result the combination of the selected parameters is analyzed, to do so 250 parameters sets are obtained using a sobol sequence for a dimension equal to the number of varying parameters. The sobol sequence defines a quasi-random sequence of values from 0 to 1 for each of the defined parameters, these values are then adjusted to the limits shown in table 3.2 for each parameter. An example of how the parameters are distributed is shown in figure 3.11, where it is possible to see that the space is evenly covered for a given combination of two parameters, however it is important to consider also that we are working with 7 dimensions, one per parameter, so 250 sets of parameters might not be enough to cover the entire space, although it is a number of sets similar to what can be found in the literature [70]. Each of these 250 parameters are evaluated over 5 different seeds that are generated randomly and kept constant for all of the sets. This leads to performing 1250 1-hour simulations.

After this first run of 250 simulations the impact of the parameters was reanalyzed, concluding that the *actionStepLength* parameter needed to be further limited (not exceeding 0.9 s), since it yielded unexpected and unrealistic long queues. On top of this, some tuning of the network geometry and characteristics of the junctions was performed. This involved changing the geometry to better represent the geometry of the different lanes instead of the overall dimensions of the roundabout.

Later, more simulations were launched again using a sobol sequence. In this case, the results of the simulations were compared against the real data available. To perform this comparison as a measure of performance the Mean Squared Error was used. The results were split between periods 2, 3 and 4 and periods 5 and 6. Then the sum of the MSE for all the lanes were computed for each of the periods. Between all the tested set of parameters the one that provided an optimum compromise between minimizing the error when looking at periods 2, 3 and 4 and the error when looking at periods 5 and 6. This was done in order to guarantee that the later periods were more heavily weighted when finding the best set of parameters, considering that they should be closer to reality since

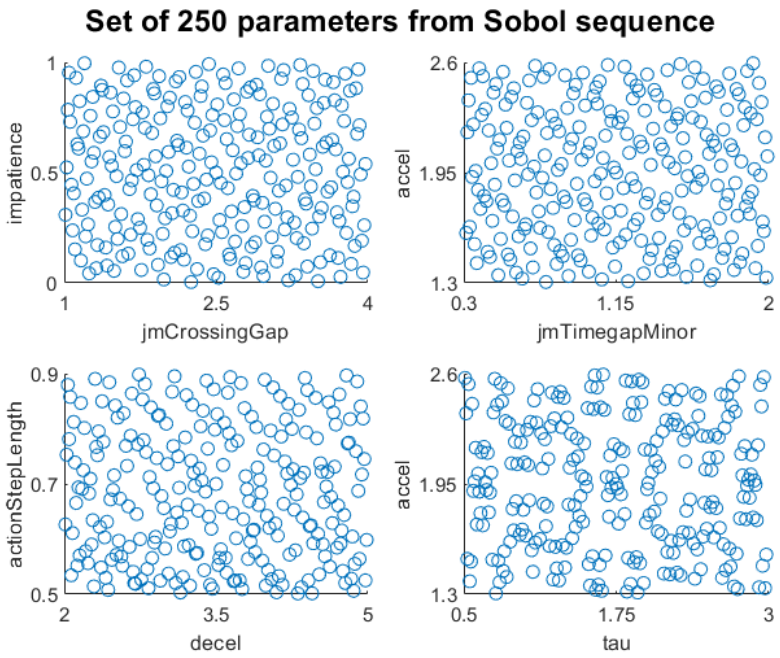


Figure 3.11: Example of parameter distribution for a 250 sobol sequence

the anomalies of the first period should have been already been dissipated and also to check that the model performs well for different flow levels as the ones seen in the different periods. It is important to highlight that the first period was neglected due to the great outlier that it represents, especially when looking at the north entry, as has already been discussed and can be seen in figure 3.5.

3.4. Results and discussion

After performing the simulations, the best solution was found for the set of parameters shown in table 3.3.

After obtaining the results, it is possible to compare the value of the obtained parameters with the original parameters of the IDM implemented in SUMO. The first parameter, *minGap*, was fixed prior to the calibration process to perform the conversion from queue length to number of vehicles in the queue. The *jmCrossingGap* parameter is reduced when compared to the default value, this would mean that the drivers are more aggressive when dealing with pedestrians, the pedestrian needs to be closer to the car for it to stop. The required time gap to a prioritized vehicle inside the roundabout is represented by the parameter *jmTimegapMinor* and it is higher than the default value, being closer to values known from previous experience using different software and seen in the literature for

Best set of parameters

Parameter	Value	Default
minGap	1.0 m	2.5 m
jmCrossingGap	1.3545 m	10 m
jmTimegapMinor	1.7792 s	1 s
impatience	0.1182	0
accel	1.7634 m/s^2	2.6 m/s^2
decel	4.2939 m/s^2	4.5 m/s^2
tau	1.3472 s	1 s
actionStepLength	0.505 s	NA

Table 3.3: Calibrated IDM parameters

comparable parameters of different models [78]. The *impatience* parameter represents the willingness of a driver to enter the roundabout even if this means that a vehicle inside needs to brake, by default this parameter is deactivated but our observations indicate that it is relevant in queue formation. The *accel* and *decel* parameters in SUMO are by default the maximum acceleration and deceleration that the vehicle can provide, however, when looking at these parameters from the perspective of a CFM they are better interpreted as the maximum acceleration and deceleration that the driver is willing to perform and that are lower than the maximum value of the vehicle. The parameter *tau* represents the time headway between vehicles, it is, in conjunction with *jmTimegapMinor*, the parameter that leads to a vehicle entering the roundabout, in this case after a car using the same entry. The last parameter that was modified is the *actionStepLength* that tries to simulate the reaction time of the driver, the obtained value is lower than the expected values for human reaction time, however the literature offers an explanation to why this is possible when looking at macroscopic outputs using microscopic simulation in the form of other aspects that are not included in this model such temporal and spatial anticipation [45].

This physical interpretation of the parameters is one of the main advantages of using the IDM, however it is important to consider that the mathematical formulation of the model (see 1.3.1) introduces interactions among the different parameters. Although these interactions reduce the interpretability of the model, it is important to consider that these values - particularly the ones referring to acceleration and deceleration - need to be below certain thresholds because of physical constraints, this has been consider prior to the calibration process when imposing limits to all of the parameters.

To further assess the results, it is possible to represent the total queue length for different periods as shown in figure 3.12. The figure shows the sum of the queues among the different legs for the experimental and the simulated results. The experimental queue representation includes an error bar to consider possible deviations. The figure excludes the first period since it was not considered when calculating the error due to its condition of outlier. In the figure it can be seen that the third period is the one that is worst represented by the simulation, meanwhile the fifth and sixth period are the ones with the smallest error. It is important to consider that the simulated results are the result of averaging the maximum queue output over 5 simulations, which may lead to a higher impact of abnormally high or low results for specific periods. It is also significant, that the measurement of the experimental queue length is also subjected to error, both when measuring it in reality and when transforming it from distance to number of vehicles. Another problem arises from the fact that the queue length is aggregated for all the lanes which does not provide a complete picture since one entry could be outperforming the others, to better analyze the results this data is separated for all the different entries.

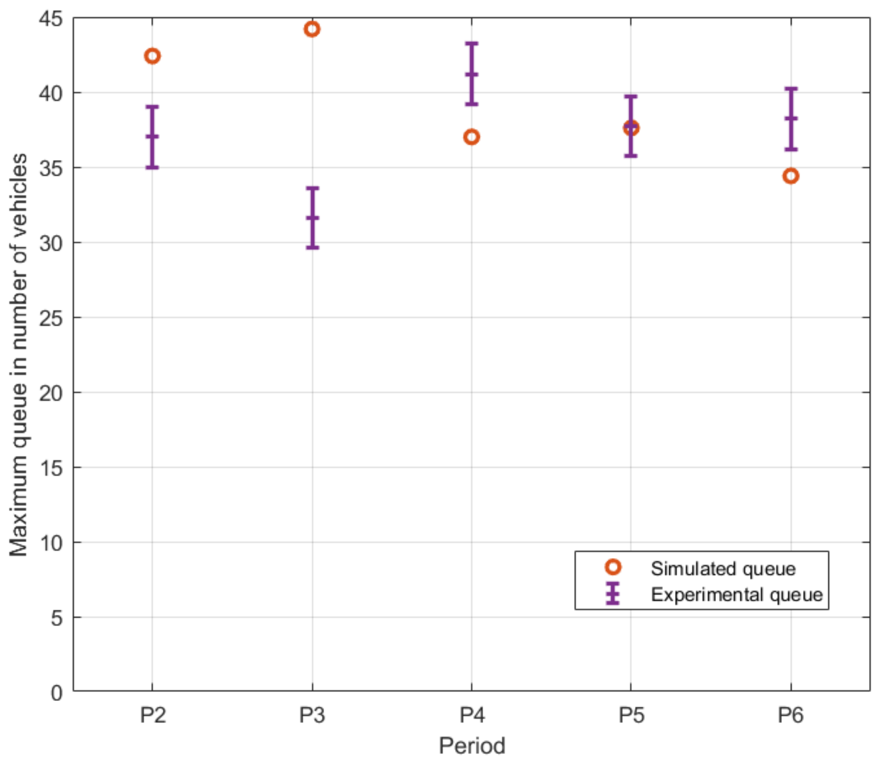
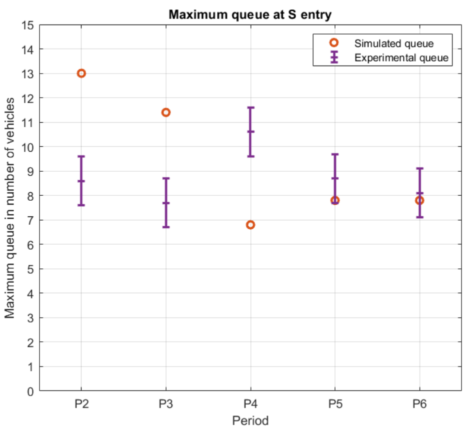
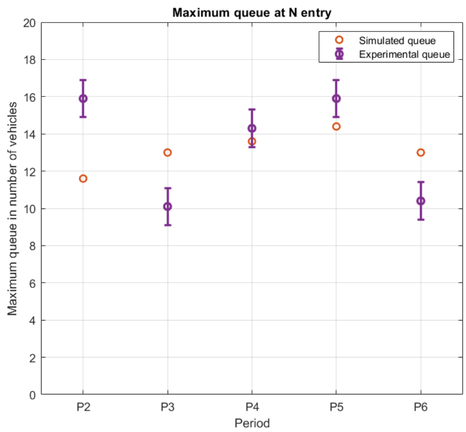


Figure 3.12: Total queue for each time period

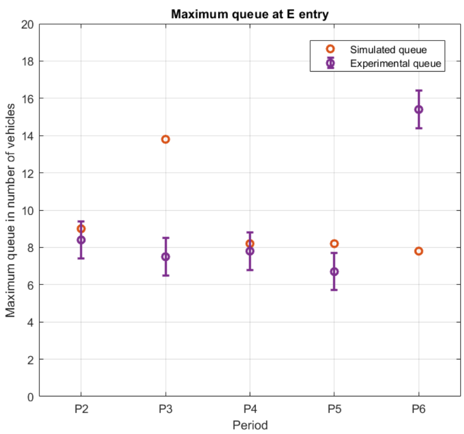
The next figures (3.13 a-d) show the simulated and the experimental queue for the different entries of the roundabout. These figures show that the error of the simulation is different for all the entries. It can be seen that the overall error is reduced for the last two periods, with the exception of the last period on the east lane. When looking at the south entry (figure 3.13a) we can see that the simulation overestimates the queue in the first two periods and is not able to capture the change in trend during the fourth period. Figure 3.13d displays the conditions on the west approach, in this case, excluding the second period (P2), the simulation is able to capture the trend that was seen in reality.



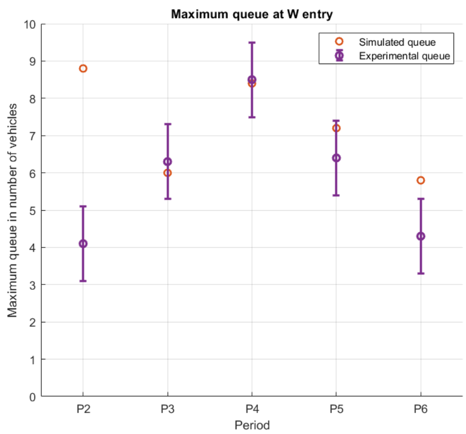
(a) Maximum queue on the south entry



(b) Maximum queue on the north entry



(c) Maximum queue on the east entry



(d) Maximum queue on the west entry

Figure 3.13: Simulated and experimental queue length for different lanes and periods

It is clear that these parameters are not able to reproduce the observed queue with no error, however expecting this from the simulation would be unrealistic. It is important to consider that the roundabout is not isolated from the network, so it is possible that some of the conditions that led to higher queues at specific times came from other characteristics of the network or the traffic flow conditions and to control for all the conditions of the network is nearly impossible. Nonetheless, if the goal were to furhter diminish the error it could be possible to loose the constraints of having maximum and minimum values of the parameters, or to use different CFM parameters for different approaches or even for different routes, since it can be seen in the literature that the behavior of the driver changes depending on the traffic conditions as well as a function of the desired exit of the roundabout [20, 21].

Regarding the lack of data to perform further validation of the parameters, it could have been helpful to repeat the data acquisition process for different traffic conditions and in different days, to better capture the impact of the infrastructure and the interaction between vehicles which changes for different flow levels. Also, having the data further divided in smaller time periods would have yielded more data-points to compare and allowed us to better capture the reality of the scenario, leading, however, to a more cumbersome data acquisition process. It is also important to note that all the results are necessarily influenced by the fallibility of human perception when collecting the data. This last aspect could be corrected using available technologies such as cameras and image recognition to asses the queue, the number of vehicles and even the travel time, but these solutions are very expensive, not only from the point of view 0f equipment price, but also considering the data management computational costs.

However, the main constraints during the calibration process was the high computational effort of performing 1-hour simulations with a time-step of 0.005 seconds. This constraint responds to the necessity of having a real-time simulation when performing tests in the driving simulator. This parameter greatly affects the behaviour of the CFM so calibrating the parameters with a higher time-step would lead to significant differences in the output of the simulation. This is one of the issues when trying to match macroscopic outcomes with microscopic behaviour. This condition constrained the way in which the data was acquired from SUMO using the TraCI option of getting the number of halted vehicles for each time step since using the option of asking SUMO directly for the queue output file led to the necessity of managing huge files for each of the simulations, further slowing down the process.

Despite all the considerations made, the obtained results are deemed to be satisfactory and are the ones provided to *Fondazione Bruno Kessler* (FBK) to perform the training of the AI policy that guides the behavior of the automated vehicles.

4 | Co-simulation: Driving simulator and traffic simulator

This chapter deals with the process of creating the scenario that is seen in the driving simulator and its alignment with the SUMO scenario. The first section explains the requirements for both scenarios and the differences and similarities with the scenario used for the calibration process as well as some of the constraints related to the co-simulation process. The second section describes the specific process undergone both to create the scenarios and to guarantee its alignment.

4.1. Requirements and constraints

A basic requirement for a co-simulation to work is that the different software involved replicate the same events in a comparable way. In our specific case, this means guaranteeing that the vehicle information (position at each time-step) coming from SUMO is correctly displayed in the Worldsim scenario that is seen in the driving simulator; and, vice versa, to introduce the human-controlled vehicle behavior, coming from the driving simulator software, in the SUMO simulation obtaining the desired interactions with the traffic.

The first thing that needs to be considered is that the two software involved do not use the same reference to locate the vehicles' positions (i. e: the vehicle reference in SUMO is in the middle of the front bumper, meanwhile Worldsim uses the rear axle position as input and provides the position of the front axle as output). Furthermore, the angle in SUMO is biased 90º with respect to WorldSim. These systematic differences had already been addressed to perform the previous tests introducing an intermediate step that performs the required modifications to the coordinates coming from SUMO before feeding them to WorldSim.

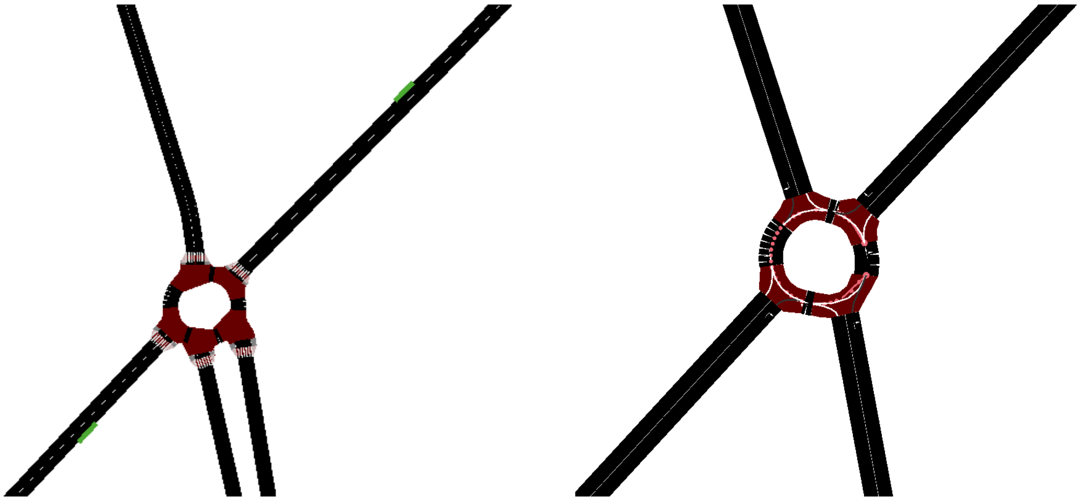
In this specific case, it is important to include the constraint that the scenario is based on a real location and that the calibration has been based on this, so changing the geometry and other aspects could certainly influence the outcomes of the simulation, being this important to get a certain degree of realism from the simulations. Other constraints and requirement are:

- The scenario needs to be easy to interpret by persons on the driving simulator with no previous knowledge of it, therefore, it is necessary to avoid possible sources of driver mistakes (e.g: exiting on an incorrect exit).
- The width of the lanes needs to be close to reality while still allowing the vehicle in the simulator to remain in the same lane as in SUMO. This is particularly important for the circulatory roadway, which needs to be wide enough to allow for realistic trajectories to be performed by the human in the driving simulator.
- The position at which the vehicles stop in the SUMO simulations need to be realistically displayed in the driving simulator, this is, vehicles should be stopped close to circulatory roadway, as it is seen in reality.
- The conjunction of the two software does not allow yet to introduce an control pedestrians as it is done with cars.
- The visual environment needs to offer a sense of realism so that the person in the driving simulator is able to perceive speed and behaves as close as possible to how they would behave in a real environment.

Once this constraints were clear, it was possible to start the scenario alignment process as an iterative process trying to match all the requirements that is described in the next section.

4.2. Scenario alignment

The starting point for the scenario alignment is the SUMO scenario obtained for the Car Following Model and Junction Model calibration based on OpenStreetMap data, the scenario had to be modified to avoid possible errors during the assessment of the scenario by real drivers This meant deleting the gap that can be seen on the South legs. Since it was decided that representing pedestrians was not inside the scope of the project at this phase, it is not required to introduce this crossing in the SUMO scenario, which could lead to a slightly different behaviour. The last modification consists in changing the junction shape, so the vehicles in SUMO stop closer to the circulatory roadway and are correctly displayed in the Worldsim scenario. These modifications are firstly implemented in SUMO, getting a working scenario, and can be seen in figure 4.1.



(a) Network used for the calibration of the model (b) Network used for the simulation in the driving simulator

Figure 4.1: SUMO network modification to interface the driving simulator

Afterward, the SUMO scenario is exported to an intermediate format that is compatible with *RoadRunner*, the software used to create the graphic environment of the driving simulator. Even though the intermediate format is compatible with both software some adjustments are still needed to guarantee that the roads seen in SUMO and RoadRunner are the same lengths and shapes. In addition, RoadRunner is used to introduce all the required visual elements, such as driving lanes, sidewalks, traffic signs, and pedestrian crossings, that make the simulation more realistic

After this, the scenario can be exported to Worldsim and check running a simulation with trajectories generated by SUMO if the scenarios are aligned. The main source of misalignment is the different reference systems used by the software, this is managed modifying the trajectories from SUMO adequately. On top of this, an offset needs to be added to the original SUMO scenario to perfectly match the WorldSim scenario, guaranteeing that the cars are effectively on the road. To perform this, the SUMO's *netconvert* is used since it allows to take the original SUMO network and offset it using as reference the position of the centre of the roundabout in both RoadRUnner and SUMO. The original missalignment can be seen in figure 4.2, where the black lanes represent the elements of the roundabout in Worldsim, the red lanes represent the SUMO trajectories before the offsetting and the blue ones show the final position of the SUMO trajectories after offsetting the network. This offsetting could be directly introduced in the Simulink model that controls the input data to the driving simulator scenario, but it would lead to other difficulties down the line, so it is more convenient to have the two scenarios previously aligned.

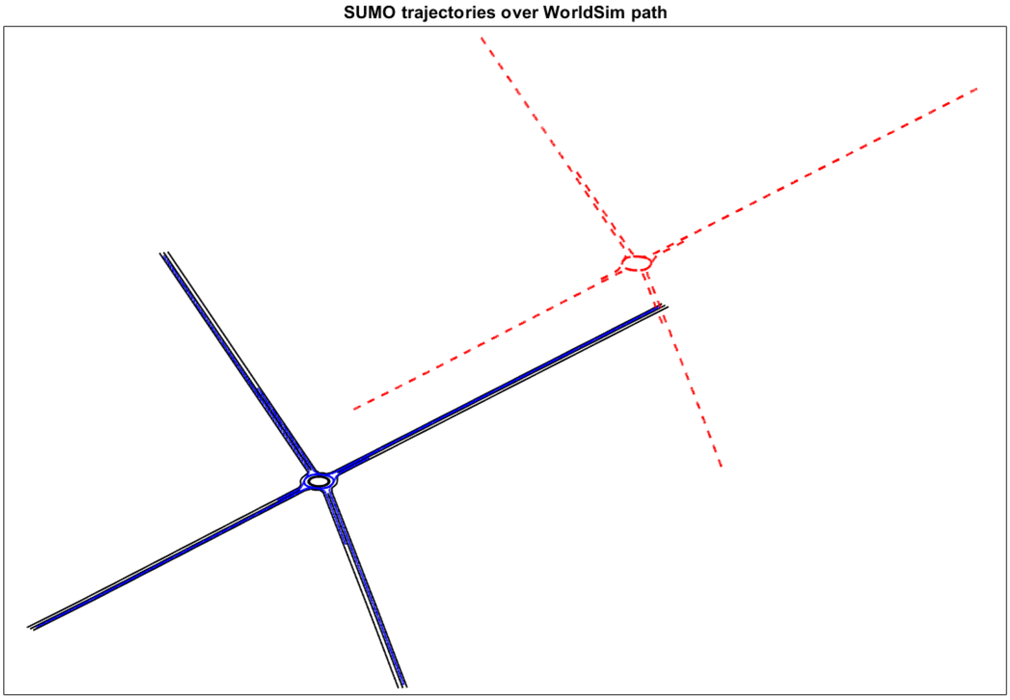
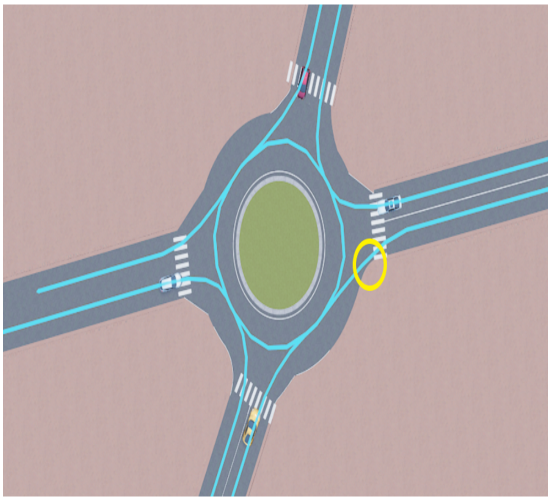
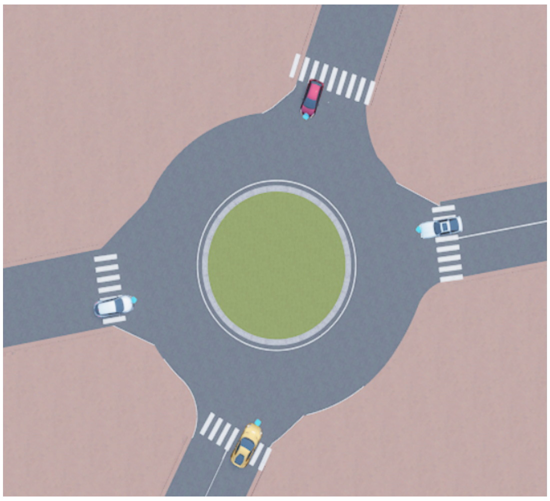


Figure 4.2: Alignment offset between WordlSim and SUMO

After guaranteeing that the trajectories are aligned, it is required to look at how the vehicles going over this trajectories are displayed in the driving simulator environment, to do so the trajectories coming from SUMO can be displayed in the WorldSim Studio software were it is possible to force vehicles to move over this trajectories to check if the scenario needs further modification to perform correctly. An example of this can be seen in figure 4.3a where one of the preliminary iterations of the scenario is shown and we can assess that on the highlighted exit the flare needs to be modified to accommodate the trajectory. This figure also makes clear on the of the limitations of SUMO when representing circular trajectories, since they are given by a number of indivual points joined by a straight line. It is possible to modify the network file so that the number of points is increased to obtain higher resolution, however, SUMO is still limited by the fact that the trajectories inside the junction are unique independently of the complete trajectory of the vehicle (i.e: both a vehicle making a U-turn and a vehicle turning right on the first exit exhibit the same trajectory in the part of the path they share). Since this behavior is different from what can be seen in reality, it is decided to modify how the trajectories are displayed on the driving simulator presetting the visual trajectories knowing the origin and destination of the vehicle, although it is out of the scope of this thesis. This approach of using WorldSim to preview the position of the vehicles is also helpful to understand were they would stop to wait to enter in the roundabout, figure 4.3b shows the stopping position for the 4 different entries, where the blue dot represent the SUMO stopping lane (i.e: the position of the middle of the front bumper when waiting to enter a junction).



(a) SUMO trajectories displayed on the Wordl-Sim scenario



(b) Position of the SUMO stopping lines displayed on the WorldSim scenario

Figure 4.3: Alignment checking using WorldSim Studio

Despite using WorldSim Studio to perform this alignment analysis, it is still required to check on the driving simulator if the feeling obtained is sufficiently realistic or if some modifications are still needed. This is true both for the visual elements of the network, as well as for the trajectories of the vehicles and other elements of the simulation environment and to check if SUMO is correctly retrieving the data provided by the driving simulator.

After having the scenarios aligned it is possible to add all the visual elements that render the simulation environment more realistic, adding references so that the driver can have a sense of speed. This is shown in figure 4.4.

(a) Aereal view

(b) Ground view

Figure 4.4: Final WorldSim scene used in the driver simulator

4.3. Tests and results

After creating the scenario, we performed more tests to asses how drivers perceive the safety and the fluency of the simulation. In this case, the drivers are asked to enter the roundabout and leave the roundabout using the third exit. The drivers experience two different penetration levels of automated vehicles - 80 % and 20 % - for each of the legs of the roundabouts but they do not know the order in which they are exposed to this scenarios. In total each driver does 8 runs before assessing the fluency and safety they experienced. They are asked to do this answering a questionnaire. The expected result, according to the KPIs used for the training of the policy, is that the fluency of the traffic is improved when the percentage of automated vehicles is increased, while maintaining a satisfactory level of perceived safety.

The questionnaire starts with some control questions regarding age, km driven per year, need of glasses to drive and years of driving experience. Then two questions about if vehicles respected the rules of precedence and if they waited for too long at the entry of the roundabout are introduced. This questions is repeated two times, considering the two different scenarios presented (different automated vehicle percentage). The final questions refer to the preference of drivers about the first or the second scenario.

1. Regarding safety perception, which of the following statements do you agree with the most?
    - Traffic in the scenario with 20% of CAVs was definitely safer than in the scenario with 80% CAVs
    - Traffic in the scenario with 20% of CAVs was partially safer than in the scenario with 80% CAVs
    - Traffic in the scenario with 20% of CAVs was partially less safe than in the scenario with 80% CAVs
    - Traffic in the scenario with 20% of CAVs was definitely less safe than in the scenario with 80% CAVs
    - No difference perceived
2. Regarding traffic smoothness, which of the following statements do you agree with the most?
    - Traffic in the scenario with 20% of CAVs was definitely smoother than in the scenario with 80% CAVs
    - Traffic in the scenario with 20% of CAVs was partially smoother than in the scenario with 80% CAVs
    - Traffic in the scenario with 80% of CAVs was partially smoother than in the scenario with 20% CAVs
    - Traffic in the scenario with 80% of CAVs was definitely smoother than in the scenario with 20% CAVs
    - No difference perceived

3. Globally, which of the 2 scenarios did you prefer?

In this round of tests, 10 drivers were asked to perform the test. The results obtained regarding scenario preference can be summarized and reported as in table 4.1. In this table the previous questions are synthesised as preference in terms of safety, smoothness and overall preference.

Questionnaire answers

Answer\Question	1. Perceived safety	2. Perceived smoothness	3. Overall preference
Definitely 20% over 80%	0	1	0
Partially 20% over 80%	2	3	3
Partially 80% over 20%	2	4	3
Definitely 80% over 20%	6	1	4
No difference	0	1	0

Table 4.1: Preference questionnaire results

The results also show that in the simulation with a higher number of vehicles, the yield rules are perceived as more respected. In the 80 % scenario 6 people perceive that yield is respected, and this number goes down to 2 for the 20% scenario. Regarding the fluency of the queue, drivers generally don't perceive either of the scenarios as slow to enter the roundabout.

In conclusion, it can be seen that the vehicles governed by the policy not only increase the fluency of the traffic, but also are perceived as safer. With an overall preference for the scenario with a higher percentage of CAVs.

5 | Conclusions and future works

The objective of this thesis was to create a calibrated and aligned scenario for performing realistic co-simulations using a traffic simulator (SUMO) and a driving simulator (DriSMi) in an urban environment. This co-simulation will enable to evaluate the impact of automated vehicles on drivers' perception of safety and traffic fluency, as well as assess the comfort of the AI policy guiding the automated vehicle that is trained using the calibrated scenario.

The calibration process of the traffic simulation involved several steps. First, we collected real data on vehicle flow and queues at the real roundabout. Then, we designed the network to closely resemble the real environment. After that, we identified the most influential parameters through previous experience and sensitivity analysis, and various parameter sets were tested to determine the best solution.

The biggest constraint that we found was the requirement of using a time step of 0.005 seconds for the driving simulator. This increased significantly the computational cost of the traffic simulation, which limited the number of repetitions per simulation to deal with stochasticity. To address this issue, regression techniques using neural networks were proposed as a potential solution, which would then allow to have a faster process where different optimization techniques could be tested, but more work needs to be done in this area.

To improve the calibration process, gathering more data on the same roundabout for different traffic conditions would be helpful for model validation purposes. It could also be possible to compare the performance of different Car Following Models (CFMs), although the Intelligent Driver Model (IDM) was selected because it outperforms more complex models according to the literature. However, exploring other options for modeling driver behavior, such as using AI to develop a policy that mimics real driver behavior, could be worthwhile. Using the capabilities of the driving simulator, it could be possible to explore the speed and acceleration profiles performed by real drivers and compared them with the ones obtained using different CFMs or the automated vehicle policy. This information could be then used to further refine the model and improve its accuracy in reproducing real-world traffic conditions.

Once the calibration of the traffic scenario was completed, it was possible to introduce it in the co-simulation environment and perform some tests with non professional drivers on how they perceived the safety and fluency of the traffic. From the obtained results, we can conclude that the higher number of automated vehicles not only reduces the crossing time in the roundabout but also is perceived as safer and smoother. This is in line with the expected results, but more testing will be done to validate this conclusions.

It is important to consider the impact of automated vehicles not only in other vehicles and drivers, but in vulnerable road users. To do so, it could be interesting to, increasing the complexity of the traffic simulation used in the co-simulation, include other actors such as pedestrians and cyclists taking advantage of the already implemented co-simulation architecture.

In the coming months the AI@EDGE project will go on carrying diverse tests using this roundabout scenario to check the impact of the automated vehicles in the driver's perception of safety and traffic fluency. The comfort of the policy for passengers of the automated vehicle will also be evaluated, in this case the driver in the driving simulator will have no control over what the vehicle does and the perceived safety and comfort will be assessed. On top of this, it could be possible to analyze the perception of different policies with cooperative and individual objectives.

Bibliography

[1] Department for Transport. Research on the impacts of connected and autonomous vehicles (cavs) on traffic flow summary report department for transport, 2016.

[2] Stefano Secci et al. A secure and reusable artificial intelligence platform for edge computing in beyond 5g networks. d2.1 use cases, requirements, and preliminary system architecture, 2021.

[3] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *IEEE Intelligent Transportation Systems Conference*, 2018.

[4] Lee A. (Lee August) Rodegerdts, Bruce W. Robinson, National Research Council (U.S.). Transportation Research Board., National Cooperative Highway Research Program., and United States. Federal Highway Administration. *Roundabouts : an informational guide.* Transportation Research Board, 2010.

[5] Werner Brilon. Studies on roundabouts in germany: Lessons learned. In *Proceedings of 3rd International TRB-roundabout Conference*, 2011.

[6] Ministero delle infrastrutture e dei trasporti. Decreto 19 aprile 2006 - norme funzionali e geometriche per la costruzione delle intersezioni stradali. (gu n. 170 del 24-7-2006), 2006. URL: `https://www.gazzettaufficiale.it/eli/id/2006/07/24/06A06657/sg`.

[7] Porter, Gooch, Peach, Chestnutt, Moore, Broeren, and Tigelaar. Advancing turbo roundabouts in the united states: Synthesis report fhwa safety program, 2010.

[8] Lambertus Bertus and G Fortuijn. Turbo roundabouts estimation of capacity. 01 2003.

[9] Christer Hydén and András Várhelyi. The effects on safety, time consumption and environment of large scale use of roundabouts in an urban area: a case study. *Accident Analysis and Prevention*, 32:11–23, 2000.

[10] Lucy Tamsin Downey, Wafaa Saleh, Deepti Muley, and Mohamed Kharbeche. Pedestrian crashes at priority-controlled junctions, roundabouts, and signalized junctions: The uk case study. *Traffic Injury Prevention*, 20:308–313, 3 2019.

[11] Dago Antov, Kirstiina Abel, Peep Sürje, Harri Rõuk, and Tiia Rõivas. Speed reduction effects of urban roundabouts. *Baltic Journal of Road and Bridge Engineering*, 4:22–26, 2009.

[12] Bram De Brabander, Erik Nuyts, and Lode Vereeck. Road safety effects of roundabouts in flanders. *Journal of Safety Research*, 36:289–296, 2005.

[13] Orazio Giuffrè, Anna Granà, Maria Luisa Tumminello, Tullio Giuffrè, Salvatore Trubia, Antonino Sferlazza, and Marko Rencelj. Evaluation of roundabout safety performance through surrogate safety measures from microsimulation. *Journal of Advanced Transportation*, 2018, 2018.

[14] Yadeta Chimdessa, Semu M Kassa, and Legesse Lemecha. Efficiency of roundabouts as compared to traffic light controlled intersections in urban road networks. *Momona Ethiopian Journal of Science (MEJS)*, 5:81–100, 2013.

[15] Hooi Ling Khoo, Chun You Tang, and Qiang Meng. A novel control strategy for roundabout system with origin-destination flow pattern. volume 25, pages 1556–1567. Elsevier B.V., 2017.

[16] Vincenzo Gallelli and Rosolino Vaiana. Safety improvements by converting a standard roundabout with unbalanced flow distribution into an egg turbo roundabout: Simulation approach to a case study. *Sustainability (Switzerland)*, 11, 1 2019.

[17] P. Fernandes, R. Tomás, F. Acuto, A. Pascale, B. Bahmankhah, C. Guarnaccia, A. Granà, and M. C. Coelho. Impacts of roundabouts in suburban areas on congestion-specific vehicle speed profiles, pollutant and noise emissions: An empirical analysis. *Sustainable Cities and Society*, 62, 11 2020.

[18] Alfonso Montella, Shane Turner, Salvatore Chiaradonna, and Dave Aldridge. International overview of roundabout design practices and insights for improvement of the italian standard. *Canadian Journal of Civil Engineering*, 40:1215–1226, 2013.

[19] Linh Thanh Trinh. *Development of Microscopic Simulation Model for Heterogeneous Traffic at Roundabout*. PhD thesis, 2020.

[20] Gregory W Davis, Vaughan W Inman, Ted Shafer, and Bryan J Katz. A simulation study of path and speed through double-lane roundabouts. 2005.

[21] Zhenliang Jiang and Said Easa. Analysis of driving characteristics at roundabout using high definition vehicle trajectory data design implications of roundabout sight distance. view project. In *CSCE Annual Conference*, 2020.

[22] Ana Bastos Silva, Sílvia Santos, Luís Vasconcelos, Álvaro Seco, and João Pedro Silva. Driver behavior characterization in roundabout crossings. volume 3, pages 80–89. Elsevier, 2014.

[23] Susan A Shaheen and Rachel Finson. *Intelligent Transportation Systems*. PhD thesis, 2013.

[24] SAE International. Surface vehicle recommended practice taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles, 2021.

[25] Yiyun Wang, Rongjie Yu, Shuhan Qiu, Jian Sun, and Haneen Farah. Safety performance boundary identification of highly automated vehicles: A surrogate model-based gradient descent searching approach. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[26] Jack Olsson and Michael W. Levin. Integration of microsimulation and optimized autonomous intersection management. *Journal of Transportation Engineering, Part A: Systems*, 146, 9 2020.

[27] Vindula Jayawardana and Cathy Wu. Learning eco-driving strategies at signalized intersections. 4 2022.

[28] Razvan Solea and Urbano Nunes. Trajectory planning with velocity planner for fully-automated passenger vehicles. pages 474–480. Institute of Electrical and Electronics Engineers Inc., 2006.

[29] David González, Joshué Pérez, and Vicente Milanés. Parametric-based path generation for automated vehicles at roundabouts. *Expert Systems with Applications*, 71:332–341, 4 2017.

[30] Peng Hang, Chao Huang, Zhongxu Hu, Yang Xing, and Chen Lv. Decision making of connected automated vehicles at an unsignalized roundabout considering personalized driving behaviours. *IEEE Transactions on Vehicular Technology*, 70:4051–4064, 5 2021.

[31] Narayana Raju and Haneen Farah. Evolution of traffic microsimulation and its use for modeling connected and automated vehicles. *Journal of Advanced Transportation*, 2021, 2021.

[32] Qiong Lu, Tamás Tettamanti, Dániel Hörcher, and István Varga. The impact of autonomous vehicles on urban traffic network capacity: an experimental analysis by microscopic traffic simulation. *Transportation Letters*, 12:540–549, 9 2020.

[33] Amirhosein Karbasi and Mahmoud Saffarzadeh. Impact of connected and automated vehicles on capacity of roundabout using idm car following model. 2020.

[34] Stefan Krauß. *Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics*. PhD thesis, 1998.

[35] Ding wei Huang. Phase diagram of a traffic roundabout. *Physica A*, 383:603–612, 2007.

[36] A Nakayama, K Hasebe, and Y Sugiyama. Optimal velocity model and its applications.

[37] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama. Dynamical model of traffic congestion and numerical simulation. *Physical Review E*, 51:1035–1042, 1995.

[38] P G Gipps. A behavioural car-following model for computer simulation, 1981.

[39] Biagio Ciuffo, Vincenzo Punzo, and Marcello Montanino. Thirty years of gipps' car-following model. *Transportation Research Record*, pages 89–99, 12 2012.

[40] Guanghui Liu. W99 car following model - how it works. http://w99demo.com (18-11-2022).

[41] Hesham Rakha and Yu Gao. Calibration of steady-state car-following models using macroscopic loop detector data final report prepared for virginia transportation research council, 2010.

[42] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 62:1805–1824, 2000.

[43] Dominik Salles, Stefan Kaufmann, and Hans-Christian Reuss. Extending the intelligent driver model in sumo and verifying the drive off trajectories with aerial measurements, 2020.

[44] Martin Treiber and Arne Kesting. *Traffic flow dynamics. Data, models and simulation.* 2013.

[45] Martin Treiber, Arne Kresting, and Dirk Helbong. Delays, inaccuracies and anticipation in microscopic traffic moels. *Physica A*, 360:71–88, 2005.

[46] Arne Kesting, Martin Treiber, and Dirk Helbing. Enhanced intelligent driver model to access the impact of driving strategies on traffic. *Source: Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 368:4585–4605, 2010.

[47] Jianping Wu, Mark Brackstone, and Mike Mcdonald. Fuzzy sets and systems for a motorway microscopic simulation model. *Fuzzy Sets and Systems*, 116:65–76, 2000.

[48] Tomer Toledo, Haris N. Koutsopoulos, and Moshe Ben-Akiva. Estimation of an integrated driving behavior model. *Transportation Research Part C: Emerging Technologies*, 17:365–380, 2009.

[49] Samer H. Hamdar, Vinayak V. Dixit, Alireza Talebpour, and Martin Treiber. A behavioral microeconomic foundatioin for car-following models. *Transportation Research Part C*, 2019.

[50] Gowri Asaithambi, Venkatesan Kanagaraj, and Tomer Toledo. Driving behaviors: Models and challenges for non-lane based mixed traffic. *Transportation in Developing Economies*, 2, 10 2016.

[51] Vicente Milanés and Steven E. Shladover. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. *Transportation Research Part C: Emerging Technologies*, 48:285–300, 11 2014.

[52] Boris S. Kerner. Criticism of generally accepted fundamentals and methodologies of traffic and transportation theory: A brief review. *Physica A*, 329:5261–5282, 2013.

[53] Sara Moridpour, Majid Sarvi, and Geoff Rose. Lane changing models: A critical review. *Transportation Letters*, 2:157–173, 2010.

[54] P G Gipps. A model for the structure of lane-changing decisions. *Transportation Research B*, 20:403–414, 1986.

[55] Peter Hidas. Modelling lane changing and merging in microscopic traffic simulation. *Transportation Research Part C*, 10:351–271, 2002.

[56] Ping Sun, Xuesong Wang, and Meixin Zhu. Modeling car-following behavior on freeways considering driving style. *Journal of Transportation Engineering, Part A: Systems*, 147, 12 2021.

[57] Martin Treiber and Ame Kesting. An open-source microscopic traffic simulator. *IEEE Intelligent Transportation Systems Magazine*, 2:6–13, 9 2010.

[58] J. Ludmann, D. Neunzig, and M. Weilkes. Traffic simulation with consideration of driver models, theory and examples. *Vehicle System Dynamics*, 27:491–516, 1997.

[59] Michal Maciejewski. A comparison of microscopic traffic flow simulation systems for an urban area. *Transport Problems*, 5, 2010.

[60] Sadullah Goncu, Ismet Goksad Erdagi, Mehmet Ali Silgu, and Hilmi Berk Celikoglu. Analysis on effects of driving behavior on freeway traffic flow: A comparative evaluation of two driver profiles using two car-following models. volume 2022-June, pages 688–693. Institute of Electrical and Electronics Engineers Inc., 2022.

[61] Duo Zhang, Xiaoyun Chen, Junhua Wang, Yinhai Wang, and Jian Sun. A comprehensive comparison study of four classical car-following models based on the large-scale naturalistic driving experiment. *Simulation Modelling Practice and Theory*, 113, 12 2021.

[62] Vincenzo Punzo, Marcello Montanino, and Biagio Ciuffo. Do we really need to calibrate all the parameters? variance-based sensitivity analysis to simplify microscopic traffic flow models. *IEEE Transactions on Intelligent Transportation Systems*, 16:184–193, 2 2015.

[63] Aleksandra Deluka Tibljaš, Tullio Giuffrè, Sanja Surdonja, and Salvatore Trubia. Introduction of autonomous vehicles: Roundabouts design and safety performance evaluation. *Sustainability (Switzerland)*, 10, 4 2018.

[64] Martin Treiber and Venkatesan Kanagaraj. Comparing numerical integration schemes for time-continuous car-following models. *Physica A: Statistical Mechanics and its Applications*, 419:183–195, 2 2015.

[65] German Aerospace Center (DLR) et al. Sumo documentation, 2022. Accessed December 12, 2022. `https://sumo.dlr.de/docs/index.html`.

[66] Jakob Erdmann. Sumo's lane-changing model.

[67] Jakob Erdmann and Daniel Krajzewicz. Sumo's road intersection model, 2011.

[68] Vincezo Punzo, Constantinos Antoniou, Jaume Barcelo, and Mark Brackstone. Traffic simulation: Case for guidelines, 2014.

[69] Elmar Brockfeld, Reinhart D Kühne, and Peter Wagner. Calibration and validation of microscopic models of traffic flow. *Journal of the Transportation Research Board*, pages 179–187, 2005.

[70] Palak Maheshwary, Bhattacharyya Kinjal, Maitra Bhargab, and Boltze Manfred. A methodology for calibration of traffic micro-simulator for urban heterogeneous traffic operations. *Journal of Traffic and Transporttation Engineering*, 7:507–519, 2020.

[71] Bryan Higgs, Virginia Tech, vtedu M Montasir Abbas, and Alejandra Medina. Analysis of the wiedemann car following model over different speeds using naturalistic data. 2011.

[72] Johannes Asamer, Henk Van Zuylen, Bernhard Heilmann, and Henk J Van Zuylen. Calibrating vissim to adverse weather conditions. 2011.

[73] Meixin Zhu, Xuesong Wang, Andrew Tarko, and Shou'en Fang. Modeling car-following behavior on urban expressways in shanghai: A naturalistic driving study. *Transportation Research Part C: Emerging Technologies*, 93:425–445, 8 2018.

[74] Rebecca L. Olson, Richard Joseph Bocanegra, J. Hanowski, and Jeffrey S. Hickman. Driver distraction in commercial operations, 2009.

[75] Vicki L. Neale, Thomas A. Dingus, Sheila G. Klauer, and Jeremy Sudweeks. An overview of the 100-car naturalistic study and findings, 2005.

[76] FHWA. Us highway 101 dataset, 2007.

[77] Steven Smith. Freeway data collection for studing vehicle interactions - technical report, 1985.

[78] F. Clara Fang and Hernan Castaneda. Computer simulation modeling of driver behavior at roundabouts. *International Journal of Intelligent Transportation Systems Research*, 16:66–77, 1 2018.

[79] James W. S. Longhurst and C. A. Brebbia. *Urban transport XVIII : urban transport and the environment in the 21st century*. WIT Press, 2012.

[80] Catur Yudo Leksono Tina Andriyana. Roundabout microsimulation using sumo a case study in idrottsparken roundabout norrkoping, sweden, 2012.

[81] Richard Dowling, Alexander Skabardonis, and Vassili Alexiadis. Traffic analysis toolbox volume III: Guidelines for applying traffic microsimulation modeling software, 2004.

[82] Ankit Anil Chaudhari, Karthik K. Srinivasan, Bhargava Rama Chilukuri, Martin Treiber, and Ostap Okhrin. *Calibrating Wiedemann-99 Model Parameters to Trajectory Data of Mixed Vehicular Traffic*, volume 2676. SAGE Publications Ltd, 2022.

[83] Umair Durrani and Chris Lee. Calibration and validation of psychophysical car-following model using driver's action points and perception thresholds. *Journal of Transportation Engineering, Part A: Systems*, 145, 9 2019.

[84] Qiao Ge and Monica Menendez. Exploring the variance contributions of correlated model parameters: A sampling-based approach and its application in traffic simulation models. *Applied Mathematical Modelling*, 97:438–462, 9 2021.

[85] Kayvan Aghabayk, Majid Sarvi, William Young, and Lukas Kautzsch. A novel methodology for evolutionary calibration of vissim by multi-threading, 2013.

[86] Hesham Rakha and Yu Gao. Calibrtion of steady-state car-following models using macroscopic loop detector data, 2010.

[87] Serge P. Hoogendoorn and Raymond Hoogendoorn. Generic calibration framework for joint estimation of car-following models by using microscopic data. *Transportation Research Record*, pages 37–45, 12 2010.

[88] Chris van Hinsbergen, Hans van Lint, Serge Hoogendoorn, and Henk van Zuylen. Bayesian calibration of car-following models. 2009.

[89] Gábor Pótári, Zoltán Vincellér, and Zoltán Ács. A method for real-time adaptation of weather conditions within a traffic simulation. pages 41–47. University of Debrecen/ Debreceni Egyetem, 5 2015.

[90] Miguel Catalan-Cid et al. A secure and reusable artificial intelligence platform for edge computing in beyond 5g networks. d2.2 preliminary assessment of system architecture, interfaces specifications, and techno-economic analysis, 2022.

[91] Babak Mafakheri et al. Leonardo Goratti. A secure and reusable artificial intelligence platform for edge computing in beyond 5g networks. d5.1 testing and validation methodology, planning and preparation, 2022.

[92] Transport for London. Traffic modelling guidelines, 2021.

[93] Manuel G Romana Emilio Moreno González and Oscar Martínez Álvaro. Bus dwell-time model of main urban route stops: case study in madrid, spain. *Transportation research record*, 2274:126–134, 2012.

[94] Taqwa AlHadidi and Hesham A. Rakha. Transportation research interdisciplinary perspectives. *Transportation research record*, 2019.

A | Appendix A - Templates for the collection of data

Codice Posizione: N2

Giorno:

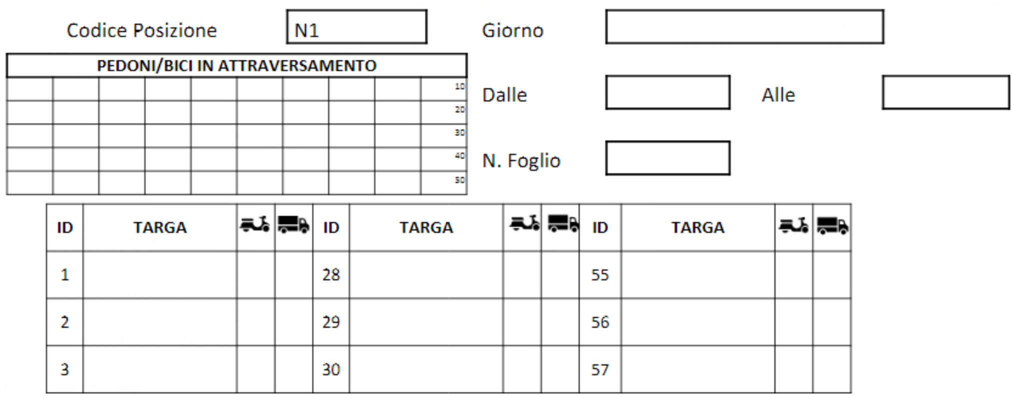
Dalle: Alle:

N. Foglio:

ID	TARGA			ID	TARGA			ID	TARGA		
1				28				55			
2				29				56			
3				30				57			
4				31				58			
5				32				59			
6				33				60			
7				34				61			
8				35				62			

CODA MEDIA	CODA MAX
ATTESTAMENTO	
PALESTRA	

Figure A.1: Data acquisition template entry



Codice Posizione N1 Giorno

PEDONI/BICI IN ATTRAVERSAMENTO

									10
									20
									30
									40
									50

Dalle Alle

N. Foglio

ID	TARGA			ID	TARGA			ID	TARGA		
1				28				55			
2				29				56			
3				30				57			

Figure A.2: Data acquisition template exit

B | Appendix B - OD matrices and SUMO demand

OD matrices for cars

PERIOD 1					
OD	E	N	S	W	Total
E	0	7	24	36	67
N	21	1	56	7	85
S	20	50	4	8	82
W	31	0	13	0	44
Total	72	58	97	51	278

PERIOD 2					
OD	E	N	S	W	Total
E	0	6	39	34	79
N	9	0	59	11	79
S	18	27	1	8	54
W	26	5	9	1	41
Total	53	38	108	54	253

PERIOD 3					
OD	E	N	S	W	Total
E	0	6	28	33	67
N	18	1	66	6	91
S	17	40	5	7	69
W	26	0	15	0	41
Total	61	47	114	46	268

PERIOD 4					
OD	E	N	S	W	Total
E	0	4	20	41	65
N	17	1	46	8	72
S	16	29	3	9	57
W	25	0	10	0	35
Total	58	34	79	58	229

PERIOD 5					
OD	E	N	S	W	Total
E	0	5	23	34	62
N	17	1	54	6	78
S	17	38	4	8	67
W	26	0	12	0	38
Total	60	44	93	48	245

PERIOD 6					
OD	E	N	S	W	Total
E	0	5	28	43	76
N	14	1	66	8	89
S	13	38	5	10	66
W	21	0	15	0	36
Total	48	44	114	61	267

Table B.1: OD matrices for cars

OD matrices for motorcycles

PERIOD 1					
OD	E	N	S	W	Total
E	0	0	2	24	26
N	2	0	5	4	11
S	2	4	0	5	11
W	2	0	1	0	3
Total	6	4	8	33	51

PERIOD 2					
OD	E	N	S	W	Total
E	0	0	2	29	31
N	3	0	5	5	13
S	3	4	0	7	14
W	4	0	1	0	5
Total	10	4	8	41	63

PERIOD 3					
OD	E	N	S	W	Total
E	0	2	2	32	36
N	4	0	5	6	15
S	4	9	0	7	20
W	6	0	1	0	7
Total	14	11	8	45	78

PERIOD 4					
OD	E	N	S	W	Total
E	0	0	2	30	32
N	3	0	4	5	12
S	3	1	0	7	11
W	4	0	1	0	5
Total	10	1	7	42	60

PERIOD 5					
OD	E	N	S	W	Total
E	0	0	3	26	29
N	2	0	6	5	13
S	1	2	0	6	9
W	2	0	1	0	3
Total	5	2	10	37	54

PERIOD 6					
OD	E	N	S	W	Total
E	0	0	3	13	16
N	2	0	6	2	10
S	2	2	0	3	7
W	4	0	1	0	5
Total	8	2	10	18	38

Table B.2: OD matrices for motorcycles

OD matrices for bicycles

PERIOD 1					
OD	E	N	S	W	Total
E	0	1	2	12	15
N	0	0	3	2	5
S	0	5	0	3	8
W	0	0	1	0	1
Total	0	6	6	17	29

PERIOD 2					
OD	E	N	S	W	Total
E	0	1	1	13	15
N	1	0	2	2	5
S	1	4	0	3	8
W	1	0	1	0	2
Total	3	5	4	18	30

PERIOD 3					
OD	E	N	S	W	Total
E	0	1	2	14	17
N	1	0	5	3	9
S	1	5	0	3	9
W	2	0	1	0	3
Total	4	6	8	20	38

PERIOD 4					
OD	E	N	S	W	Total
E	0	0	2	16	18
N	1	0	5	3	9
S	1	3	0	4	8
W	2	0	1	0	3
Total	4	3	8	23	38

PERIOD 5					
OD	E	N	S	W	Total
E	0	1	2	6	9
N	1	0	4	1	6
S	1	6	0	1	8
W	1	0	1	0	2
Total	3	7	7	8	25

PERIOD 6					
OD	E	N	S	W	Total
E	0	0	3	12	15
N	1	0	6	2	9
S	1	3	0	3	7
W	2	0	1	0	3
Total	4	3	10	17	34

Table B.3: OD matrices for bicycles

OD matrices for heavy vehicles

PERIOD 1					
OD	E	N	S	W	Total
E	0	0	0	2	2
N	3	0	1	0	4
S	3	1	0	0	4
W	5	0	0	0	5
Total	11	1	1	2	15

PERIOD 2					
OD	E	N	S	W	Total
E	0	0	1	4	5
N	3	0	1	1	5
S	3	3	0	1	7
W	5	0	0	0	5
Total	11	3	2	6	22

PERIOD 3					
OD	E	N	S	W	Total
E	0	0	1	3	4
N	3	0	1	1	5
S	3	1	0	1	5
W	4	0	0	0	4
Total	10	1	2	5	18

PERIOD 4					
OD	E	N	S	W	Total
E	0	0	0	2	2
N	2	0	1	0	3
S	1	1	0	0	2
W	2	0	0	0	2
Total	5	1	1	2	9

PERIOD 5					
OD	E	N	S	W	Total
E	0	0	1	3	4
N	2	0	2	1	5
S	2	1	0	1	4
W	3	0	1	0	4
Total	7	1	4	5	17

PERIOD 6					
OD	E	N	S	W	Total
E	0	0	1	2	3
N	2	0	1	0	3
S	1	3	0	0	4
W	2	0	0	0	2
Total	5	3	2	2	12

Table B.4: OD matrices for heavy vehicles

OD matrices using coefficients for motorcycles and bicycles

PERIOD 1					
OD	E	N	S	W	Total
E	0	7	25	48	80
N	22	1	59	9	90
S	21	53	4	11	88
W	32	0	14	0	45
Total	74	61	101	68	304

PERIOD 2					
OD	E	N	S	W	Total
E	0	6	40	48	94
N	10	0	61	13	85
S	19	29	1	11	61
W	28	5	10	1	43
Total	58	41	112	74	284

PERIOD 3					
OD	E	N	S	W	Total
E	0	7	29	49	85
N	20	1	69	9	99
S	19	45	5	10	79
W	29	0	16	0	44
Total	67	53	119	68	307

PERIOD 4					
OD	E	N	S	W	Total
E	0	4	21	56	81
N	18	1	49	11	79
S	17	30	3	13	63
W	27	0	11	0	38
Total	63	35	83	79	261

PERIOD 5					
OD	E	N	S	W	Total
E	0	5	25	46	75
N	18	1	57	8	84
S	18	40	4	11	72
W	27	0	13	0	40
Total	63	46	98	64	272

PERIOD 6					
OD	E	N	S	W	Total
E	0	5	30	51	85
N	15	1	70	9	95
S	14	39	5	12	70
W	23	0	16	0	39
Total	52	45	120	72	289

Table B.5: OD matrices using coefficients for motorcycles and bicycles

This demand is represented in SUMO with a flow element per each cell of the OD matrix, representing one pair of origin destination. In the simulation the used OD matrices are the one refering to heavy vehicles (table B.4) and the combination of cars, motorcycles and bicycles using different coefficients (table B.5). Following, an example of the code representing the demand for one period is shown:

```xml
<!-- <flow id="EE_P1" type="1" begin="120" from="-e" to="e" end="720" departSpeed="avg" period="exp(0)"/> -->
<flow id="EN_P1" type="1" begin="120" fromJunction="E_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0.012)"/>
<flow id="ES_P1" type="1" begin="120" fromJunction="E_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0.042)" />
<flow id="EW_P1" type="1" begin="120" fromJunction="E_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0.08)" />
<flow id="NE_P1" type="1" begin="120" fromJunction="N_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.0363333333333333)" />
<flow id="NN_P1" type="1" begin="120" from="-n" to="n" end="720" departSpeed="avg" period="exp(0.00166666666666667)"/>
<flow id="NS_P1" type="1" begin="120" fromJunction="N_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0.0976666666666667)"/>
<flow id="NW_P1" type="1" begin="120" fromJunction="N_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0.015)"/>
<flow id="SE_P1" type="1" begin="120" fromJunction="S2_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.0346666666666667)"/>
<flow id="SN_P1" type="1" begin="120" fromJunction="S2_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0.0876666666666667)"/>
<flow id="SS_P1" type="1" begin="120" fromJunction="S2_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0.00666666666666667)"/>
<flow id="SW_P1" type="1" begin="120" fromJunction="S2_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0.0176666666666667)"/>
<flow id="WE_P1" type="1" begin="120" fromJunction="W_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.053)"/>
<!-- <flow id="WN_P1" type="1" begin="120" fromJunction="W_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<flow id="WS_P1" type="1" begin="120" fromJunction="W_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0.0226666666666667)"/>
<!-- <flow id="WW_P1" type="1" begin="120" from="-w" to="w" end="720" departSpeed="avg" period="exp(0)"/> -->

<!-- <flow id="EEh_P1" type="HEAVY" begin="120" from="-e" to="e" end="720" departSpeed="avg" period="exp(0)"/> -->
<!-- <flow id="ENh_P1" type="HEAVY" begin="120" fromJunction="E_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<!-- <flow id="ESh_P1" type="HEAVY" begin="120" fromJunction="E_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0)" /> -->
<flow id="EWh_P1" type="HEAVY" begin="120" fromJunction="E_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0.000933333333333334)" />
<flow id="NEh_P1" type="HEAVY"  begin="120" fromJunction="N_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.005)" />
<!-- <flow id="NNh_P1" type="HEAVY" begin="120" from="-n" to="n" end="720" departSpeed="avg" period="exp(0)"/> -->
<flow id="NSh_P1" type="HEAVY" begin="120" fromJunction="N_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0.00166666666666667)"/>
<!-- <flow id="NWh_P1" type="HEAVY" begin="120" fromJunction="N_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<flow id="SEh_P1" type="HEAVY" begin="120" fromJunction="S2_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.005)"/>
<flow id="SNh_P1" type="HEAVY" begin="120" fromJunction="S2_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0.00166666666666667)"/>
<!-- <flow id="SSh_P1" type="HEAVY" begin="120" fromJunction="S2_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<!-- <flow id="SWh_P1" type="HEAVY" begin="120" fromJunction="S2_S" toJunction="W_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<flow id="WEh_P1" type="HEAVY" begin="120" fromJunction="W_S" toJunction="E_S" end="720" departSpeed="avg" period="exp(0.00593333333333333)"/>
<!-- <flow id="WNh_P1" type="HEAVY"  begin="120" fromJunction="W_S" toJunction="N_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<!-- <flow id="WSh_P1" type="HEAVY" begin="120" fromJunction="W_S" toJunction="S1_S" end="720" departSpeed="avg" period="exp(0)"/> -->
<!-- <flow id="WWh_P1" type="HEAVY" begin="120" from="-w" to="w" end="720" departSpeed="avg" period="exp(0)"/> -->
```

Figure B.1: Definition of vehicle flow in SUMO

```xml
<personFlow id="Pe1" begin="120.00" end="720" period="exp(0.0183)">
    <personTrip from="-e1" to="e1" />
</personFlow>
<personFlow id="Pe1_op" begin="120.00" end="720" period="exp(0.0183)">
    <personTrip from="e1" to="-e1" />
</personFlow>
<personFlow id="Pn1" begin="120.00" end="720" period="exp(0.0358333333333333)">
    <personTrip from="-n1" to="n1" />
</personFlow>
<personFlow id="Pn1_op" begin="120.00" end="720" period="exp(0.0358333333333333)">
    <personTrip from="n1" to="-n1" />
</personFlow>
<personFlow id="Ps1" begin="120.00" end="720" period="exp(0.0341666666666667)">
    <personTrip from="-s1" to="s1" />
</personFlow>
<personFlow id="Ps2" begin="120.00" end="720" period="exp(0.0341666666666667)">
    <personTrip from="-s2" to="s2" />
</personFlow>
<personFlow id="Ps1_op" begin="120.00" end="720" period="exp(0.0341666666666667)">
    <personTrip from="s1" to="-s1" />
</personFlow>
<personFlow id="Ps2_op" begin="120.00" end="720" period="exp(0.0341666666666667)">
    <personTrip from="s2" to="-s2" />
</personFlow>
<personFlow id="Pw1" begin="120.00" end="720" period="exp(0.0141666666666667)">
    <personTrip from="-w1" to="w1" />
</personFlow>
<personFlow id="Pw1_op" begin="120.00" end="720" period="exp(0.0141666666666667)">
    <personTrip from="w1" to="-w1" />
</personFlow>
```

Figure B.2: Definition of pedestrian flow in SUMO

List of Figures

List of Tables