Technische Universität München
TUM School of Engineering and Design

Collective Perception using Roadside ITS Stations: Simulations on Traffic Safety and Efficiency of Urban Automated Driving

Mathias Pechinger

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung eines
Doktors der Ingenieurwissenschaften (Dr.-Ing.) genehmigten Dissertation

Vorsitz: Prof. Dr. Constantinos Antoniou

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Klaus Bogenberger
2. Prof. Dr.-Ing. Carsten Markgraf

Die Dissertation wurde am 11.04.2023 bei der Technischen Universität München eingereicht und durch die *TUM School of Engineering and Design* am 28.09.2023 angenommen.

Executive Summary

This thesis evaluates the incorporation of roadside Intelligent Transport Systems Stations (ITS-S) in urban road networks to enable Collective Perception (CP), which increases the field of view of Automated Vehicles (AVs) and reduces the number of Vulnerable Road User (VRU) accidents while keeping the traffic fluent and efficient. The primary research questions concerning roadside ITS-S impacts discuss the safety, reliability, and economic impact of AVs in urban areas. Safety and reliability are evaluated from a technical and scientific point of view. The financial aspect is put in perspective to enable policymakers to decide whether it is economically reasonable to utilize this technology.

A series of Hardware in the Loop (HIL) simulations is the methodology to derive significant results. A combination of a microscopic and sub-microscopic simulation was used. The microscopic part generates a realistic traffic flow, and the sub-microscopic part is used for realistic 3D environment simulations. Open-source automated driving algorithms that have proven maturity in real-world driving tests were used. This improves the reproducibility and transparency of the simulations for other research and the significance of the results.

Recent work indicates that the most significant impact of roadside ITS-S can be achieved considering the vulnerability of bicyclists. Simulation results provide evidence that collective perception improves the safety of bicycle riders. As a result, major conflicts resulting in severe accidents could be resolved if roadside ITS-S is used by a Connected Automated Vehicle (CAV). Additionally, the findings suggest that automated vehicles may disrupt traffic flow if additional sensing systems on the side of the road are not utilized at critical locations in an urban road network. AVs without the support of roadside ITS-S show an increased travel time of 19.8% in simulations evaluated in this work.

Roadside ITS-S can improve traffic efficiency and road safety in a mixed-traffic environment of human-driven vehicles and AVs. Further research in real-world tests in an urban area that implements the architecture used in this work is needed to validate the results of this thesis.

Zusammenfassung

In dieser Arbeit wird der Aufbau von straßenseitigen Infrastruktur Stationen (ITS-S) für kollektive Wahrnehmung (CP) im städtischen Straßennetz bewertet. Das Sichtfeld von AVs wurde vergrößert und dadurch die Zahl der VRU-Unfälle reduziert, während der Verkehrsfluss effizient bleibt. Die primären Forschungsfragen in Bezug auf die Auswirkungen von straßenseitiger ITS-S beziehen sich auf Sicherheit, Zuverlässigkeit und wirtschaftliche Auswirkungen von autonomen Fahrzeugen, sowie VRU Verkehrsunfälle im urbanen Raum. Sicherheit und Zuverlässigkeit werden unter technischen und wissenschaftlichen Gesichtspunkten bewertet. Der finanzielle Aspekt wird betrachtet, damit die politischen Entscheidungsträger beurteilen können, ob es wirtschaftlich sinnvoll ist, diese Technologie einzusetzen.

Eine Abfolge verschiedener Simulationen ist die Methodik, um aussagekräftige Ergebnisse zu erzielen. Es wurde eine Kombination aus einer mikroskopischen und einer submikroskopischen Simulation verwendet. Der mikroskopische Teil erzeugt einen realistischen Verkehrsfluss, und der submikroskopische Teil wird für realistische 3D-Umgebungssimulationen verwendet. Es wurden Open-Source-Algorithmen für automatisiertes Fahren verwendet, die sich in realen Fahrversuchen bewährt haben. Dies verbessert die Reproduzierbarkeit und Transparenz der Simulationen für andere Forschungsarbeiten und die Aussagekraft der Ergebnisse.

Jüngste Daten und Ergebnisse deuten darauf hin, dass der maximale Nutzen von straßenseitiger Infrastruktur bei der Betrachtung von Radfahrerschutz erzielt werden kann. Die Simulationsergebnisse belegen, dass die kollektive Wahrnehmung die Sicherheit von Fahrradfahrern verbessert. Folglich konnten schwere Unfälle verhindert werden, wenn eine straßenseitige Infrastruktur von einem automatisierten Fahrzeug eingesetzt wurde. Darüber hinaus deuten die Ergebnisse darauf hin, dass automatisierte Fahrzeuge den Verkehrsfluss stören können, wenn an kritischen Stellen in einem städtischen Straßennetz keine zusätzlichen Erfassungssysteme am Straßenrand eingesetzt werden. AVs ohne die Unterstützung von straßenseitiger ITS-S zeigen in Simulationen, die in dieser Arbeit durchgeführt wurden, eine erhöhte Fahrzeit von 19,8%.

Straßenseitige ITS-S können die Verkehrseffizienz und die Verkehrssicherheit in einer gemischten Verkehrsumgebung aus von Menschen geführten Fahrzeugen und AVs verbessern. Weitere Untersuchungen in realen Tests in einem städtischen Gebiet, welche die in dieser Arbeit verwendete Architektur implementieren, sind erforderlich, um die Ergebnisse zu validieren.

Contents

Chapter 1

Introduction

Automated driving is a state-of-the-art challenge investigated by the world's leading scientists, who are taking on related questions from different perspectives. In addition to the scientific community, the industry is working on solutions to bring this powerful technology to the market. There are difficult questions to answer, like the ethics of programming AVs or even the feasibility of this technology. Nevertheless, AVs have unique potential for the economy, ecology, and society. If we can solve these problems, we will advance toward a better future.

1.1 Context

One of the most pressing challenges for today's society is the climate of our planet. CO_2 emissions influence climate, and the land transport sector is one of the significant emitters, responsible for 20.19% of global CO_2 emissions [Statista Research Department, 2022] and one of the hardest to decarbonize [Giannakis et al., 2020]. Optimizing the traffic flow can reduce the overall energy consumption of urban transportation. An example is the utilization of green wave scenarios, which can reduce travel time as well as energy consumption of vehicles [Dong et al., 2019]. Utilizing a 100% penetration rate of AVs might significantly reduce induced CO2 emissions by urban transportation.

One can spot another practical use for AVs by looking at the worlds' demography. The population has passed its so-called "peak child," and their "demographic dividend" is disbursed [Hannah Ritchie and Max Roser, 2017]. Germany is an example of a country that surpassed its "peak child" several years ago. Furthermore, the number of people older than 67 will increase by 22% from 2020, with 16 million, up to 20 million people in 2035 [Federal Office of Statistics Germany, 2021]. This age group relies on mobility as a matter of their quality of life. AVs can improve the access of the older population to mobility at a point where they do not feel comfortable anymore, driving on their own.

Concerning improving access to mobility for older people, one must also consider the demands of disabled people. The "Committee on the Rights of Persons with Disabilities" of the United Nations (UN) formulated the following request. In "Convention on the Rights of Persons with Disabilities": "States Parties shall take effective measures to ensure personal mobility with the greatest possible independence for persons with disabilities (Article 20)" [Committe on the Rights of Persons with Disabilities, 2021]. Additionally, state parties need to promote and perform research in favor of the mobility of disabled people.

Another primary task is finding a solution for AVs that meets the requirement of driving safely and reliably. Therefore, AVs should be compliant with the International Organization

for Standardization (ISO) standard number 26262, functional safety for road vehicles [International Organization for Standardization, 2018] (ISO 26262). Even though their developers have evaluated automated test vehicles for several years, one still struggles to fulfill major guidelines. For example, in the product development section of ISO 26262 at the hardware level, a random hardware failure target value below 10^{-8} critical failures per hour is defined. Applying the random hardware failure of a motion control computer that generates the cars' trajectory is equivalent to a single car driving 1.2 times 10^{10} km without a critical failure at a mean velocity of 120 km/h (1.1). Suppose this measurement is applied to an autonomous system that must drive without driver interaction. In that case, one can quickly realize how hard the path to Society of Automotive Engineers (SAE) Level 5 [SAE International, 2014] fully automated driving will be.

$$\frac{v}{FailureRate} = \frac{120\,km/h}{10^{-8} failure/hour} = 1.2 \cdot 10^{10}\,km/failure \tag{1.1}$$

Besides random hardware failure, ISO26262 covers software development safety standards regarding e.g. Verification and Validation (V&V). Another aspect is Safety of The Intended Functionality (SoTIF) [International Organization for Standardization, 2019] of software running on a motion control computer. A standardized software homologation process for automated driving must be developed.

The number of car accidents rises with the amount of fully automated driven kilometers, as the reaction time of a safety driver to a system failure increases with the distance he travels with the car [Dixit et al., 2016]. Considering this relationship, another fallback for the driver is indispensable. Another solution is needed if the automated vehicle cannot perform the motion control task and the safety drivers' reaction time is too slow. Dixit et al. [2016] suggests, among other possibilities, providing adequate roadway infrastructure.

In a white paper Safety First for Automated Driving (SaFAD) written by established OEMs, namely Audi, BMW, Daimler, VW, and additional technology partners like Intel or Continental, steps were defined for developing and validating a safe automated driving system [Wood et al., 2019]. SaFAD addresses the whole audience involved in the future of automated driving. Among those are the press and media, regulatory officials, individuals from the AV industry, insurance companies, and people involved in standardizing this technology. They point out that it will be impossible to guarantee 100% safety concerning current V&V standards. One must instead try to achieve a so-called "positive risk balance," meaning that the AV achieves fewer hazards and crashes than the average human driver.

Automated driving has the potential to help to cope with climate change, societal challenges, and road safety. The next question is how roadside ITS-S can be used in this context and the motivation to use such systems.

1.2 Motivation

This dissertation deals with the broad question: How can roadside infrastructure contribute to the future of automated driving? Additional perception sensors are placed at the side of the road as roadside infrastructure. Perceived objects are transmitted to AVs in real-time, which can include these in their driving algorithms. Essential topics are outlined by safety and reliability. In automated driving, safety is measured by traffic accidents or critical situations encountered by an AV. For reliability aspects, two factors are considered. One is the reliability of the perception systems of the ego vehicle itself, which an additional sensor outside the car can increase. The other factor is the reliability of the macroscopic traffic flow of an urban road network.

Increasing either safety or reliability may lead to a reduction of the other. A fundamental example is the velocity of a vehicle. Considering a cost function designed to minimize accident fatality, the lower the speed of a car, the lower the fatality probability of an accident. If one approaches zero meters per second or a standstill, accident fatality will come to its minimum. This setup achieves maximum safety for an AV in a road network. Considering traffic efficiency or flow while the vehicles approach a standstill, traffic efficiency will be at its worst, as safety and reliability usually influence each other in an opposing manner.

The utilization of roadside infrastructure can improve automated driving safety as well as reliability. This work explores to which extent Road Side Infrastructure (RSI) can help AVs in specific urban road networks to complete both goals: Safety and Reliability. Additionally, an analysis of economic feasibility is given.

1.2.1 Safety

For automated driving safety, safety hazards must be identified in the first place. Understanding the cause of an AVs' problem enables work towards practical solutions. To understand the safety aspect, the latest driving statistics of AVs (i), historical data from human-driven vehicles (ii), and reasons for the issues elaborated in driving statistics and historical data are considered (iii).

i) The "Crash and disengagement data of autonomous vehicles on public roads in California" [Sinha et al., 2021] summarizes crashes and disengagements of AVs from 2014 until 2019. Or in the case of crashes only, until March 2020. This data grants insights into causes for disengagements and influencing factors for crashes. The work provides valuable information on locations where AVs are prone to safety issues. Additionally, conflict parties like VRUs or cars are mentioned in this report. Here a tendency for more crashes to happen inside intersections (63%) can be observed. Regarding intersections, it is interesting to see the share between signalized and unsignalized intersections. At signalized intersections, AVs experienced 49% of the recorded crashes compared to the unsignalized ones (51%), showing no inclination towards either of those. One might think that "simple" four-way crossings should lead to fewer incidents than more complex ones. Nevertheless, common "simple" crossings make up their share with 59%. Two-wheeler-like bicycles take part in 12% of crashes.

ii) Staubach [2010] elaborates in his thesis on factors influencing human driving errors to see possible issues for driver assistance systems. Regarding occlusion, he shows that a significant impact is parked vehicles. A total of 56% of accidents at intersections result from occlusion by parked or stopped vehicles, followed by 28% as a result of buildings and planting of vegetation. The summary of his results are shown in Figure 1.1. In addition to the mentioned causes, the ego vehicle itself with its pillars and windows can occlude the Field of View (FoV) of a human driver, creating blind spots. This is indicated by the bar named ego vehicle in Figure 1.1.

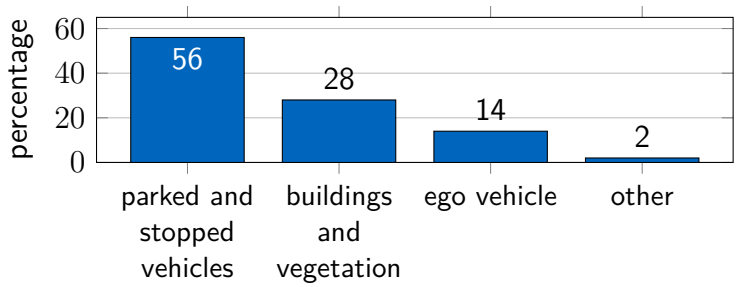


Figure 1.1: Distribution of causes for accidents at intersections

From [Gerstenberger, 2015], we can derive who is involved in accidents regarding urban intersections. His evaluation is based on German In-Depth Accident Study (GIDAS) data about accidents in urban intersections in Dresden and Hannover, Germany. Figure 1.2 illustrates the distribution of accident perpetrators and crash victims. The highest amount of crashes are observed for car/bike and car/car mishaps.

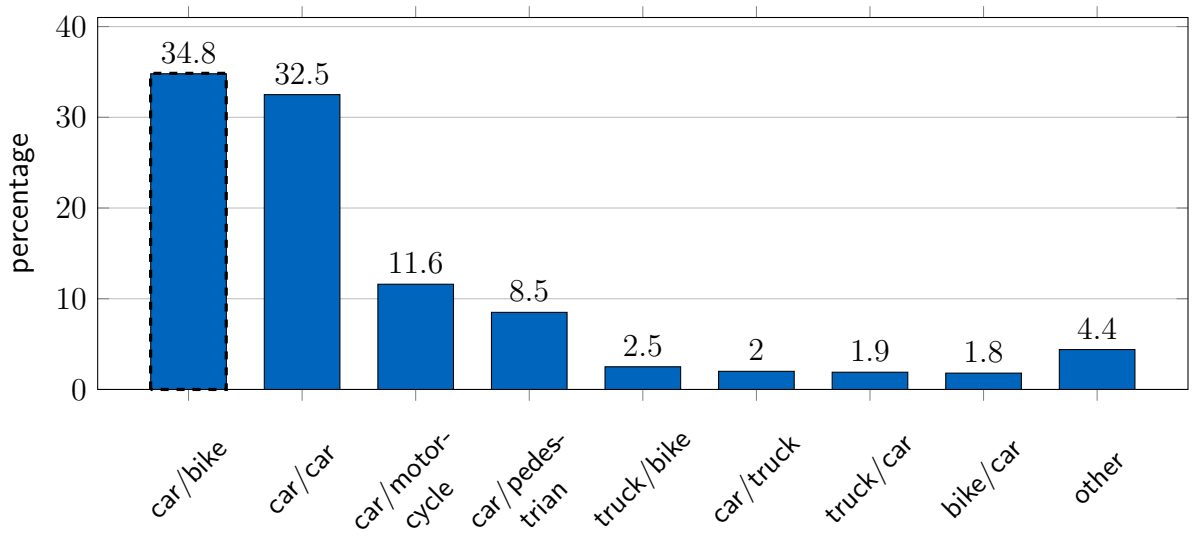


Figure 1.2: Distribution of the combination of different road users in an accident regarding urban intersections (accident perpetrator/crash victim).

iii) Overall Gerstenberger concludes in his work that only 10% of all driveways are not imposed by reduced visibility to neighbor roads. Most accidents happen at intersections and T-junctions. He strongly suggests occlusions as a significant driver for conflicts in urban driving.

For human-driven vehicles, roundabouts and intersections with more than four driveways play a minor role. Looking at the AV crash data, we can see the same tendency. Common intersections pose a slightly higher risk for crashes.

Comparing human-driven vehicles and Automated Vehicles, it is still being determined whether AVs will make the same mistakes as humans. Looking at data from both, one can say that common urban intersections show a fair likelihood of possible issues for AVs. Table 1.1 shows Gerstenberger's findings on reducing major accidents using roadside infrastructure and driver assistance systems. This illustrates well that driver assistance systems can resolve most crash issues. An actual benefit may be achieved by looking at occlusion and roadside infrastructure utilization.

Table 1.1: Summary about where infrastructure and driver assistance systems are beneficial for road safety from [Gerstenberger, 2015]. Infrastructure has an advantage considering occlusion, highlighted by the green "+" symbol.

Cause	Infrastructure	Assistance systems
Wrong attention focus	o	++
Occlusion	++	+
Distraction	+	+
Misjudged distance/velocity	o	++
Violation of priority rule	+	+

Based on the reviewed literature, one can conclude the following. If RSI is used to aid the operation of automated vehicles, increasing the FoV of an AV may be most advantageous. Additionally, the extension should cover occlusions affecting the VRU group of cyclists. Using information from the given literature, the most reasonable starting point for implementing roadside infrastructure is an urban intersection with occluded cyclists.

1.2.2 Reliability

In the context of this thesis, reliability is reflected as a reliable driving AV (i) or a reliable traffic flow (ii). An unreliable AV will ultimately impact the traffic flow. Therefore this work considers the impact of roadside infrastructure on the road networks' traffic efficiency.

i) The reliability of an AV has several meanings. It is vital to ensure an object is detected reliably with a low false positive rate under different circumstances. For example, a vehicle's sensor setup might have difficulties detecting an obstacle from a specific angle, whereas high-mounted roadside infrastructure sensors have a good view of the situation. Multiple systems must perform reliably in order for the system to operate with high reliability.

ii) The impact of an malfunctioning vehicle on the traffic flow is severe. This applies to highways and urban areas. If an automated vehicle fails to operate correctly and would therefore block one lane of a major city network, the impact on the whole transportation system is high. RSI may suitable to mitigate such risks.

Regarding the reliability of the AVs' perceptions system (i), the statistical detection accuracy can be increased using multiple sensors. In an actual application, sensors will have a certain chance of detecting an object with a specific accuracy. This detection accuracy is summarized within the Mutli Object Tracking Accuracy (MOTA) [Bernardin and Stiefelhagen, 2008]. Considering several sensors with a MOTA of a defined percentage, one can derive a formulation under the assumption that detection probabilities are independent for different sensors. Using Equation 1.2, the statistical accuracy for detecting an object by the AV itself using its accuracy $MOTA_{AV}$ and input from additional observing infrastructure sensors, with their accuracy $MOTA_{Infra}$ is evaluated. Variable n defines the number of infrastructure sensors.

$$MOTA(n) = 1 - (1 - MOTA_{AV}) \cdot (1 - MOTA_{Infra.})^n \tag{1.2}$$

Figure 1.3 highlights the detection accuracy improvement using several sensors. The number of sensors is shown on the x-axis, and the accuracy is on the y-axis. There is no differentiation between $MOTA_{AV}$ and $MOTA_{Infra.}$ for Figure 1.3, as Equation 1.2 is evaluated using a different amount of sensors utilizing the same MOTA value. The illustration shows the results for this setup. The right side is a zoomed-in version of the left side.

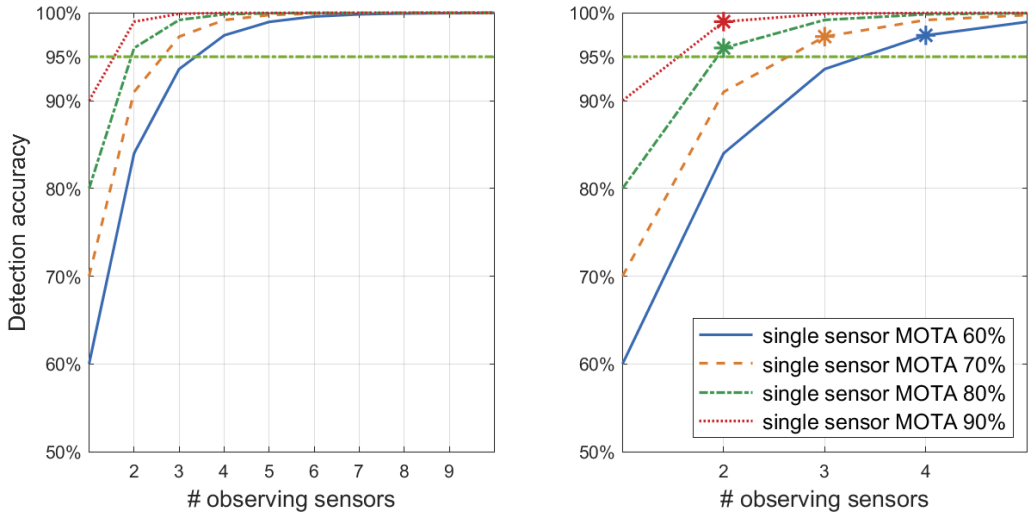


Figure 1.3: Analysis of using several sensors and the statistical increase of the chance of detecting an object where $MOTA_{AV}$=$MOTA_{Infra.}$

On the left side of Figure 1.3, the exponential increase indicated by Equation 1.2 is seen. If one assumes that 95% accuracy is good enough for perception systems, either one infrastructure sensor with a $MOTA$ of 80% or 90% would achieve proper operation. This is shown by the red (on the dotted line) and the green (on the dash-dotted line) star-shaped markers on

the right side. With a decrease in accuracy, the amount of sensors increases, as shown by the dashed and solid lines. This example shows the significant improvement of a single sensor on the statistical detection accuracy, even though 95% accuracy of perception systems may not be enough for safe AV operation.

One could argue that using two sensors on an AV will achieve the same result instead of utilizing infrastructure perception systems. Nevertheless, having two sensors at the same spot running the same algorithm may result in a common-cause error or a common dependence of the detection probability. Observing an object using a different approach, especially from another physical point of view, separated from the AV, is not prone to the AVs' common cause issue and may be considered with an independent detection probability.

1.2.3 Economic Prospects

Looking at the economic prospects for AVs, several aspects must be considered. One is producing and selling these vehicles for individuals (i). Another is building a fleet of AVs and offering shared mobility as a service (ii). Moreover, one can consider the economic impact of driving accidents (iii).

i) Vehicles have been sold to individuals for several years. This is state-of-the-art for OEMs all around the globe. Commercial car manufacturers build and sell cars to individuals based on their specific needs and preferences.

ii) Shared mobility has increased over the last few years. Shared vehicle fleets are growing and are being accepted on a broader scale. Especially for AVs, shared mobility, like sharing an AV by several people, may become even more interesting from an economic point of view.

iii) Driving accidents significantly impact the economy: "Costs of accidents make up an essential part of the total external cost of traffic" [Blaeij et al., 2003]. With the high traffic volume, these costs make up a high prospect for improvement if it is assumed that AVs will decrease the overall amount of accidents.

For the economic perspective, this work emphasizes the impact of roadside infrastructure on driving accidents. The deployment of roadside infrastructure is expensive, and this cost must be carefully weighed against the economic benefit. Infrastructure is usually funded by the government and therefore paid by the whole society. One must consider what benefits society can derive from the technology they indirectly pay for. Looking at safety, this comes down to an ethically tricky problem. Fatal traffic accidents result in the decease of a human being[1]. In a theoretical approach, the number may be quantified using the Value Of Statistical Live (VOSL). Nevertheless, it cannot be calculated easily, as elaborated by Blaeij et al. [Blaeij et al., 2003]. Therefore, this work considers the European Union's rule regarding the analysis of safety cost-benefits considering euros per human life. This rule is called "one-million-euro rule", given in [Marc Despontin et al., 1997]. A more detailed analysis is given in Chapter 4.

[1] This work explicitly does not consider ethical questions.

1.3 Structure of this Thesis

Figure 1.4 is designed to give an overview about the structure of this dissertation. After the introduction, current state of the art is discussed within Chapter 2. State of the Art is divided into the subsections CP, HIL testing, and the automated vehicle itself. Subsequently, the research gap is described, and Research Questions (RQs) are given.

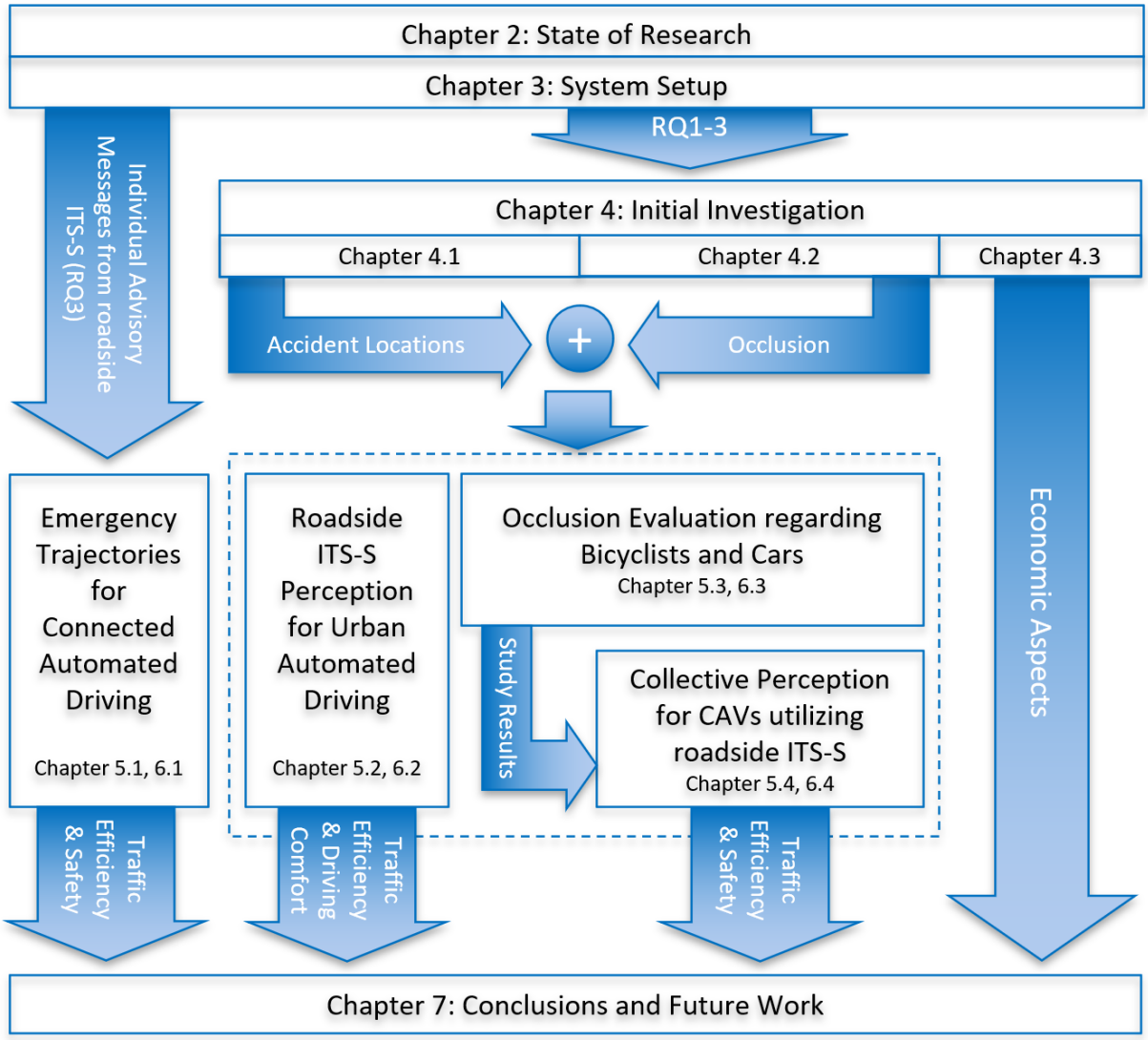


Figure 1.4: An overview introducing the structure of this thesis

Chapter 3 describes the setup of the simulation system and explains three parts in detail. First is the HIL system setup architecture, second is the automated vehicle setup and, third, the roadside ITS-S solution, used in this work. Before central simulation studies are introduced, a initial investigation about accidents and occlusion in urban areas is discussed in Chapter 4. This study is threefold and considers accident locations in Munich, occlusion in urban areas and economic aspects of roadside ITS-S.

Chapter 5 and 6 are linked closely. While Chapter 5 explains the simulations, in Chapter 6 the results of these simulations are evaluated and discussed. With reference to Figure 1.4, e.g., the left most box describes the study about "Emergency Trajectories for Connected Automated Driving." Its input is based on RQ3 from the state of the art and its results deal with traffic efficiency, indicated by a blue arrow that departs the box on the lower end.

The whole work is summarized in Chapter 7 "Conclusions and Future Work", including a summary of this work's main contributions and limitations. Finally, an outlook toward future work is given.

Chapter 2

State of the Art

This Section is structured as follows. First, a state-of-the-art AV design is described. Including an overview of naming conventions, SAE automated driving levels, and vehicle testing procedures such as HIL testing. Second, the current standards and related work on CP is introduced. This is summarized by the definition of the research gap leading to the research questions of this thesis.

2.1 Automated Vehicle

Replacing the various capabilities of a human driver is a challenge. Even if one can sense the ego AVs' environment to some degree of certainty, the response to the perceived environment is hard to evaluate. One can argue that AVs may be accomplished using an implementation based on a finite number of states the vehicle must incorporate, but this cannot be applied to "chaotic" urban driving situations. A finite state approach may be feasible in structured environments utilizing a closed space like a highway. In this case, variables that must be considered could be limited to a minimum by removing VRUs from the system. When AVs encounter pedestrians, bikes, animals, or unknown object types, a simple finite state approach must handle an intractable number of states.

To understand AVs, SAE levels of automated driving are introduced. Then HIL testing of AVs is discussed, including an introduction to automotive development standards. Next, the concept of sense, plan, and act with a more in-depth view of the current status of each sub-category is described.

2.1.1 SAE Levels of Automated Driving:

This section summarizes relevant information from "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," [SAE International, 2021] which describes a complete taxonomy of all levels of driving automation for on-road vehicles. The SAE document does not intend to provide specifications. Instead, a more descriptive formulation for automated driving is given, including the following:

1. A clarification about the role of the human driver in an AV
2. A scope for questions regarding regulations, policies, and standards
3. A taxonomy for automation specifications and technical requirements
4. A proper standard for communication when talking about AVs

Definitions

As this work utilizes actual automated driving algorithms a common understanding about specific terminology and the system functions is needed. Therefore, SAE definitions for automated vehicles are introduced.

DDT & OEDR: The Dynamic Driving Task (DDT) and Object and Event Detection and Response (OEDR) are explained in this paragraph based on Figure 2.1. The DDT includes lateral and longitudinal operation, monitoring the vehicle's environment and the response to it, namely OEDR. All parts of the DDT are given inside the dashed box in Figure 2.1. Tactical maneuver planning actuation of the headlight, horn, or other gesture responses are not given in Figure 2.1 but are part of the DDT as well. Solutions such as Electronic Stability Control (ESC), Automatic Emergency Braking (AEB), or Lane Keeping Assistance (LKA) may also be a part of the DDT.

There are three types of functions when operating a vehicle. **Operational functions** include basic vehicle motion control, **tactical functions** include planning and execution for object avoidance and route following, and **strategic functions** utilize route and destination selection, as well as timing matters.

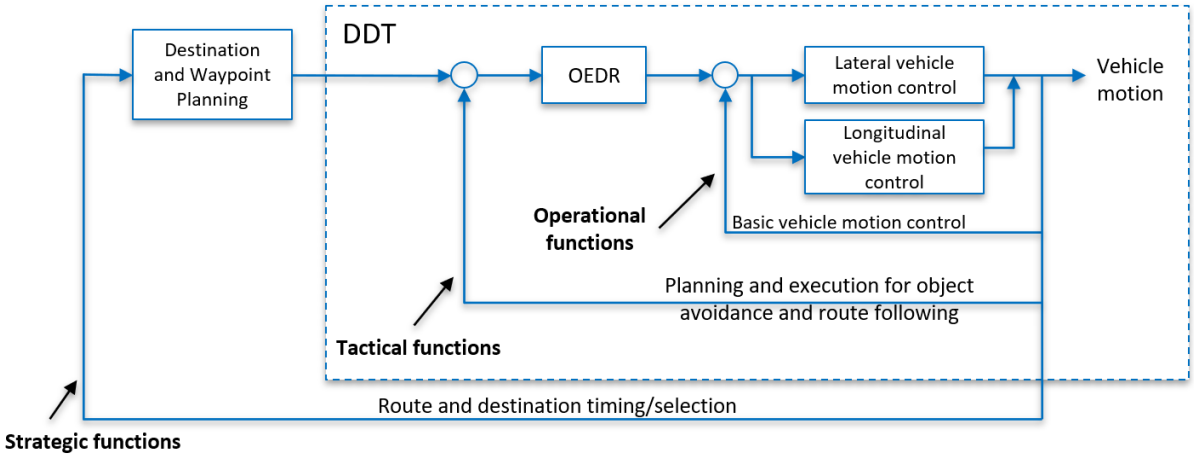


Figure 2.1: Schematic showing the principles on what is part of the DDT based on a graphic from [SAE International, 2021]

ADS: When speaking of Automated Driving Systems (ADSs), one must be aware of the difference compared to driving automation systems, as these have different meanings. Driving automation systems is used while talking about levels of automation from one to five, whereas a ADS applies to systems from levels three to five. The ADS is the wording for a system that can perform the DDT continuously within a specific Operational Design Domain (ODD).

ODD: According to SAE J3016 [SAE International, 2021], the ODD defines the operating conditions in which a driving automation system is designed to operate. Conditions

include but are not limited to the vehicle's environment, geographical area of operation, daytime, and presence of specific traffic or roadway characteristics.

Coordinate Frame: The vehicle coordinate frame is shown in Figure 2.2, where longitudinal movement is defined along the x-axis and lateral movement along the y-axis. The center point of the coordinate system is not defined in J3016. In most works, the reference of a vehicle is defined in the center of the rear axle, called "base_link."

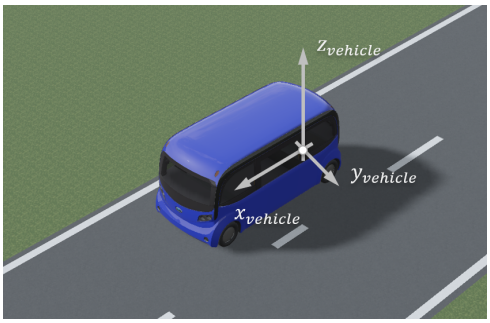


Figure 2.2: Vehicle coordinate system

Fallback & minimal risk condition: If a driving automation system fails for any reason, a DDT fallback should be triggered. Such a fallback should achieve a minimal risk condition. A minimal risk condition is a stable stopped condition that reduces the risk of an accident if a trip cannot be continued. An example of a level three system's fallback is a driver who proceeds manual driving after several seconds of preparation time. Level four or five systems can achieve a minimal risk condition independently. If manual control by a user is possible in a level four or five ADS, it may also be used to achieve a minimal risk condition.

Taxonomy of driving automation

The taxonomy of driving automation considers six levels. These levels are based on the role of the user/human driver. More precisely, the relation between the driving automation system and the user. The levels are given in Table 2.1 and Table 2.2. Table 2.1 represents cases where the driver performs parts or all of the DDT, whereas Table 2.2 cases where the ADS performs the entire DDT while it is engaged. As of the date this thesis was written, the highest level of automation achieved by an OEM is SAE level three.

Table 2.1: Summary of the level of driving automation, from level zero to level two, given in [SAE International, 2021], where the driver performs parts or all of the DDT while the driving systems utilize driver support features.

Level	**Name**	**Narrative Definition**	**DDT**			**ODD**
			Motion Control	OEDR	Fallback	
0	No Driving Automation	The driver performs the entire DDT, even when enhanced by active safety systems.	Driver	Driver	Driver	n/a
1	Driver Assistance	The sustained and ODD-specific execution by a driving automation system of the DDT's lateral or longitudinal vehicle motion control sub-task (but not both simultaneously) with the expectation that the driver performs the remainder of the DDT.	Driver and System	Driver	Driver	Limited
2	Partial Driving Automation	The sustained and ODD-specific execution by a driving automation system of both the DDT's lateral and longitudinal vehicle motion control sub-tasks with the expectation that the driver completes the OEDR sub-task and supervises the driving automation system.	System	Driver	Driver	Limited

Table 2.2: Summary of the level of driving automation, from level three to five, given in [SAE International, 2021], where the ADS performs the entire DDT while it is engaged. Levels three to five are considered automated driving.

Level	**Name**	**Narrative Definition**	**DDT**			**ODD**
			Motion Control	OEDR	Fallback	
3	Condit. Driving Automation	The sustained and ODD-specific performance by an ADS of the entire DDT with the expectation that the DDT fallback-ready user is receptive to ADS-issued requests to intervene, as well as to DDT performance-relevant system failures in other vehicle systems, and will respond appropriately.	System	System	Fallback ready user	Limited
4	High Driving Automation	The sustained and ODD-specific performance by an ADS of the entire DDT and DDT fallback without any expectation that a user will need to intervene.	System	System	System	Limited
5	Full Driving Automation	The sustained and unconditional (i.e., not ODD-specific) performance by an ADS of the entire DDT and DDT fallback without any expectation that a user will need to intervene.	System	System	System	Unlimited

2.1.2 Hardware in the Loop Testing

An AV must be tested before it can get a road permit. Regarding SAE levels, the testing procedure for different levels of automated driving varies. The HIL testing approach is a crucial method to work on research questions because it can achieve more significant results than pure Software in the Loop (SIL) simulations. Therefore, it is vital to have a look at previous work on HIL setups incorporating Intelligent Transport Systems (ITS) and CP. This subsection will give an overview about general automotive development standards and HIL testing.

Standards

There are different standards concerning the road safety of automated vehicles during their development process. Functional safety and SoTIF e.g., address different safety aspects. Considering functional safety, the driving system must be able to behave safely if an error in the vehicle occurs. Therefore the focus is on mitigating safety hazards resulting from malfunctions. SoTIF is related to system failures not covered by the functional safety resulting from the system design. An intended function verified by functional safety standards can still create a safety hazard if the intended function itself is not safe. If the system does not operate safely, the functional safety checks will ensure safe operation. At the same time, SoTIF is beneficial if the system works as intended but runs into a safety-critical situation.

HIL testing is a part of the functional design process given by SoTIF [International Organization for Standardization, 2019], which is one step in V&V. Figure 2.3 shows the different types of HIL simulations based on a line graph. A HIL simulation comprises many different system setups. On the left side of Figure 2.3, no hardware is used in the simulation. Depending on the simulated system, one can assume that the simulation errors are high, as no actual hardware is used in the simulated system. Moving on to a Co-simulation HIL (CoHIL), a subset of different components that are part of the existing system are included. This leads to an increased realism and significance of the simulation results. In contrast, the simulation errors decline as more parts of the setup are replaced by the parts used in the tested system.

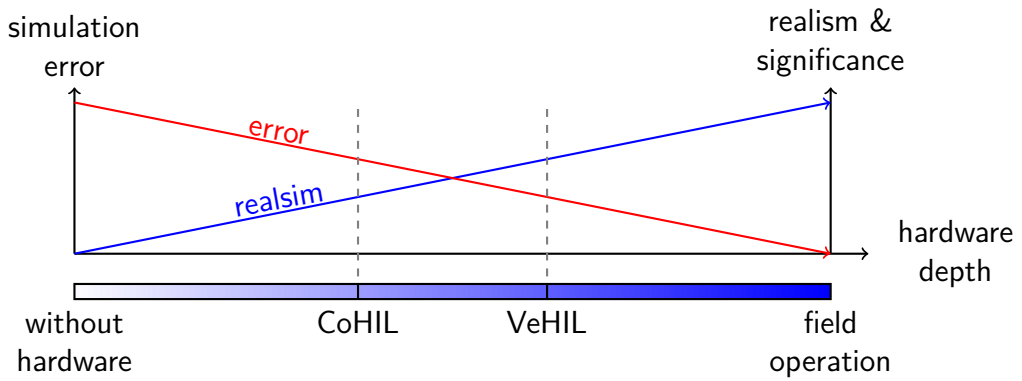


Figure 2.3: An overview about hardware depth and its impact on the realism and significance of simulation results, as well as simulation errors.

This trend progresses in a Vehicle Hardware in the Loop (VeHIL) until a complete field test removes the simulation and actual hardware is tested independently. Simulation errors decline as the hardware depth rises. Nevertheless, sensors add errors to data processing, increasing realism. However, the overall error of data sent to the tested setup increases, which is not shown in Figure 2.3.

The following description gives insight into the SoTIF process and is explained using Figure 2.4. During the analysis step, indicated by the arrow in the middle of Figure 2.4, possible scenarios for automated vehicles are elaborated. Such a scenario could be a right-turn maneuver in an urban intersection. The next step is verifying if and how an AV may solve this scenario. The scenario defined in the analysis and the proposed solution given in the verification is evaluated in the validation process. Such an evaluation may be done by real or virtual testing. The final step is field operation considering testing on public roads. It is essential to understand that not every possible encounter of an AV can be foreseen during the analysis step. Therefore, V&V will not be able to ensure functional design safety of 100%.

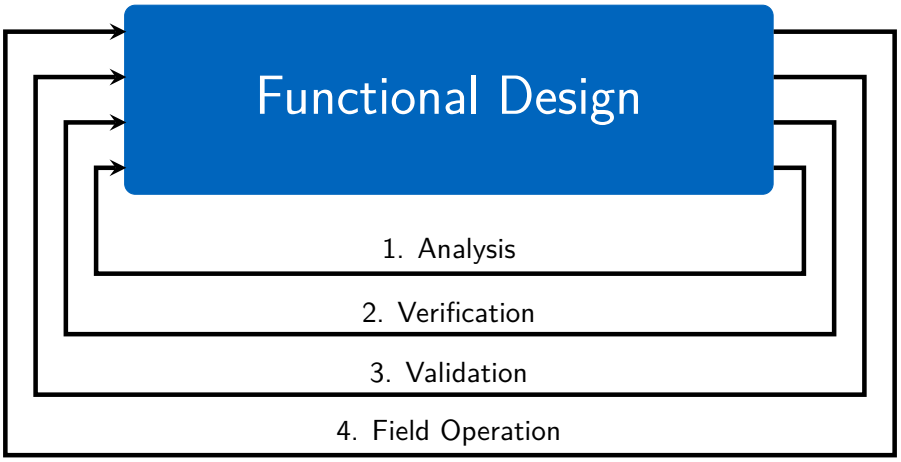


Figure 2.4: Functional Design based on ISO/PAS 21448 SOTIF [International Organization for Standardization, 2019]

Foreseeable possible scenarios could be comprised of a set of hundreds of imaginable right-turn maneuvers. Every time a scenario in the analysis section changes, the following sections must be reevaluated from top to bottom. Real-world testing may not be feasible when approving automated vehicles on public roads, as 100% safety cannot be guaranteed. Uncertainties that create new scenarios in the actual world are almost infinite. Virtual testing is an exciting strategy to achieve meaningful field operation tests to approve the functional design. If a scenario defined in the first step, the analysis, misses out on a significant scenario variation, this could have been discovered in the validation by extensive use of virtual testing. Suppose the validation phase fails to show such an issue. In that case, it will likely come up during field operation testing, leading to a costly reevaluation of all four functional design steps.

To approve an AV according to V&V standards, a certain amount of accidents may be accepted. For example, the existence of airplane crashes and the users' acceptance may be applied. Another comparison would be an AV which is statistically ten times safer than the average human driver. SaFAD [Wood et al., 2019] deals with the issue of approving AVs on

public roads. Five significant challenges are outlined, shown in Table 2.3. Especially the first challenge may be confronted by the extensive use of HILTesting. Challenges two, three, four, and five can be addressed with virtual tests to some extend as well.

Table 2.3: Major challenges for V&V according to [Wood et al., 2019]

Challenge	
#1	Statistical demonstration of system safety and a positive risk balance without driver interaction
#2	System safety with driver interaction (especially in takeover maneuvers)
#3	Consideration of scenarios currently not known in traffic
#4	Validation of various system configurations and variants
#5	Validation of (sub)systems that are based on machine learning

Simulation types

Table 2.4 gives an overview of different simulation types, where the amount of simulation detail increases from top to bottom. There are several types of traffic simulations, each with pros and cons.

A macroscopic simulation captures a whole city or region and is used to evaluate, e.g., traffic flow or average speed of road users. The movement of each road user is not considered, as this type of simulation evaluates the overall traffic inside a large area. Mesoscopic simulations can include, e.g., interactions between vehicles and lane changes. It is used to analyze specific traffic scenarios in a sub region of a road network to achieve more detailed and precise results than in the macroscopic simulation. In a microscopic simulation, the amount of detail is increased further. The user of such a simulation can extract more detailed information on the individual movement of road users. If the simulation detail is increased by a 3D environment or very detailed vehicle dynamics simulations, one could refer to this type as sub-microscopic simulation. Using a sub-microscopic simulation makes achieving the most accurate and detailed results possible.

Essentially, as the simulation type changes from macro to sub-micro, the detail level increases while the analyzed traffic area decreases.

Table 2.4: Overview of different traffic simulation types

Simulation Type	Description
Macro	Large model of a metropolitan area
Meso	Large sub-area of the metropolitan area
Micro	Several intersections inside the meso area with an increased amount of details
Sub-Micro	Increased number of detail of the microscopic simulation (e.g., detailed vehicle dynamics model, 3D Simulation)

Related Work

Hardware-In-The-Loop Testing of Connected and Automated Vehicle Applications: A Use Case For Cooperative Adaptive Cruise Control In [J. Ma et al., 2018], a proof of concept state-of-the-art Vehicle to Vehcile (V2V) HIL is proposed. The main question discussed in the article is how V2V HIL testing of automated vehicles helps develop algorithms for automated driving.

Overall, the authors argue that this method will accelerate the development and testing of automated driving software. In their experiment, a Cooperative Adaptive Cruise Control (CACC) is evaluated using their HIL setup. This setup consists of a real car supplied with virtual vehicles by the VIS-SIM traffic simulator. The CACC algorithm was tested without requiring several physical vehicles by integrating the real car into the simulation. Furthermore, the actual vehicle increased the accuracy of the test compared to a standard simulation. The standard simulations' approximated vehicle model is replaced by the actual vehicle.

Different driving scenarios were shown, and the influence on the CACC system was presented, proving the feasibility, and illustrating the beneficial effects of a V2V HIL-Simulator.

Hardware-in-the-Loop Testing of Connected and Automated Vehicle Applications: A Use Case for Queue-Aware Signalized Intersection Approach and Departure Article [J. Ma et al., 2018] demonstrates a V2I HIL setup. This paper, in addition to [J. Ma et al., 2018] shows how V2I can help the development of automated driving algorithms.

Model accuracy, missing calibration data, and experimental setup assumptions are limiting factors to the validity of the evaluation results. The articles experiment presents a Queue Signalized Intersection Approach and Departure (Q-SIAD) algorithm, which generates recommended speed profiles based on the vehicle's status (e.g. location, velocity), Signal Phase and Timing (SPaT), system constraints like maximum acceleration and other vehicles, provided by VISSIM. Q-SIAD was demonstrated on an automated vehicle driving through an intersection controlled by a fixed-timing traffic signal. Additionally, multiple traffic conditions, such as congestion were applied to the test.

This article presents another example of a state-of-the-art HIL setup, where a real car is introduced into virtual traffic to thoroughly test an algorithm for automated driving.

Use of Hardware in the Loop (HIL) Simulation for Developing Connected Autonomous Vehicle (CAV) Applications The publication [Cantas et al., 2019] presents a HIL simulator for the development of applications for CAV. The article describes a green wave use case, enabling a continuous traffic flow through several intersections by adjusting the traffic light control.

A dSPACE Scalexio HIL Computer runs the vehicle dynamics and the environment simulation. This component is connected to a dSPACE MicroAutoBox (MABX) that runs the control software in the vehicle. The vehicle itself is not included in the HIL. A traffic light controller is sending Signal Phase and Timing (SPaT) information through a Road Side Unit (RSU) to the CAV's On Board Unit (OBU). The traffic simulation software Simulation of Urban MObility (SUMO) was used to simulate the actual traffic. The environment simulation was performed in CarSim. As a result of sending SPaT messages to the CAV, its idle time at a junction was

reduced. Consequently, the cars' fuel economy is improved in the simulation.

The article suggests that using HIL simulations help develop automated driving algorithms. It is unnecessary to prove that HIL testing is necessary. This tool is widely used for many reasons. The development time is reduced by incorporating HIL test software development, and the accuracy of simulations is increased. Real-time delays between wireless connections can replace assumptions of a virtual model when the actual components are used in the loop. Vehicle model imprecision's accepted in pure SIL setups are entirely removed from the equation if the actual car is used in a HIL simulation. Summarizing a HIL simulation is an efficient method to develop and test algorithms for highly automated driving.

A Connected and Autonomous Vehicle Hardware-in-the-Loop Simulator for Developing Automated Driving Algorithms This work presents a HIL architecture and results from real-world tests. Their system incorporates a vehicular ITS-S and a roadside ITS-S, which are connected using Dedicated Short Range Communication (DRSC) based on the 802.11p wireless standard. Their article considers testing of Cooperative Adaptive Cruise Control applications using their HIL systems. [Gelbal et al., 2017]

Summary on HIL Testing

HIL tests are a viable state-of-the-art technology in industry and research. Driving algorithms can be tested in more detail if HIL setups are utilized. Replacing virtual parts of the simulation architecture increases the accuracy and significance of simulation results. Nevertheless, most research is restricted to SIL simulations, as those are more convenient to implement in the first place due to increased efforts needed when HIL tests are added to the system. Especially when complex algorithms, such as automated driving algorithms, are tested in real-time simulation environments, utilizing a HIL framework adds value. A pure SIL setup will likely not generate results as reliable and significant as a HIL setup.

2.1.3 Sense, Plan, Act

In SaFAD V&V methods for SAE automated driving levels 3 and 4 are summarized. System operation is given by the Sense Plan Act paradigm, illustrated in Figure 2.5. This fundamental design derived by well-established companies and research institutes explained by Wood et al. [2019] is used throughout this thesis and forms the basic system design for the AV.

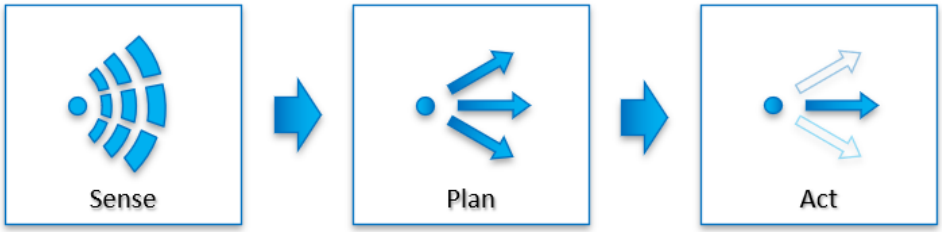


Figure 2.5: Sense, Plan, Act system design

Sense

Sensing consists of perceiving static and dynamic objects near the automated vehicle. Additionally, the behavior of detected objects may be predicted. Furthermore, the location of the ego is evaluated by the Sense level. Figure 2.6 gives a typical data flow for a perception system. Raw sensor data is processed by an object detection algorithm, generating detected objects. Detected objects can contain their position in the world, their size, and possible classification information, whether, e.g., pedestrian, bicyclist, or car. Detected objects are sent to a tracking algorithm which can add or improve the velocity information and heading of the detected object. A prediction algorithm then evaluates a tracked object. This setup will most likely vary in different systems, as some solutions may combine two blocks into one single system or even all three algorithms into one.

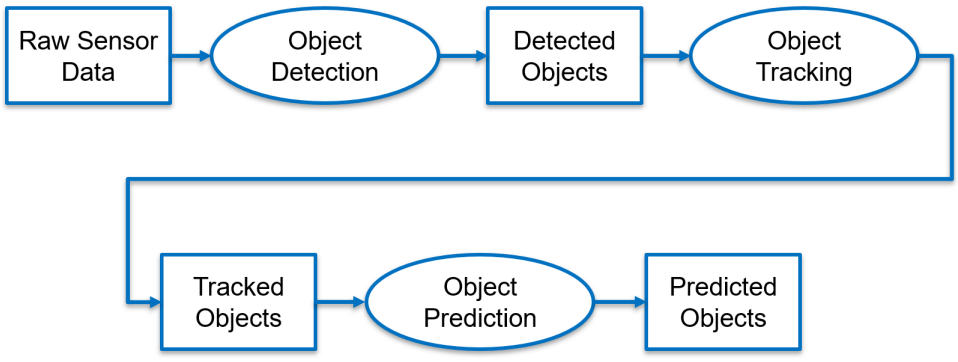


Figure 2.6: Example of the data flow of a perception system

Sensing systems: Different sensors are available for perceiving objects in an AVs' environment. Waymo[1] is using a Light Detection And Ranging (LiDAR), and camera setup for their AV test vehicles [Sun et al., n.d.]. Other competitors, like Tesla, e.g., are relying solely on camera and Radio Detection And Ranging (RADAR). Besides Waymo and Tesla, several other companies are working on the future of automated driving. Here is more than one strategy to achieve high levels of automated driving, and one must judge the best solution. As of the day this thesis is written, Daimler is the first and only one to have achieved certification of a SAE Level 3 system. Their system is called DRIVE PILOT[2] and uses a combination of RADAR, LiDAR, camera, road moisture, ultrasonic sensors, and microphones to perceive the environment. Essentially all major sensor technologies are used.

Sensing systems may be categorized into three major technologies: RADAR, LiDAR, and camera. Each of these systems has its subcategories. Where a LiDAR system overall performs very well in the accuracy range, and FoV, LiDAR is affected by the weather. On the other hand, a RADAR sensor may not have the highest accuracy or FoV but is the most reliable sensor regarding weather effects. A camera contributes special features to the system through color perception capabilities.

Rosique et al. [2019] did a review of perception systems focusing on four types: RADAR, Camera, LiDAR and Ultrasonic. They have introduced sensor Key Performance Indicators (KPIs) and a scoring system. Firstly, KPIs will be explained, followed by and introduction to the KPI scores.

- **FoV:** This is the area that a sensor can observe. Figure 2.7 shows the FoV of an AV that utilizes a forward-facing sensor with an FoV sensor angle of 80 degrees. This is a typical setup for a camera in an AV that is used for long-range detection.
- **Range:** Refers to the maximum distance a sensor can detect objects.
- **Accuracy:** This KPI can have different meanings, e.g., MOTA [Bernardin and Stiefelhagen, 2008]. Accuracy, in technical terms, is the value that describes how precise a sensors' measurement is compared to the real value or ground truth data.
- **Frame Rate:** Most modern sensing systems capture data at a specific rate. A camera running 30 Hz will capture 30 images per second. This principle is also applied to all other sensing systems, e.g., a LiDAR sensor running at the same frame rate will capture 30 point-cloud images per second. The frame rate defines the sampling input of sensor data for the system. Usually, the higher the frame rate a sensor can achieve, the better, as long as the computing units can handle the amount of data considering high-resolution images.
- **Resolution:** The higher the resolution of a sensor, the more detail it will be able to capture. This can result in higher detection ranges but will directly increase the need for computational power as data increases.

[1] Waymo LLC is the self-driving car project of Google

[2] https://group.mercedes-benz.com/dokumente/innovation/sonstiges/2019-02-20-vssa-mercedes-benz-drive-pilot-a.pdf

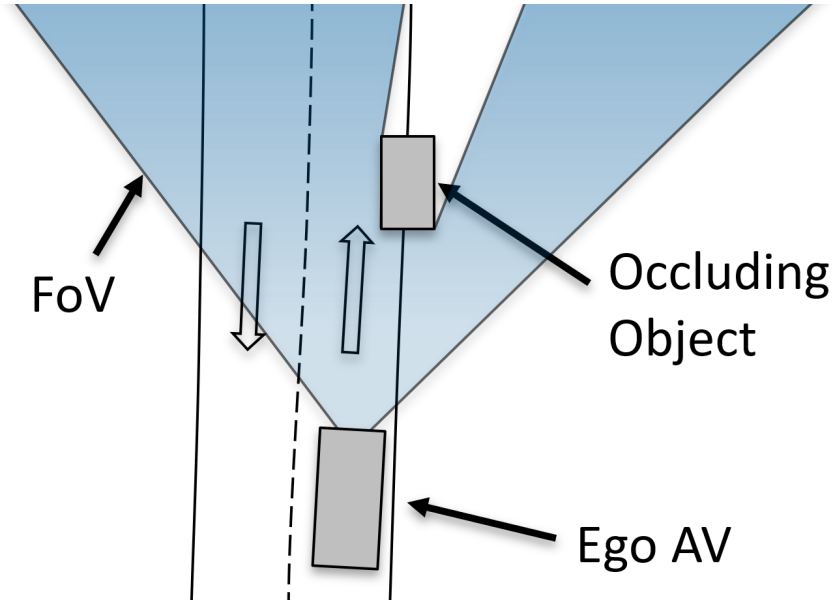


Figure 2.7: Image that shows the FoV of an AV that has, e.g., a forward facing camera.

- **Color Perception:** The sensors' capability in detecting color.
- **Size:** The physical size of a sensor and the amount of space it needs when it is utilized in a system.
- **Weather Affection:** The effect of weather on the sensor. Various conditions, such as rain, snow, ice, or fog, can interfere with the sensor's capabilities. This can result in minor accuracy derogation up to a complete system failure.
- **Maintenance:** This is an issue for solutions that should last for an extended period, e.g., rotating LiDARs include mechanically moving parts that are more likely to fail than solid state LiDARs. In case of a failure, these systems must undergo maintenance.
- **Visibility:** In the automotive industry, it is advantageous to have sensors that can be mounted behind covers (e.g., RADAR). This influences, e.g., the aerodynamics of vehicles or the design.
- **Price:** This aspect considers the price of the sensor at which it can be bought on the market.

To evaluate KPIs a scoring system must be established. In the work of ROSIQUE et al. [2019] a numerical score from zero to four is introduced. In this number based score system, a high value does not automatically indicate good performance. The scoring numbers translate as follows:

- **0 = None:** The KPI cannot be applied because the sensor is not capable of directly perceiving color information, e.g., RADAR or LiDAR.
- **1 = Low:** This KPI score must be considered twofold. For the price, scoring one would be good, while for the FoV, a low scoring (1) is bad.
- **2 = Medium:** A medium score is achieved if there is no tendency if the sensor performs rather well or badly.
- **3 = High:** A high score for the price meaning a high price is not ideal, while a high resolution is.

Evaluated sensing systems are shown at the top of Table 2.5 and are explained in more detail using polar plots below.

Table 2.5: Summary of the findings from ROSIQUE et al. [2019], showing significant AV sensors types and KPIs. The KPIs are measured with numbers from zero to three (0 = none, 1 = low, 2 = medium, 3 = high).

	RADAR	**3D LiDAR**		**Cameras**			**Ultrasonic**
		Rotating	Solid State	VIS	IR	ToF	
FoV	2	3	2	3	3	2	1
Range	3	3	3	2	3	2	1
Accuracy	2	3	3	3	2	2	1
Frame rate	2	2	2	2	3	3	2
Resolution	1	2	2	3	1	1	1
Color perception	0	1	2	3	1	1	0
Size	1	2	1	1	1	1	1
Weather affections	1	2	2	3	1	3	1
Maintenance	1	2	1	2	2	2	2
Visibility	1	3	2	2	2	2	2
Price	2	3	1	1	3	2	1

Figure 2.8 highlights the capabilities of a radar sensor. The solid black line shows an ideal sensor that is compared to the dashed blue line corresponding to the capabilities of the RADAR sensor. If RADAR was perfect, it would overlay the black line in the whole diagram. In the review of ROSIQUE et al. [2019], typical automotive grade RADAR sensors had a comparable low resolution and a medium frame rate and accuracy. Today, new radar types are already being tested, utilizing a higher RADAR frequency, enabling higher resolution and accuracy. A recent article even claims that major localization algorithms like Simultaneous Localization and Mapping (SLAM) may be performed on a lower cost level than using LiDAR if state-of-the-art RADAR sensors and algorithms are used [VENON et al., 2022]. RADAR sensors have an advantage in detection range and are not very expensive. Those sensors are perfect for visibility and size inside a car, as they can be placed behind covers and are very small. There is almost no weather effect on RADAR, and maintenance is low, as automotive RADAR systems usually do not include moving antennas.

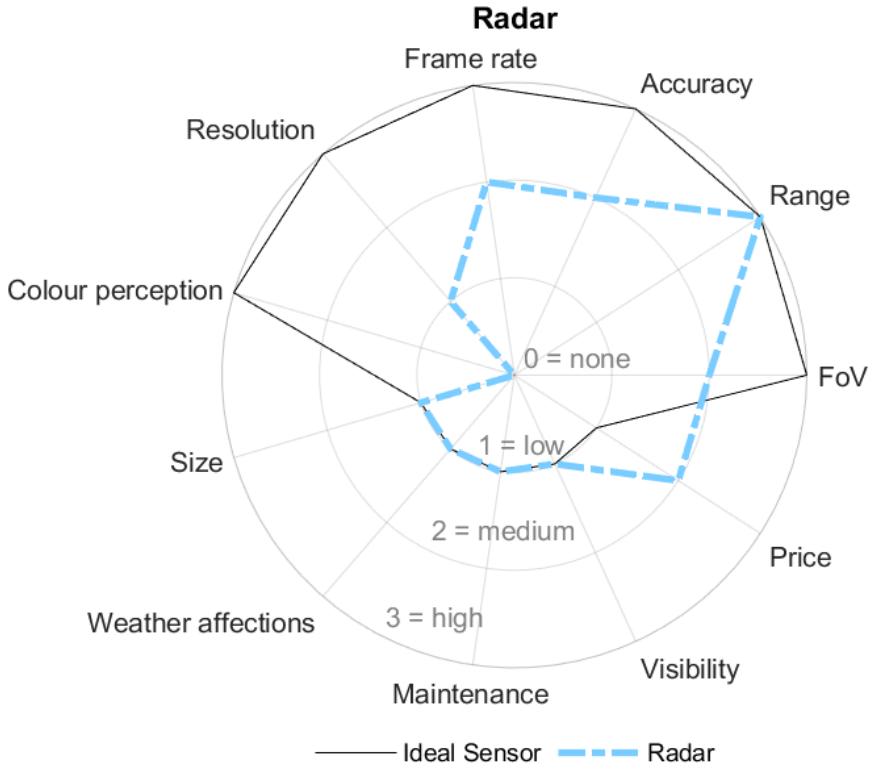


Figure 2.8: An overview about radar capabilities based on findings from ROSIQUE et al. [2019]

Figure 2.9 shows different types of camera systems. The typical visual camera (VIS) generates images based on light falling into the lens and image sensor. These cameras are outstanding in color perception, resolution, and accuracy, enabling them to be particularly advanced in object classification using Deep Neural Networks (DNNs), for example. A significant downside is the effect of weather. E.g., fog will severely influence the capabilities of a VIS camera. An infrared (IR) camera has high accuracy and range. It can detect temperature changes and is

useful in applications where data privacy is essential, as people cannot be identified as easily as on a VIS camera image. A Time of Flight (ToF) camera is usually used in, e.g., augmented reality or gesture recognition.

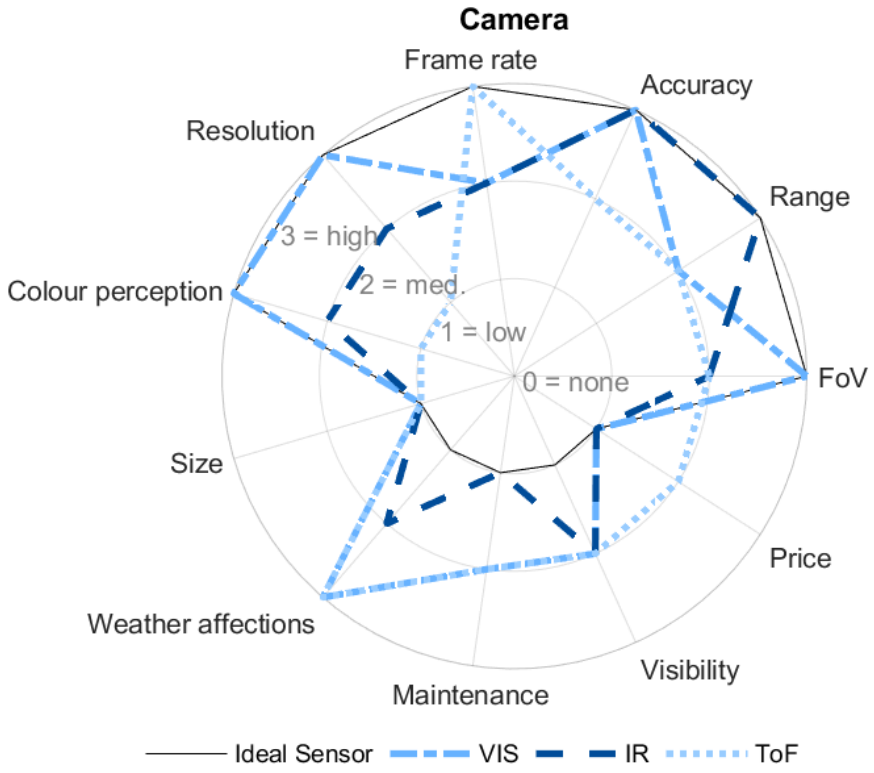


Figure 2.9: An overview about camera capabilities based on findings from Rosique et al. [2019]

LiDAR sensors are available in two major system types. There are rotating and solid state LiDARs. Currently, most systems used in research vehicles use rotating ones, as those are available and easy to install. In contrast, modern solid-state systems lack availability and require increased installation effort for a 360-degree FoV around the car. It is projected that new LiDARs systems, will reduce the price [Raj et al., 2020].

Currently, a LiDAR usually captures point cloud data at about 20 Hz. Using the point pillar DNN, an inference time of 16 ms is achievable. Therefore, an overall turnaround time of perceived data until receiving a viable object list is as fast as 66 milliseconds [Lang et al., 2019]. Cameras can achieve at least an object list capture rate of 30 Hz, resulting in a time needed to extract objects from a single image of 33 milliseconds. Sistu et al. [2019] used the You Only Look Once (YOLO) Convolutional Neural Network (CNN) from Redmon et al. [2016], to achieve this results. R. Pérez et al. [2019] extracted object lists from RADAR within 30 milliseconds, using YOLO [Redmon et al., 2016]. These numbers may change quite a lot as the development of sensing systems progresses.

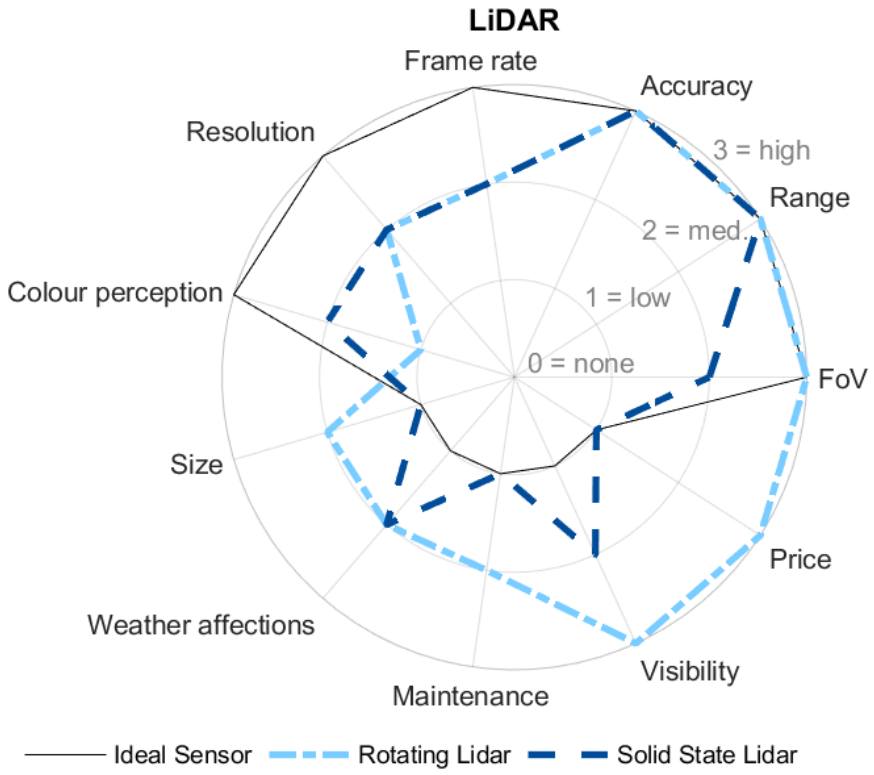


Figure 2.10: An overview about LiDAR capabilities based on findings from Rosique et al. [2019]

Object Detecting and Tracking Measurement: A significant measure for sensing systems is the accuracy available to an ADS to detect objects in its environment. At first, one must define how accuracy is measured. A standard wording used by Sun et al. [n.d.] is the MOTA defined by Bernardin and Stiefelhagen [2008]. In their work, they have issued the problem of needing more ability to compare object-tracking algorithms. There are too many different metrics to measure the performance of such systems. They have defined two metrics to intuitively and objectively evaluate object trackers. The first metric is the Mutli Object Tracking Precision (MOTP) given in Equation 2.1, where c_t is the number of matches found for time t and d_t^i the distance between the objects evaluated position and actual position.

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{2.1}$$

Equation 2.2 describes the MOTA metric, where m_t stands for the number of misses or false positives respectively, fp_t describes false positives, mme_t the mismatch error and g_t the current number of objects, evaluated at time t. A false positive, in this context, occurs when a system detects a pedestrian at a position where no pedestrian is. The counterpart of a false positive is the false negative, where a pedestrian is at a position, but the system does not detect him. In addition, one can also define a true positive and true negative indicating proper

operation characteristics of the system.

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{2.2}$$

Equation 2.3,2.4 and 2.5 highlight the miss, false positive, and mismatch ratio over time.

$$\overline{m} = \frac{\sum_t m_t}{\sum_t g_t} \tag{2.3}$$

$$\overline{fp} = \frac{\sum_t fp_t}{\sum_t g_t} \tag{2.4}$$

$$\overline{mme} = \frac{\sum_t mme_t}{\sum_t g_t} \tag{2.5}$$

Occlusion and Sensing: Road vehicles are generally exposed to many uncertainties when driving in urban street networks. In many situations, an occluded intersection can challenge an ADS due to urban building structures. There are several approaches dealing with the matter of occlusion-aware intersection handling for ADSs.

In the work "Crossing Blind Intersections from a Full Stop Using Estimated Visibility of Approaching Vehicles" by NARKSRI et al. [2019] the authors considered crossing intersections with low visibility. They have considered an urban road layout with heavy occlusion imposed on the ego vehicle. Figure 2.11 shows one of the low-visibility intersections mentioned in their work.

Figure 2.11: Image showing one of the low visibility intersections used in work by NARKSRI et al. [2019]

They state that AVs can encounter deadlock situations if they have to stop at the stop-line and are not allowed to move forward if the scene cannot be observed properly, such as the one given in Figure 2.11. In their method, they propose the following strategy:

1. Drive to the stop line and execute a full stop.
2. If the occlusion situation allows the AV to see all possibly approaching vehicles, it shall proceed with everyday driving. Otherwise, the AV should drive slowly into the intersection, even if the visibility does not allow it.
3. By proceeding with the driving maneuver, the ego's visibility to other traffic participants is increased.
4. Finally, the ego AV can sense its environment properly and cross the intersection safely.

In their article, a simulation setup is presented, which is based on logged data from the actual world. They use a virtual vehicle to test the proposed approach and compare vehicle dynamics, such as the vehicle's velocity, to the results of an actual human driver. The outcome shows that their systems can compare to the driving behavior of a human driver and can handle the tested situations safely.

In this work, the issue of deadlock situations for AVs is highlighted, and an approach to handle such situations is given. Even though they can cope with occluded situations, they state that more difficult low-visibility situations do not work properly and safely. In context to this thesis, one can derive from their results that an AV must drive slowly, relying solely on its sensor setup if the scene includes a high amount of occlusion. In addition, it may end up in a deadlock situation if a worst-case situation is encountered.

Orzechowski et al. [2018] implemented an algorithm to account for occlusion and limited sensor range in urban scenarios. They state that their approach can prove whether the trajectory of an AV is safe concerning occlusion. From their point of view, state-of-the-art AV driving algorithms must integrate a comfortable driving behavior and a traffic flow improvement. Additionally, algorithms not taking into account occlusion are not likely to guarantee the safety of their maneuvers.

They propose a set-based safety method to verify the AV's planned course of action in their work. A high-definition map of the environment, including, e.g., road boundaries is needed. They perceive relevant information from the AV sensor readings and the road network. In Figure 2.12, one can see the ego vehicle in blue at the bottom of the picture and a grey obstacle that creates occlusion. The transparent blue area around the ego indicates the area observed by the ego AV. Numbers in this figure show relevant edges of the area perceived by the vehicle's sensing systems. Relevant edges are shown in red, which is edge number three in this case. They place an object at this position and predict the occupancy of a possible object using the road network from their map. In order to achieve computationally feasible results, they use a set-based prediction.

Even though they could prove the trajectories' safety, only three specific intersections were used to test their system. Considering more challenging scenarios such as the one shown by Narksri et al. [2019], they are likely to end up in a deadlock because their scenarios do not

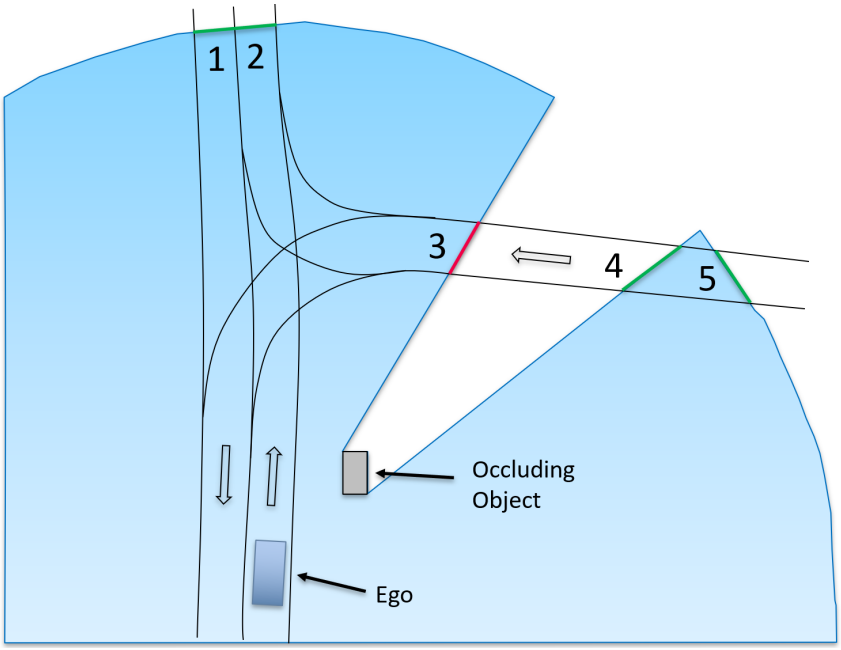


Figure 2.12: Illustration showing the qualitative approach from Orzechowski et al. [2018]

consider severely occluded scenes such as the one shown in Figure 2.11. Additionally, they state that, for now, their approach reduces driving comfort and increases driving safety. This solution is a trade-off between both aspects. Increasing one of them reduces the other. They suggest using fail-safe trajectories as a compromise between both or tracking occluded areas. In this thesis, occluded areas are tracked, and the impact on the egos' driving behavior is analyzed.

In "Limited Visibility and Uncertainty Aware Motion Planning for Automated Driving" by Tas and Stiller [2018] the authors presented their approach for a motion planning system that handles scenarios including limited visibility, weather effects, and other uncertainties collision-free. Their approach utilizes an optimization-based planning algorithm that takes occlusion into account. They claim to be able to drive safely and comfortably for all their scenarios, running in a proprietary closed-loop simulation system. They do not state anything about the traffic flow impacts of their approach. The only minor indication of issues with traffic flow due to their approach is a comparison of sensor detection range. The lowest possible sensor detection range they simulated was 30 meters, which significantly impacts the speed profile of their vehicle. Considering the perception system capabilities of AVs from Sun et al. [n.d.], which states that a reliable detection distance of 30 meters is hard to achieve, this approach must be considered with caution if we look at the maneuver time of such an AV.

All three article's approaches show similar behavior to a human driver, which might not be the best method for a non-human system. None of these approaches consider extending the FoV of the AV to overcome this problem. Additional sensors can be mounted in roadside ITS-S to enhance the perception system of vehicles. FoV extension is discussed by related work for CP in Chapter 2.2, as this section considers solely AVs.

Plan & Act

From to the output of the sense level, a collision-free and lawful driving plan is generated in the planning section. This plan includes maintaining a safe lateral and longitudinal distance to other road users, abiding by traffic rules, object occlusion, and prioritizing traffic rules to avoid accidents. The last part is the restriction to harm third parties. Planning is another very challenging task for AVs.

To get a better insight into state-of-the-art AV planning systems, the work "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles" by PADEN et al. [2016] is introduced. Their article focuses on decision-making, motion planning, and control for self-driving vehicles within the SAE levels three to five. They state that the decision-making of AVs is structured into route planning, behavioral decision-making, local motion planning, and feedback control. This structure is reviewed in Figure 2.13. Nevertheless, PADEN et al. [2016] argues that this structure is somewhat blurry in other literature. There are a lot of solutions that can be used to solve automated driving challenges, for sensing, e.g., different approaches by car manufacturers are chosen, for AV software, e.g., several architectures may be feasible as well.

The naming conventions used in Figure 2.13 are derived from PADEN et al. [2016]. If we consider SAE naming conventions, shown in Figure 2.1, route planning is part of strategic planning of an AV. Terminology used in PADEN et al. [2016] is more intuitive than the one from SAE. Table 2.6 presents a quick overview about both naming conventions.

Table 2.6: A comparison between the naming conventions used by SAE INTERNATIONAL [2021] and PADEN et al. [2016].

Paden et. al	**SAE**
Route Planning	Strategic Functions
Behavioral Layer Motion Planning	Tactical Functions
Local Feedback Control	Operational Functions

In the top left box inside Figure 2.13, a user enters a destination for the ADS that is evaluate by **Route Planning** functions. Such functions may be compared to a common navigation system used in human driven cars, creating a strategic route plan for the whole path to the vehicles' destination. Using information from the **Road Network**, way points are generated and sent to the **Behavioral Layer**. In the **Behavioral Layer**, a motion specification is derived based on reasoning for the route, perceived objects, and road signs. This information is sent to the **Motion Planning** where the AVs' pose, and free space is used to generate a collision-free trajectory. In the last step, **Local Feedback Control** uses the estimated vehicle state and generates steer throttle and brake commands.

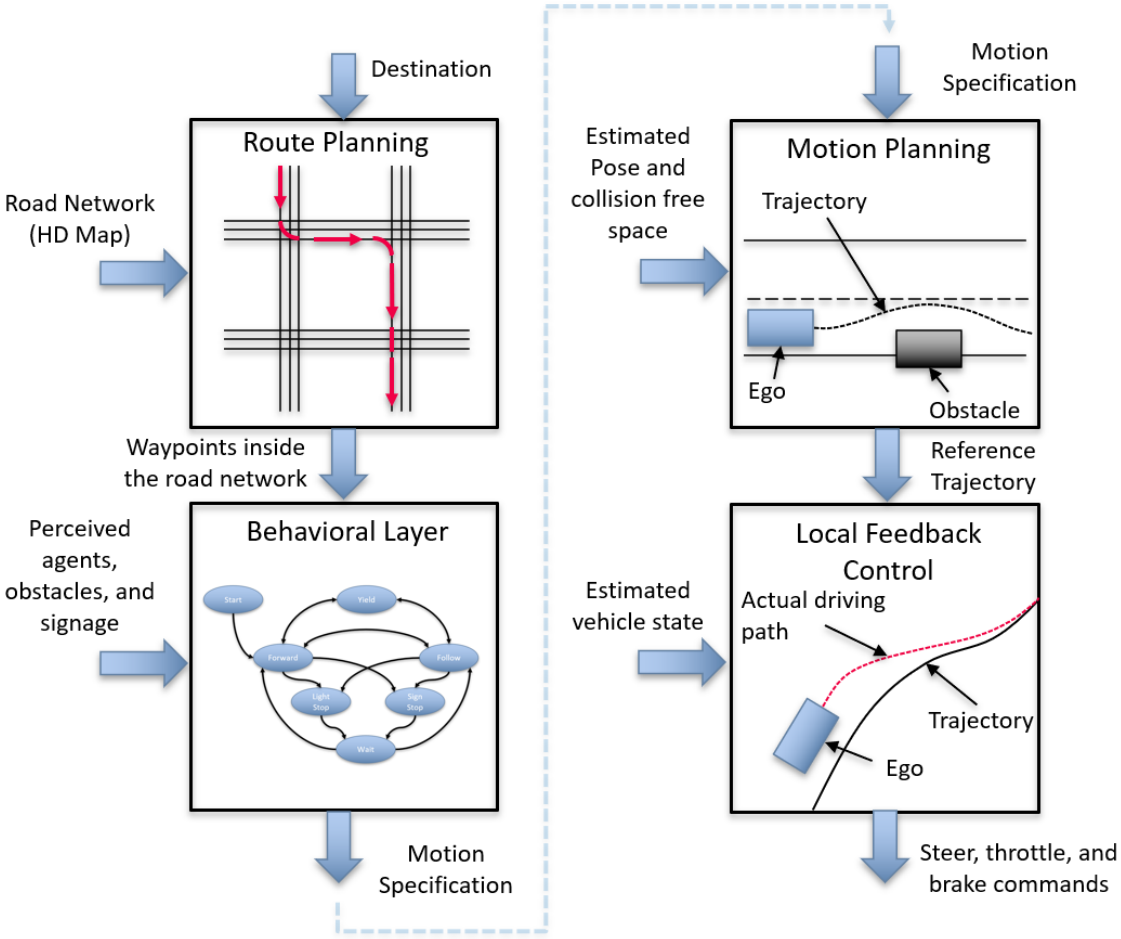


Figure 2.13: The structure of the decision-making process for AVs based on [Paden et al., 2016]

Route Planning: Route planning describes a function that selects the best vehicle route for the costs assigned to specific parts inside the road network. The basic function of this layer can be compared to a human-driven vehicle's navigation system. Applicable route search algorithms might be Dijkstra [Dijkstra, 1959] or A* [Goldberg, Andrew V. and Harrelson, Chris, 2005]. A minimal route cost could be achieved using more complex optimization-based search algorithms in more complex cases.

Besides the vehicles' goal destination, a road network is needed to derive a route. In most cases, the road network is based on an HD Map. Such an HD Map contains information about, e.g., lane geometries, lane boundaries, lane relations, signs, poles, walls, traffic lights, or road topology. Additionally, landmarks or point cloud maps for localization could be part of an HD Map definition. Among various HD Maps, Table 2.7 presents three of the most significant ones, such as Navigation Data Standard (NDS), Lanelet2, and OpenDrive.

Table 2.7: A short overview about three of the most significant HD Map definitions.

Definition	Description
NDS	This type is popular among car manufacturers and provides standardization under the NDS association. Over 30 automotive companies are already using it. A downside for research purposes is that it needs a license. [Hubertus, 2019]
Lanelet2	This is an open standard that the user may extend to cover more road network features. Compared to the other two, this map definition is the easiest one to apply, as the road network is based on geometric points. [Poggenhans et al., 2018]
OpenDrive	This standard is provided by Association for Standardization of Automation and Measuring Systems (ASAM) and is as Lanelet2 also freely available. This definition is more detailed than Lanelet2 and provides more complex definitions of road networks. [ASAM e.V., 2021]

Furthermore, Seif and Hu [2016] highlight HD Maps as a key challenge for the automotive industry. Previous work on AVs was focus on highway driving and the necessity of HD Map data was not known as core competence for automated driving.

Behavioral Layer: As explained, SAE notations combine behavioral decision-making and motion planning as tactical functions. Paden et al. [2016] seperates those into two separate systems. Both make sense because in practice behavior and motion planning hard to separate as they need inputs from the perception system of the AV and significantly impact tactical decisions of ADSs. In this setup, the behavior layer generates a broad behavior for the driving automation system, e.g., following a leading vehicle, changing lanes or stopping at a stop line. Behavioral decisions are mostly handled in a finite state machine that represents foreseeable scenarios that the AV might encounter. States could also include specific behaviors for urban driving, highway driving or parking lot situations. An reduced example of a state machine that is used in the behavioral layer is explained in Chapter 3.2.4, System Setup.

Motion Planning: Motion planning functions generate a dynamically feasible path or trajectory for the current task based on inputs from the behavior layer. Path and trajectory must not be mixed for motion planning. The path is a resulting path a vehicle can use to maneuver through its environment. This path must include information on how the AV should follow. Different strategies, e.g., a velocity profile, can be used to drive using a path. A trajectory considers a dynamically feasible movement of the vehicle. This also allows including vehicle dynamics of the ego vehicle and other vehicles. In summary, the path describes waypoints, while the trajectory specifies the future positions of the ego vehicle in time.

Several Planning Methods are presented in Table 2.8:

Table 2.8: A list of planning methods based on findings from PADEN et al. [2016]

Method	Description
Geometric	An example of geometric methods are the visibility graph [NILSSON, 1969] or voronoi diagrams [BERG et al., 1997]. These are efficient algorithms for polygonal models. Especially the visibility graph can be optimal for finding the shortest path.
Variational	A variational planning method is a complex solution not discussed here. For further reading refer to BETTS [1998]
Graph-search	This includes, e.g., a road lane graph or a lattice of motion primitives in combination with a Dijkstra [DIJKSTRA, 1959] shortest path algorithm. Both are not optimal, based on a predefined path or set of motion primitives.
Incremental Search	A example of incremental search path planning is the RRT [LAVALLE and KUFFNER, 2001], which is capitalistically complete.

Local Feedback Control: Operational functions utilize lateral and longitudinal control to accurately maneuver the ego AVs' pose as close to the reference path as possible. PADEN et al. [2016] defined a path stabilization problem for an AV. Using the differential equation $\dot{x} = f(x, u)$ and a reference path $x_{ref} : \mathbb{R} \rightarrow \mathbb{R}^n$ to find a feedback law $u(x)$. Additionally, an instance-dependent constant is used. Equation 2.6 forms the feedback law for AV control algorithms. Essentially the feedback is the closest point to the reference path.

$$u(x) = f\left(\arg\min_{\gamma} \parallel x - x_{ref}(\gamma) \parallel\right) \tag{2.6}$$

Local feedback control gets tricky when disturbances, e.g., strong winds or icy roads, are added. Different control architectures such as Model Predictive Control (MPC) or more simplistic Pure Pursuit Control (PPC) [RICHARD S. WALLACE et al., 1985] approaches can be used. There are various implementations of PPCs available. RICHARD S. WALLACE et al. [1985] was the first to publish the PPC. This thesis utilizes the PPC, which is explained in more detail than other control approaches.

The PPC is based on the principle that a vehicle drives on a semi-circle that starts on the rear axle of the ego vehicle. This circle must be tangential to the heading of the vehicle that is controlled. The semi-circle intersects with the path at a lookahead distance L. Figure 2.14 presents the basics of the PPC approach. The ego vehicle is given by a bicycle model, indicated by the front and rear wheels inside the transparent gray box in the middle of Figure 2.14. The lookahead distance L is used to draw a circle with its center on the vehicle's rear axle. This circle intersects with the path at two points. The intersection in front of the car is used as a lookahead point.

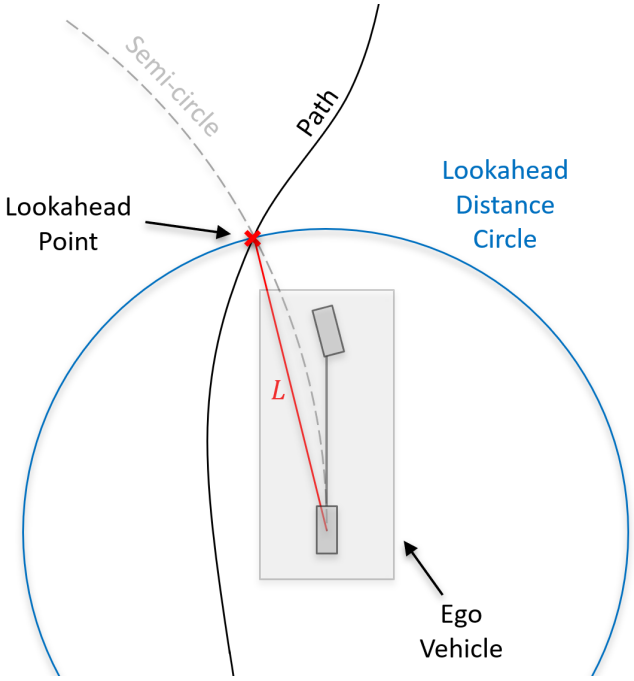


Figure 2.14: Basic description of the PPC approach, including the lookahead point and the ego vehicles' driving arc.

The geometrical description shown in Figure 2.15 is a close-up of Figure 2.14. The lookahead distance is given by a red line tagged with L, starting from the rear axle of the ego vehicle. Firstly the two basic mathematical Equations 2.7 and 2.8 are applied.

$$L^2 = \sqrt{x^2 + y^2} \tag{2.7}$$

$$x + D = r \tag{2.8}$$

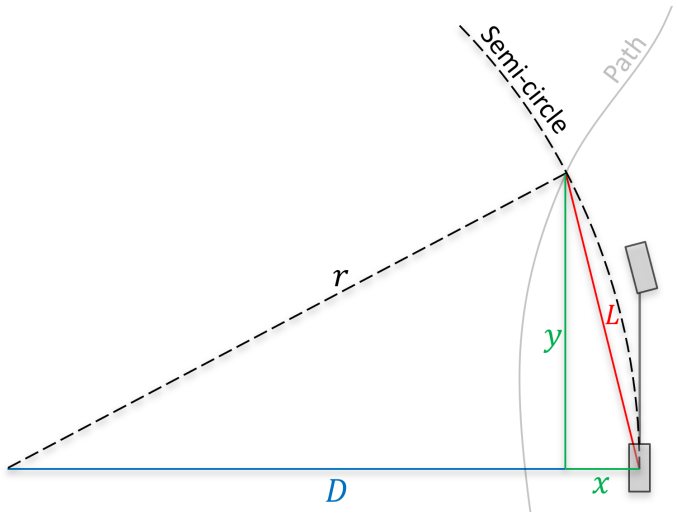


Figure 2.15: Geometrical description of the PPC defined by R. Craid coulter [1992].

Secondly, the following equations from R. Craid coulter [1992] are used to evaluate the semi-circle curvature that starts at the ego vehicle's rear axle.

$$D = r - x \tag{2.9}$$

$$(r - x)^2 + y^2 = r^2 \tag{2.10}$$

$$r^2 - 2rx + x^2 + y^2 = r^2 \tag{2.11}$$

$$2rx = L^2 \tag{2.12}$$

$$r = \frac{L^2}{2x} \tag{2.13}$$

As the radius r is the reciprocal value of the curvature κ we get:

$$\kappa = \frac{2x}{L^2} \tag{2.14}$$

The curvature κ is used to set the steering angle. The implementation from R. Craid coulter [1992] is only one of several other possibilities to utilize a PPC. In summary, this approach is defined by the following steps:

1. Find the lookahead point on the path.
2. Calculate the curvature κ.
3. Set the steering angle of the vehicle to the evaluated curvature.

The path feedback law $u(x)$ given by Equation 2.6 can be satisfied for a path with no curvature. For a path with curvature, it must be mentioned that during a typical path following the approach from R. Craid coulter [1992] will result in slight but acceptable path-tracking errors. The controllers' path tracking error or stability issues are considered using an adaptive lookahead distance adjusted by the vehicles' velocity.

2.1.4 Summary of the Automated Vehicle

As of now, the reader of this thesis should be able to understand the following matters:

- Terminology and Jargon related to AVs
- SAE levels of automated driving
- Basic development standards in the automotive industry
- HIL testing
- The automated vehicle structure according to: sense, plan, act

In the next chapter, the reader will be introduced to the current state of CP, focusing on AVs in urban environments.

2.2 Collective Perception

CP with special consideration of roadside ITS-S is a primary topic of this thesis. At first, naming conventions and background knowledge about roadside ITS is introduced, followed by related work. Several scientific articles: Smart infrastructure, Vehicle to Infrastructure (V2I), roadside infrastructure, hardware in the loop simulation, and automated driving are reviewed. Additionally, occlusion-aware path planning approaches for AVs are reviewed. Furthermore, related work regarding the research of CP is considered. A broad introduction to current work on CP is given.

2.2.1 Standards

ISO and European Telecommunications Standards Insitutute (ETSI) have developed the common architecture for ITS. Standardization is done in [International Organization for Standardization, 2020] while requirements for standardization activities are evaluated in the ETSI standard [European Telecommunications Standards Institute, 2010]. CP is summarized in Collective Perception Service (CPS) [European Telecommunications Standards Institute, 2019a]. CPS is defined as detecting road users or other obstacles by local perception sensors. These sensors are not restricted to one technology; camera, radar, or lidar systems are considered within CPSs. The goal of CPS is increasing the awareness of ITS-S. An ITS-S may be located at the roadside but is not restricted to it. As a result, a ITS-S can also be located within any road users' system. Concerning this thesis, vehicular ITS-S usually includes an OBU and roadside ITS-S a RSU.

CPS defines the Collective Perception Message (CPM) to properly utilize Intelligent Transport Systems Communications (ITSC). The CPM includes detailed information on the ITS-S sensor capabilities, such as position on the ITS-S, detection range, or its detected objects, including object classes such as vehicles, persons, and animals. A person sub-class is, e.g., a pedestrian or cyclist. Provided objects are given in the ITS-S reference frame. CPM is sent cyclically utilizing adaptive message generation rates to incorporate adaptive load scheduling. The broad focus of the CPM is to report changes in the dynamic road environment.

Another message type used in CP is the Cooperative Awareness Message (CAM). This message is designed to create awareness among road users. The CAM includes information about the traffic participants' position, dynamics, and other attributes. Road users are defined as, e.g. cars, trucks, bicycles, and pedestrians. [European Telecommunications Standards Institute, 2019b]

To summarize, collective perception utilizes, as the wording suggests, collectively intercepting the environment. This includes roadside perception as well as vehicle-based perception systems.

2.2.2 Related Work

Several articles on roadside infrastructure's impact, utilization, and feasibility are available. This Section summarizes the review of V2I technologies. After that, another review about AVs in a smart urban environment is presented. Then, several articles utilizing V2I and Vehicle to Everything (V2X) technologies are discussed.

Mapping and Deep Analysis of Vehicle-to-Infrastructure Communication Systems: Coherent Taxonomy, Datasets, Evaluation and Performance Measurements, Motivations, Open Challenges, Recommendations, and Methodological Aspects

This article from 2019 is about the mapping and deep analysis of V2I communication systems [Malik et al., 2019] elaborated on the benefits and challenges of these systems. Their findings are based on a review of a final set of 70 articles about V2I communications and data exchanges between vehicles and infrastructure. Four significant benefits are identified. Enhancing transport management and road safety can increase traffic flow efficiency and improve road safety. For privacy and security relations, anonymous authentication is provided, potentially reducing data privacy risks. Service applications are provided, including road safety services, support applications, and information about the traffic situation. On connection and communication topics, the direction routing protocol can be enhanced. This also applies to information exchange and improving communication efficiency between vehicles and infrastructure.

They describe four major concerns:

1. First, general traffic concerns related to the control of traffic flow, traffic information data management, and the reassurance of safety and efficiency of transportation.
2. Another concern is the reliability of mobility and connectivity. This includes achieving an efficient privacy scheme with low delays, establishing a steady connection, designing protocols that can adapt to frequent changes in network topology, a reliable line of sight communication, and the seamless handovers of vehicles between access points.
3. Concerns about privacy and security for data confidentiality, defense against misbehaving participants, cybersecurity risk mitigation, and preserving privacy.
4. The last of the four major concerns is related to safety. Data fusion and gathering, collision avoidance maneuvers, road safety applications, poor infrastructure, and intelligent vehicular control systems pose possible safety issues.

Their conclusion assumes that V2I communication is a fundamental technology for different vehicular applications. Furthermore, they suggest studies conducted on V2I communication systems to work on the elaborated benefits and concerns. [Malik et al., 2019]

Automated vehicles in smart urban environment: A review

The review article from 2017 about "Automated Vehicles in Smart Urban Environment" [Pereira et al., 2017] explains trends for the future of automated vehicles in smart urban environments. The current focus of research on highway-based automated driving and the growing role of infrastructure as an information provider is pointed out. They propose the integration of connected automated driving into a traffic system, where smart infrastructure interacts with a CAV to negotiate intentions and make intelligent decisions based on the goals of all road users.

Their review is based on 72 articles on automated vehicles, urban intersections, and cooperative intelligent transportation systems. As a result of their article, they highlight five major upcoming challenges that must be solved:

1. An accurate traffic state estimation system is available for urban areas, enabling the management of priorities for different types of vehicles. This can be done by individual advisory messages in traffic jams in emergency situations.

2. The second aspect is the necessity of algorithms that can deal with mixed traffic for trajectory optimization to generate routes and speed recommendations. Therefore, systems must be developed and evaluated for accurately predicting vehicle arrival times at intersections.

3. Another aspect is known by the keyword "interoperable algorithms," which means the collaboration of different algorithms like a traffic signal optimization and a green light speed advisory while still handling vehicle priorities.

4. The next trend is the change in the role and responsibility of traffic managers. Their work will increase the livability of cities regarding congestion, air quality, and a walkable city.

5. Future work focuses on multiagent systems, using data fusion and V2X communication to run advisory-based algorithms, allowing real-time traffic intersection control optimization.

These issues must be solved subsequently to achieve automated driving in urban areas. [Pereira et al., 2017]

Developing a Coordinated Signal Control System for Urban Ring Road Under the Vehicle-Infrastructure Connected Environment

Their work on advanced infrastructure systems shows how a coordinated signal control system can improve the traffic flow in an urban ring road system. The key parts of their solution are the generation of a guidance speed and adjustments of signal timings. They have simulated their system with the simulation software VISSIM. Their green wave algorithm shows how ITS can reduce delays in travel time on a ring road by 50%. [C. Ma et al., 2018]

Research on Intelligent Vehicle Infrastructure Cooperative System Based on Zigbee

In this work, Dong et al. [2019] propose an Intelligent Vehicle Infrastructure Cooperative System (IVICS). Their system is built upon an RSU and an OBU. The OBU is placed inside the car and transmits the vehicle's positions and speed using a Zigbee module. The RSU uses this information alongside traffic light information to generate speed recommendations for the vehicle. Zigbee is a wireless high-level communication protocol defined in IEEE 802.15.4, given in [Krausse and Konrad, 2014]. This technology consumes comparable low amounts of power, resulting in transmission distance of up to 100 meters, and is usually used in smart home systems. They evaluated their approach using a stochastic simulation where the vehicle drove across ten signalized intersections. They have used a co-simulation consisting of OMNeT++ and SUMO. Using their technique, they claim to be able to optimize the effect of a green

wave system. Overall, their ITS promises energy and emission savings of 26% to 27%. [Dong et al., 2019]

A Hybrid V2I and V2V Approach for Urban Traffic Management in Vehicular Networks

The framework used in this article is called DEcentralized System for Traffic Management (DESTINy). They have created a hybrid V2I/V2V system. An urban road network is equipped with several RSUs. A vehicle can request information from the closest RSU whenever necessary. If the RSU is not within the transmission reach of a vehicle, it uses another vehicle to increase its transmission distance. The RSUs act as a database for other road vehicles, which these can use to update their internal database. They have evaluated their approach based on SUMO and OMNET++ simulations. In their article, they propose to solve problems related to single server-controlled maintenance of whole urban road networks, which creates challenges for data processing and communication. Therefore, they show a decentralized solution. Their results highlight a reduction of the number of messages sent of 70% and a reduction in travel time of 26%. [Lourenco et al., 2019]

TRADER: Traffic Light Phases Aware Driving for Reduced Traffic Congestion in Smart Cities

Their approach shows how advanced traffic light systems, utilizing V2X communication, can reduce travel time in the urban road network. Their system is called TRaffic Light Phases Aware Driving for REduced tRaffic Congestion (TRADER). They simulate a solution that communicates traffic-light phases to "smart cars," using SUMO. Their approach enables these cars to limit their delays in the road networks by avoiding stopping for red lights by adjusting their velocity. As communication technology, they suggest IEEE 802.11p WiFi. In all their scenarios, they have achieved an average travel time reduction of 7.94%. The maximum reduction achieved in a scenario was 31.4%. [Rhodes and Djahel, 2017]

Infrastructure Assisted Automation of Lane Change Manoeuvre for Connected and Autonomous Vehicles

An infrastructure-assisted automated lane change maneuver for CAVs is presented in this article. Their computer simulations show promising results toward a sophisticated lane change protocol for CAVs. In this approach, infrastructure means several RSUs are used to communicate with the vehicles. These are then connected to a Local Traffic Controller (LTC), which manages the data. As a result, the LTC should inform the vehicles about open spaces to safely and efficiently perform lane change maneuvers on highways. Their initial simulative results, evaluated using SUMO and Traffic Control Interfrace (TraCI), only showed a slight increase in the average trip time of vehicles performing a lane change. Nevertheless, they have proven that their system can full fill its central goal: Increasing the safety of lane change maneuvers by ensuring an open space the vehicle can use to change lanes. [Lissac et al., 2019]

An Intelligent V2I-Based Traffic Management System

The authors of this work have performed simulations and real-world driving tests to see the impact of V2I technologies on traffic flow and safety. They suggest they are the first to test a situation, including four actual vehicles approaching an intersection, utilizing IEEE-802.11p communication technology. Their test vehicles were automated and controlled based on the magnetic marker technology presented by Lu et al. [2000]. Their results show a good performance regarding traffic flow in urban areas and decreasing the potential of collisions. They suggest that future work should include a test with traffic light control and lateral maneuvers. [Milanes et al., 2012]

Cooperative RADAR Sensors for the Digital Test Field A9 (KoRA9) – Algorithmic Recap and Lessons Learned

In the KoRA9 project, a three-lane highway was equipped with ten radar sensors, observing a total length of 180 meters of this highway. The information gathered by the roadside sensors was sent to a vehicle to enhance the vehicle's perception systems. A unique feature of their setup is the position of the roadside perception sensors. Several radar sensors were placed on the actual side of the road, behind the highways' guardrails. They have used two test vehicles to test and analyze their system. The demonstrator showed the feasibility of enhancing the perception of a vehicle. Nevertheless, they suggest shifting this technology from highways to urban areas to increase the benefits of the solutions by protecting VRUs, for example. [Sören Kohnert et al., 2021]

Providentia – A Large-Scale Sensor System for the Assistance of Autonomous Vehicles and Its Evaluation

This article shows a system that extends the FoV of AVs on highways. This work is similar to Sören Kohnert et al. [2021] as both extend the FoV and are implemented and tested at the same highway in Germany. The authors assume that the limited FoV of AVs increases the likelihood of accidents. To overcome limited observable areas of AVs, they install radar and camera sensors as roadside perception ITS-S. These are mounted on highway gantries and therefore enable proper observability as they can observe traffic from a top-down perspective. The implementation and demonstration at the actual highway show that tracking road users from roadside ITS-S is possible with high accuracy and reliability. [Krämmer et al., 2019]

Towards Cooperative Perception Services for ITS: Digital Twin in the Automotive Edge Cloud

A digital twin of the urban traffic environment may be used for cloud-assisted/cloud-controlled ADSs. Presented work is limited to proving the feasibility of their yore concept of "Cooperative Perception". Concerning naming conventions, this may be called CP. A digital twin of the real world is built in a simulation environment that includes detected objects from the actual world. A vehicle and several roadside sensing systems are used as ITS-S that share their information

with the digital-twin/simulation. As a result, the authors claim that their system achieves a more accurate representation of the traffic environment than any of its sub-systems. This article presents another example of an actual CP system that has proven its feasibility in the actual world. [Tihanyi et al., 2021]

Improving Vulnerable Road User Safety: Existing Practices and Consideration for Using Mobile Devices for V2X Connections

The authors of this work consider protecting VRUs using their so-called Cooperative-ITS (C-ITS). A gross number of severe VRU accidents are given as motivation for their approach. C-ITS or rather called CP in this thesis will contribute to reducing VRU accident according to the authors. A general system architecture that utilizes smartphones and mobile broadcasting technologies is proposed. Their main contribution is the framework that is assumed to reduce VRU accidents. In their work smartphones are used to localize the position of VRUs, such as a pedestrian or bicycle. Nevertheless, they state that VRU localization is one of the main challenges that must be solved to enable their architecture. The authors additionally emphasize that ITS applications are implemented based on cost-benefit principles. Therefore, they suggest deploying roadside perception at the most required sections of roads or in urban areas instead of everywhere. Vehicle to VRU (VRU) systems should be cost-efficient with consideration of the whole community by minimizing initial implementation costs. Much work is put into the vehicle's perspective in solving VRU accident issues. The given article criticizes this approach and suggests that ITS solutions shall be used. Work on V2X to protect VRUs is very limited. Three main challenges are given:

1. The incorporation of VRUs such as bicyclists and pedestrians in CP
2. Consideration of V2VRU communication standards.
3. Assessing the main challenges of V2VRU integration in ITS

The authors highlight VRU incorporation challenges (1). Especially in consideration with Section 2.2.1 the technical report [European Telecommunications Standards Institute, 2019a] considers so-called ITS-S concerning vehicles and roadside systems. They do not limit ITS-S to those but looking at the report, this is no proper standard for V2VRU communication (2). Lastly, the assessment of missing information on the challenges of integrating V2VRU systems into ITS is given (3). [Dasanayaka et al., 2020]

Cooperation of V2I/P2I Communication and Roadside Radar Perception for the Safety of Vulnerable Road Users

Similar to the work of Dasanayaka et al. [2020] this article considers the safety of VRUs. Their work utilizes an actual test setup with an actual pedestrian and car. A 79 GHz radar sensor is mounted at the roadside. This sensing system is connected to a 700 MHz communication terminal that can be considered a RSU in the context of CP. The combination of radar and transmission unit may be called ITS-S in this context. Pedestrians are equipped with 700 MHz terminals to enable Pedestrian to Infrastructure (P2I) communication and vice

versa. Their system can warn the VRU about a car occluded to the pedestrian. They have shown the feasibility of an actual roadside CP alike system. [Liu et al., 2018]

Cooperative-Intelligent Transport Systems for Vulnerable Road Users safety

In this study, the safety of bicycle drivers is highlighted. Actual bicycles are equipped with ITS-S, also called OBU. Their system sends CAM messages that can be received by other traffic participants such as cars. Information that is transmitted in the CAM is generated using a combination of Global Navigation Satellite System (GNSS) and Ultra Wide Band (UWB) localization system. The authors argue that during their tests, the safety of bicycle drivers was increased as the CAM was received by cars. These were warned if the bicycle was in a dangerous zone. Their C-ITS system was shown, and the feasibility of the setup was proven in a real-world environment. [Casademont et al., 2019]

Infrastructure-supported Perception and Track-level Fusion using Edge Computing

This work was published as a result of the MEC-View project [Robert Bosch GmbH, 2019]. In one of the work packages, roadside infrastructure perception was deployed at an intersection. Perception systems used in this work were cameras and lidar. These were connected to an edge computer that used the information to track objects in the road environment. The given article highlights the capabilities of MEC-View and presents different tracking algorithms that are evaluated on the edge computer. They proposed to send tracked objects to an AV and fused the AVs' tracks and the roadside perception tracks to achieve CP. This work is an example of an actual CP system that utilizes roadside ITS-S and vehicular ITS-S to enhance the AVs' FoV. [Gabb et al., 2019]

Implementation of a MEC-based Vulnerable Road User Warning System

Safety of VRUs is considered a result of human driving errors. A V2X/Vehicle to Pedestrian (V2P) system was developed and tested that creates a bidirectional communication channel between cars and pedestrians, utilizing the CAM. Their article suggests the feasibility of a CAM V2P system. Additionally, latencies in the architecture are analyzed to further prove this technology's maturity. The authors suggest that warnings should be sent to cars and VRUs. They do not provide evidence on proper tracking of VRU even though they are using GNSS, which is, from a technical point of view, not suitable for accurate and reliable tracking of VRUs in an urban environment. [Napolitano et al., 2019]

Networked Roadside Perception Units for Autonomous Driving

This article shows a CP system that utilizes vehicular ITS-S and roadside ITS-S. The authors suggest that CP such as V2X systems increase VRU safety. They highlight their architecture, "AutoC2X" which is published as open-source. This architecture combines Autoware [Kato, Tokunaga, et al., 2018][Kato, Takeuchi, et al., 2015] and OpenC2X [Laux et al., 2016]. Autoware is an open source AV driving software framework, and OpenC2X is an open vehicular networking software. They performed actual tests using AutoC2X, proving the

feasibility of the architecture. The Autoware AV driving framework is connected to roadside perception systems running the Autoware framework. AutoC2X supports CPM and CAM message communication. An evaluation showed that the perception of AVs is extended, latency limited to 100 milliseconds maximum. A high delivery ratio and low latency can be achieved even during heavy traffic load situations. The authors suggest that future work on AVs should incorporate perceived objects by roadside ITS-S into the driving algorithm of the AV. Their architecture and results provide evidence that CP is feasible and must be evaluated for actual AV driving tests. [Tsukada et al., 2020]

2.2.3 Summary on Collective Perception

Several articles on roadside infrastructure were reviewed. Roadside infrastructure is a general term and includes many different systems. The main findings from the articles discussed in this section are summarized below.

- CP such as V2I communication is a fundamental technology to cope with upcoming challenges for AVs and further work on V2I must be conducted [Malik et al., 2019].
- Infrastructure support gains importance as a service provider for ADS. It may help solve issues in traffic state estimation, speed recommendation, trajectory optimization, traffic light optimization, congestion, air quality, walkable cities, and real-time intersection control optimization [Pereira et al., 2017].
- Traffic congestion and, as a consequence, its emissions can be reduced by CPS [C. Ma et al., 2018][Dong et al., 2019][Lourenco et al., 2019][Rhodes and Djahel, 2017][Ruskin and Wang, 2002].
- In some studies driving efficiency and safety was increased by CPS [Lissac et al., 2019][Milanes et al., 2012].
- Real-world systems prove the feasibility of increasing the FoV of AVs utilizing highly accurate tracking of road users by roadside ITS [Sören Kohnert et al., 2021][Krämmer et al., 2019][Tihanyi et al., 2021].
- Several studies on VRU safety improvements using ITS suggest CP that utilizes roadside ITS-S[Dasanayaka et al., 2020][Liu et al., 2018] [Casademont et al., 2019][Gabb et al., 2019][Napolitano et al., 2019] [Denk et al., 2022].
- An open-source framework for CP ITS that enables CP for research groups is available [Tsukada et al., 2020].

Despite these articles' findings, there are many open questions to be answered, e.g., how exactly CP utilizing roadside infrastructure will help automated driving, especially in urban road networks concerning VRUs. The review showed that roadside ITS-S has vast potential and is worth exploring further.

2.3 Research Gap

Starting with a focus section on AVs, insights into automated driving, ADS (e.g., AVs, CAVs) were given. From there on, recent work on roadside infrastructure systems is given.

Most recent literature suggests that roadside infrastructure has a high potential to solve several problems for AVs, such as the need for safe and reliable operation in different environments. The authors of the work exhibited in Section 2.2 set high hopes for AVs and point out issues for them that could be tackled by ITS such as roadside ITS-Ss. Nevertheless, how such systems can aid the authors' goals is unclear. One can define the following research gap: it is unclear how exactly roadside infrastructure will help AVs in the future. Additionally, there are no experiments that explain what the measurable benefit of roadside infrastructure could be. What are the impacts of increasing AVs in mixed urban traffic environments regarding driving safety and traffic flow?

There are several ways to find conclusions on this topic. HIL testing is an accurate and effective methodology to achieve related answers and work on research gaps. Otherwise, one could use historical data to achieve assumptions. Nevertheless, no such data is available yet. Therefore, simulation is the option chosen in this work.

2.4 Research Questions

Considering the research gap, research questions are derived. With the introduction of special RSI, the safety of automated driving can be increased [Dixit et al., 2016]. The thesis intends to prove the assumption that overall driving safety in urban areas can be improved alongside a high level of traffic efficiency using the roadside CP. This is done based on HIL simulations, where parts of featured scenarios are supplied with fundamental hardware components like a vehicle computer or the automated vehicle itself.

As explained in the introduction and the state of the art, the support of AVs with additional sensors placed at the side of the road has a high potential to improve their overall driving behavior in several aspects. Additionally, one must consider the economic sensibility of CP utilizing roadside ITS-S. With careful consideration of all information given in this section, the following research questions arise:

RQ1: In which cases is a roadside ITS-S, utilizing CP reasonable for AVs?

RQ2: Is the comprehensive availability of roadside ITS-S meaningful for AVs in urban areas?

RQ3: How can smart infrastructure enhance the perception, planning, and control of an AV?

Research Question 1 CP utilizing roadside ITS-S must be validated for reasonability in several aspects, such as technical benefits for road traffic or social benefits, e.g., the safety of VRUs. Standard road safety systems must be qualified to aid a broad scale of society to justify their costs, or the investment in such systems must at least be accepted by society. RQ1 also should consider the effect of roadside ITS-S on the safety and reliability of traffic in urban road networks from a sub-microscopic point of view, meaning whether roadside ITS-S is relevant for automated driving or not.

Research Question 2 Comprehensive availability considers a full-scale deployment of roadside ITS-S to achieve full coverage of cities. RQ2 considers this technology's technical benefits regarding whether a large-scale deployment is meaningful for the operation of ADSs. The question deals with a macroscopic perspective on roadside ITS-S availability in urban environments.

Research Question 3 How exactly may roadside ITS-S for CP help CAVs in the future? There are several ideas, but validation on the foreseeable impact of roadside CP on ADS has yet to be available. RQ3 considers traffic safety, and efficiency impacts of ADSs in urban areas and the benefit of roadside ITS-S can add to CAVs.

Chapter 3

System Setup

This section describes the HIL setup used in this thesis. First, an introduction to the overall system is given, followed by a more in-depth explanation of the sub-components of the simulation framework.

3.1 Hardware in the Loop System Setup Architecture

A basic architecture of the **HIL Simulation Setup** is shown in Figure 3.1. The sytem setup is categorized as a sub-microscopic simulation. Microscopic traffic flow is simulated and combined with a 3D environment simulation, that is extended to a HIL using an actual Device Under Test (DUT).

Two significant parts are seen in Figure 3.1, the **Simulation Environment** and **DUT** indicated by dashed blue boxes on the left and right side, respectively. The primary data flow of the architecture is shown with arrows between sub-components. In combination, the simulation environment and the DUT are developed to jointly simulate a vehicle driving in a simulation environment, forming the HIL simulation setup. The simulation environment and the DUT given in Figure 3.1 are explained subsequently in this chapter.

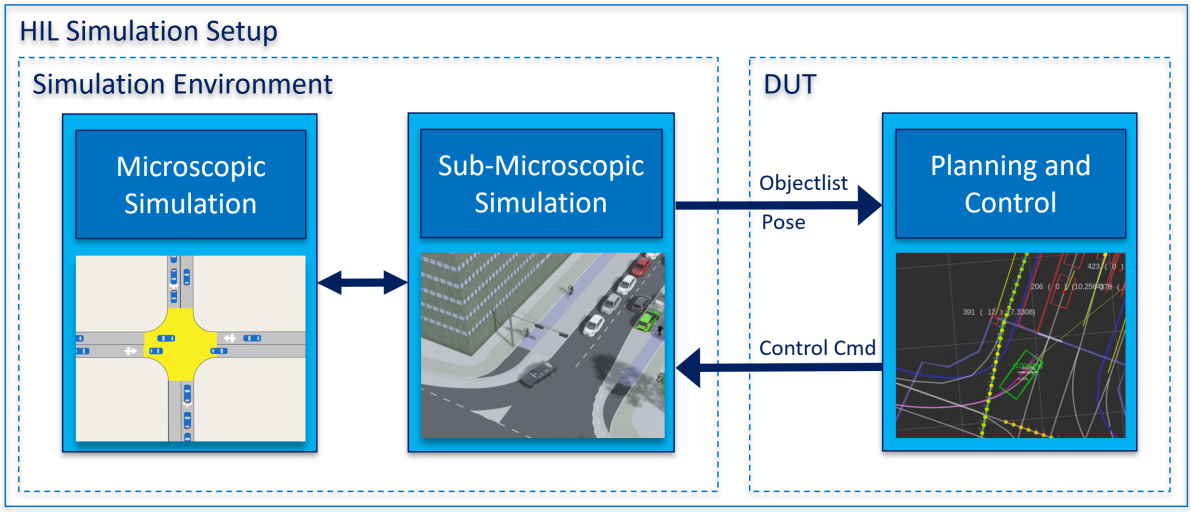


Figure 3.1: Overview of the HIL simulation framework, showing basic information about HIL components and data flow.

3.1.1 Simulation Environment

The simulation environment given by the left dashed box in Figure 3.1 is evaluated using a powerful desktop computer, explained in Chapter 3.1.2. Aimsun Next [Siemens Digital Industries Software, 2020a] is used for microscopic traffic simulation. The sub-microscopic simulation, shown in the middle of Figure 3.1 is realized using the software Simcenter Prescan [Siemens Digital Industries Software, 2020b]. This tool can simulate, e.g., different sensors or vehicle models. The sub-microscopic simulation allows perception systems to be installed on the Vehicle Under Test (VUT) and roadside ITS-S. A key advantage of combining Prescan and Aimsun Next is their communication interface. It enables the transmission of simulated vehicles in the microscopic simulation to the sub-microscopic simulation and vice versa out of the box. Therefore, this setup solves the problem of combining a complex traffic simulation with 3D perception simulations and vehicle models.

Microscopic Simulation

As mentioned before, the microscopic simulation is based on the traffic simulator Aimsun Next. This software can simulate large urban road networks for whole cities. Traffic demands are defined utilizing an Origin Destination (OD) matrix. The columns and rows of such a matrix are defined by a centroid configuration which contains several centroids. Figure 3.2 shows an image from Aimsun that shows several centroids. An OD matrix defines the number of traffic participants that travel from one centroid to another. Centroids apply to different road users, e.g., bikes, cars, or trucks.

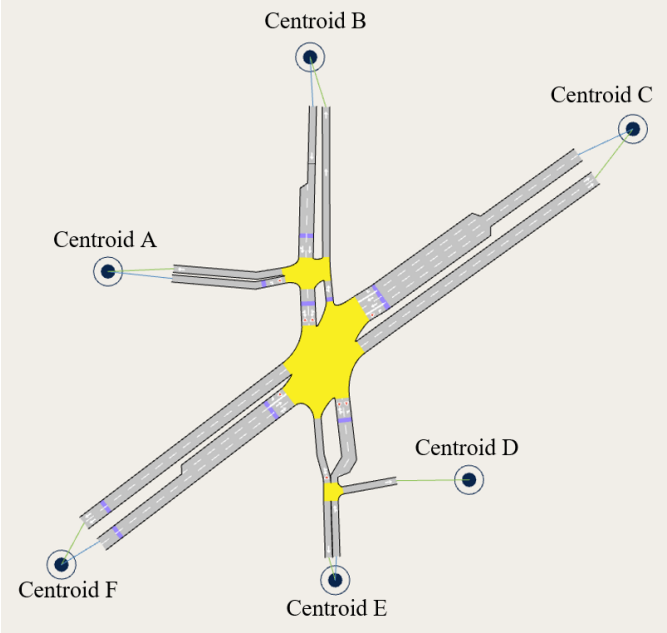


Figure 3.2: An image from Aimsun Next that shows the layout of a road network within Aimsun Next, including additional markers at the end of the roads. These markers indicate the corresponding centroids A-F used for OD matrices.

For example, in an OD matrix, the traffic demand for vehicles traveling from centroid E to centroid C can be defined. The demand for different kinds of road users is set within this matrix. The same applies to all other centroids in the road network.

When a simulation is started, the traffic participants can be visualized. Figure 3.3 shows a close-up of an intersection including cyclists, pedestrians, and cars. A closer look at the cyclists reveals that they drive on designated bike lanes unless they enter the shared space inside the intersection. For the pedestrian simulation, one can see the green dashed area inside the intersection, which indicates that this area is available to them for the time they are allowed to cross. As shown in Figure 3.3, cars are also shown in the simulations' visualization. Additionally, the marked car in Figure 3.3 renders a brake light and the turn indicator, which are observed on other vehicles entering the intersection as well.

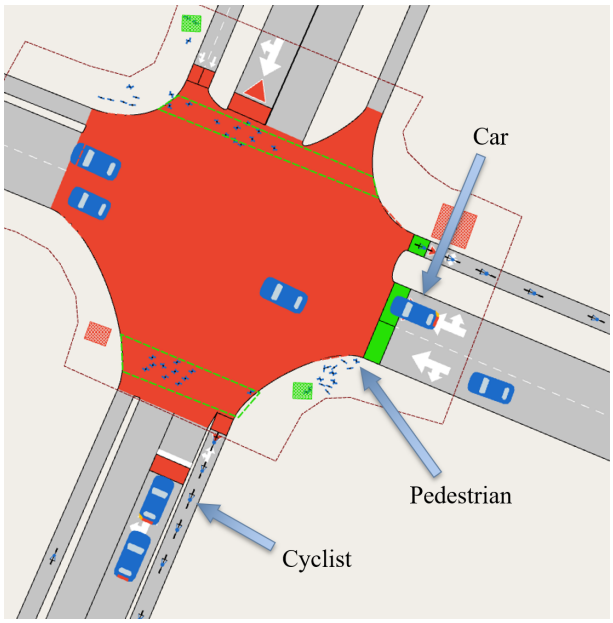


Figure 3.3: An image from the simulation software Aimsun Next, showing a simulation that highlights cars, pedestrian, and bikes.

Car-Following Model: Car-following is based on the GIPPS [1981] car-following model and has a high impact on the results of the simulations in this thesis. Most car models are derived from assumptions based on real-world driving. Such an assumption may be driving at a desired velocity, considering driving comfort for acceleration and braking maneuvers. Essential kinematic aspects are assessed, e.g., the maximum braking distance for the current velocity. In the best case, a car model reproduces the driving behavior of an Adaptive Cruise Control (ACC). A simplified car following model based on the book from TREIBER and KESTING [2010] that utilizes a complete and accident-free car model is given in the following.

The guaranteed absence of accidents is leveraged by an equation that uses the maximum distance to and velocity of a leading vehicle v_l. As a result, a safe velocity $v_{safe}(s, v_l)$ is derived as follows. The leading vehicle's brake distance Δx_l is given by Equation 3.1, where b is a constant braking acceleration.

$$\Delta x_l = \frac{v_l^2}{2b} \tag{3.1}$$

For the ego vehicle that follows the leading one, overall brake distance Δx_{ego} includes the reaction and brake distance,

$$\Delta x_{ego} = v_{ego}\Delta t + \frac{v_{ego}^2}{2b}. \tag{3.2}$$

Therefore we can say that the stopping distance s, considering that s is greater than the starting distance s_0, must satisfy,

$$s - s_0 \geq \Delta x_{ego} - \Delta x_l \tag{3.3}$$

$$s - s_0 \geq v_{ego}\Delta t + \frac{v_{ego}^2}{2b} - \frac{v_l^2}{2b} \tag{3.4}$$

to achieve a safe stopping position. For the safe velocity, one can conclude: $v_{ego} = v_{safe}$. Solving the resulting quadratic equation Equation 3.5:

$$v_{safe}(s, v_l) = -b\Delta t + \sqrt{b^2\Delta t^2 + v_l^2 + 2b(s - s_0)}. \tag{3.5}$$

Concerning Equation 3.5 for safe velocity, the Gipps-Model can be formed:

$$v(t + \Delta t) = min[v + a\Delta t, v_0, v_{safe}(s, v_l)]. \tag{3.6}$$

Considering the model's simplicity, it achieves good results and is the model of choice for several simulation frameworks.

Lane-Changing Model: Lane-changing is based on the work form GIPPS [1986]. The model analyses whether it is necessary to change the lane. This may be the case if a leading vehicle is slower but it depends on several other factors, such as the vehicle's position in the road network. Lane changing can be divided into three questions that result in a decision:

- Is it necessary to change lanes?
- Is it desirable to change lanes?
- Is it possible to change lanes?

The lane change model [GIPPS, 1986] is designed to be used to work in combination with the car-following model [GIPPS, 1981]. The flowchart of the lane change model and in-depth descriptions can be reviewed in the corresponding paper [GIPPS, 1986].

The car following model and lane change model used in this work is based on the implementation from Aimsun Next and can be found in the Aimsun User Manual [AIMSUN NEXT, 2023].

Sub-Microscopic Simulation

The simulation software covers sensing other traffic participants in the ego vehicles' proximity. Perfect detection is considered for simulations. Therefore, the simulation assumes a MOTA and MOTP of 100%. This generates an object list that the planning algorithm can evaluate without an object recognition disturbance that must be considered in real systems.

In the introduction to the HIL simulation, the depth of different types of HILs is explained. This simulation setup does not include perception sensor evaluation, meaning that this work is not intended to analyze the performance of detection or tracking algorithms. This is a different research topic. As explained, an actual sensor is prone to noise forwarded to algorithms that detect, track, and predict objects, adding disturbances to simulation results that are part of a different research field and do not aid the goals of this thesis. This work deals with the FoV extension of AVs and not object detection.

A primary feature of this sub-microscopic simulation is the 3D environment simulation. In Section 3.2.3, more details on perception system modeling from the point of view of the ego AV are explained. Additionally, Section 3.3 describes the roadside infrastructure simulated within the sub-microscopic simulation system.

Simulation Environment Co-Simulation

The simulation systems' microscopic and sub-microscopic components work closely together. Both share an interface that is delivered with the respective software. Simulation time is synchronized, enabling both to share data at appropriate time steps. Figure 3.4 shows a qualitative data flow schematic between the microscopic and sub-microscopic simulation environment. Both interchange information about the actors each of them evaluates. In detail, this means the pose of the VUT is simulated within the sub-microscopic simulation and sent to the microscopic simulation. Vehicles evaluated by the microscopic simulation can react to the VUT, evaluated by the sub-microscopic simulation. Vice versa, the VUT can react to vehicles generated by the microscopic simulation.

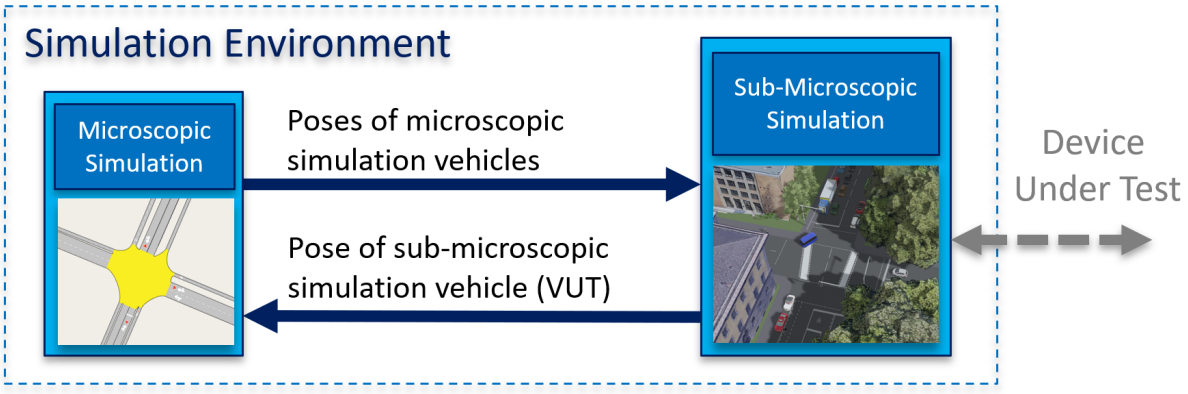


Figure 3.4: A close up image of the simulation environment setup is shown. The arrows indicate the data flow between the micro and sub-microscopic simulation and the connection to the DUT.

Simulation Software Reasoning

Although traffic simulation could have been done with SUMO [Deutsches Zentrum für Luft- und Raumfahrt, 2001] in combination with the sub-microscopic vehicle and perception simulation software CARLA [*CARLA: An Open Urban Driving Simulator* 2017], this was no choice due to the higher maturity of the Prescan, Aimsun combination. The second option is suitable and would allow running the simulation utterly open source instead of the combination used in this thesis, which employs commercial simulation solutions that speed up the development process. Pilz et al. [2019] and Tong et al. [2020] analyzed different simulators. In their work, they have compared Prescan and Carla among other simulators.

Pilz et al. [2019] evaluated simulators considering suitability for, e.g., NCAP scenario simulation. Table 3.1 shows his findings for the given aspect. **IF** considers the scenario interface compatibility, which states that Carla has an excellent interface and Prescan doesn't have one implemented yet. Further, the author suggests that implementation will take a lot of effort by adding the notation poor (-) to the table entry. This is reasonable as Carla has, e.g., an OpenSCENARIO[1] interface, and Prescan does not. The access to **Ego** vehicle data is very good for both. The access to **Non-Ego** vehicle data and **Ped**strians is considered worse in Prescan. This downside is eliminated by running Prescan in a Co-Simulation with Aimsun, which generates proper pedestrian and non-ego vehicle data. Prescan is better in **Sens**or simulation but had its downsides in **Vis**ualization, as Unreal Engine rendering was not introduced in 2019, which is available since 2021. Prescan level of **Phy**sics was not rated by the authors, and the **Cost** evaluation is precise, comparing open-source and commercial software.

Table 3.1: Suitability of Carla and Prescan for, e.g., NCAP scenario simulation. The ratings are as followed: very poor (–), poor(-), not rated or irrelevant (o), good (+), very good (++), not implemented (i)

Simulator	IF	Ego	Non-Ego	Ped	Sens	Vis	Phy	Cost
Carla	++	++	++	+	+	+	+	++
Prescan	i-	++	+	o	++	-	o	- -

Tong et al. [2020] extended the results of Pilz et al. [2019], but did not consider scenario simulation, instead they were looking at suitability for planning simulation. Additonally, [Tong et al., 2020] explicitly added Aimsun and Sumo. Their results can be seen in Table 3.2. The last two entries of Table 3.2 are added as a combination of Cara&Sumo and Prescan&Aimsun each combining the strength of the respective simulator. This work considers a **V2X** application and therefore Prescan is the exceptionally good in simulating such application, while Carla has no native interface. There is one other simulator that has scored a very good result named IPG CarMaker[2] which is a commerical solution as well. Considering scale and flow of **Traf**fic flow simulations both co-simulations Cara&Sumo and Prescan&Aimsun have a very good result.

[1] A standard for defining the content of traffic simulators https://www.asam.net/standards/detail/openscenario/
[2] https://ipg-automotive.com/de/produkte-loesungen/software/carmaker/

The Quality of **Non-Ego** driver modelling is good for both combinations. As given in PILZ et al. [2019] Prescan performs better for **Sens**or simulations, while Carla has an advantage in **Vis**ualization. The quality of **Ego** vehicle dynamics cannot be compare, because it was not rated.

Table 3.2: Suitability of Carla and Prescan for, for planning simulation. The last two rows are a combination of the best features of the corresponding simulation software. The ratings are as followed: very poor (–), poor(-), not rated or irrelevant (o), good (+), very good (++), not implemented (i)

Simulator	V2X	Traf	Non-Ego	Sens	Vis	Ego
Carla	i	+	+	+	+	+
Prescan	++	+	+	++	-	o
Sumo	i	++	+	–	–	–
Aimsun	i	++	+	–	–	–
Carla&Sumo	i	++	+	+	+	+
Prescan&Aimsun	++	++	+	++	-	o

In Summary the work of PILZ et al. [2019] and [TONG et al., 2020] showed that for this work, a co-simulation of Prescan&Aimsun is superior to a setup that uses Carla&Sumo, because:

- This work considers collective perception and Prescan performs better regarding sensor simulation.
- The main objective of this work considers simulation of V2X systems, which is available in Prescan.

The most significant downsides of the Prescan&Aimsun co-simulation are:

- Visualization does not look as good as to one of its competitors, e.g., Carla.
- No standardized scenario specification available.
- Licenses for Prescan and Aimsun must be purchased or acquired via a license agreement[3].

[3]Aimsun offers a free postgraduate edition of their software.

3.1.2 Device Under Test

The simulation environment is extended by a DUT, where perception, planning, and control algorithms are processed. For this work, two different DUT setups are used, which means that the hardware deployed during HIL test simulations varies. Each of them is explained in this section.

Vehicle Computer in the Loop

Figure 3.5 shows a simplified data flow of a HIL architecture used in this work. This setup adds an actual vehicle computer as DUT to the simulation system. Regarding HIL system depth, a vehicle computer in the loop simulation is a CoHIL, explained in Section2.1.2. Basic data flow for the **Simulation Environment** is shown in the box on the left side of Figure 3.5, where a virtual vehicle model is evaluated. The virtual vehicle model generates the current pose[4] of the VUT. In addition, the simulation environment runs virtual vehicle sensors to perceive virtual objects and send these to the communication interface.

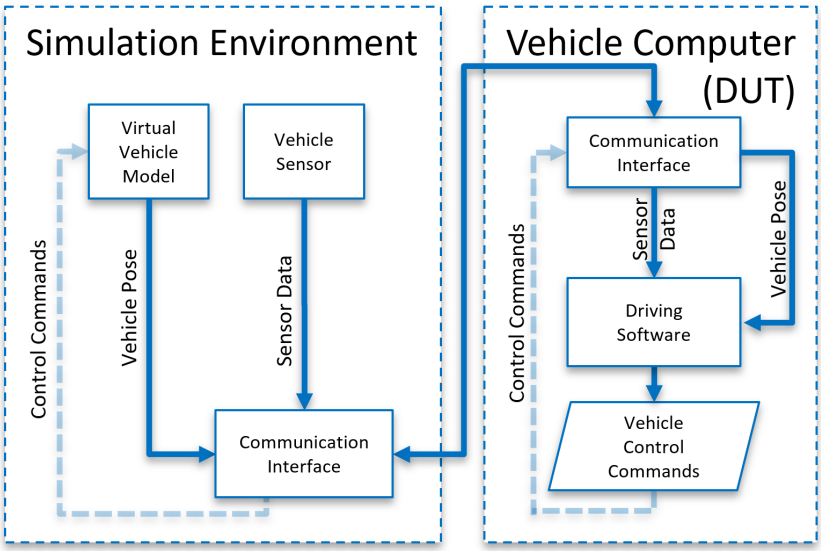


Figure 3.5: Simplified data flow of the HIL simulation with the actual vehicle computer as DUT in the loop.

The **Vehicle Computer (DUT)** is shown on the right side of Figure 3.5, extracts information from the simulation environment, and processes it using driving software. Vehicle control commands are derived and sent back to the virtual vehicle model, which can reevaluate the Vehicles' pose. To see computing capabilities used when performing these simulations, specifications of this computer are given in Table 3.3. The computer employs Ubuntu 18.04 and the middle-ware Robot Operation System (ROS) [Open Source Robotics Foundation, 2018].

The AVs' driving software is based on the Autoware Toolbox [Miura et al., 2019]. This system has already been tested by Kato, Tokunaga, et al. [2018] in real-world driving

[4]The pose of a robot describes its current position (x,y,z), velocity (x,y,z), acceleration (x,y,z), e.g.

scenarios. Additionally, the Autoware system is upgraded using the latest releases of Open-Planner from DARWEESH et al. [2017] and additional features developed for this thesis. These upgrades enable incorporating the most recent state of open-source automated driving algorithms. The software used for each simulation study is referenced in the corresponding section.

Table 3.3: Specification of the vehicle computer used for the simulations

Name	Nuvo-7160GC
CPU	Intel Core i9-9900 3,1 GHz
RAM	32GB DDR4-2666MHz
GPU	GeForce GTX 1660 Ti - 6 GB

Complete Vehicle in the Loop

The simulation system is extended for special tests by adding an actual vehicle into the loop. Figure 3.6 shows the systematic data flow of this approach. The simulation environment responsible for sensor data generation and microscopic traffic simulation is given on the left side. The Vehicle sensor extracts objects from the virtual environment as input data for the AV driving software. The DUT, which is shown in Figure 3.5, is divided into two sub-components, the vehicle computer and the actual vehicle in Figure 3.6. The actual vehicle model replaces the virtual vehicle model used in the vehicle computer test setup.

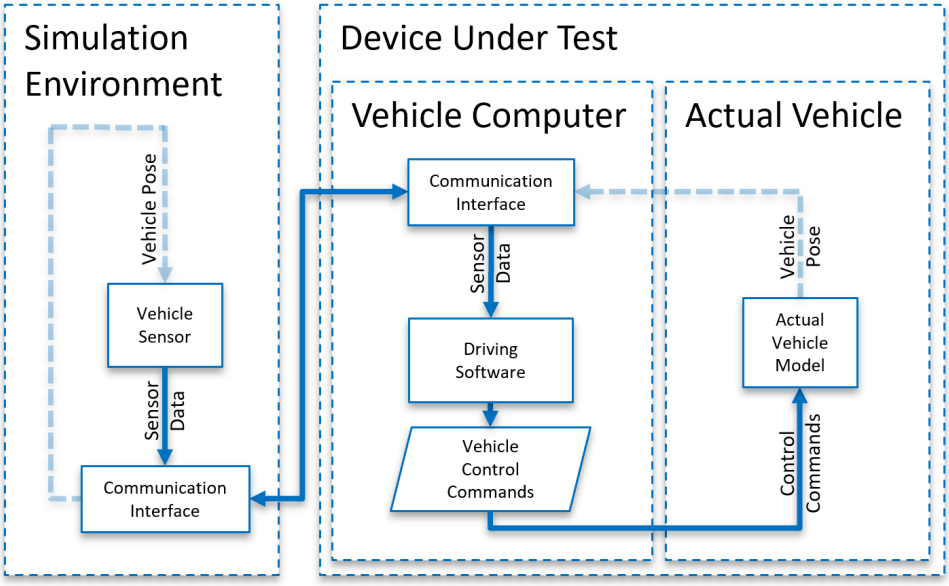


Figure 3.6: Data flow of the hardware in the loop simulation with the actual vehicle in the loop.

3.2 Automated Vehicle Setup

The automated vehicle considered in the simulations is described in this section. At first, vehicle dynamics are introduced. Then perception planning and control of the AV are described. The whole vehicle is utilized according to the state of the art section on the automated vehicle given in Chapter 2.1. The precise utilization is described below. This description explains how the overall vehicle setup is utilized for different simulations performed in this work.

3.2.1 Actual Vehicle

If an actual vehicle is used for a HIL simulation, the research vehicle of the University of Applied Sciences (UAS) Augsburg is used. This vehicle is a modified version of a BMW active hybrid 3. The modifications are introduced for the actuators and sensors of the vehicle. Several components are added to the vehicle, which enables research regarding sense, plan, and act. Figure 3.7 shows an image of the research vehicle used in this thesis. The cars' roof feature a GNSS system on the rear and a 360-degree LiDAR in the middle. Additionally, cameras covering 360 degrees surround view and a front facing long range radar is installed in the car. Considering the paragraph about sensing systems in Chapter 2.1.3 the research vehicle covers offers research possibilities for almost all mentioned systems.

Figure 3.7: Image of the research vehicle of the UAS Augsburg, which was used for the HIL simulation studies

Figure 3.8 shows the trunk of the research vehicle, containing several additional components. Within the microautobox given in Figure 3.8, lateral and longitudinal actuator control of the VUT is utilized. The vehicle computer shown in Figure 3.8 calculates the trajectory sent to the microautobox. Finally, the image shows the OBU used for RSI communication.

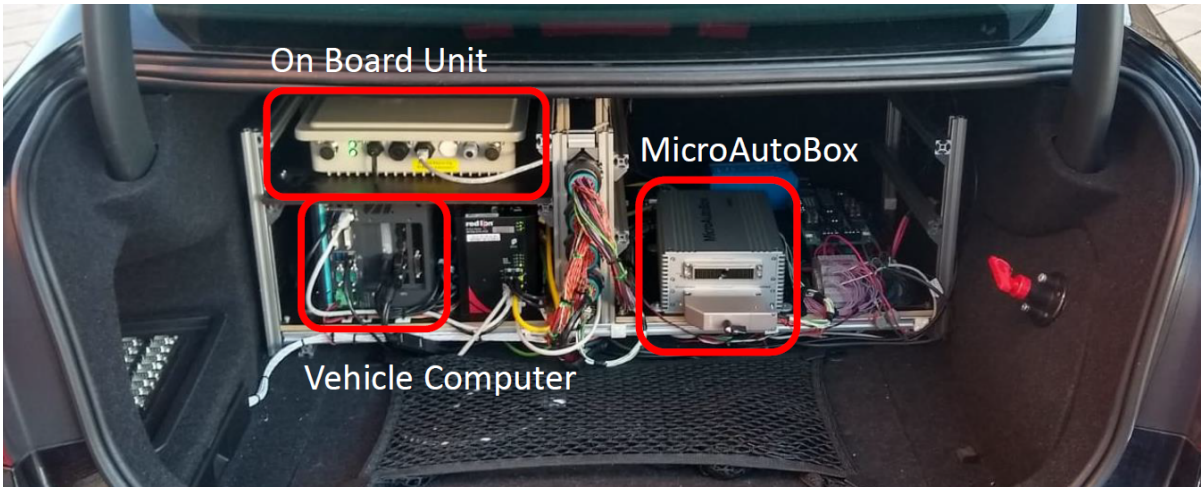


Figure 3.8: Image of the trunk of the research vehicle with the hardware configuration that was used in article [PECHINGER et al., 2020]

3.2.2 Simulated Vehicle

The evaluation model of the vehicle used for simulation is based on a simple dynamics model. It includes an engine, transmission, final drive ratio, chassis, and automatic gear shifting. The most interesting part is the chassis, which significantly impacts the vehicle dynamics, as it includes a bicycle model [KONG et al., 2015] extended by roll dynamics and a linear tire model. As an example for evaluating the vehicle state, the kinematic bicycle model from KONG et al. [2015] can be used. The continuous time equation for the position state x and y are given by

$$\dot{x} = v\ cos(\psi + \beta), \tag{3.7}$$

$$\dot{y} = v\ sin(\psi + \beta), \tag{3.8}$$

and are located at the center of mass, as shown in Figure 3.9. The continues time equations for the heading ψ, velocity v and the slip angle β are,

$$\dot{\psi} = \frac{v}{l_r} sin(\beta), \tag{3.9}$$

$$\dot{v} = a, \tag{3.10}$$

$$\beta = tan^{-1}\left(\frac{l_r}{l_f + l_r} tan(\delta_f)\right), \tag{3.11}$$

where a is the acceleration in the center of mass and has the same direction as the velocity v. The control inputs for the model are the acceleration a and front wheel steering δ_f. δ_r is zero as the vehicle utilizes no rear-wheel steering. A Table that gives an overview of all dynamics model parameters is reviewed in Annex 1.

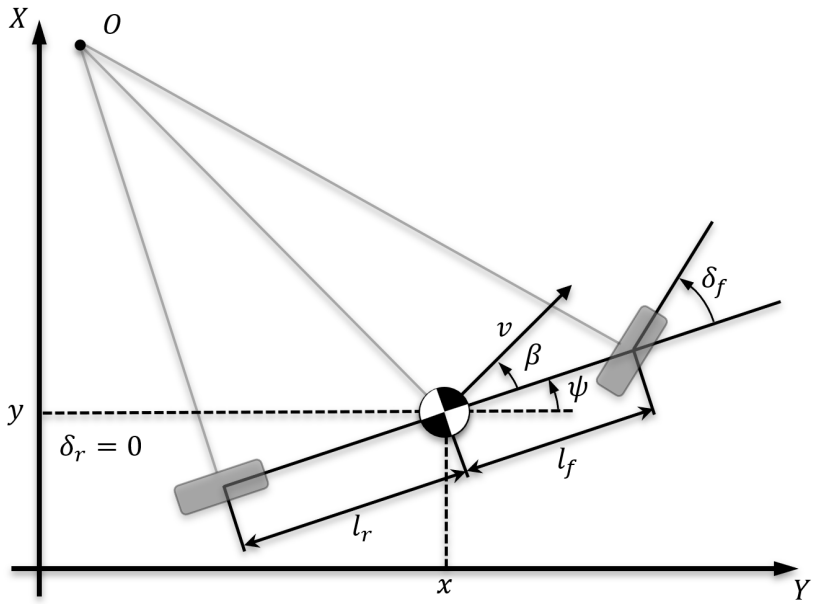


Figure 3.9: Representation of the kinematic bicycle model used to describe Equations 3.7, 3.8, 3.9, 3.10 and 3.11.

Figure 3.10 shows a simplified diagram of the complete vehicle model that includes the bicycle model. In the sub-microscopic simulation, the virtual vehicle model was given a desired throttle, brake, and steering angle to evaluate the pose of the VUT. The bicycle model, including roll dynamics and tire model, are handled inside the virtual vehicle model and are one part of the whole virtual vehicle model.

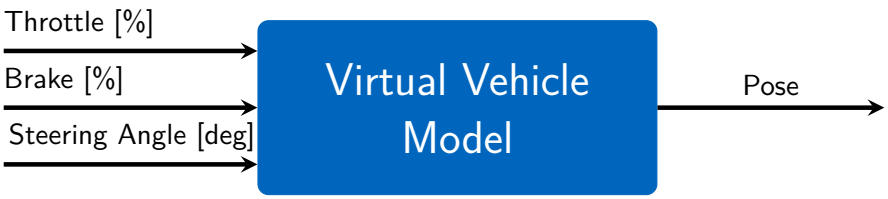


Figure 3.10: Simplified block diagram of the complete virtual vehicle model used in the simulations that utilizes throttle brake and steering angle as inputs to generate the pose as output.

3.2.3 Perception

Perception of the ego AV is based on a sensor setup with a 360-degree view. The vehicle can detect objects in its environment within a specified maximum range. The sensing system could be anything like LiDAR, Camera, or, RaDAR as a Technology Independent Sensor (TIS) is used. The output of the perceiving sensor system is given in Table 3.4. From range and detection angles Theta and Phi, an object's position in the ego vehicle's frame is derived. Object IDs are utilized for object tracking and sensor data fusion. Velocity, Heading, and

Size are crucial additional sensor readings for AVs' driving algorithms. Gaussian Noise can be applied to the detection range and angle from the ego to the detected object to simulate sensor noise

Table 3.4: Virtual sensor output data and description.

Type	**Description**
Range	Distance from the sensor axis system to the object detected [m]
Theta	Detection angle in azimuth [degs]
Phi	Detection angle in elevation [degs]
ID	Numerical ID of the object detected
Velocity	Longitudinal velocity of the object detected [m/s]
Heading	Heading of the detected object [degs]
Size	Object size in x and y direction [m]
Noise	Gaussian Noise on detection range and angle

In Figure 3.11, an image from an explanatory simulation is shown to illustrate occlusion in the simulation. The Ego vehicle is equipped with a 360-degree sensor system, as mentioned above. A building is placed in the simulation to generate occlusion. As a result, two out of five objects are not detected by the perception system. Green bounding boxes notate the other three traffic participants that are detected.

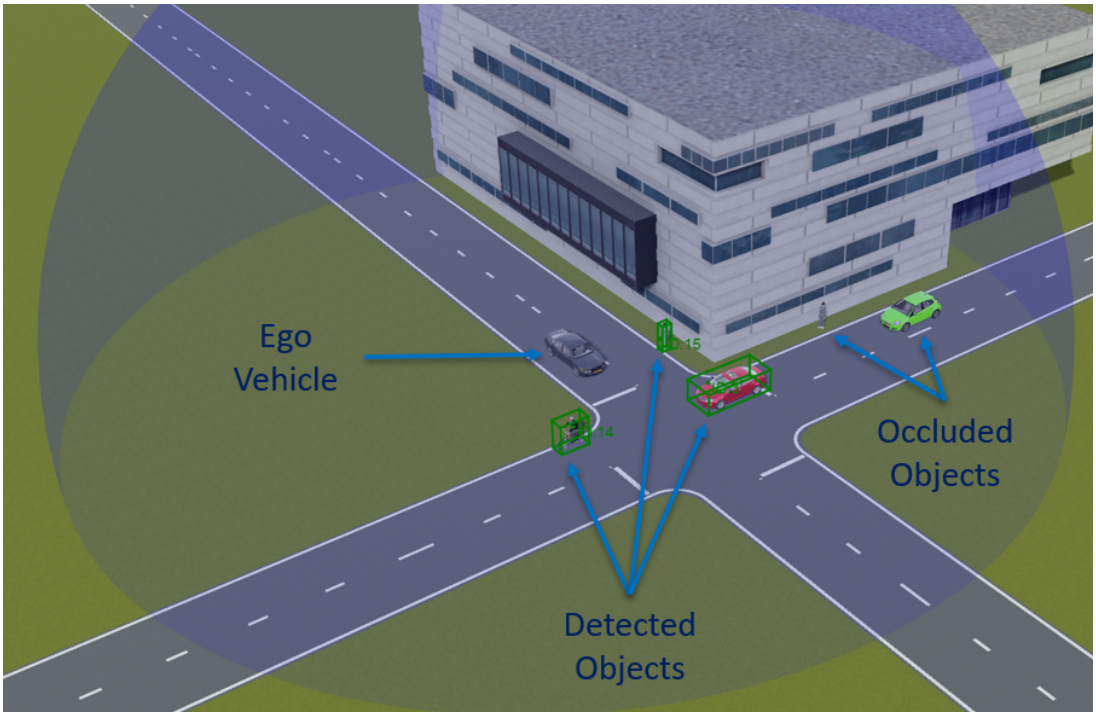


Figure 3.11: Image of a simulated environment showing perception capabilities regarding occlusion.

3.2.4 Planning

One of this thesis's most critical factors is the ADS's planning algorithm. The overall performance regarding, e.g., driving safety or driving comfort, of an ADS is highly influenced by it. In this thesis, OpenPlanner [Darweesh et al., 2017] is used and improved by new features. The significance of the AV stack cannot be understated. It ultimately has the most significant impact on overall driving behavior. As a result, this work extends the capabilities of OpenPlanner, given in a separate repository [5].

Behavior Planning

A finite state machine defines the overall behavior of the ADS using OpenPlanner. In order to give a brief understanding of the states and transitions, Figure 3.12 combined with Table 3.5 can be reviewed. The given example of the state machine is simplified. Some transition states, excluded from Figure 3.12, do not aid the understanding of behavioral planning. The behavior planning differs from the original version of OpenPlanner, with some minor and one significant change, introducing the "Yield" state. This state allows the AV to operate more cautiously inside urban intersections. An improper response to a cyclist in an intersection is a specific issue the ADS encounters without this state. Meaning: if the ADS performs a right turn while a bike drives straight, the ego vehicle will yield to the cyclist with this new state. The "Follow" state of OpenPlanner is not applicable in such a situation.

The state machine in Figure 3.12 is given qualitatively. When the vehicle is started, it can transition to the forward state. In this state, the vehicle will start driving at the maximum allowed speed given by the road network. If the AV detects an object on its planned trajectory, it will change its state to follow. As the state already says, the driving system will follow the object on its path. The VUT can transition from either of those states to a light stop, sign stop, or yielding state, depending on the vehicle's internal state. The yield state is introduced to the software for this thesis. The AV can detect whether it is inside an intersection and has an obstacle on its current path. If this is the case, the AV will transition to the yielding state, changing the driving behavior to an extra cautious mode.

Table 3.5: State machine description

State	Description
Forward	Drive at maximum allowed speed
Follow	Follow an obstacle ahead
Light Stop	Stop at the traffic light or leading vehicle
Sign Stop	Stop at stop line or leading vehicle
Wait	Wait until state transition criteria are met (traffic light green/stop line wait time)
Yield	Stop for an obstacle in our way inside an intersection

[5] github.com/MathiasPechinger/common/releases/tag/ITSC2021

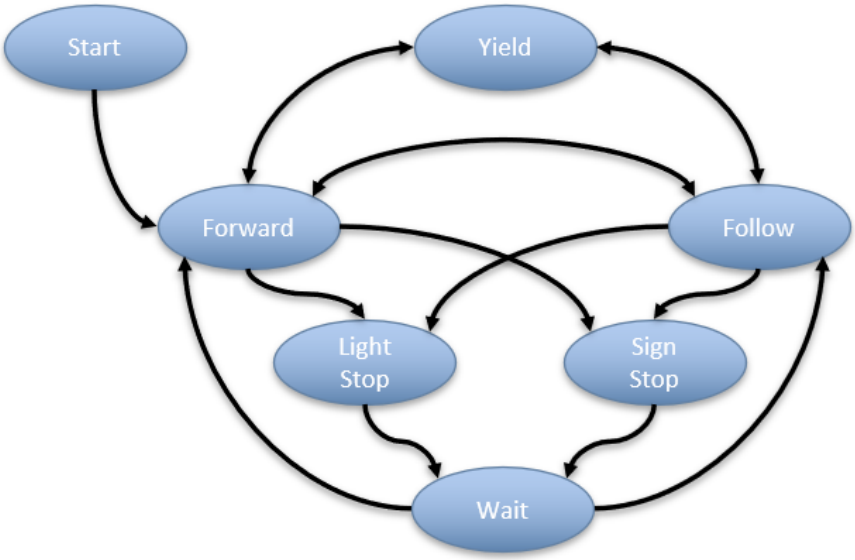


Figure 3.12: Simplified behavior state machine of the planners behavioral architecture

Global Planning

Global planning facilitates the generation of a path from the ADSs' starting point to a goal point, as explained in the state of the art on the AV (Section 2.1). Using the OpenPlanner framework, the global plan is achieved by searching for the shortest path toward the goal in an OpenDrive or Lanelet2 map.

Local Planning

Local planning is based on the approach described in [Darweesh et al., 2017]. The software used in this thesis differs from the original release. The driving software has a specific feature responsible for lateral object avoidance, called "swerving." This mode was not used in local planning; instead, the vehicle would always apply center-line driving, navigating along its predefined route, which was extracted from the urban road network. Many code sections were changed in order to account for more driving comfort. OpenPlanner version 1.13, for example, did exceed the maximum acceleration boundaries defined in the planning parameters. The parameterization of OpenPlanner is extensive. Refer to Annex 2 for an overview of all configurable values.

3.2.5 Control

The control is based on a PPC [R. Craid coulter, 1992] for the lateral control, as explained in the paragraph local feedback control in Chapter 2.1.3. A common Proportional Integral Derivative (PID) controller utilizes longitudinal control. This methodology is efficient and reliable considering path tracking of ADSs.

3.3 Roadside Infrastructure

There are several different types of RSI. In this thesis, roadside infrastructure relates to 802.11p wifi for wireless connection in combination with perception sensors as roadside ITS-S. For tests with real hardware the RSU from Siemens shown in Figure 3.13 is used.

Figure 3.13: Image of a RSU that was installed at Hamburg Hafencity, including the intersection and mounting pole.

Figure 3.14 shows a close-up image of the RSI CP installation highlighted in Figure 3.13. In the middle of Figure 3.14 the RSU can be seen. Attached to the RSU, one can see four long white antennas mounted vertically to the RSU. These provide access to the 802.11p (ITS-G5) wifi system. Radar sensors are shown on the left and right sides with respect to the RSU. The top of Figure 3.14 shows a black 360 degree lidar sensor.

Figure 3.14: Closeup of a RSU with two grey radar sensors and one black 360-degree LiDAR Sensor on top.

No actual perception sensor was used for simulations evaluated in this work. Perceived objects are always generated virtually. As mentioned in previous sections, this work considers perfect sensors; therefore, any kind or combination of sensors may be used to detect objects. All

parts of the roadside infrastructure are interchangeable by any technology available to perform either the task of data transmission or object detection. Therefore several communication technologies such as ITS-G5 or 5G may be used interchangeably.

AV or CAV Whenever an ADS without the support of a roadside ITS-S/RSI is discussed in the following chapters, it is referred to by the acronym AV. If an AV utilizes roadside ITS-S/RSI it is referred to as CAV as it is connected to the roadside system.

- CAV: Connected AV utilizing roadside ITS-S.
- AV: Standalone AV without roadside ITS-S support.

Chapter 4

Initial Investigation on Roadside ITS-S

In this Chapter, an analysis of bicycle accidents in Munich is evaluated. In Section 1.2, it was pointed out that there is a motivation to protect bicyclists by roadside ITS-S, considering occluded areas. This Chapter studies historical accident data concerning bicyclists in Munich. These locations are analyzed, looking for reasons for those incidents. Then, different types of occlusion are introduced and reviewed. Lastly, the economic aspect of roadside ITS-S is assessed. At this Chapter's end, the potentials of roadside ITS-S shall be apparent.

4.1 Accident Locations and Potential Reasons

When evaluating occlusion on a sub-microscopic 3D level, it is pleasant to choose one intersection as 3D modeling and simulation are very time-consuming. Therefore, the question of which intersection to choose arises.

Munich has hundreds of intersections, so a methodology must be applied to find the most relevant ones. The introduction explained that occlusion has the highest impact on the number of accidents in urban intersections. Additionally, cyclists have the highest participation rate in accidents. In conclusion, it is productive to focus on accidents of cyclists. This focus has the highest potential to improve safety for road users when roadside ITS-S is applied. The next question is: "Where do accidents of cyclists occur?"

4.1.1 Bicycle Accident Locations in Munich

To get an answer to this question, data from the German "Unfallatlas"[1] is used. This data contains accident information for Germany from 2016 to 2020. This information includes geo-referenced position, weekday, month, and severity category of accidents. More importantly, it includes the type of road user involved in an accident. As a result, it is possible to show all accidents that involve cyclists in Munich on a map. Accidents are categorized into three levels:

1. Slightly injured
2. Seriously injured
3. Fatal

[1] https://unfallatlas.statistikportal.de/_opendata2022.html

Figure 4.1 shows all accident locations in the area of Munich. From 2016 to 2020, there were a total of 11600 bike accidents.

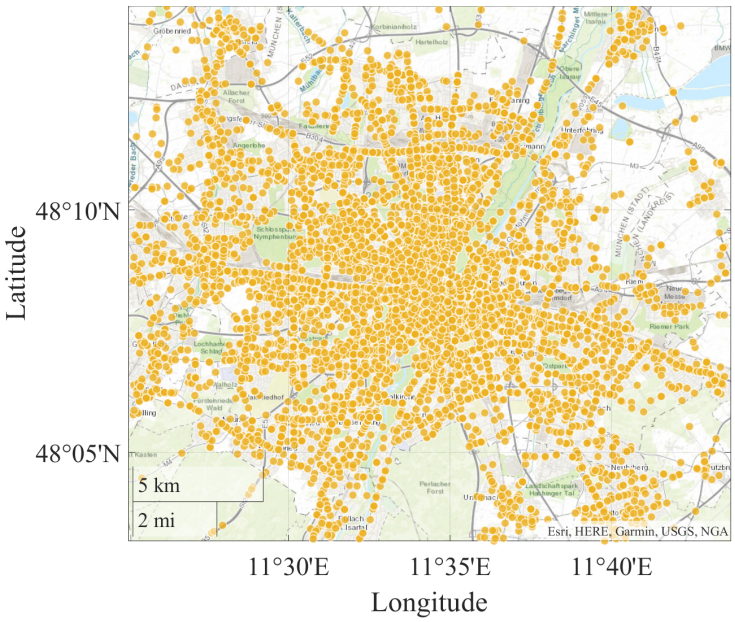


Figure 4.1: Locations all of bike accidents in the area of Munich, from 2016 to 2020.

Figure 4.2 shows all locations of serious and fatal accidents. The total number of accidents presented in Figure 4.2 is 1345, accumulated from 2016 to 2020.

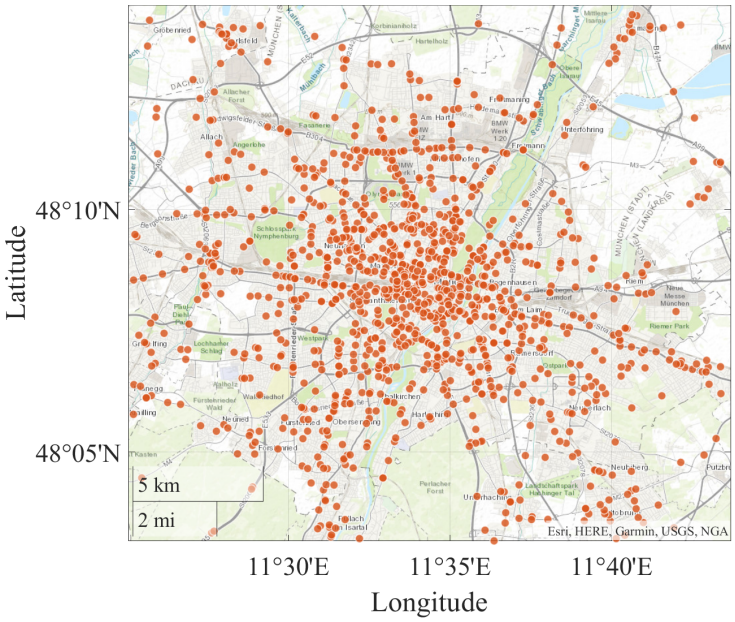


Figure 4.2: Locations of serious and fatal bike accidents in the area of Munich, from 2016 to 2020.

In Figure 4.3, fatal accidents are shown. From the visualizations of accident locations in five

following years, given in Figure 4.1, 4.2 and 4.3, no clear area or spot for incidents is obvious as "hotspot". There were 28 fatal accidents concerning cyclists, resulting in a yearly death of 5.6 cyclists in Munich. Considering Munichs' overall 11600 bike accidents, 0.24% (28) were fatal.

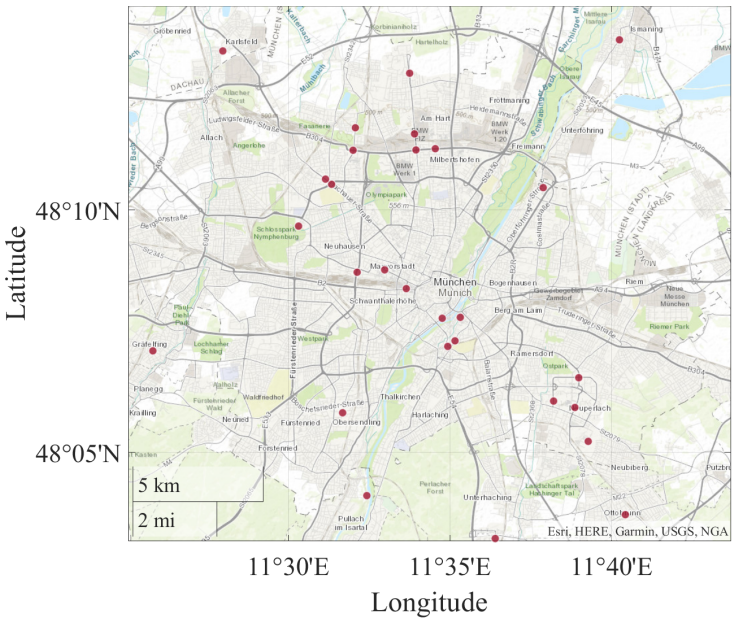


Figure 4.3: Locations of fatal bike accidents in the area of Munich, from 2016 to 2020.

For these accidents, it is interesting to see with whom cyclists have crashed in Munich. The accident parties are shown in Figure 4.4. For this analysis, data from 2016 and 2017 from the "Unfallatlas" must be ignored because, within this data, trucks are counted in the subgroup other. Therefore, a detailed analysis with a clear separation between trucks and the category "others" is impossible for 2016 and 2017. The result is given in Figure 4.4, showing that most bike accidents are linked to cars (82%).

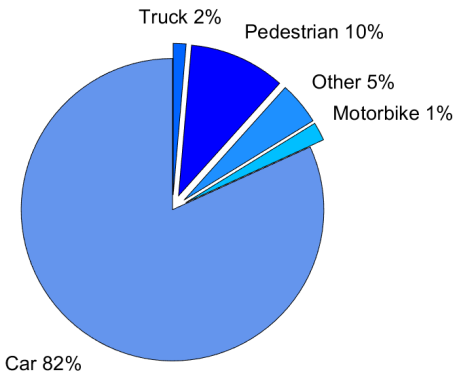


Figure 4.4: Statistical data from the "Unfallatlas" on bicycle accident participants in Munich.

4.1.2 Accident Clustering

From Figure 4.1, it is hard to derive an area where accidents are prevalent. Therefore, a clustering algorithm by Yann Macron [2022] was used to find accident clusters based on all bike accidents in Munich. The algorithm takes a set of XY coordinates and clusters these based on the distance between the points. The clustering parameters are reviewed in Table 4.1. The "max distance" is the maximum distance the points of a cluster can be separated until they are not considered one cluster anymore. "Min cluster size" describes the minimum size of a cluster. The method, in this case, is "geometric median". If a point is within the maximum allowed distance to the geometric median of a cluster, it will be added to the cluster. This analysis considers bike accidents with a maximum distance to the geometric median of the current cluster of 60 meters. Additionally, a cluster must contain at least 21 bike accidents.

The clustering result is presented in Figure 4.5. Blue points indicate the locations of the clusters, and letters next to the point are used to identify each cluster for a detailed analysis that follows this paragraph. The median cluster size was 22, while the smallest cluster had 21 accidents and the largest had 28 accidents. These incidents happened over five years, from 2015 to 2020. Therefore, each identified intersection had 4.4 accidents per year on average.

Table 4.1: Cluster algorithm parameters

Parameter	Value	Unit
Max distance	60	[meter]
Min cluster size	21	[points]
Method	"geometric median"	n.a.

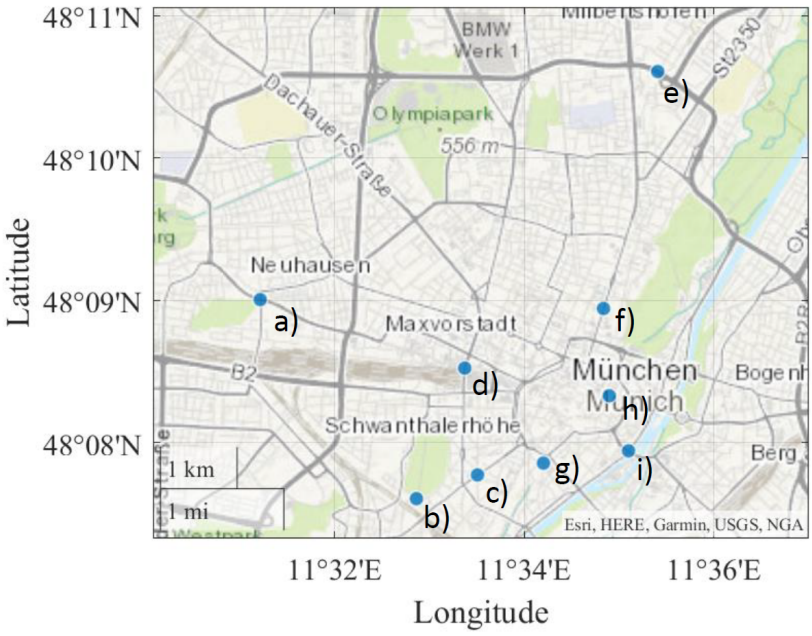


Figure 4.5: Locations of bike accident clusters in the area of Munich

The "Unfallatlas" data set does not contain subjectively perceived information about why these accidents occurred. Therefore the "Gefahrenatlas" created by Sueddeutsche Zeitung (SZ) [Linus and Dominik, 2018] is used. In this dataset, local people were asked to tell them where they see hazards in the road network. The intersections found using the clustering algorithms are used to filter "Gefahrenatlas". Filtering is done utilizing the center point of each cluster. Then a search in the "Gefahrenatlas" based on this datasets' locations of hazards is performed. As a result, it is possible to see what people think about the intersections found by the clustering algorithm.

Table 4.2 and 4.3 describe accidents types and kinds given in "Unfallatlas" and are used in the detailed analysis of the clusters in this Section.

Table 4.2: Description of accidents kinds used in the open data set "Unfallatlas"

	Kind of accident
1	Collision with approaching/stopping/resting vehicle
2	Collision with vehicle ahead / waiting vehicle
3	Collision with vehicle driving sideways in the same direction
4	Collision with oncoming vehicle
5	Collision with turning / crossing vehicle
6	Collision between vehicle and pedestrian
7	Collision with road obstacle
8	Leaving the lane to the right
9	Leaving the lane to the left
0	Other

Table 4.3: Description of accidents types used in the open data set "Unfallatlas"

	Type of accident
1	Driving accident
2	Turning accident
3	Turning/Crossing accident
4	Pedestrian crossing accident
5	Parked/Resting vehicle traffic accident
6	Longitudinal traffic accident
7	Other

Cluster a):

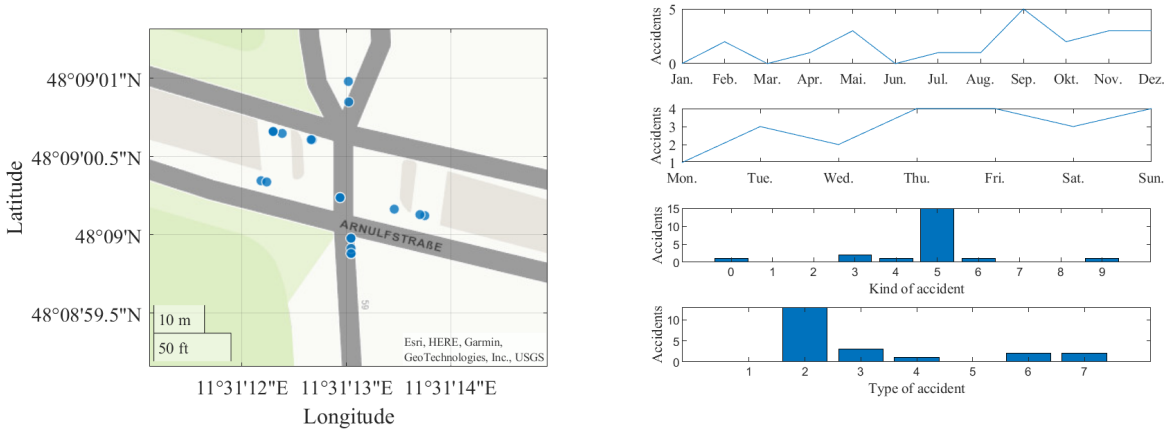


Figure 4.6: Specific cluster of accidents a)

Figure 4.6 shows that most the accidents are rather evenly spaced during the weekdays. Most of the accidents happen in September and are a result of turning incidents. In Figure 4.7 we can see that there is no heavy occlusion for cars or other road users. This means road users should be able to observe cyclists properly.

From SZ road users have reported that this intersection contains a bikeway which can be used in both directions. Two major issues are reported here. Firstly, cyclists and pedestrians complain about each other regarding space usage. Secondly, cars do not predict that cyclists are approaching the intersections in both directions on the same bikeway.

Figure 4.7: Goolge maps image of cluster a)

Cluster b):

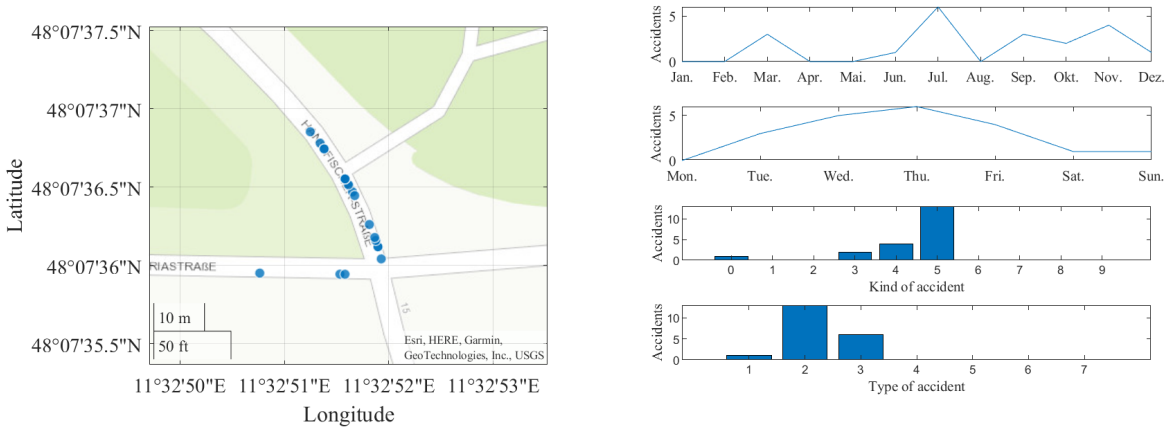


Figure 4.8: Specific cluster of accidents b)

Cluster b) is shown in Figure 4.8 and Figure 4.9. We can see that most accidents happen during weekdays and as a result of turning incidents. The urban layout does not show occlusion by, e.g., buildings.

In the "Gefahrenatlas" it is reported that cyclists which intend to go left from "Hans Fischer Straße" to the "Bavaria Ring" have no reasonable route to cross. This results in many conflicts with pedestrians and cars. It is likely that many cyclists use the main road to perform this left turn, which would explain the accidents shown in Figure 4.8.

Figure 4.9: Goolge maps image of cluster b)

Cluster c):

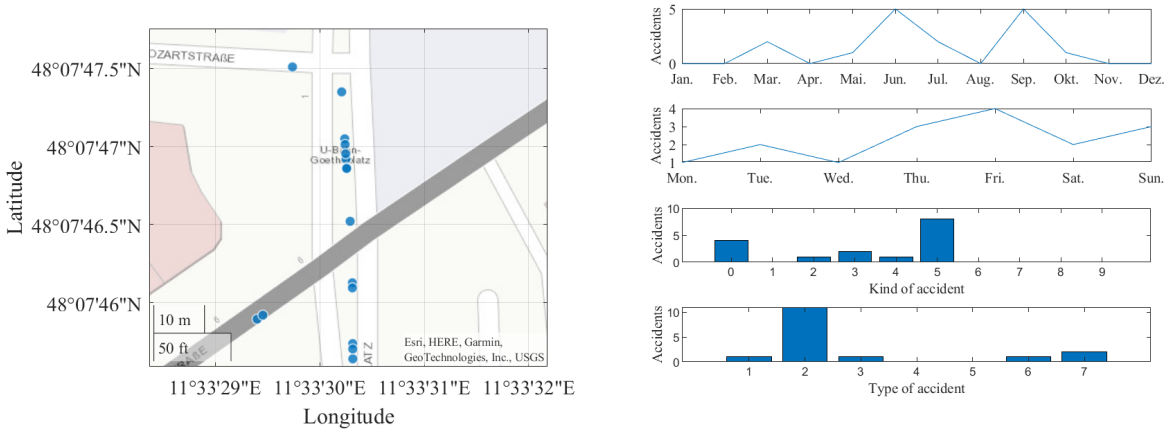


Figure 4.10: Specific cluster of accidents c)

Cluster c) points to an intersection with several accidents in an area with no separate bike lanes. Occlusion could be a problem resulting in accidents at the intersection. This could be related to dynamic occlusion, which is introduced in Section 4.2. The majority of accidents are turning accidents.

In the report from SZ we can see that cyclists complain about a notch in the road's surface on the main road.

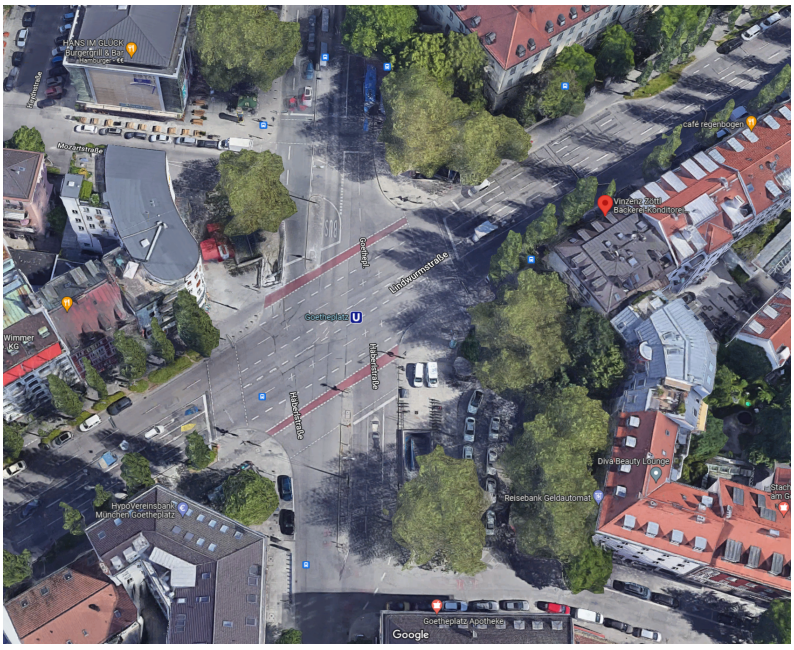


Figure 4.11: Goolge maps image of clusters c)

Cluster d):

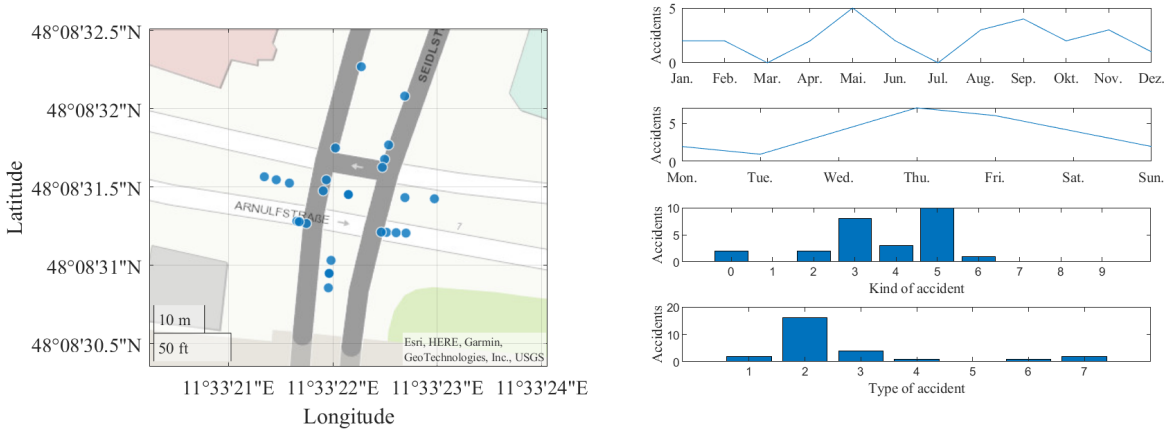


Figure 4.12: Specific cluster of accidents d)

Cluster d) points to a very complex road layout. Most recent images from google maps shown in Figure 4.13 indicate that this intersection is already being transformed and new bike lanes are added. This observation is based on the yellow markings on the intersection. Regarding occlusion, this intersectoin should not be a major issue for road users. VRUs should be observable inside the intersection.

In the report from SZ, cyclists complain about cars and buses frequently failing to yield right of way to them. One user reports that especially cars driving to the south create this issue. With respect to Figure 4.13 it can be observed that there is a new bike lane starting on the Seidlstraße, that separates right turning cars from cyclists that go straight. Overall many cyclists complain that cars or buses do not notice them in several occasions.

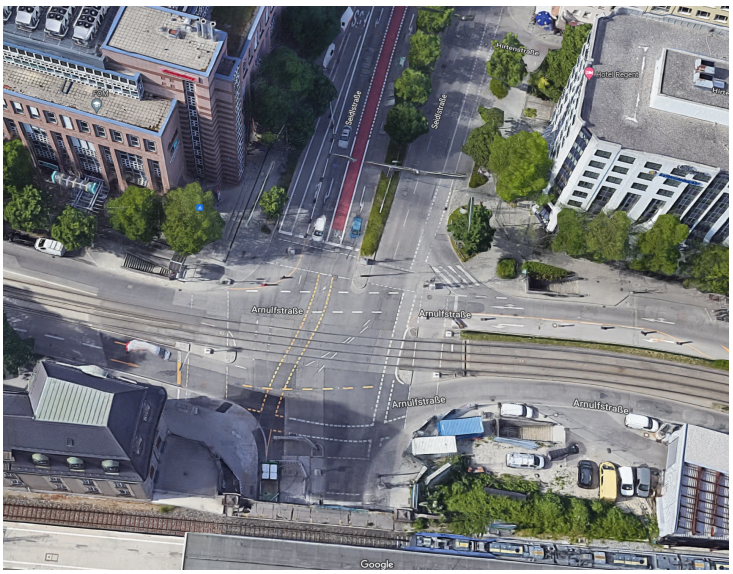


Figure 4.13: Goolge maps image of clusters d)

Cluster e):

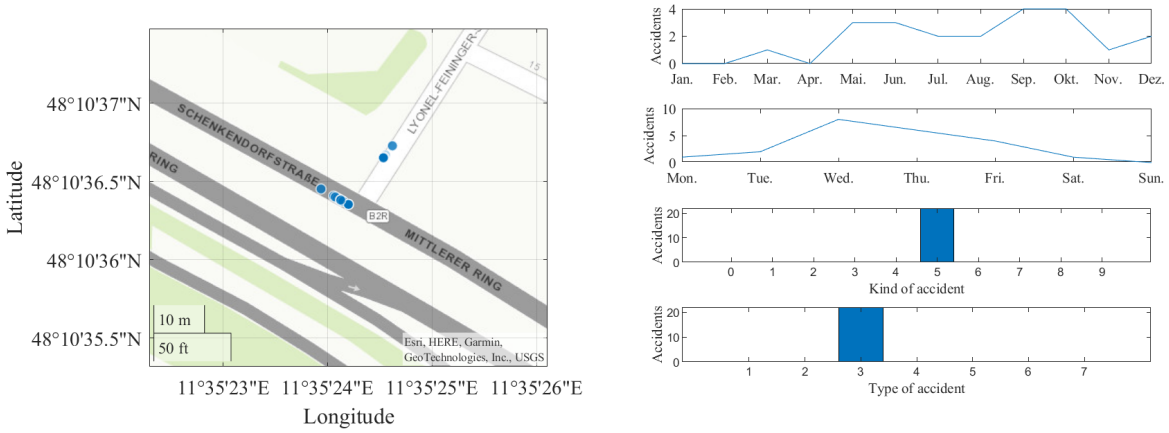


Figure 4.14: Specific cluster of accidents e)

Cluster e) clearly shows that even if there is no occlusion by buildings or parked vehicles many accidents occur. The kind and type of accidents refer to turning accidents.

From SZ cyclists report that cars try to merge in a rapid manner on to the main road. Additionally the bikeway is classified for two way usage, car drivers are reported to regularly fail to notice cyclists from both directions. The bold stop line in the middle of Figure 4.15 is consequently ignored by car drivers.

Figure 4.15: Goolge maps image of cluster e)

Cluster f):

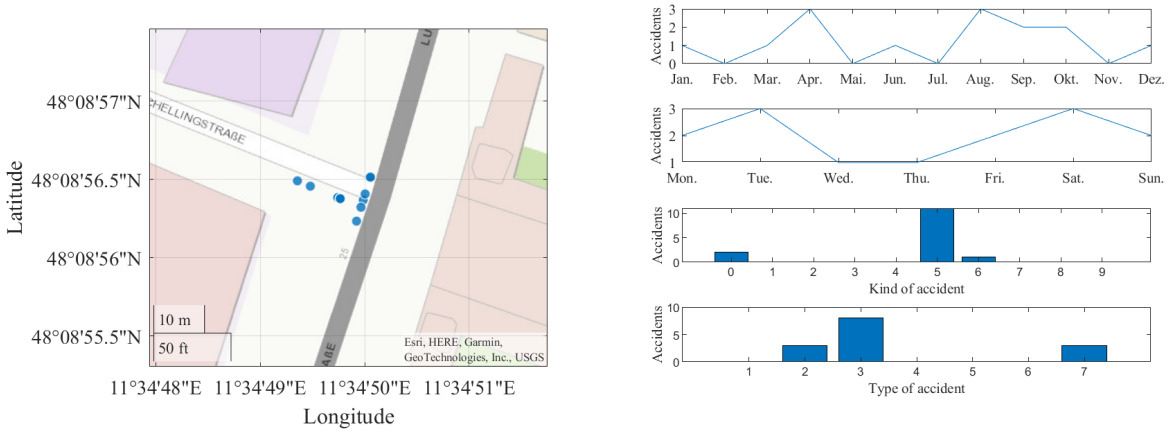


Figure 4.16: Specific cluster of accidents f)

The accidents of cluster f) do not show significant occlusion. Cyclists on bike lanes are easy to monitor by vehicles or their drivers.

The reports from SZ the cyclists complain about the necessity of this traffic light for them. They don't see a reason to stop if they want to drive from the bottom of Figure 4.17 to the top. This is reported by pedestrians, which argue that this behavior is dangerous for them. Additionally, they report that cars frequently fail to notice the red light at this intersection, which results in dangerous situations. This is very interesting: While cyclists ignore the traffic light, human drivers may not notice the red light due to how the brain interprets the whole scene in an urban area. Similar to the behavior of human drivers following the car in front instead of looking at road boundaries and traffic lights.

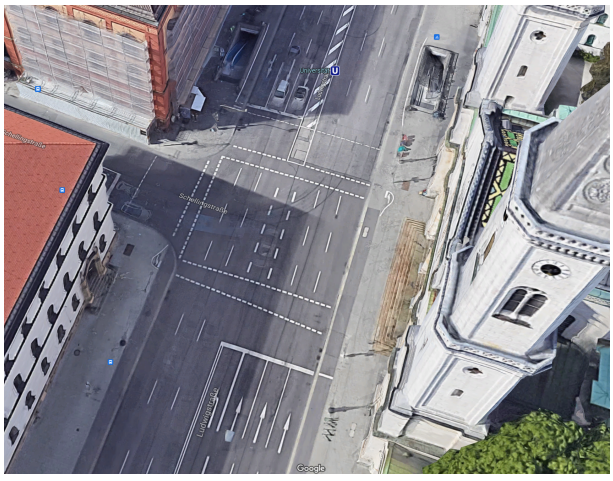


Figure 4.17: Goolge maps image of cluster f)

Cluster g):

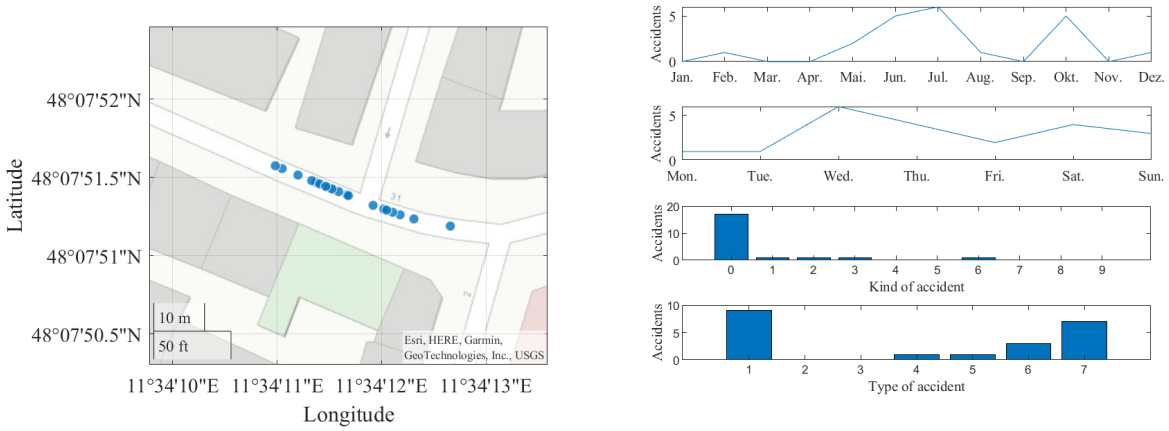


Figure 4.18: Specific cluster of accidents g)

This cluster of accidents shows a shared road for cyclists, trams, and cars. If we look at the type and kind of accident, an excessive number of accidents named other or normal driving are given. It is possible that cyclists are affected by the tram rails and do not behave as they would usually. Therefore, occlusion may not be a significant driver of cyclist accidents. There is no information about this cluster in the SZ report.

Figure 4.19: Goolge maps image of cluster g)

Cluster h):

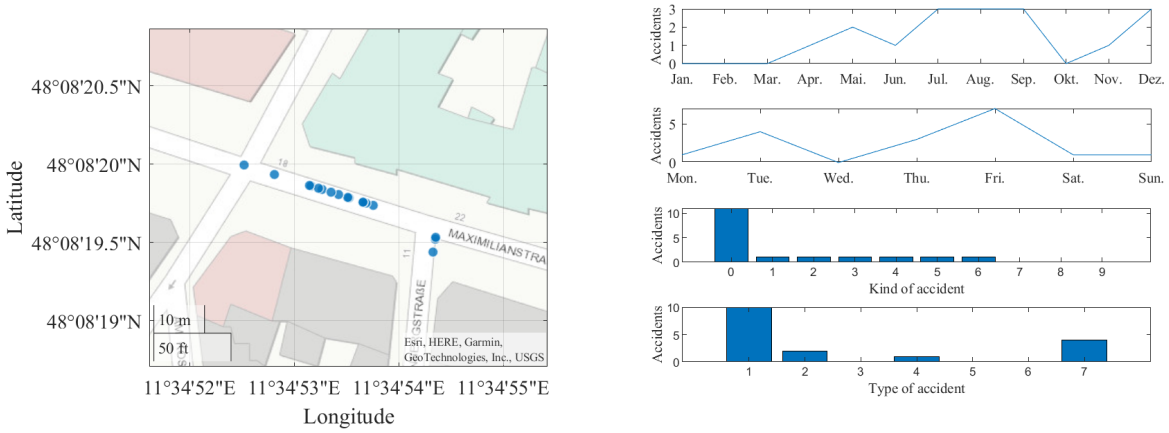


Figure 4.20: Specific cluster of accidents h)

In this cluster, a shared road for cyclists, trams, and cars is visible. No bike lane is available. A bike lane could likely reduce accidents rather than solving occlusion issues.

In the report from SZ a cyclist mentioned that there is no bikeway and urged to remove one car parking lane to create one. Many vehicles are reported to stop where they are not allowed to. Pedestrians suggest using the parking spaces or moving large plant pods because they block their sidewalks.

Figure 4.21: Goolge maps image of cluster h)

Cluster i):

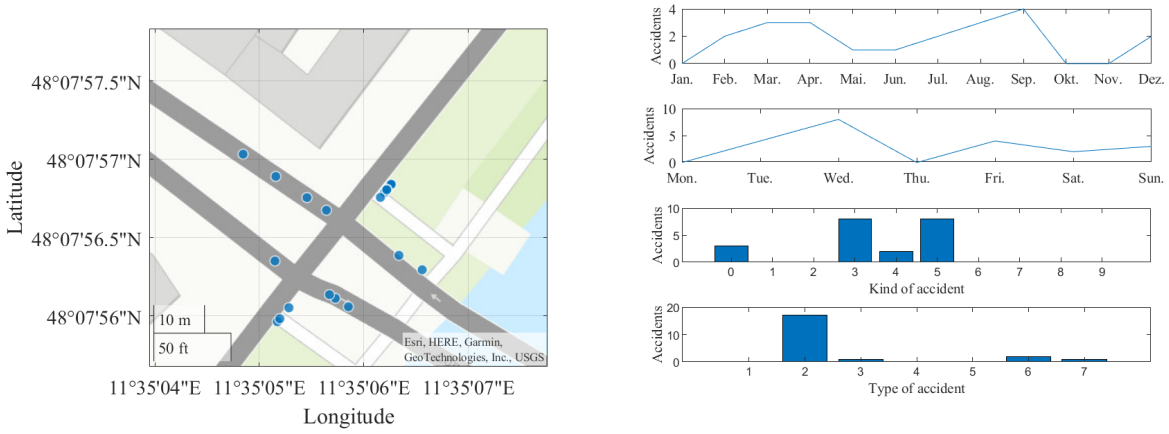


Figure 4.22: Specific cluster of accidents i)

Most recent aerial imagery from cluster i) shows that this intersection is being worked on. This cluster showed a high amount of turning accidents even if there was no occlusion that could harm the visibility of cyclists to common vehicles.

The SZ report shows statements of cyclists which say that the bikeways in this area are to small and hard to observe. Two people report that driving from Ludwigsbrücke to Erhardstraße (left turn maneuver) is straight forward "a threat to your live" and say that it should be forbidden to have green light for bikes and cars at the same time for this intersection. Several people complain about the signal timings at this intersection, concluding that it is "insane" to have the green light for both cars and bicycles at this intersection at the same time.

Figure 4.23: Goolge maps image of cluster i)

4.1.3 Summary

This section considered accident locations and potential reasons for bicyclists. Accidents in the area of Munich are clustered to find specific intersections that show the highest number of bike accidents. The clustering presented nine intersection and each of them had 4.4 accidents per year on average. Information about accidents in the clustered regions are considered and discussed. Additonally, data from the "Gefahrenatlas" is used to get further insights into given intersections.

From the given analysis the following conclusions can be drawn:

1. In several situations accidents may be related to issues concerning the bikeway. From the clustering in combination with the results from SZ it can be observed that a bikelane that is approved for bicyclists traveling in both directions is a cause for accidents. Furthermore in some cases there was not bikeway or route for bicyclists available.

2. The number accidents for accidents that are a result of turning incidents is prevalent in the data. For several situations the report from SZ suggests that this is a result of inattentive driving of cars.

3. A shared signal phase for bicyclists and vehicles is another issue that can cause accidents.

In the data analysis, it is observed that the city of Munich does already take action and is aware of some critical intersections. As a result, the road network is being changed in order to solve hazardous spots at intersections. Overall the accidents observed are mostly related to turning maneuvers. The given analysis does not give a clear answer on whether occlusion has a significant impact on given traffic incidents. Buch and Jensen [2017] argue that several questions about factors that lead to right-turn accidents, cannot be answered using accident data. Further analysis is needed to find reasons for turning accidents of bicyclists.

4.2 Static and Dynamic Occlusion

In Section 1.2, the influence of occlusion on accidents in urban areas is explained. Therefore, it is essential to point out that for this work, two different kinds of occlusion are considered:

- Dynamic Occlusion: Moving objects, e.g., cars/trucks driving on the road
- Static Occlusion: Urban environment, e.g., buildings, bushes, trees, parked cars

An occlusion evaluation software that uses simulation data from Aimsun Next as input was developed to evaluate possible dynamic occlusion. Figure 4.24 helps to understand how the occlusion evaluation approach works. A calibrated Aimsun simulation of an intersection is run, and the vehicle positions throughout the simulation are collected. In the next step, the collected positions of the vehicles in the road network are used to evaluate occlusion generated by moving objects.

An illustration of this occlusion evaluation is given in Figure 4.24. This Figure shows a top-down view of an intersection, including cars and buildings. The occlusion of one specific time step on the left side is shown. This occlusion is based on all cars and buildings in the road network at this time step. Small black boxes indicate cars driven manually and therefore do not have sensors that observe their environment. These road users generate dynamic occlusion for CAVs. Small red boxes on the left side of Figure 4.24 highlight CAVs that are currently in the road network. Around each CAV, a circle is drawn that highlights occlusion it is objected to by non CAVs. This is illustrated by red circles on the left side of Figure 4.24.

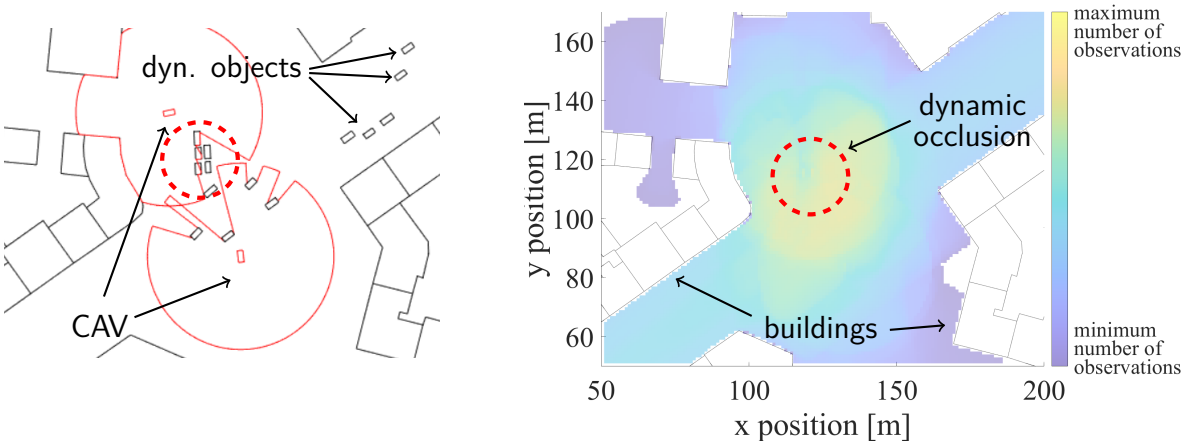


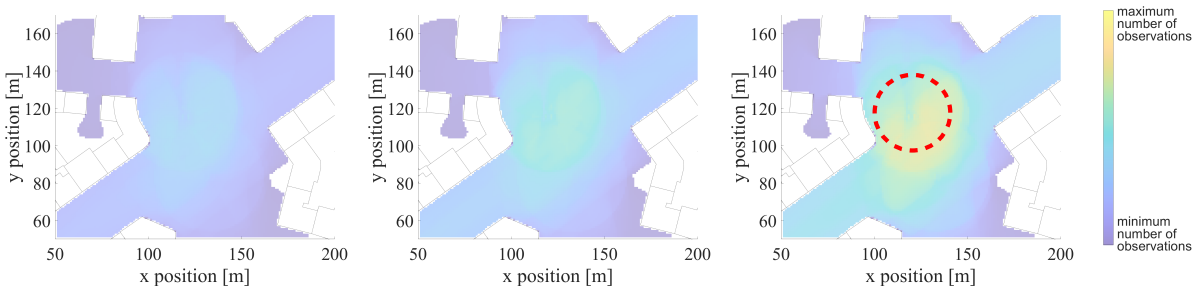
Figure 4.24: Image of the occlusion simulation evaluation based on vehicle pose data from an Aimsun simulation. The left side shows a single frame of the simulation, while the right side results from an accumulated bin-based heatmap.

One can see the resulting heat map on the right side of Figure 4.24. This map is generated based on an occupancy grid map with a one-meter resolution. Every time step of the simulation, a bin of this grid map is observed by one or more CAVs the bin counter is increased by one. After the whole simulation run, a heat map is generated that shows which locations of the road network are dynamically occluded.

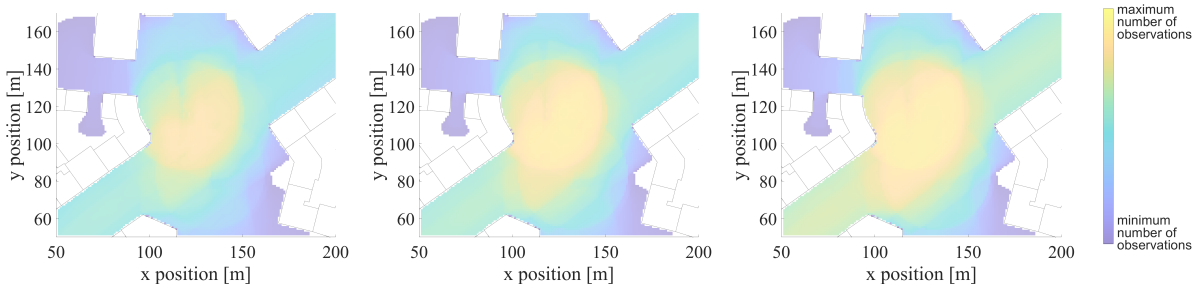
Moreover, this approach does not just highlight dynamic occlusions in a road network. It is also possible to change the penetration rate of CAVs in a mixed traffic scenario. The red arrow in Figure 4.24 shows where a dynamically occluded spot in the road network was observed. To sum up, this approach is based on three main steps: Run Aimsun simulation of a road network (1), evaluate occlusion based on generated data (2), and generate an occlusion heat map (3).

4.2.1 Dynamic Occlusion - Intersection from accident clustering

One intersection from the accident clustering was chosen for analysis on dynamic occlusion. This specific intersection has increased bike accidents and does not show significant occlusion based on visual observations of, e.g., buildings. For a closer look at this intersection, refer to intersection cluster c). The Aimsun Next road network was calibrated based on the work of Dandl et al. [2020]. The occlusion simulation (2) was evaluated with eight CAV penetration rates. The experiment assumes that dynamic occlusion will become visible at a specific penetration rate of CAVs in the road network. At an upper limit, the density of CAVs will be high enough that no dynamic occlusion is observed anymore because if all cars share their information ideally, no occlusion can occur. Concerning Figure 4.25, one can see a specific dynamically occluded area. A bluish spot in the middle of the box can be seen at an CAV penetration rate of 30% and 40%. Around this spot, a yellow area is displayed, showing the dynamic occlusion spot explained in Figure 4.24c. At an increasing penetration rate of CAVs the occlusion gets less visible in the heatmap. Above 30% penetration rate, the dynamic occlusion is almost invisible in the heat map, indicated by 4.24d and 4.24e.



(a) $p_{ref} = 10\%$ (b) $p_{ref} = 20\%$ (c) $p_{ref} = 30\%$

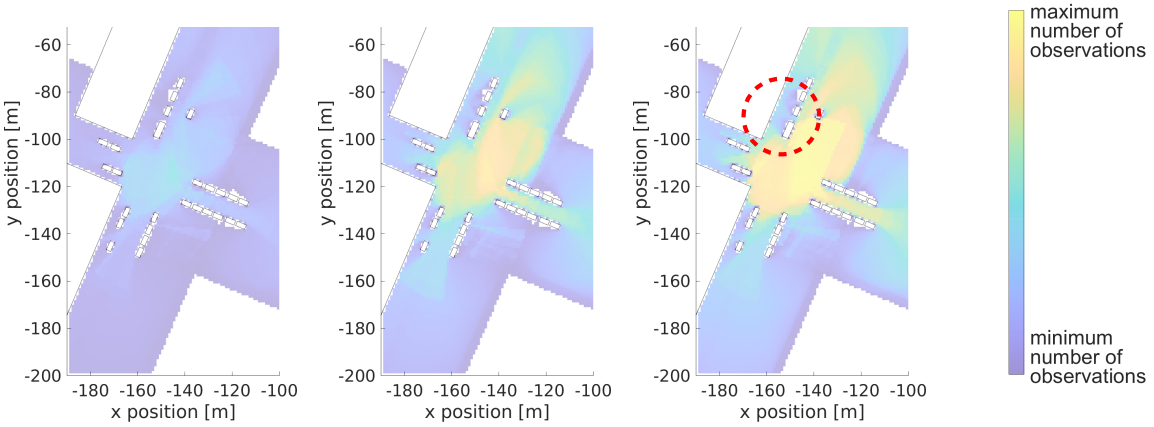


(d) $p_{ref} = 40\%$ (e) $p_{ref} = 50\%$ (f) $p_{ref} = 60\%$

Figure 4.25: Heat maps from the dynamic occlusion simulation for increasing CAV penetration rates (p_{ref}) from 10 to 60%.

4.2.2 Static Occlusion - Intersection at TU Munich

The literature, discussed in previous Chapters shows that static occlusion is a general issue in urban road networks. The framework explained in this section was used to evaluate static occlusion based on an intersection at the TU Munich. Therefore, parked vehicles were added to the road network, and occlusion was evaluated. From Figure 4.26, one can see that static occlusion will stay an issue even if 100% CAVs is achieved. If it was possible to have the vehicles at the parking spots powered on, these could share their information with moving cars. Occluded scenarios could be avoided. Nevertheless, this would require infrastructure such as a power supply at every parking spot, which may only be realized at some point in the far future. Additionally, it is essential to understand that the vehicle's perception systems use a significant amount of power to detect objects in their environment.



(a) $p_{ref} = 10\%$ (b) $p_{ref} = 50\%$ (c) $p_{ref} = 100\%$

Figure 4.26: Heat maps from the static occlusion simulation on the intersection of Arcisstraße and Theresienstraße for increasing CAV penetration rates (p_{ref}) from 10 to 50%

4.2.3 Static and Dynamic Occlusion Findings

One main finding of this analysis is that dynamic occlusion is a problem for CAVs until approximately more than 30% of vehicles in mixed traffic are CAVs. For dynamic occlusion, one can argue that roadside infrastructure is an interim solution until the threshold of CAVs in urban areas is achieved.

Considering static occlusion, externally powered parked vehicles or roadside ITS-S could be used to observe occluded areas. Even though roadside ITS-S perception could be replaced by parked vehicles utilizing their perception sensors, such sensing systems must be reliable and accurate. It is hard to grasp that a parked vehicle could be used for a safety-critical task such as detecting VRUs.

Dedicated roadside ITS-S may be a rather achievable system to help in occluded scenarios, at least as an interim solution for several years until a threshold in the number of CAVs is achieved.

4.3 Economic Standpoint

For this analysis, the accident count must be put into perspective. Considering a, purely economical analysis, the following question arises: is the number of 5.6 fatal cyclist accidents acceptable?

4.3.1 Accident Costs

As introduced in Section 1.2 there is an economic motivation to save human lives. This motivation can be based on the VOSL. In Europe, though, the one-million-euro-rule is applied, meaning a human life should be valued at about 1 million Euros, regarding the introduction of safety applications. If one would only calculate the fatal accidents, a budget for improvement may be estimated at around 5.6 million Euros for Munich, per year.

However, one could also use the "Handbook on the external costs of transport" issued by the European Union [VAN ESSEN et al., 2019]. They have evaluated the external costs of transport accidents for European and non-European countries. This contains more than fatal accidents, e.g., human costs of a serious and slight injury are included. Table 4.4 shows their result for the external costs of transport accidents in Germany. For more information about the meaning of, e.g., human costs, production loss, or medical costs, please refer to VAN ESSEN et al. [2019].

Table 4.4: External accident cost of transportation in Germany per casualty

Human costs

Fatality	Serious Injury	Slight injury
3.067.253,00 €	503.575,00 €	38.737,00 €

Production loss

Fatality	Serious Injury	Slight injury
383.018,00 €	25.497,00 €	1.560,00 €

Medical costs

Fatality	Serious Injury	Slight injury
2.885,00 €	8.883,00 €	765,00 €

Administrative costs

Fatality	Serious Injury	Slight injury
2.023,00 €	1.391,00 €	598,00 €

4.3.2 Bicycle Accident Costs for Munich

Based on Table 4.4, external bicycle transport accident costs for Munich are evaluated. This evaluation is given in Table 4.5, where the accident counts from 2016 to 2020, introduced before, are used to get the overall annual costs. It is remarkable to see the price of accidents,

including all external costs. Especially when the annual costs of serious and slight injuries are compared to fatal accident costs in Table 4.5, a significant difference is visible.

Table 4.5: External accident cost of bicycle accidents for Munich

	Fatality	Serious Injury	Slight injury
Amount (2016-2020)	28	1345	10227
Amount (annually)	5,6	269	2045,4
Costs (annually)	19.349.002,40 €	145.084.074,00 €	85.211.364,00 €
Overall (annually)		249.644.440,40 €	

4.3.3 Economically Reasonable Measures

In the European Union, the rule to decide whether to implement a safety feature is based on the one-million-euro-rule. What could the city do with those 5.6 million Euros solely based on fatalities? In "Cost analysis of V2I Deployment" by Ricardo Energy & Environment, the costs of deploying roadside infrastructure are analyzed [Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi, 2020]. They have estimated a deployment cost for roadside infrastructure up to "full coverage". The most straightforward setup used a cellular network for complete coverage and estimated a cost per kilometer shown in Figure 4.27. According to the city of Munich, their road network has an overall length of 2382 km [2]. Using that road length and 1250€ per kilometer the result is 2,98 million Euros. Therefore, using the most sparse approach for V2I sounds reasonable considering the one-million-euro-rule.

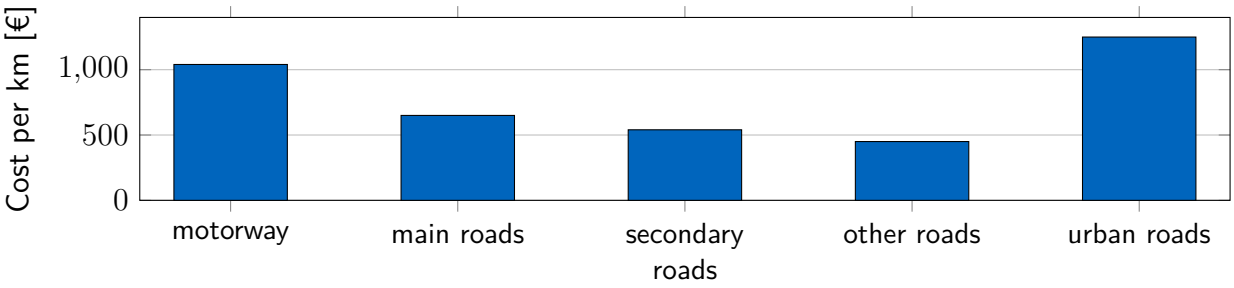


Figure 4.27: Costs for deployment of cellular V2I technology, based on [Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi, 2020]

The cost per RSU deployment is estimated at around 35.000,00€, depending on the variant. Three variations are given in Table 4.6 If one installs a fiber backhaul, the costs might be beyond 55.000,00€. These values are estimated for a lifetime of 10 years [Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi, 2020].

[2] https://stadt.muenchen.de/dam/jcr:459cce24-3894-4d4d-b144-4185b750a310/LHM.Stat_Faltkarte_-2019_englisch.pdf

Therefore, for a rough estimate using the one RSU per kilometer in Munich without fiber connection, one can estimate a total full coverage deployment cost of 83.37 million Euros for ten years.

Table 4.6: Estimates for deployment of RSU V2I technology for different expansion stages, based on [Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi, 2020]

System	Cost per Unit
RSU without backhaul	18.000,00€
RSU with wireless backhaul	35.000,00€
RSU with fiber backhaul	45.000,00€

As the lifetime of a RSU is set to ten years, the annual cost results in 8.3 million Euros, compared to a European cost-benefit evaluation of 5.6 million Euros. Considering that, and the likelihood of having no fatal bicycle accidents in traffic anymore, the cost-benefit advantage of this technology will become even less. Additionally, Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi [2020] have evaluated in their work that full coverage with RSU infrastructure for Europe is not feasible due to a total cumulative cost of 70 billion Euros.

Only considering the total annual external costs (249.6 million Euros) compared to deployment costs utilizing full coverage (8.3 million Euros annually), deployment of roadside ITS-S sounds reasonable. Figure 4.28 highlights the RSU costs as a small slice of the total external cost.

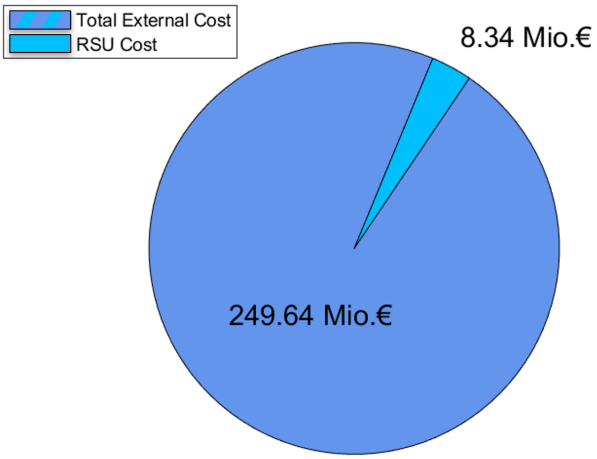


Figure 4.28: The total external cost of bicycle accidents, compared to the RSU cost in the area of Munich.

4.4 Summary

It is essential to mention that fatal accidents cannot be clustered in a specific geologically defined spot. Therefore, it is not possible to find one or more countable places where one would deploy infrastructure to prevent fatal traffic incidents. It is possible to find hotspots when all types of accidents are clustered. Concerning all external costs of bicycle accidents, including serious and slightly injured bicyclists, deployment of roadside ITS-S should be considered.

The benefit of comprehensive deployment of roadside infrastructure for cyclist safety may not be the primary driver to pursue this technology in Munich. Using specific locations, e.g. location of accident clusters, to deploy roadside ITS-S should be considered.

From literature, one can derive that road users can benefit from infrastructure if occluded spots are observed. The counterpart of safety is the reliability of the urban road network or, in other words, traffic efficiency. From historical data, we cannot derive the impact of roadside infrastructure on traffic efficiency, especially considering an increasing number of CAVs in the future. Therefore, these topics are covered by simulations in Chapter 5 and 6.

Chapter 5

System Simulation

This chapter describes all simulations performed for this thesis, whereas the results are evaluated in Chapter 6. For more details on the structure of this thesis, refer to Figure 1.4 given in the introduction. This chapter starts with findings of a HIL test with an actual vehicle are given in Section 5.1. Then more complex HIL simulator studies incorporating several road users such as cyclists and cars are introduced in Section 5.2, 5.3 and 5.4.

5.1 Emergency Trajectories for Connected Automated Driving

The study "Hardware in the Loop Test Using Infrastructure based Emergency Trajectories for Connected Automated Driving" [Pechinger et al., 2020] was presented at the Intelligent Vehicles Conference 2020. The system setup shown in Figure 5.1 consists of an actual **Research Vehicle** in combination with a virtual **Sub-Microscopic Simulation** environment, forming a VeHIL, as explained in Section 3.1.2. The VUT was driving in a parking lot while the complete environment was augmented. Individual advisory messages for automated vehicles are proposed to cope with critical situations where an AV cannot drive anymore. As a result, it is believed that a CAV may be able to resolve critical situations. The VUT used in this setup, utilized backup information from a roadside ITS-S to navigate safely to a stopping position.

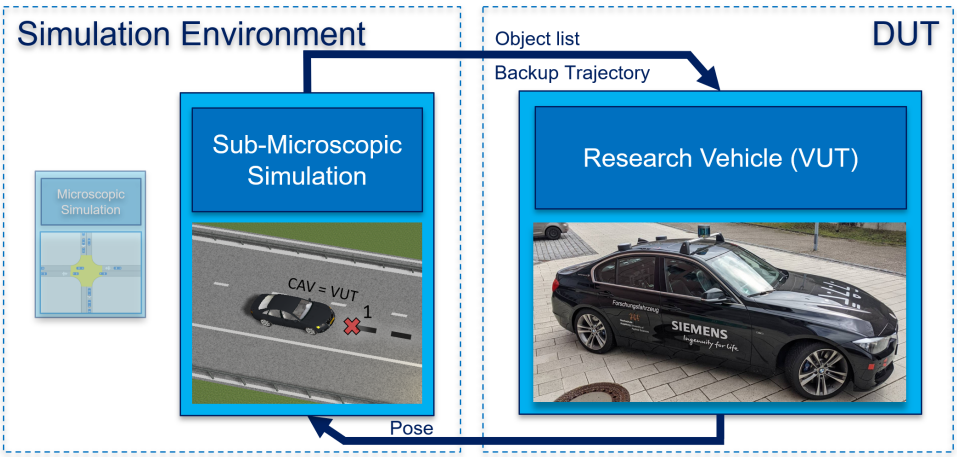


Figure 5.1: An overview showing the system setup used in this simulation.

In this work, an actual VUT also referred to as actual CAV that is moving in an augment environment. This system is the DUT. Additionally, a digital twin namely virtual VUT/CAV that moves inside the simulation environment is utilized.

Figure 5.2 shows the system architecture used for this HIL test. The **infrastructure** system with the digital twin is given on the left side of Figure 5.2, and the actual **CAV** on the right side. Both systems are connected using an 802.11p wireless bidirectional communication, utilizing a RSU and OBU, shown in the middle of Figure 5.2. The physical CAV can drive independently using its sensing system, indicated by the box "Lane Detection Sensor" on the right side of Figure 5.2. From here on, the "Motion Planner" generates a trajectory and sends it to the "motion control", which uses this information to maneuver the CAV. If the CAVs' lane detection system or the vehicle computer and motion planner fails, the CAV will fail.

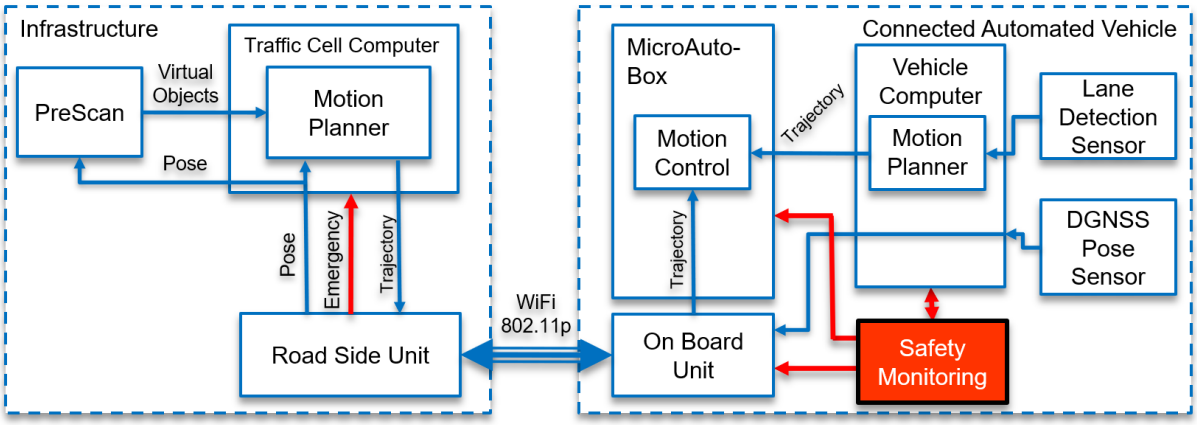


Figure 5.2: System architecture and qualitative data flow of the HIL system used for this analysis

A safety monitoring system is introduced that monitors critical components, and as such, it can detect issues within the actual CAVs' system. The red box in Figure 5.2 represents this system. At this point, the infrastructure is utilized to mitigate the problem of the CAV driving in the real world. The box named PreScan in Figure 5.2 indicates the simulation environment that generate the augmented reality for the actual VUT. The pose of the actual vehicle, driving in an real parking lot, is sent through the 802.11p wireless connection to the computer running the virtual PreScan simulation environment. As a result, the digital twin is moving in the virtual environment and the actual CAV in the real world.

Figure 5.3 shows the visualization of the simulation framework. The black car denoted with CAV moves inside the virtual environment, using the pose information from the actual VUT. To make more use of the virtual environment, a "Virtual Vehicle" is placed on the hard shoulder. This can be seen in Figure 5.3. Additionally, the dashed black line shows the qualitative trajectory, sent from roadside ITS-S, that the actual VUT should drive to safely maneuver in the augmented reality.

The red cross (1) shown in Figure 5.3 indicates the position of the ego vehicle in the simulation where an intentional failure of the vehicle computer is introduced. The RSI system is informed and takes over the trajectory generation for the motion control system of the AV. As the RSI detects an obstacle on the hard shoulder, the actual VUT is maneuvered until it

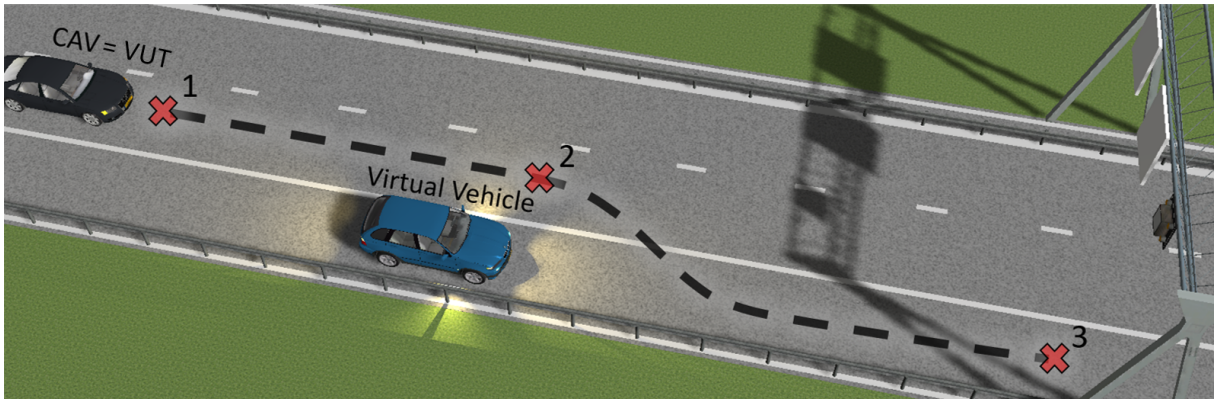


Figure 5.3: Visualization of the simulation framework's output with additional markers to show at which point the vehicle performed which specific maneuvering task [Pechinger et al., 2020]

reaches the next red cross (2) in Figure 5.3. At this point, the hard shoulder is detected as unoccupied, and the CAV is maneuvered to reach the last cross marker (3), where it should come to a safe stop without significant influence on the overall traffic flow.

In Figure 5.4 the CAV in the configuration used in the given article and the RSU can be seen.

Figure 5.4: Image of the CAV and the RSU from article [Pechinger et al., 2020]

5.2 Roadside ITS-S Perception for Urban Automated Driving

In this simulation, the actual vehicle was not used as part of the hardware in the simulation loop. Instead, the vehicle computer is removed from the vehicle and placed in a HIL simulation setup, forming a CoHIL as explained in Section 3.1.2. The work presented in this paragraph was published and presented at the Intelligent Vehicles Conference 2021, with the title "Benefit of Smart Infrastructure on Urban Automated Driving - Using an AV Testing Framework" [Pechinger et al., 2021a]. This simulation aims to provide insights into the impact of increasing the FoV of a CAV. In conclusion, statistical data shall be gathered to conclude on the impact of an extension of the VUTs' perceivable area in an unsignalized intersection.

The broad system setup is shown in Figure 5.5, presenting the main parts of the simulated system. Compared to the simulation in Section 5.1, a microscopic traffic simulation is added, and the DUT is replaced by the vehicle computer, shown inside the right box in Figure 5.5.

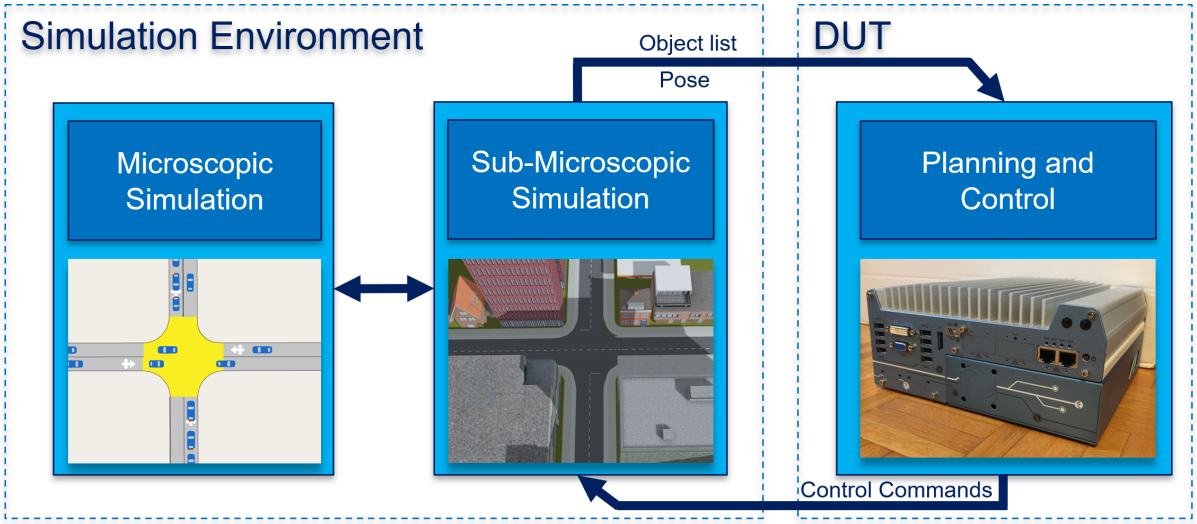


Figure 5.5: Illustration showing the broad system setup for the simulations performed in this simulation study [Pechinger et al., 2021a]

In this work, the FoV of a single CAV was extended by roadside ITS-S. Figure 5.6 shows the simulated environment where the CAV was supposed to navigate itself. The unobstructed intersection is given in the top image of 5.6. Additionally, the roadside ITS-S sensor poles and their respective FoV are highlighted by green cone-shaped areas indicated in Figure 5.6. The egos' FoV is shown by a red spherical-shaped transparent area in the middle. The road layout is indicated by green arrows highlighting minor and major roads.

Overall the CAV was set to perform a right turn maneuver from the minor road onto the major road. The simulation conducted 180 virtual HIL test drives of the CAV performing a right turn. The roadside ITS-S support was turned on and off throughout these tests. In addition, traffic demand in the road network varied from minimum to maximum, indicating no traffic up to heavy traffic jams. In Figure 5.6, the comparison between the top and bottom image shows the occlusion by urban building structures and the RSI observed areas.

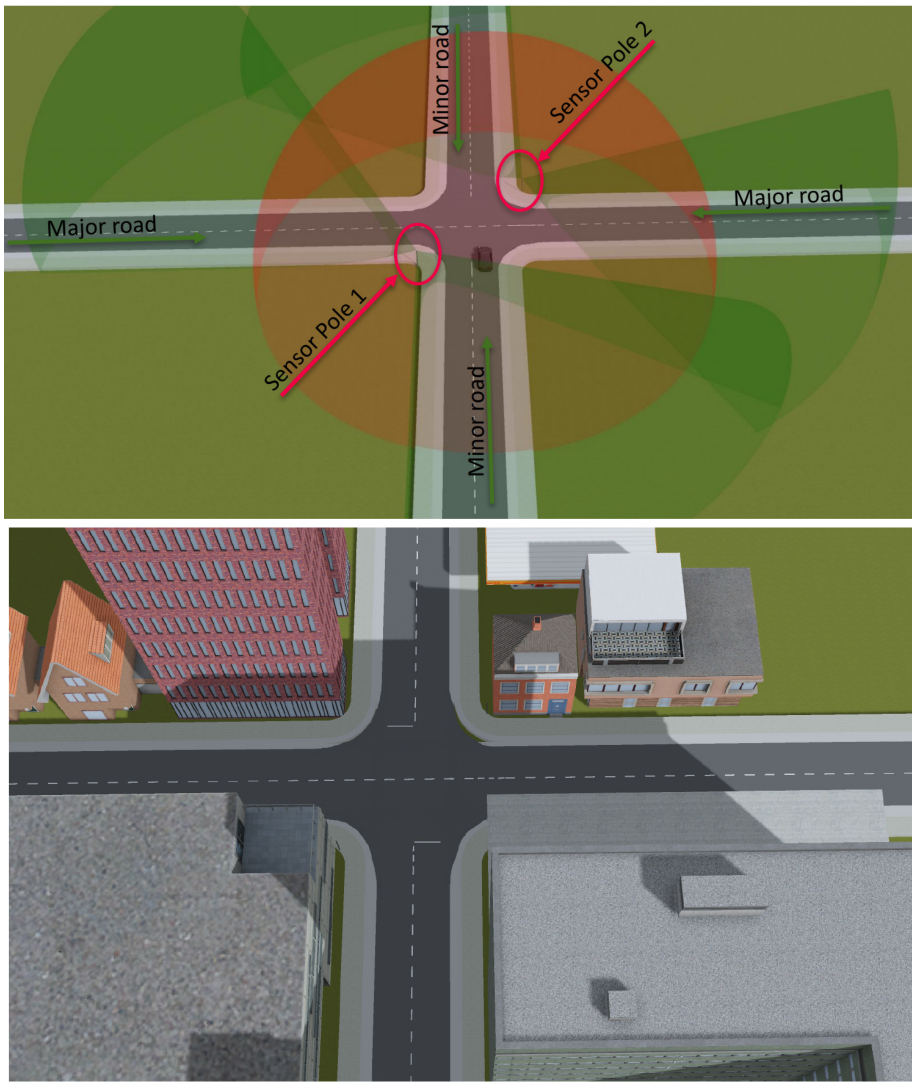


Figure 5.6: The top part shows the FoV of different perception systems and includes additional information about the system setup. The complete environment can be observed in the lower part, showing an urban road with buildings that add occlusion to the intersection.

Figure 5.7 shows the **Simulation Environment** on the left and the **DUT** on the right side. In the simulation environment, simulated cars driving in the road network are detected by **Infrastructure Sensors**, mounted on **Sensor Poles**. Furthermore, the ego vehicles' simulated **Vehicle Sensors** can also detect objects. All these are sent through an interface to the vehicle computer (DUT). Object lists are unpacked and fused to one object list on the DUT. After this step, the information is fed to the vehicles' **Driving Software**, utilizing Open Planner. Vehicle control commands are evaluated, returned to the simulation environment, and provided to the **Virtual Vehicle Model**. Additionally, the vehicle state is shared with the vehicle computer to be used by control and planning that is evaluated on the vehicle computer (DUT). The vehicle state includes information such as the ego vehicles' position, orientation, and velocity in the simulated world.

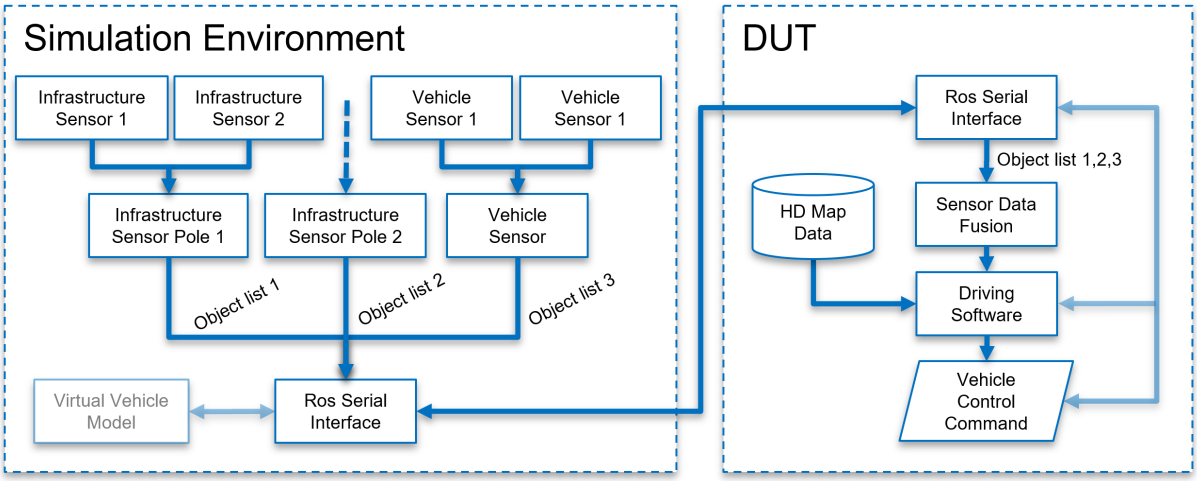


Figure 5.7: Illustration showing the data flow for simulations performed in [Pechinger et al., 2021a]

The simulation system is created to study the impact of roadside ITS-S perception. Therefore, an AV without RSI support and a CAV with RSI support are observed maneuvering a right turn onto the major road. The driving software of the AV and CAV was the same except for one change in the state machine. The CAV must only yield to traffic on the major road, while the AV must perform a brief stop to be able to observe obstacles on the major road. The CAV can see traffic on the major road utilizing roadside ITS perception. The stopping maneuver can be seen in the system evaluation.

5.3 Occlusion Evaluation regarding Bicyclists and Cars

This simulation setup analyzed the impact of static occlusion on AVs. The approach and the results were published at Intelligent Transport Systems Conference (ITS-C) 2021 with the title "Cyclist Safety in Urban Automated Driving - Sub-Microscopic HIL Simulation" [Pechinger et al., 2021b]. As explained in a previous setup, Aimsun is used for microscopic traffic, and Prescan for 3D environment simulation. This is a CoHIL setup consisting of a vehicle computer (DUT) running the driving algorithms, as explained in Section 3.1.2. The whole setup is given in Figure 5.8.

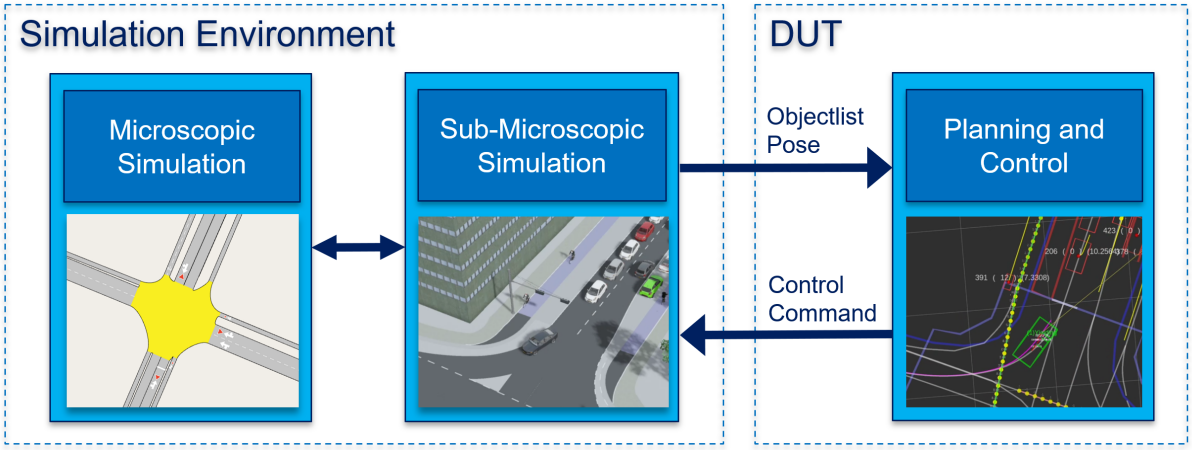


Figure 5.8: Illustration showing the broad system setup for the simulations performed in [Pechinger et al., 2021b]

The setup consists of a signalized four-way crossing. In this intersection, parked cars occlude the ego AV's FoV. As simulations are reproducible, a set of simulation iterations was performed, each varying with one prominent feature: a parked truck in the parking spot. Figure 5.9 shows two simulations that are the same except for the occlusion by the truck indicated by the red circle on the left side of Figure 5.9. On the right side of Figure 5.9, the cyclist can be seen, that is occluded by the truck on the left side, highlighted by the red circles.

Figure 5.9: A comparison between two simulation configurations, each showing the setup with additional occlusion and without [Pechinger et al., 2021b]

Figure 5.10 shows a side-by-side view of the microscopic simulation (left) and the visualization of the driving stack (right). The green rectangle corresponds to the black car in the

middle of the right part of Figure 5.10 - the ego AV. The combination of both sides highlights a right-turning maneuver. The pink line starting at the green box and leaving the image to the left indicates the planned path of the ego vehicle. The red boxes show detected objects, and the yellow dots on the green line show the predicted path by future positions of each corresponding object.

A detected bicycle object is given as a tiny red box above the ego AV on the left side of Figure 5.10. This bicyclist is predicted to pass through the egos' path. They have a conflicting area as both share a green signal phase and can enter the intersection. Therefore, the ego yields way of right to the cyclist.

In addition, the intersection boundary, parking areas, and all paths of the road network are shown on the left side of Figure 5.10

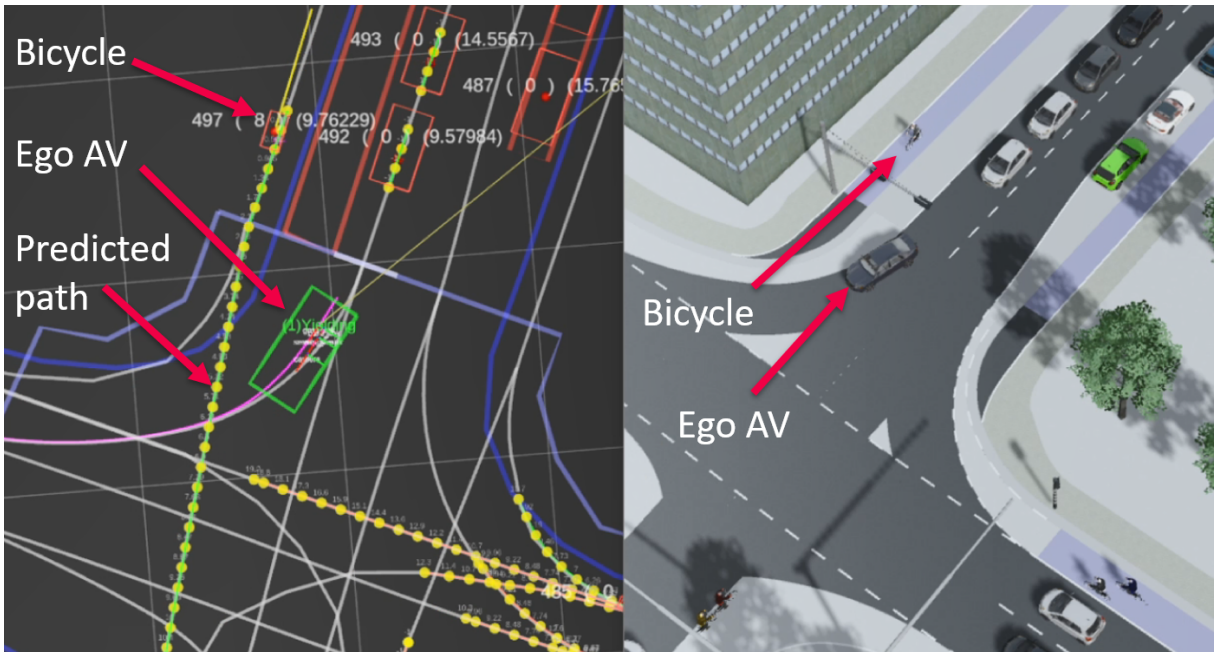


Figure 5.10: Visualization of the driving stack scene evaluation and the sub-microscopic simulation [Pechinger et al., 2021b]

The assembled intersection was aligned with the Arcis- and Theresienstraße as shown in Figure 5.11. The image shows conflicting bike lanes for right-turning vehicles and parked cars that occlude the FoV for on-road vehicles. The black car in the middle of this image is compared to the one given in Figure 5.10, highlighting the ego AV, that yields its way of right to pedestrians and a cyclist. In Figure 5.12 the location of this intersection in Munich, is shown by the blue rectangle.

Figure 5.11: Image from Google Streetview of the intersection that was used as a role model for the simulated intersection [Google, 2021]

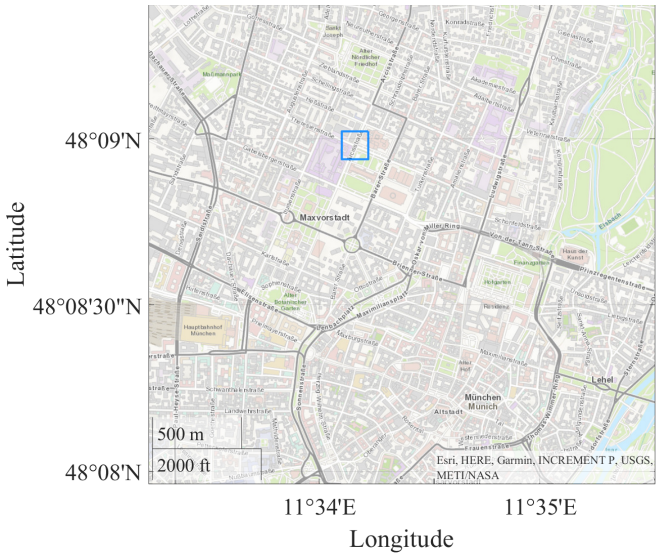


Figure 5.12: The location of the simulated intersection in Munich.

Figure 5.13 shows the simulation setups' data flow. A **Virtual Vehicle Model** inside the **Simulation Environment** (left dashed box) is evaluated. Objects from the simulation are detected by the **Vehicle Sensor** and sent to the **DUT** through the **ROS Serial Interface**, indicated by the right dashed blue box. The vehicle computer (DUT) uses the **Object list**, tracks and fuses it, and generates control commands using its **Driving Software**.

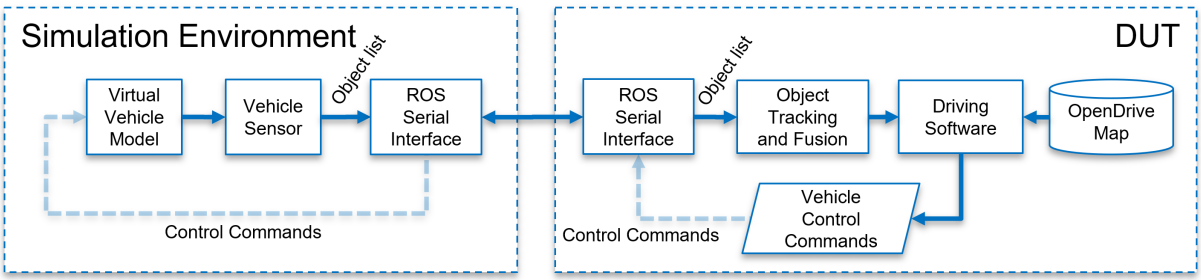


Figure 5.13: Detailed data flow of this simulation setup

Traffic demand is calibrated using actual traffic counts of this specific intersection through morning peak hours from 8:30 to 9:30 AM. V_1, V_2 and V_3 are the traffic demands in vehicles per hour (V/h), shown in Table 5.1. Combining Table 5.1 and Figure 5.14 shows corresponding traffic flows in the road network. C_1, C_2, and C_3 show the driving directions of cyclists. The cyclist traffic demand was set for C_1, C_2 and C_3 to 300 cyclists per hour going straight and 100 per hour left or right if applicable. Traffic lights are switched to green for cyclists two seconds before cars.

Table 5.1: Traffic demand of cars in the simulation in V/h [Pechinger et al., 2021b].

V1		V2			V3	
left	straight	left	straight	right	straight	right
32	95	46	784	54	537	69

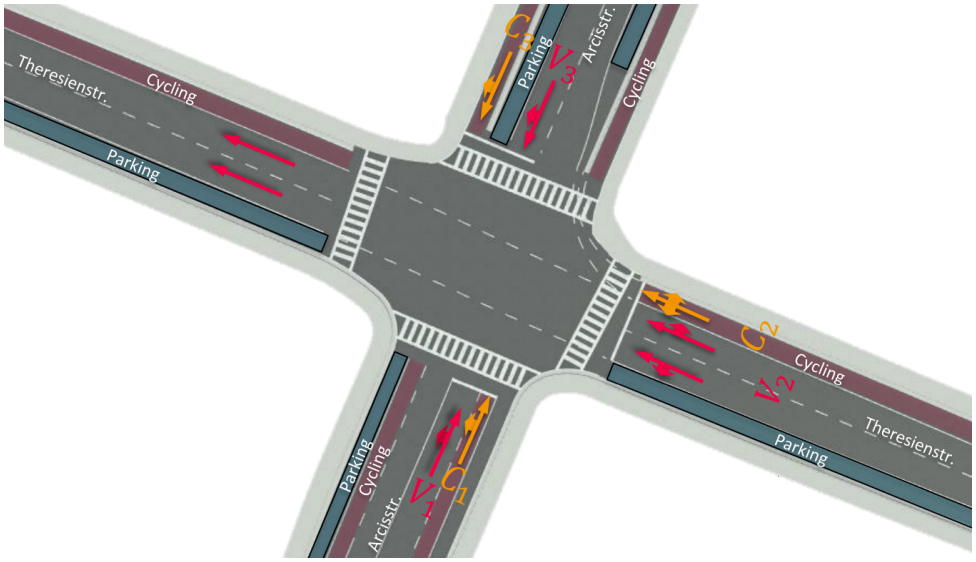


Figure 5.14: Illustration of the road network with an indication for traffic demand and urban road structure/layout [Pechinger et al., 2021b]

5.4 Collective Perception for CAVs utilizing roadside ITS-S

This analysis is built upon and extends Section 5.3. Figure 5.15 shows the simulation setup. Similar to previous simulations, a co-simulation of Aimsun is used in the microscopic simulation section and Prescan for sub-microscopic simulation. Essentially the setup describes a CoHIL explained in Section 3.1.2. Compared to the previous study in Section 5.3, the sub-microscopic simulation was improved in several aspects, such as CAV performance and adding roadside ITS-S. Additionally, the open-source driving stack has received significant updates, as CP was introduced.

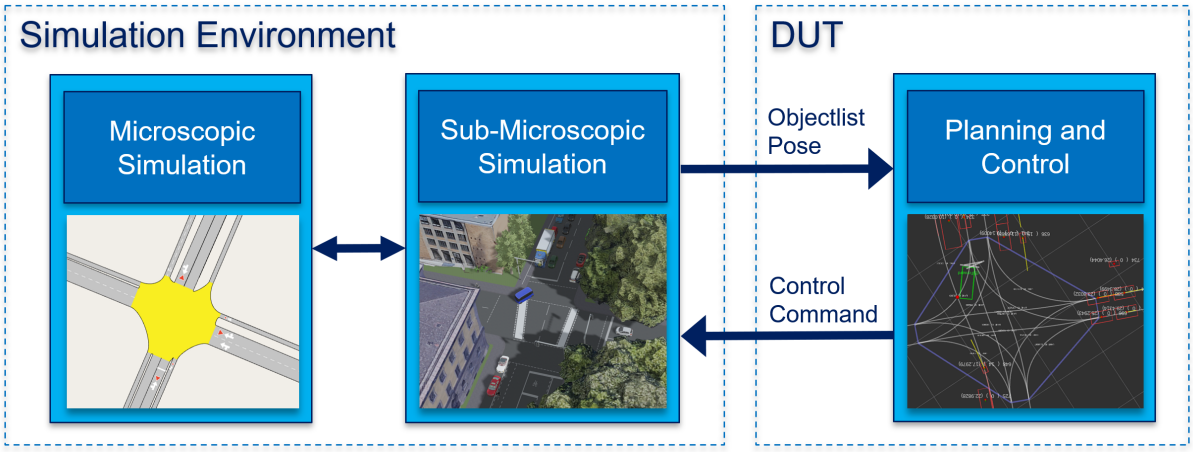


Figure 5.15: Illustration showing the basic system architecture connections between major parts of the simulation system.

The driving performance in urban environments was enhanced in cooperation with Autoware Foundation and its open planner working group. The driving software used for this simulation is available on GitHub[1]. Additionally, the driving software was evaluated using an automated testing framework which can be found on GitHub[2] as well.

Figure 5.16 highlights the data flow for the simulation setup. Compared to simulation study 5.3, a **Roadside ITS-S sensor** is added to enable CP. **Object list 1** and **2** are sent to the vehicle computer and fused to be evaluated by the **Driving Software**. The rest of the dataflow stays as explained in 5.3.

[1]https://github.com/MathiasPechinger/planner_ws/releases/tag/HILSimulator-Software-v1.0
[2]https://github.com/arminstr/ros_scenario_simulation

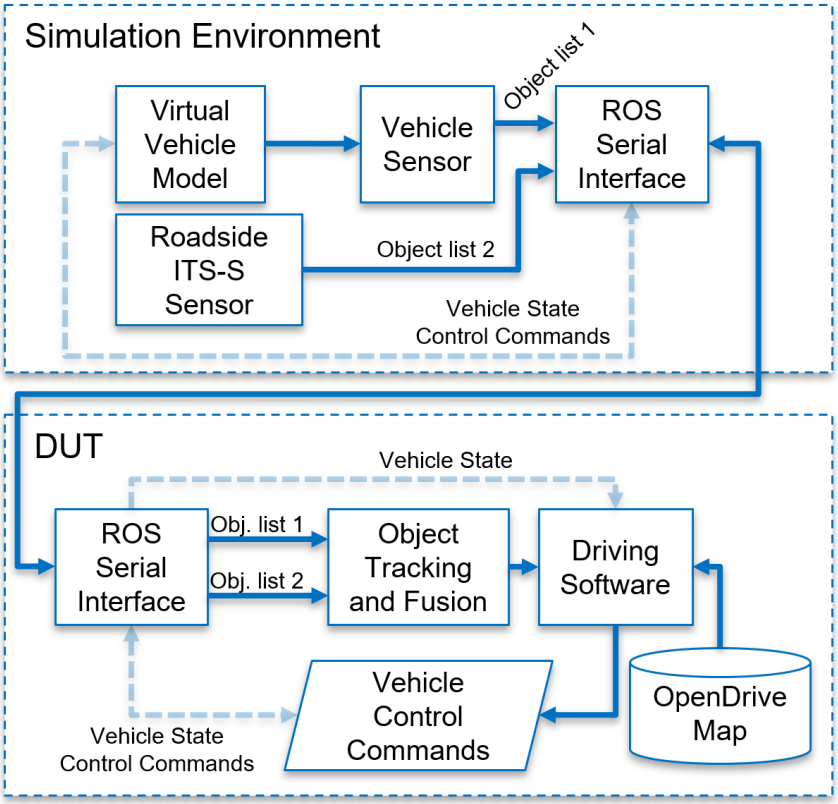


Figure 5.16: Data flow of simulations done for this study.

As mentioned at the beginning of this section, the sub-microscopic simulation was enhanced. In Figure 5.17, a side-by-side comparison of a google earth image (left side) and the simulation framework (right side) is given. The blue VUT is observed while taking a right turn in the middle of the simulation image. In addition, other driving and parked vehicles can be seen inside the simulated scene.

Figure 5.17: A side-by-side comparison showing Google's illustration of the intersection on the left and the simulation systems' environment on the right, comparing the level of detail of the digital twin.

Figure 5.18 shows a top-down view of the intersection modeled for this study. On the left side, buildings are visible, while buildings and trees were removed on the right to show the intersection without occlusion. Extra attention should be paid to the bike lane starting at the top of Figure 5.18 on the left side of the road. This bike lane is directly occluded by parking cars and a truck.

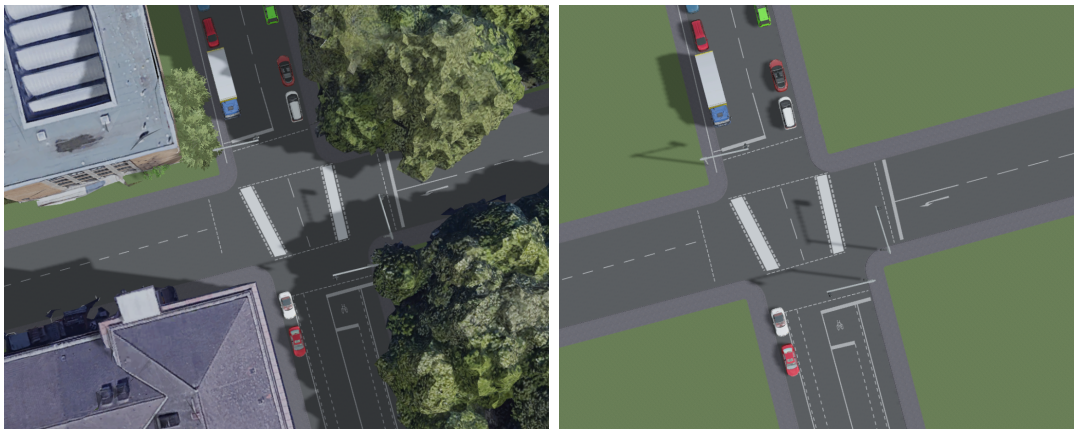


Figure 5.18: The road network with and without occluding objects, e.g., buildings or trees.

In the previous illustration, a critical bike lane was introduced. Figure 5.19 gives a closeup of the critical spot analyzed in this work. The truck is displayed in the middle of Figure 5.19, the VUT on the right side. On the left side of the truck, a cyclist is shown that cannot be observed by the VUT itself. Starting from the traffic light mount, a blue cone-shaped transparent area is shown in Figure 5.19. This corresponds to the FoV of a sensor mounted inside a roadside ITS-S. In contrast to the VUT's, this sensor can see the VRU. The flow chart (Figure 5.16) shows that this information is sent to the CAV to enhance its perception.

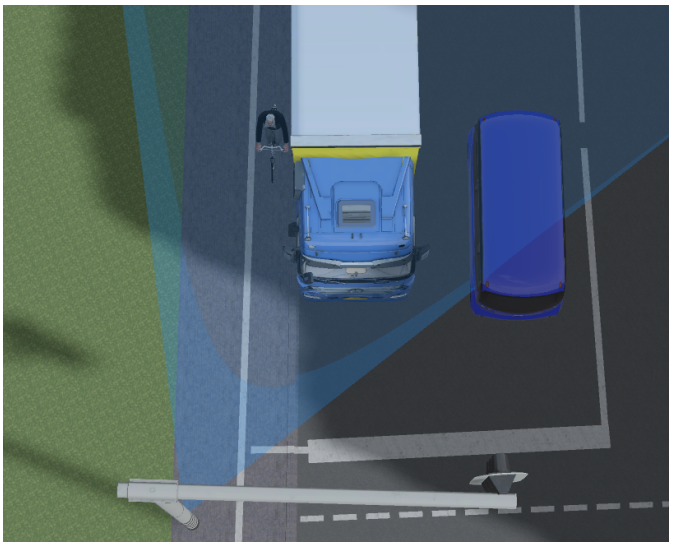


Figure 5.19: Illustration of the road network with indication for traffic demand and urban road structure/layout

Figure 5.20 shows another top-down view of the intersection. This representation is rotated 90 degrees clockwise, compared to Figure 5.17 and 5.18. A detailed qualitative description of the scenarios is given. Arrows notated $V_{1,2,3}$ correspond to Table 5.1 from study 5.3. There are two different routes defined for the VUT. The ego vehicle can perform a right turn if it starts on the left side on Figure 5.20. It approaches the intersection from P_1 and drives along $R_{\text{Car 1}}$. If the ego vehicle performs a left turn, it approaches from P_2 and drives along $R_{\text{Car 2}}$ towards Theresienstraße. Transparent boxes notate parking and cycling areas. Cyclists can travel along several routes. In Figure 5.20, two important routes $R_{\text{Bike 1}}$ and $R_{\text{Bike 2}}$, are shown by the dotted line. These conflict with the egos routes that are indicated by dash-dotted lines. As a result, the intersection includes an area of conflict A_{Conflict} shown in the middle of Figure 5.20 with a red oval area.

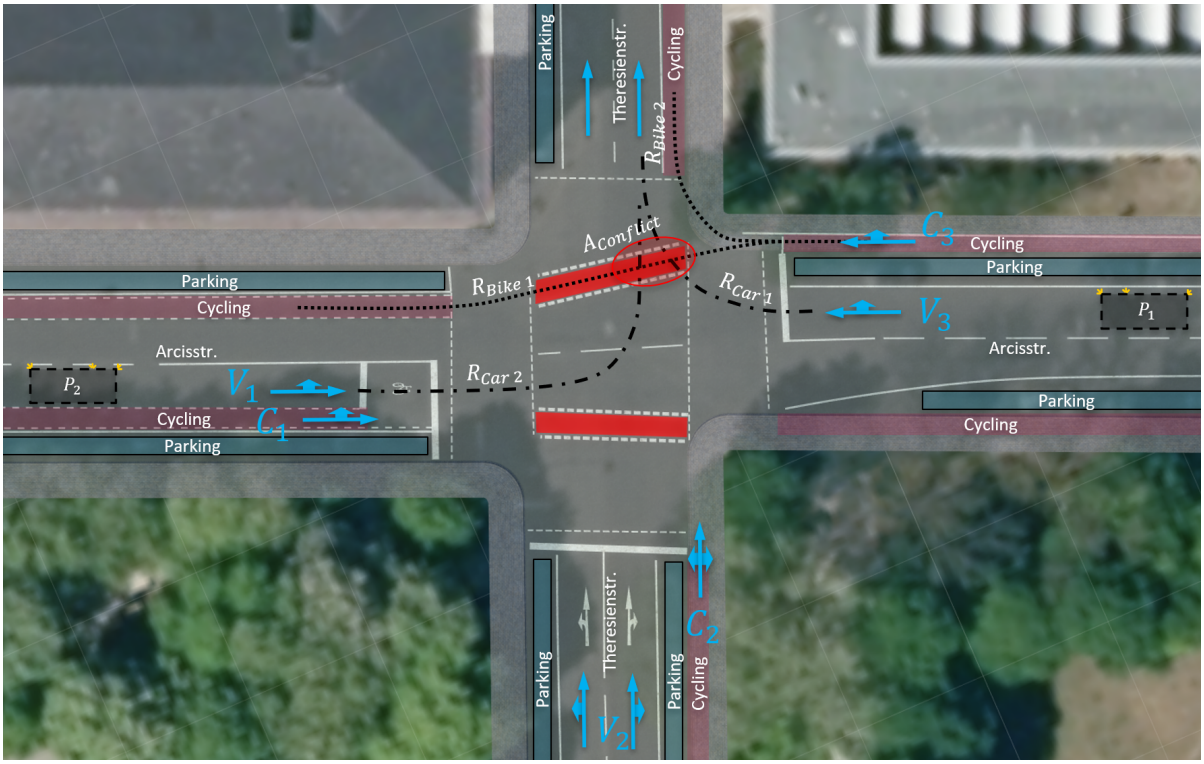


Figure 5.20: Illustration of the road network with an indication for traffic demand and urban road structure/layout

Considering this intersection's original signal phase timings, a signal program for the simulations was developed. Figure 5.21 shows the signal phase timings for vehicles $V_{1,2,3}$ and cyclists $C_{1,2,3}$. Notations are aligned with the ones given in Figure 5.20. Starting at zero seconds, cyclists receive a head start of two seconds before cars are allowed to start. The phase for cyclists ends at 25 seconds to allow cars to cross intersections if there is a high volume of bikes because they would always block right-turning cars. After the phase for cars ends at 42 seconds, the intersection is timed to clear for 3 seconds until cyclists C_2 are allowed to start, indicated by a green light for them. From here on, the phase cycle repeats.

Four scenarios are simulated, and the impact of roadside CP to cover the occluded bike lane is analyzed. Therefore, RSI support was switched on or off.

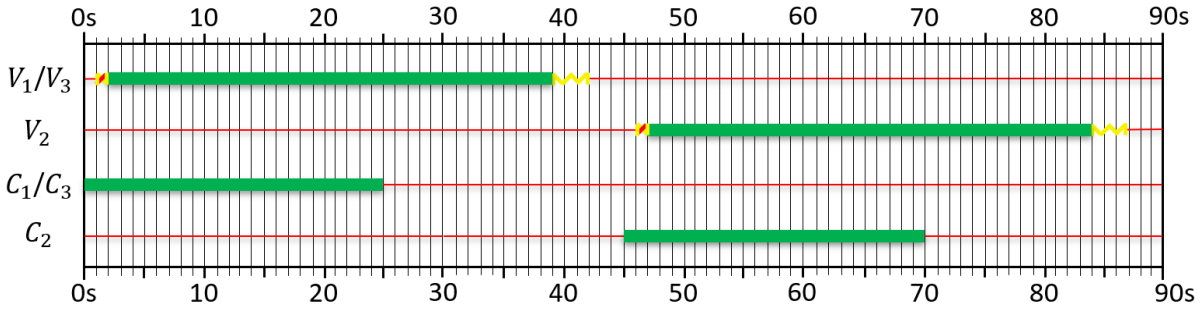


Figure 5.21: Illustration of the traffic light signal program with a cycle time of 90 seconds used in the simulation. Small thin red lines indicate a red signal, and green bars a green signal phase. Yellow indications on the line represent a yellow signal phase and a yellow/red mixed marking if both lights are active simultaneously.

5.4.1 Right Turn - Scenario$_{1,2,3}$

Scenarios one, two, and three consist of a right turn. Each consists of 50 simulation iterations where the ego vehicle drove along $R_{Car\ 1}$, which is a right turn. Each of them is compared. While the CAV is allowed to drive the maximum speed limit of the road, based on its observations, the AVs' speed is restricted while inside the intersection. The restriction varies with the scenario number.

1. Scenario$_1$ does not restrict the AV to a maximum velocity inside the intersection.
2. Scenario$_2$ restricts the AV to a maximum velocity of 2.5 meter per second
3. Scenario$_3$ restricts the AV to a maximum velocity of 0.8 meter per second

Speed restrictions are introduced subsequently as a result of evaluating Scenario$_1$ and Scenario$_2$, explained in Chapter 6.4.

5.4.2 Left Turn - Scenario$_4$

Scenario$_4$ describes a left turn, where the only difference is RSI support. Meaning that no CP is used by the AV for simulations where the ego vehicle drives along $R_{Car\ 2}$. No speed restriction other than the road's speed limit was introduced.

Chapter 6

System Evaluation

The following sections evaluate all simulations, introduced in Chapter 5 and are aligned in the same order as in Chapter 5.

6.1 Emergency Trajectories for Connected Automated Driving

In this HIL study, it was possible to maneuver a CAV solely based on infrastructure information in the real world. The results discussed in this section are the driven track of the VUT, data transmission time, and data loss.

Figure 6.1 shows a red polynomial-shaped line indicating the driven track of the VUT. Lane markings are added in black to aid the readability of the graphic. Blue straight lines between the lane markings show reference tracks that the ego CAV can use to maneuver itself. A miniature version of Figure 5.2 is added for comprehension.

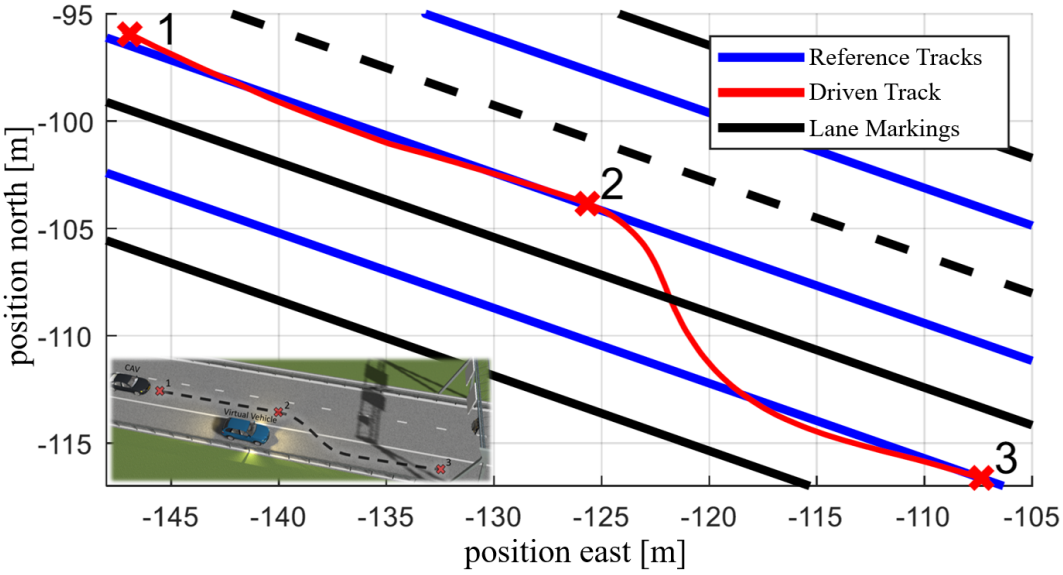


Figure 6.1: Illustration of the driven track of the ego vehicle during the VeHIL test. The actual driven track is indicated by the red line.

At the red cross marked with number one, the VUT switched to the trajectory sent by roadside ITS-S. As **Reference Track**, the middle of the three blue lanes was chosen as it is closest to the vehicle. The vehicle then kept driving along this trajectory until it reached the red cross marked with the number two, as indicated in the small image and explained in Section 5.1. A vehicle initially blocks the hard shoulder until the VUT reaches position two. After this position, the **Reference Track** is changed to the lower one on the hard shoulder. At position three, the test vehicle came to a safe stop.

The red line showing the **Driven Track** is a result of the CAVs' **Reference Track** change and proves that the proposed system could maneuver the VUT onto the hard should as it was supposed to. When the VUT reaches track point two given in Figure 6.1, the reference track is switched to the lower one, resulting in a smooth polynomial-shaped driven track that is a result of the vehicle control system. Data captured for these results were recorded using a differential GNSS system that can accurately perceive the actual vehicle position with a maximum error of two centimeters.

To validate proper communication operation between the CAV and RSU, the turnaround time between sending and receiving a message from the RSU to the CAV and back was measured. The turnaround time is shown in Figure 6.2, where the average time is 32.3 milliseconds, while a maximum of 172.9 milliseconds was observed.

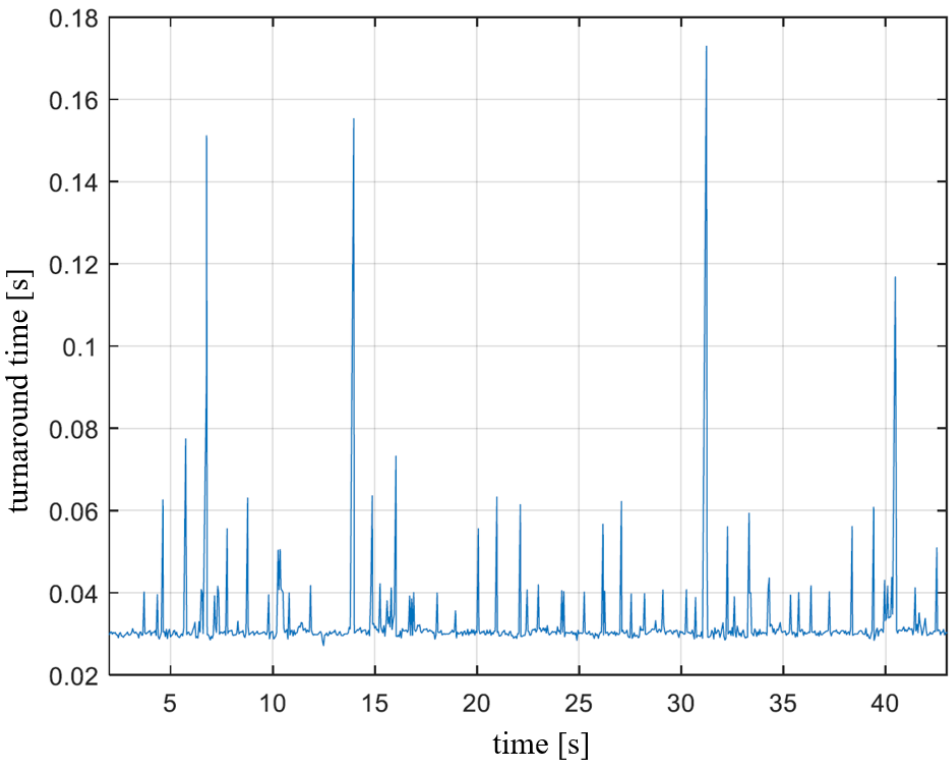


Figure 6.2: Turnaround time measurement for sensing a message to the roadside ITS-S from the vehicle and back.

For the maximum transmission time, an analysis of data consistency was performed. Figure 6.3 shows data packages on the y-axis send/receive timings on the x-axis. Blue circles indicate

packets that were sent, and red circles packets that were received. The experiment is designed so that a float number is sent and received, and the time in between is measured. Figure 6.3 shows the send and receive timestamps with circles. On the y-axis, one can see that packets 722, 723, and 724 were sent and received correctly, while packet 725 was received much later than the others before. When a packet is sent too late, the following packet may be lost entirely in our setup. A lost packet is shown by the red dot (packet 726) on the blue send line, as it was sent but never received. Nevertheless, results from the successful test show that this timing issue does not critically influence our system. A timeout observer could shut down the system in case of too much packet loss.

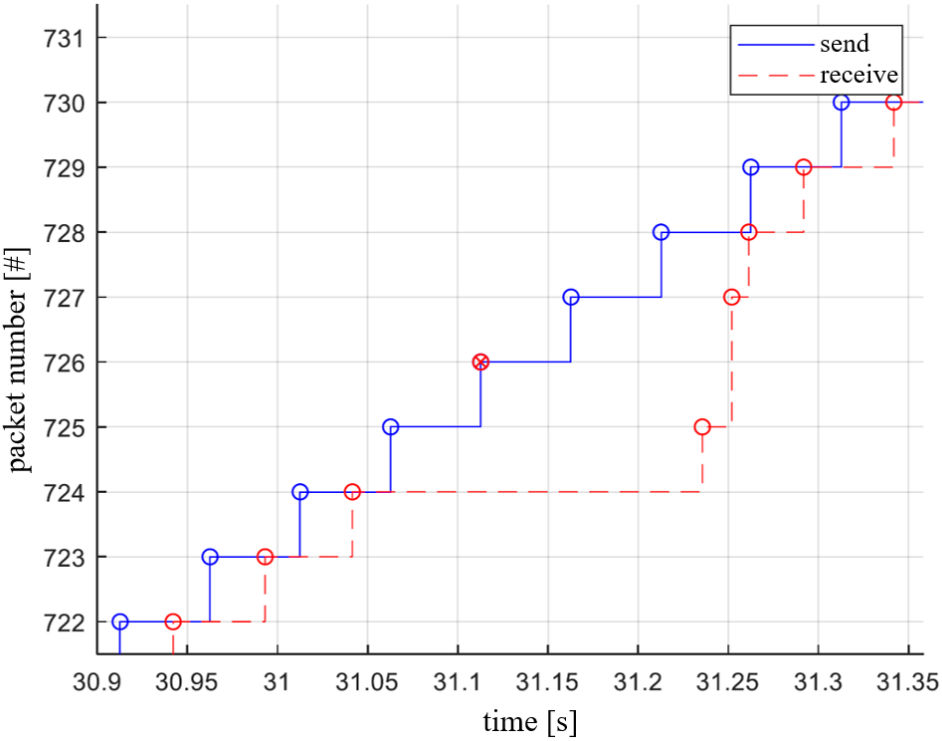


Figure 6.3: Detailed analysis of jitter and its effect on message transmission between the ego vehicle and the roadside ITS-S.

The study shows how roadside infrastructure can help increase the safety and reliability of CAVs. This technology has great potential for the future of automated driving. The solution was also patented with the title: "INFRASTRUCTURE-SUPPORTED PROCESS OF ASCERTAINING TRAJECTORIES FOR AUTONOMOUS VEHICLES [Markgraf et al., 2022]."

6.2 Roadside ITS-S Perception for Urban Automated Driving

For this simulation study, 180 iterations were evaluated, where one iteration consists of a single driving maneuver of a CAV. The work aimed to find the impact of roadside infrastructure in an urban signalized intersection. Therefore, RSI support was enabled and disabled during different simulation runs. Another parameter change is the traffic demand on the major and minor roads.

Figure 6.4 shows the driven path of the AV and CAV for their simulation runs. One of the path plots indicates the egos-driven path with RSI support enabled and one with RSI disabled. The VUT starts on the curb at the bottom of Figure 6.4, the minor road. Then it drives towards the stop line, turning right onto the major road. The driven path of the VUT does not show a significant difference when RSI support is added. This absence in the path difference is expected as the lateral control behavior should not be impacted by roadside ITS-S information used in this study.

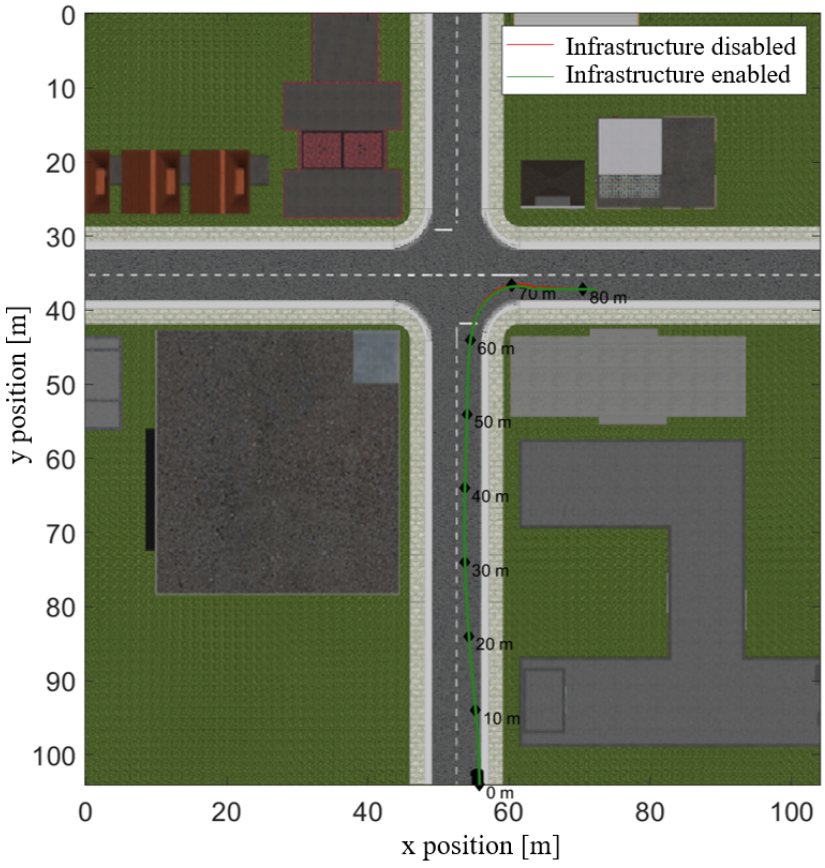


Figure 6.4: Top down view of the simulated scenario with the driven path of the simulated vehicle. In red the path of the AV is given and in green the path of the CAV.

Besides a driven path, the acceleration and velocity of the ego are compared. Figure 6.5 and Figure 6.6 show velocity and acceleration of the VUT with and without RSI support. The

x-axis highlights the driven distance corresponding to the rear axle of the ego AV.

In Figure 6.5 the travel time over driven distance for an AV and a CAV are given. As shown in Figure 6.4 the slow-down point for merging onto the major road is at around 60 meters. Until the ego vehicle arrives at the intersection, both lines overlay each other, indicating no difference in driving behavior until this point. It can be observed that a vehicle utilizing RSI support takes less time than a vehicle with no support from RSI systems. This time difference accumulates as the car - illustrated by the solid red line - slows down at a driven distance of 56 meters in Figure 6.5. The time on the y-axis increases while the distance on the x-axis does not, describing a short-stopping maneuver. At the end of the track at 81 meters, a difference between the dash-dotted and solid lines of 0.9 seconds on the y-axis remains.

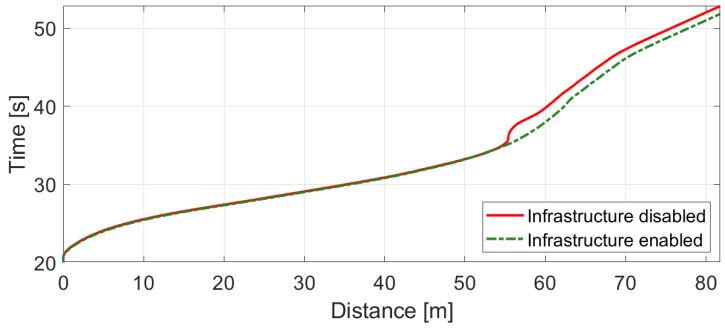


Figure 6.5: Travel time over driven distance for one single simulation, comparing an AV without roadside ITS-S and a CAV with roadside ITS-S support

These 0.9 seconds are the difference that is achieved using road ITS-S perception that increases the FoV of the CAV. It observed the major road, which is the differentiating factor compared to the AV that must stop briefly to observe traffic on the major road to proceed with its turning maneuver.

Figure 6.6 shows VUT acceleration over driven distance. Almost no difference in acceleration measurements is observed until the vehicle reaches the intersection.

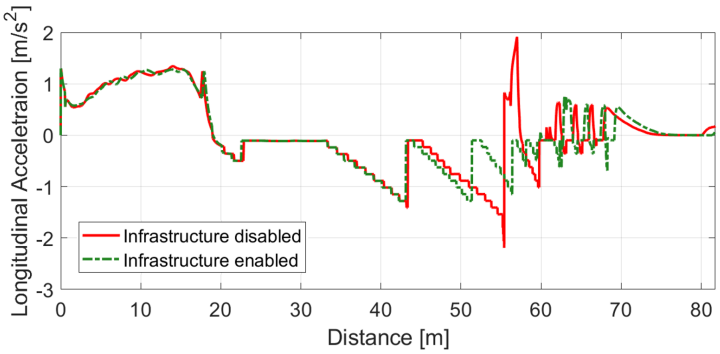


Figure 6.6: Acceleration over driven distance for a simulation, comparing a AV without RSI and a CAV with RSI support

Comparing RSI enhancements, the braking maneuver of the AV shown by the solid red line

at about 56-meter spikes first to the negative and afterward to the positive, because the AV without RSI must accelerate again. As the CAV uses roadside support, there are no similar peaks in this acceleration data plot. These peaks experienced by the AV could be reduced by slowing down even more cautiously, leading to a higher travel time. Overall both systems operate within a rather comfortable range regarding acceleration measurements.

A data summary of the whole study is given in Figure 6.8, 6.9, 6.10 and, 6.12. In Figure 6.8, 6.9 and 6.10 traffic demand is varied from 10% to 90% of the maximum value which corresponds to the maximum traffic load in an urban area on the x-axis. Box plots present each traffic demand. Ten simulations are combined into a single box plot where simulations with infrastructure support are shown in green and without RSI support in red, respectively. At first, all traffic demands are explained with a detailed representation of all simulations, concluding with a summary box plot.

Figure 6.7 describes the interpretation of the box plot used in this work. An outlier is observed at the top of Figure 6.7. Maximum and minimum values are given by extending lines from the box. These correspond to the most extreme sample data value, which is not an outlier. A data point is classified as an outlier if it exceeds 99.3% or $+/-$ 2.7σ coverage of the rest of the data. The 75th and 25th percentiles are the corresponding 25th and 75th percentile of sample data, respectively. The median is given by the dot in the middle as named in Figure 6.7 and may not be confused with the average value. For small data sets with very few sample points, there may be no most extreme maximum value, and the box plot can end at the 75th percentile and will, in this case, have at least one outlier beyond the corresponding percentile.

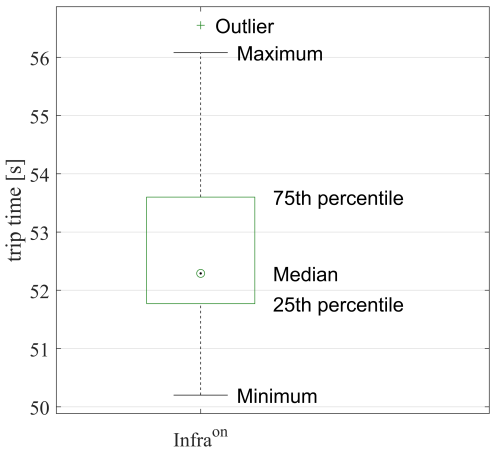


Figure 6.7: Box plot description showing percentiles, median, minimum, and maximum

Box plots in Figure 6.8 show the overall simulation results for trip time for specific traffic demands. Whether RSI support was used within the ego vehicle or not is shown by the superscript on or off in x tick labels.

When infrastructure support was used by the vehicle, the box plot is indicated in green, and if no RSI is used in red. There are distinctive observations worth mentioning. If RSI is utilized by the VUT, trip time is reduced. For 50% traffic demand with RSI support enabled, the box plot does not have a maximum value. This results from low samples in the simulation data set when comparing single-demand simulations. Samples are within the 75th percentile, but outliers are beyond a suitable maximum value. The sample data size used for the box plots is ten. Nevertheless, this illustration achieves a comprehensive representation of the simulation results. On average, the trip time is reduced by 1.63 seconds if RSI enhancements are used by the CAV.

Figure 6.9 shows the maximum acceleration experienced by the ego vehicle during a simulation iteration. Each demand consists of ten simulation iterations. Overall, for almost every traffic demand, the maximum acceleration of the VUT is lower when RSI is used comparing

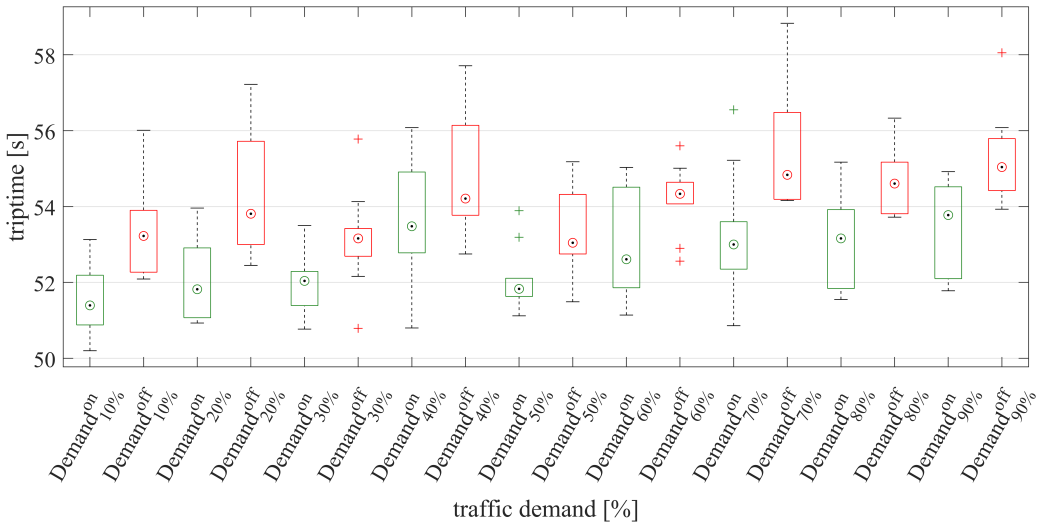


Figure 6.8: Box plot data highlighting the trip time of the VUT with respect to different traffic demands and utilization of roadside ITS-S.

the median. At 40% traffic demand, the AVs' median without RSI is slightly lower than the CAVs'. The same is observed for a traffic demand of 70%. It is believed that this is a result of an edge case. Overall, one can see that maximum acceleration decreases when RSI is utilized by the VUT.

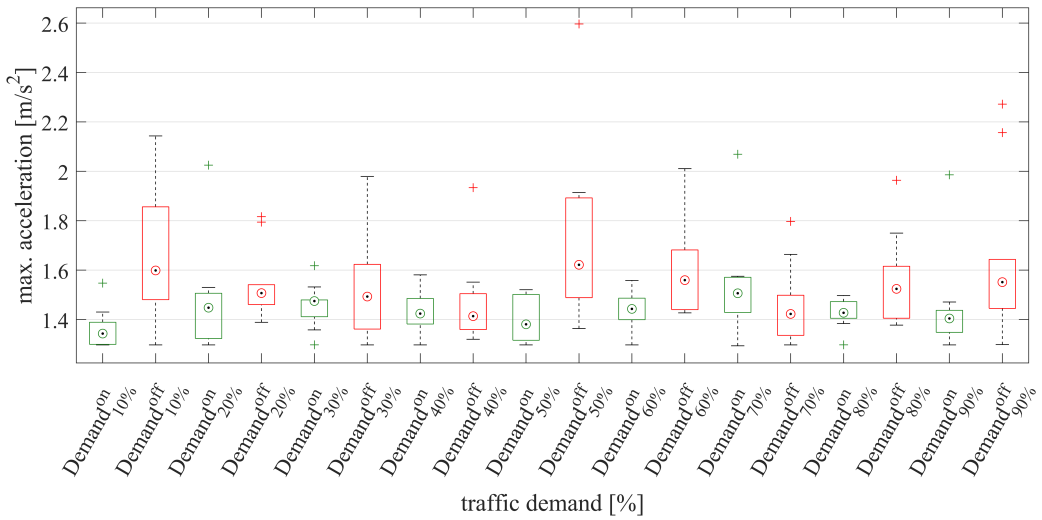


Figure 6.9: Box plot data highlighting the maximum acceleration of the VUT with respect to different traffic demands and utilization of roadside ITS-S.

Figure 6.10 indicates the maximum deceleration measurement for traffic demands with and without RSI support. The maximum deceleration data recorded in simulations is lower comparing the median of simulations with and without RSI support. The variation in maximum deceleration sample data becomes even more visible in this representation. For demand 10% and RSI^{off}, the maximum and minimum values are too close to the 25th and 75th percentile

to be visible in this representation. The same applies to other box plots where upper or lower values can not be observed, and no outlier is found in the data.

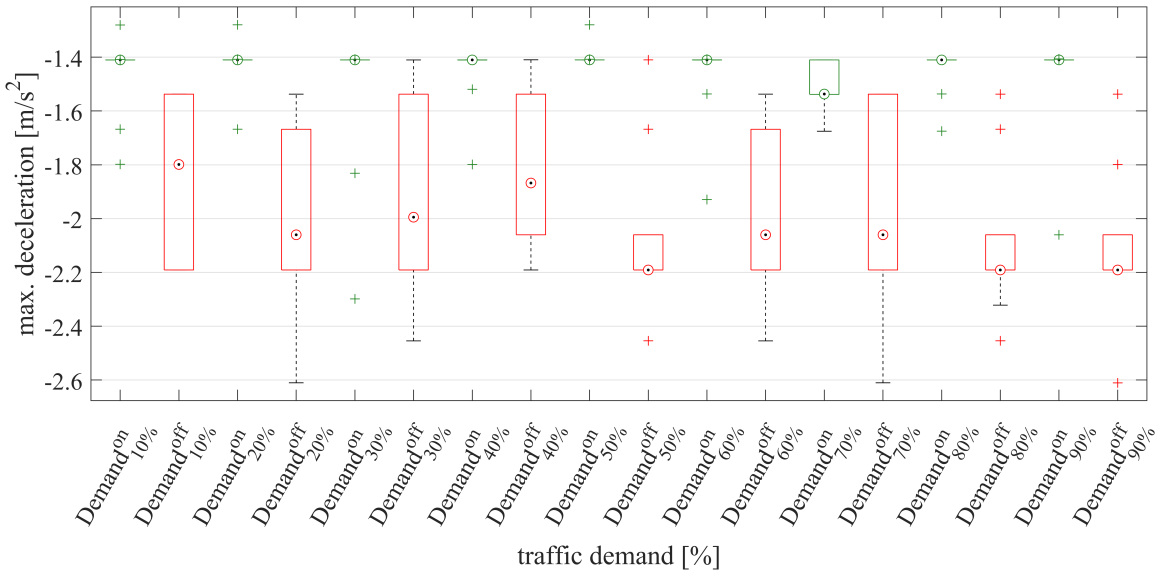


Figure 6.10: Box plot data highlighting the maximum deceleration of the VUT with respect to different traffic demands and utilization of roadside ITS-S.

The 25th and 75th percentile is close to the median for almost all simulations. In Figure 6.11, a close-up of the 90% traffic demand with RSI^{on} is shown. The median is very close to the upper and lower percentile values compared to simulations without RSI support, considering values on the y-axis. The maximum value is not shown in the close-up as an upper outlier is slightly visible in the overall boxplot illustration on the left. This representation of the boxplot suggests that a low variation and a high consistency of the maximum deceleration is achieved when RSI is used.

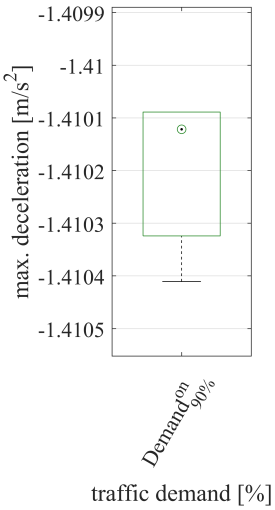


Figure 6.11: Close up illustration of the box plot showing the simulations with a traffic demand of 90% and RSI support enabled.

In Figure 6.12 box plots featuring all traffic demands with roadside ITS-S support on and off are shown. On the left side, trip time is displayed. Table 6.1 shows the median of each plot. The trip time, as explained in Table 6.1, is reduced by 1.86 seconds, looking at the median of all simulations when RSI is utilized in a CAV. Maximum acceleration measurements throughout the study show that the CAV has a slightly lower median value of 0.0771 m/s^2. A significant difference is observed in the maximum deceleration measurements, where the median for a CAV has a reduced acceleration value by 0.65 m/s^2 compared to a common AV. Outliers given in the max. deceleration plot are a result of several different situations, encountered by the CAV. While the AV with RSI^{off} will always slow down and observe the intersection before proceeding with its right turn, the CAV will only have to stop in some cases. Considering 90 iterations with RSI^{on}, most iterations are within the 25th and 75th percentile, even though it looks like a high amount of outliers.

Table 6.1: Summary of median values from measurements of the simulation study.

	trip time	max. acceleration	max. deceleration
RSI^{on}	52.29 s	1.43 m/s^2	-1.41 m/s^2
RSI^{off}	54.15 s	1.50 m/s^2	-2.06 m/s^2
difference	1.86 s	0.08 m/s^2	0.65 m/s^2

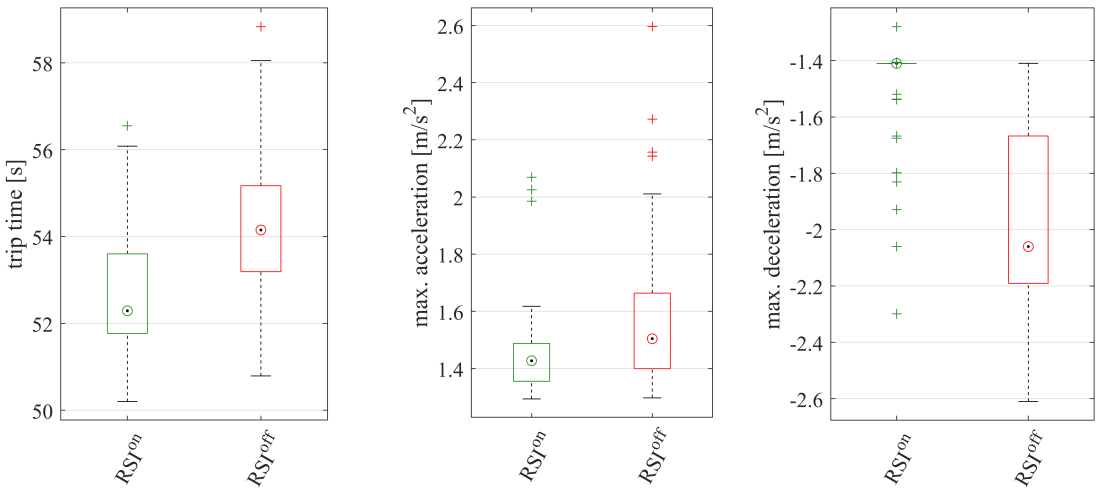


Figure 6.12: Boxplot summarizing all data given in Figures 6.8, 6.9, and 6.10

To summarize the findings of this study, one can draw two major conclusions. Firstly, RSI systems can increase traffic efficiency by decreasing the time a CAV needs to turn right. Even at peak hours, the time a CAV needs to perform its driving task is decreased. Nevertheless, the decrease in maneuver time is minor compared to the time that an AV without RSI support needs. Another finding is the impact on driving comfort, indicating an improvement when RSI is used. Incorporating more traffic participants such as VRUs will impact the retrieved data significantly. As a result, the simulation framework is extended in the following sections to incorporate this topic.

6.3 Occlusion Evaluation regarding Bicyclists and Cars

As explained in Chapter 5, the simulation framework was enhanced in various aspects. For example, VRUs were added, a more realistic intersection setup was chosen, and many changes to the simulation system were added. The evaluation of this study starts with an in-depth analysis of a single simulation iteration. After that, comprehensive simulation data is discussed.

Figure 6.13 shows the velocity over distance on the driven path (Figure 6.15) of the ego vehicle during one single simulation iteration. The line shows a smooth velocity profile. The VUT accelerates until it reaches the driven distance of 17 meters, then it starts coasting toward the intersection where it slows down and stops at a stop line related to a red signal (54 meters, Figure 6.15). After the signal changes to green, the AV briefly accelerates until it slows down again to avoid a VRU cyclist to that the VUT must yield its way of right. Then the turning maneuver is resumed and concluded with accelerating to the maximum desired velocity.

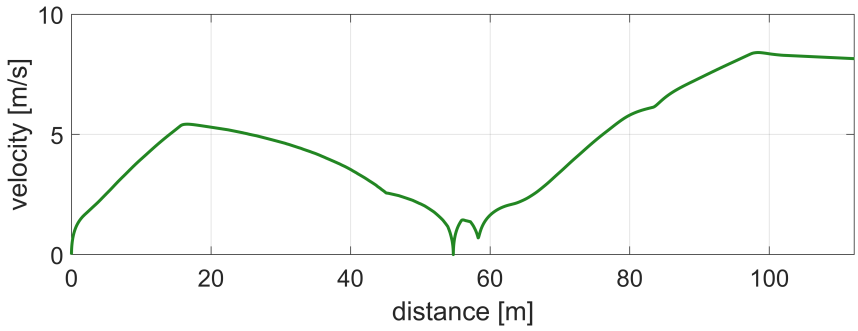


Figure 6.13: The velocity of the VUT plotted over track distance given in Figure 6.15

Figure 6.14 corresponds directly to Figure 6.13. In the beginning, the VUT starts accelerating until it starts coasting toward the signalized intersection. At the intersection, the VUT stops, which is visible as the most negative value in Figure 6.14 at 54 meters. Then the vehicle accelerates and decelerates again, as the described VRU before approaches.

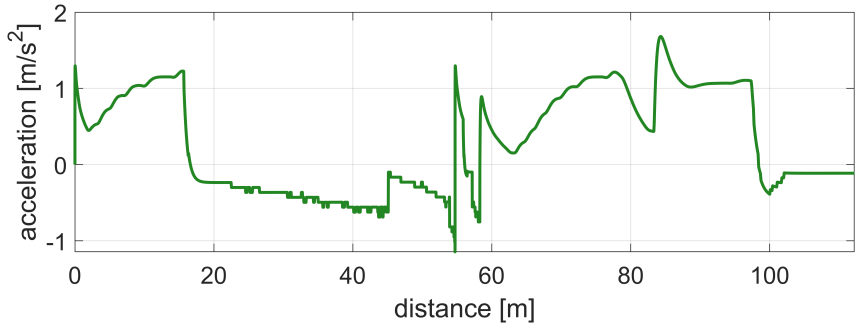


Figure 6.14: The acceleration of the VUT plotted over track distance given in Figure 6.15

In Figure 6.15, the driven path of the ego vehicle is shown in a top-down view. In the top of Figure 6.15, the VUT starts and drives towards the major road. The driven path is given by a green line with position markers on the line, showing the driven distance in meters. The

path is the actual driven path of the ego AV during a specific simulation. The driven path corresponds to the speed and acceleration profile plots in Figure 6.13 and 6.14.

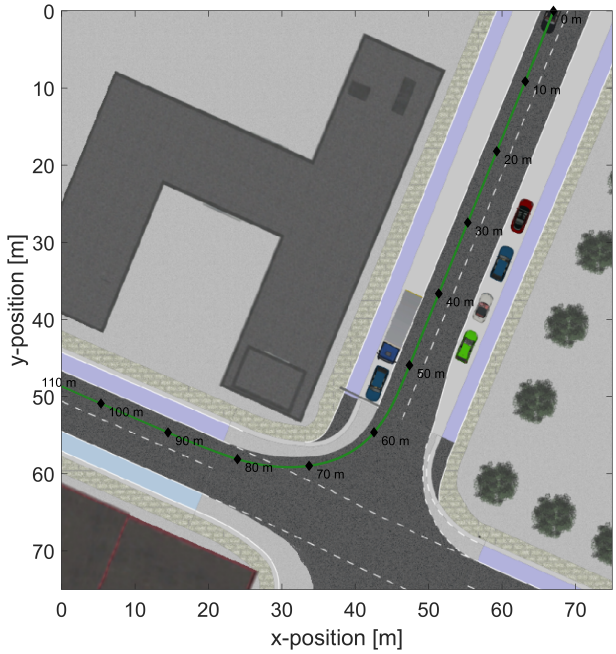


Figure 6.15: The driven path of the VUT during the simulation as reference to compare acceleration and velocity along the track.

Figure 6.16, 6.17 and 6.18 each show a comparison between the AV driving with a FoV of 50 meter, 30 meters and 30 meter and additional occlusion. On the y-axis, the corresponding value is given on the x-axis seed number. Each seed belongs to a unique traffic scenario. Regarding Figure 6.16, maximum acceleration is shown for each simulated seed. There is no significant difference between different FoVs or occlusion. Some outliers are observed for all scenarios, but there is no unusual behavior for AVs whether the FoV is 30 meters, 50 meters, or there is much occlusion.

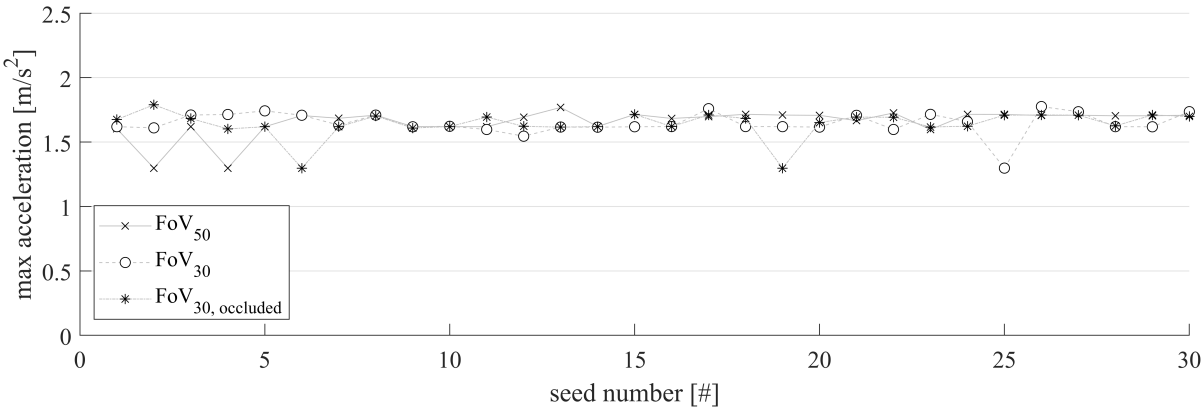


Figure 6.16: Maximum acceleration values for all simulation seeds. The three lines highlight the difference between FoV and occlusion parameter variation.

Concerning maximum deceleration in Figure 6.17, one can see very few outliers when FoV is reduced to 30 meters, except for simulations with seed numbers 2, 5, and 17.

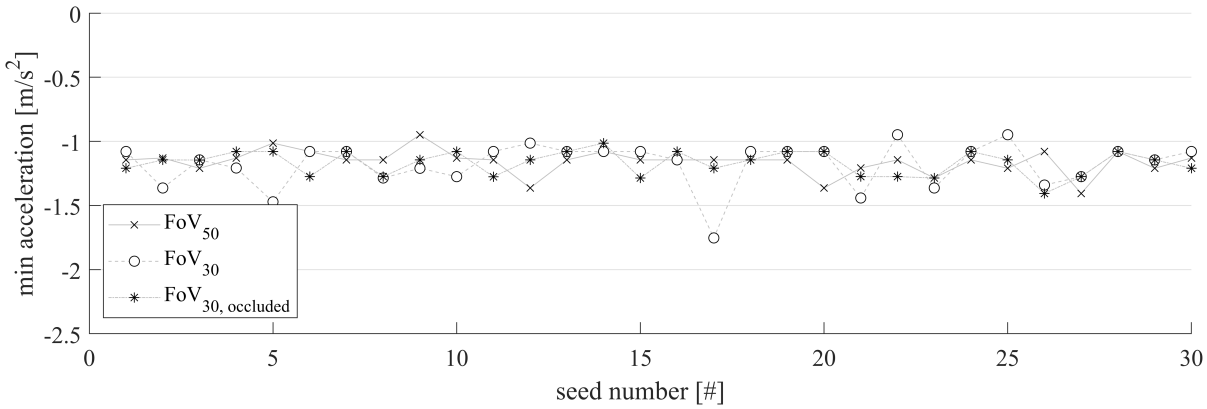


Figure 6.17: Maximum deceleration values for all simulation seeds. The three lines highlight the difference between FoV and occlusion parameter variation.

Figure 6.18 shows trip times of all simulations. In this representation, only one significant outlier is observed: the trip time of an CAV with a FoV to 50 meters. This results from the cautious behavior of the vehicle's driving system as it can see further than a AV with a FoV of 30 meters. The vehicle with a lower FoV is more likely to have a conflict with VRU cyclists as it cannot see them as early as a AV with a FoV of 50 meter.

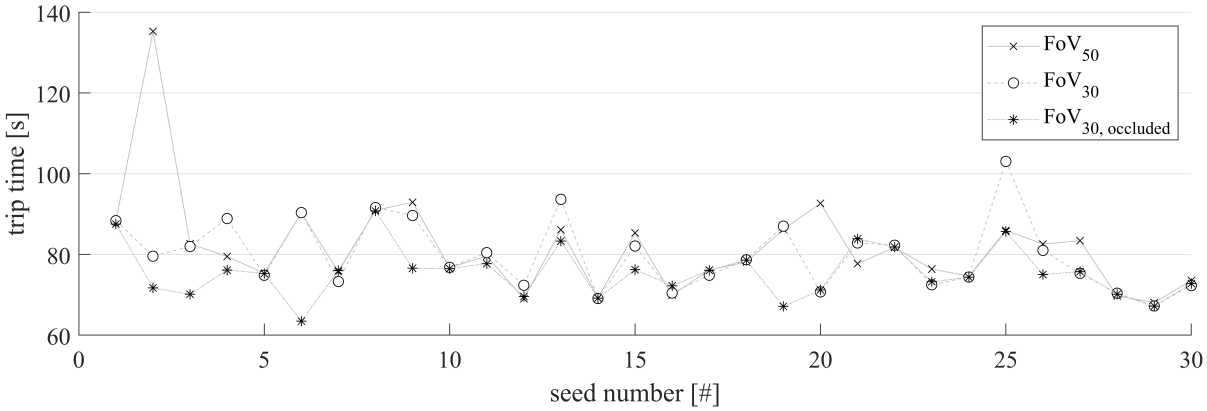


Figure 6.18: Trip time values for all simulation seeds. The three lines highlight the difference between FoV and occlusion parameter variation.

Line charts given in Figure 6.16, 6.17 and 6.18 show the whole data, while Figure 6.19, 6.20 and 6.21 uses box plots to explain results on a statistical basis. The box plot FoV$_{30,\text{occluded}}$ is added for the entirety of simulation results but cannot be compared to the other two scenes directly as it has additional occlusion in its scenario. This scenario becomes crucial concerning traffic safety. Firstly acceleration data is summarized in Figure 6.19. The median acceleration shows a maximum difference of 0.08 m/s^2. This difference is relatively low and, therefore, unlikely to be noticed by passengers.

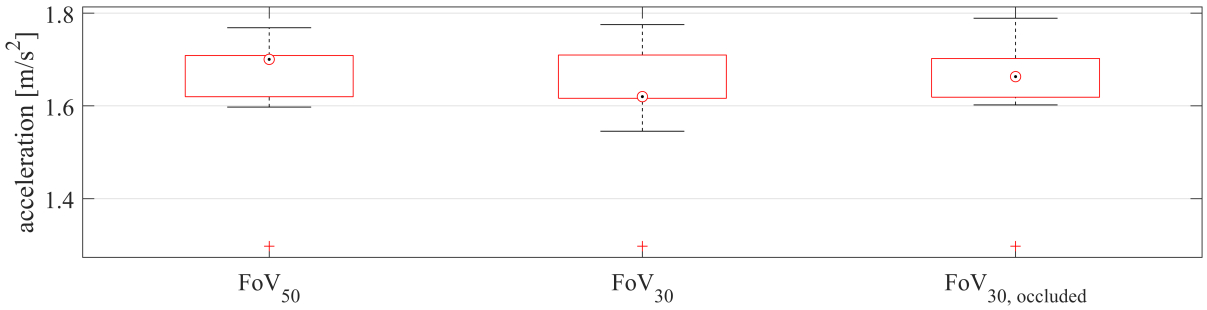


Figure 6.19: Box plots for the maximum acceleration of the VUT for scenarios evaluated in this study.

Figure 6.20 shows maximum deceleration measurement using box plots. There is no significant difference in the median values of each dataset, as the maximum difference is 0.065m/s^2. Nevertheless, a slight reduction in variance occurs if the AV uses a FoV of 50 meters instead of 30.

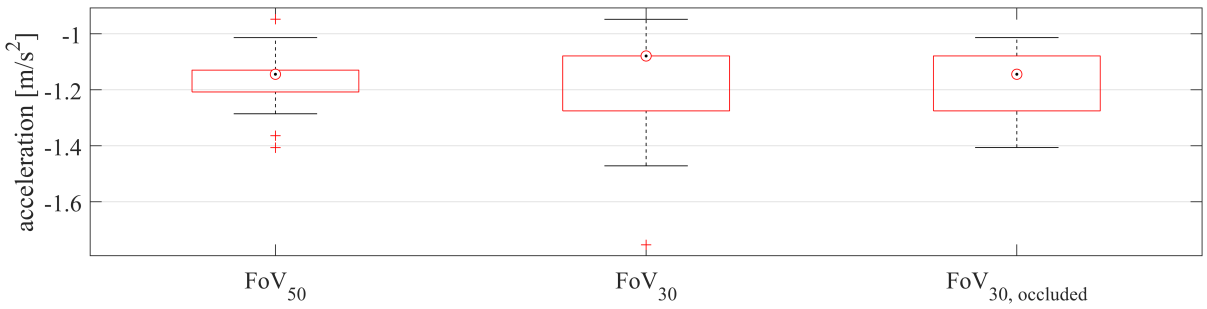


Figure 6.20: Box plots for the maximum deceleration of the VUT for scenarios evaluated in this study.

Box plots for trip time are given in Figure 6.21. On the left side, the box plot for FoV$_{50}$ shows less variation considering maximum and minimum whiskers compared to FoV$_{30}$. Additionally, the outlier for FoV$_{50}$ described in Figure 6.18 can also be seen here.

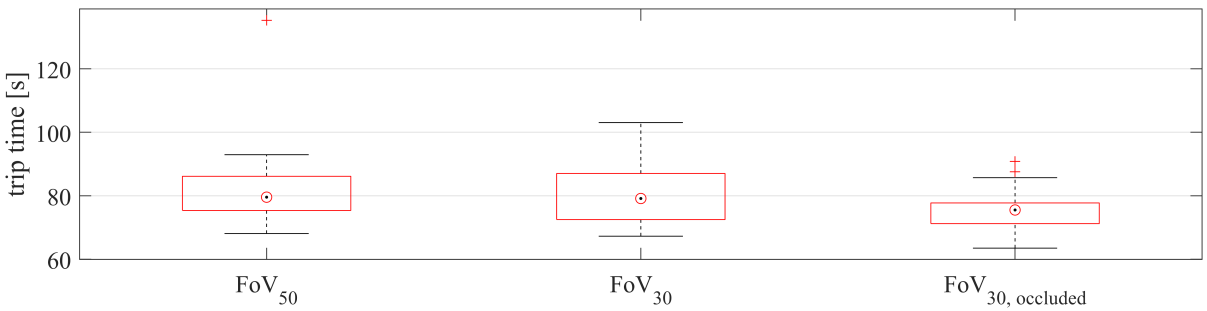


Figure 6.21: Box plots for the trip time of the VUT for scenarios evaluated in this study.

The pie chart given in Figure 6.22 shows the central finding of this study. This pie chart considers simulations of the AV utilizing the setup FoV$_{30,\text{occluded}}$. As the notation indicates, the

VUTs' FoV was occluded by parked vehicles that explicitly reduce the visibility of cyclists on the bike lane. The AV cannot slow down with an acceleration beyond 1.5 m/s^2, implemented as a soft constraint for the system. The vehicle may exceed the target acceleration slightly due to control errors. More severe braking acceleration measurements are unacceptable as these could result in injured passengers or decrease driving comfort. Such behavior with high braking acceleration is not acceptable and therefore restricted.

As the VUT is not allowed to decelerate beyond comfortable limits, several unsafe situations were observed (Figure 6.22). These unsafe situations are divided into issues where the AV had a **Collision** (6.7%) with a cyclist and **Dangerous** (23.3%) situations where the AV did not yield way of right to the cyclist. The rest of the simulations are considered **Safe** and make up 70 percent of all iterations.

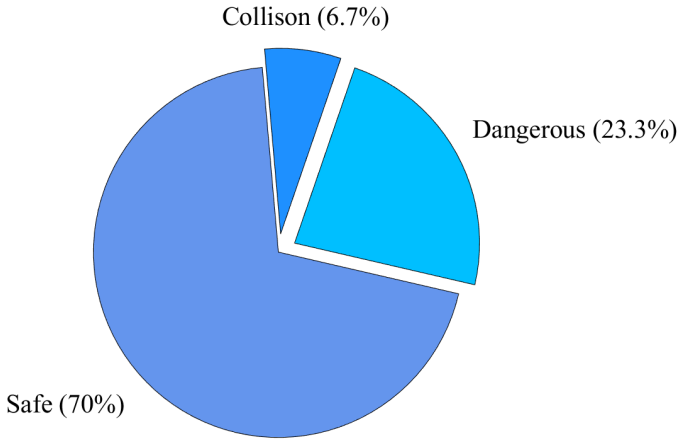


Figure 6.22: Pie charts to highlight the number of safe, unsafe and dangerous situations considering VUT and VRU interactions. The pie chart on the left side is

Firstly this study showed that the maximum detection range of an AV has a minor effect considering a comparison of 30 and 50 meters. The AV had to decelerate to take the right turn and therefore was driving at a velocity where the detection range of sensors comparing 30 and 50 meters did not make a major difference. Driving comfort regarding maximum and minimum acceleration measurements is not impacted significantly. Only slight differences were observed concerning maximum deceleration variation in the data set.

Secondly, occlusion impact on cyclists' safety was analyzed by replicating an actual intersection. The gathered data suggests that an AV cannot maneuver safely and comfortably through the intersection utilizing the driving software deployed in this study. An AV utilizing emergency maneuvers such as high dynamic braking may navigate through the intersection without a collision. Nevertheless, such a system will encounter blind spot scenarios, leading to safety-critical situations and severe drawbacks for passenger driving comfort and VRU safety. This study concluded with several open questions, considered in a follow-up simulation study in Section 6.4.

6.4 Collective Perception for CAVs utilizing roadside ITS-S

As explained in Section 5.4 the bike lane is occluded for either AV and CAV. 50 HIL simulations were executed and evaluated. While it is expected that a CAV might be able to maneuver the situation properly, the AV will end up in several conflicting scenarios as evaluated in Section 6.3 already. The driving stack was enhanced and a direct comparison between each simulation seed was performed to achieve significant results.

Figure 6.23 shows two captured frames from the simulation. This image should give the reader a visual understanding of the simulation. On the left side, the AV can be seen, and on the right side the CAV. Both images are captured exactly at the same time with the same traffic conditions. It is observed that the CAV did already finish while the AV is still performing its turning maneuver.

Figure 6.23: A side-by-side simulation comparison of the AV on the left side and the CAV on the right side, showing the same simulation time with and without the support of roadside ITS-S.

In Section 5.4, right and left turn maneuvers that are simulated — both with and without the support of RSI — were explained. Before results can be discussed, a shared understanding of different kinds of conflicts in urban traffic must be established. In this study, the term major and minor conflict is used.

1. Major conflict: Whenever a cyclist and the ego vehicle are involved in a collision or were close to a collision. Such a collision is considered to have a likelihood of fatality[1].

2. Minor conflict: If the VUT did not yield the way of right to the cyclist. Nevertheless, the cyclist can avoid a severe outcome by braking for the VUT.

The following results are divided into right and left turn maneuvers.

[1] An explicit statement on accident fatality is not possible in this simulation setup. As a result, major conflicts are conflicts where the likelihood of a fatal accident is increased and it "could" be fatal

6.4.1 Right turn - Scenario$_{1,2,3}$

Figure 6.24 shows results regarding conflicts in percentage. 100% corresponds to all 50 simulation iterations of a scenario. The most left bar in Figure 6.24 shows the baseline for simulations utilizing RSI/roadside ITS-S support. Only for one out of 50 iterations a minor conflict was observed. If scenario one without RSI is considered, conflicts happened for 48% of simulation runs. 24% may be considered major and 20% minor.

For scenario two, the maximum velocity of the ego while it is inside an intersection is reduced to 2.5 meters per second. As a result, the number of conflicts was reduced to 32% overall. 20% of those may be considered major and 12% minor.

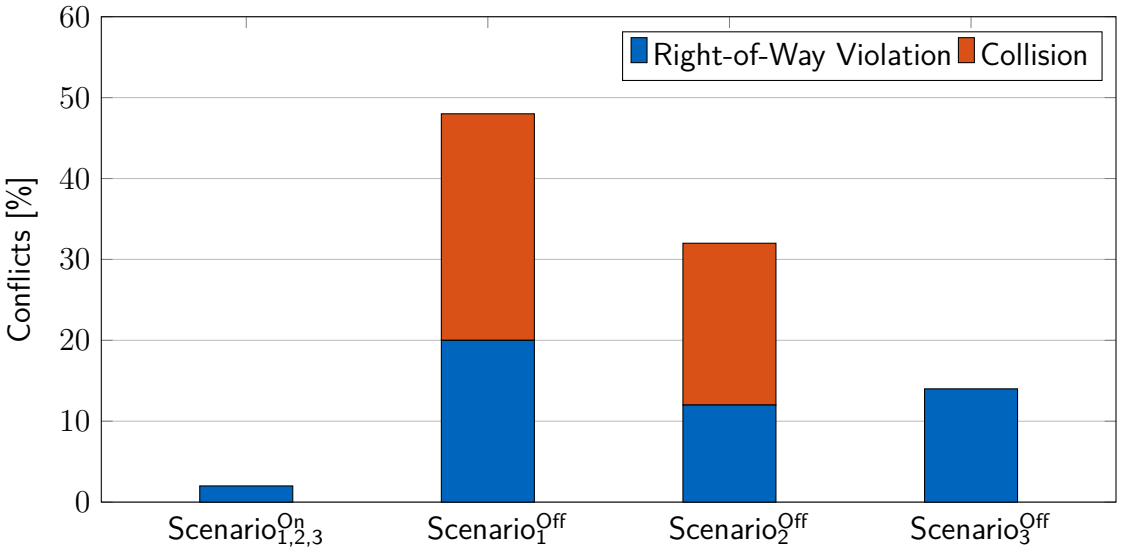


Figure 6.24: Conflicts of the VUT regarding bicycles for the different simulation scenarios.

In the next step, scenario three was defined, and as such, the maximum velocity while the AV is inside the intersection was restricted to 0.8 meters per second. This speed results in no more major conflicts because the ego vehicle is driving through the intersection at such a low velocity that a VRU that the VUT cannot observe will be able to see it and avoid a conflict itself. This kind of conflict is considered minor, as explained before. Summarizing Figure 6.24, the simulations provide evidence that even an exceptionally cautious AV performs worse regarding conflicts than a CAV.

Box plots for all maximum deceleration values can be seen in Figure 6.25. The CAV with RSI support is shown on the left side (Scenario$^{On}_{1}$). In the middle, Scenario$^{Off}_{1}$ and Scenario$^{Off}_{2}$ are given. These are displayed slightly in transparent grey to indicate that these simulations resulted in major conflicts with cyclists. This behavior is unacceptable for an AV. Therefore, it is most relevant to compare only the first and last box plots as these only show minor conflicts and may be acceptable for maneuvering the intersection. Comparing these box plots, the median maximum deceleration for an AV is 0.183 m/s^2 lower than for the CAV. Additionally, variation in sample data is increased, which is observed by larger percentiles and more extreme outliers. Overall the difference between both is insignificant, except for AVs' outliers that exceed 3.0 m/s^2. These outliers indicate uncomfortable driving behavior. Maximum acceleration is not

discussed further, as it is 2.0m/s^2 for all simulations.

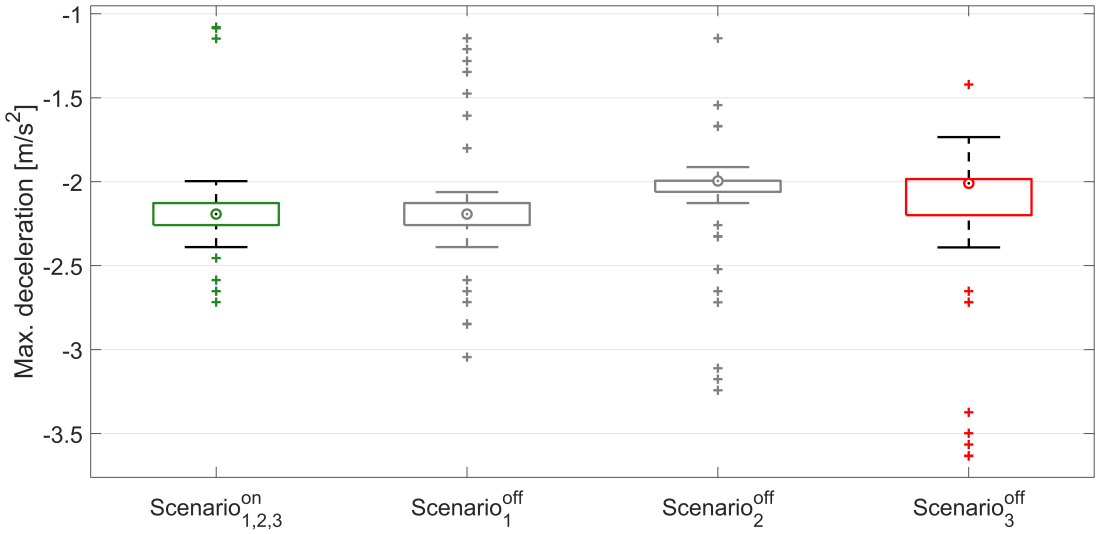


Figure 6.25: Illustration of the road network with an indication for traffic demand and urban road structure/layout.

The most significant difference in this simulation setup is observed in trip time. Figure 6.26 shows box plots for all travel times of the ego vehicle. On the left side RSI enhanced driving is given. The trip time of the AV is shown on the right side. Scenario three is given by two box plots. The first is the simulation's raw trip time, while *Scenario$^{off}_{3}$ corresponds to a corrected travel time.

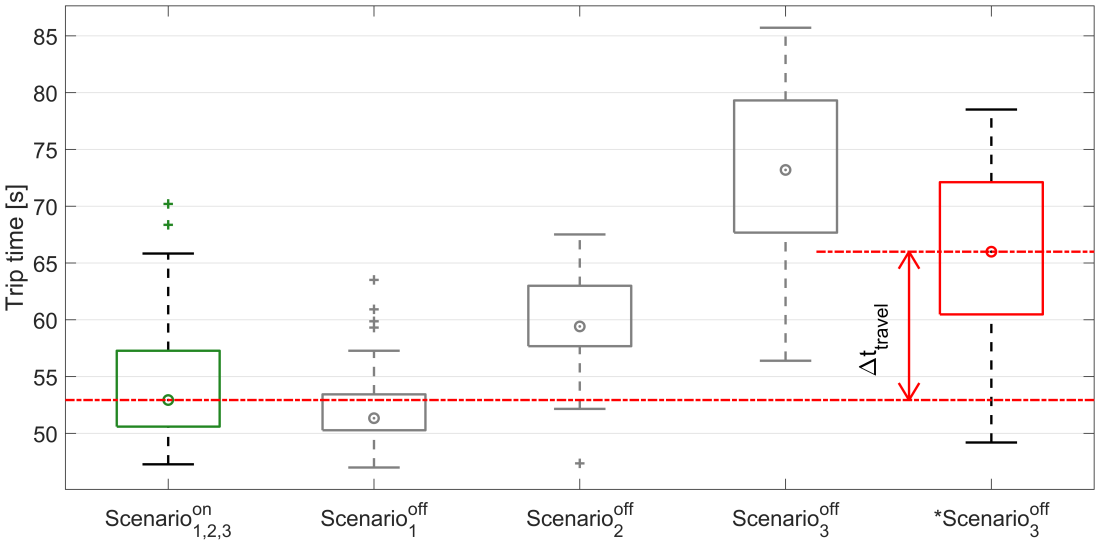


Figure 6.26: Illustration of the road network with the indication for traffic demand and urban road structure/layout.

The ego AV is forced to drive slowly through an intersection to achieve a driving behavior that does not include major conflicts. This is achieved by geofencing for intersections. Its

maximum velocity is restricted as long as the AV is inside the intersection. It is reasonable to assume that an AV will speed up right after it has passed the blind spot, meaning it can observe the bike lane. The distance it is driving after the blind spot is 7.2 meters with a velocity of 0.8 meters per second, resulting in a travel time of 9.0 seconds. The VUT could drive at 4.0 meters per second, resulting in a travel time of 1.8 seconds. Ultimately a time difference of 7.2 seconds must be subtracted. This correction must be applied for a fair comparison between utilizing RSI support and not.

Using corrected trip times, a Δt_{travel} of 13.26 seconds for the median is achieved, increasing traffic efficiency by 19.8%.

6.4.2 Left turn - Scenario$_4$

For the right turn, it was established that there are almost no conflicts when infrastructure is enabled for the VUT. For the left turn maneuver, the driving software could not create acceptable behavior concerning conflicts with other traffic participants. The improper behavior results from the planner missing the left turn feature. The planning system or driving stack cannot correctly maneuver an unprotected left turn. Therefore, no traffic efficiency analysis based on travel time was done.

Figure 6.27 compares conflicts during the right turn maneuver (Scenario 3) and the left turn (Scenario 4). The percentage of conflicts with infrastructure support is already relatively high (36%), but there is a significant improvement when RSI is utilized, as no RSI usage results in 70% conflicting simulation scenarios.

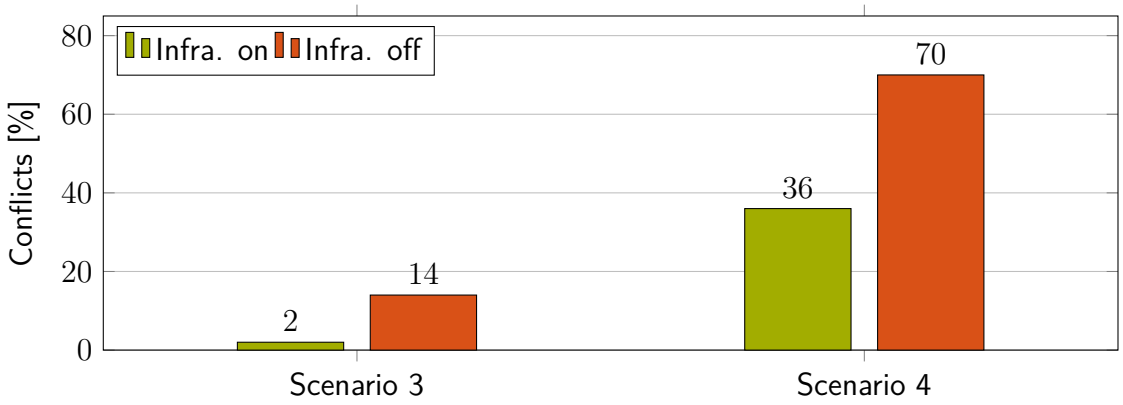


Figure 6.27: Comparison of the safety of a blind right (Scenario 3) turn and a blind left turn (Scenario 4) at the signalized intersection.

This study highlights the impact of roadside ITS-S CP in an urban area. While driving comfort is not influenced in a significant way, traffic efficiency is. With the boundary constraint that a CAV or AV must drive safely and comfortably, meaning no aggressive braking maneuvers are allowed, the AV must drive significantly slower to achieve at least a somewhat acceptable performance regarding conflicts. Even though the AV drives slower, the minor conflicts are still several magnitudes above the CAVs'. Additionally, it will be challenging for almost any urban road network to compensate for slow AVs, considering the increased travel time of 19.8%, when no roadside ITS-S is available.

Chapter 7

Conclusions and Future Work

This chapter provides an overview of the findings of this thesis, divided into the sections summary, contributions and future work.

7.1 Summary

From the analysis of the most recent work in the field of roadside ITS-S, a research gap and related research questions were defined. Several HIL simulator studies were performed and evaluated to provide new insights on RQ1, RQ2, and RQ3. A VeHIL system was used to show the feasibility of RSI sending an emergency backup trajectory that enhances traffic efficiency and safety of the road network. In the initial investigation on roadside ITS-S, research questions and state of the art were evaluated further to achieve a meaningful starting point for submicroscopic simulations of AVs and CAVs in urban environments. Additionally, valuable insights on economic aspects of roadside ITS-S are presented. Roadside ITS-S CP benefits were identified and measured, utilizing a CoHIL architecture. First, one section showed that RSI systems aid traffic efficiency and driving comfort, regarding CAVs. Second, two subsequently evaluated simulations showed how roadside ITS-S utilizing CP increases traffic efficiency and safety in an urban road network.

7.2 Contributions

From this thesis, one can derive three significant findings. First, whether roadside ITS-S utilizing CP is reasonable for AVs (RQ1). Second, if a comprehensive deployment is meaningful for urban automated driving (RQ2), and finally, how roadside ITS-S can enhance perception planning and control of an AV (RQ3). Concerning the research questions explained in Section 1, the following can be derived:

RQ1: In which cases is a roadside ITS-S, utilizing CP reasonable for AVs?

Roadside ITS-S is reasonable in cases where the ego vehicle itself cannot monitor the complete scene due to occlusion induced by other road users or urban structures. Urban structures, e.g., buildings, occur naturally in most road networks as they are developed over centuries. It is unrealistic that occlusion will be removed by changing the urban environment, e.g., redesigning a building. The same principle applies to parked cars: It is unlikely that parking spots will be removed at every location where occlusion occurs, as

parked cars, in too many cases, are a reason for occlusion. Removing the parking spots is the most convenient way. Nevertheless, this will impact the already limited parking spaces for citizens.

Comparing road safety and a positive risk balance for AVs and human-driven vehicles, AVs must be more cautious regarding occlusion, as society accepts that humans make driving errors. The easiest way to achieve higher safety for AVs is by reducing the velocity of the AV. Roadside ITS-S can contribute to a more efficient traffic flow by enabling AVs to drive at a higher velocity in given situations. Roadside ITS-S utilizing CP is a feasible and meaningful for urban automated driving.

RQ2: Is the comprehensive availability of roadside ITS-S meaningful for AVs in urban areas?

To account for occlusion, the comprehensive deployment of additional sensors in the infrastructure adds little value to the overall performance of AVs in urban areas. A comprehensive deployment would incorporate equipping every city intersection with roadside ITS-S capable of CP.

One must consider whether the specific intersection is essential for the macroscopic traffic flow in a city. Based on previous work and the evaluation of simulations done in this thesis, it can be assumed that AVs will significantly impact the traffic flow of urban road networks. Results of this work further suggest that an AV without the utilization of CP will harm traffic flow for high demand intersections.

RQ3: How can smart infrastructure enhance the perception, planning, and control of an AV?

Roadside infrastructure can enhance perception, planning, and control regarding safety and traffic efficiency by deploying additional sensors at specific locations to cover critical blind spots in an urban road network. This work uses HIL simulations on a submicroscopic level utilizing CP. The FoV of the CAV is increased by roadside ITS-S, and simulation studies provide evidence for improvements in the AVs' performance regarding perception and therefore downstream steps of planning and control.

From the data, it can be seen that bicyclists make up a vast amount of accidents in urban areas. This work showed how actual automated driving algorithms running in real-time are leveraged by using roadside ITS-S CP. Detecting VRUs that are hard to observe by AVs has excellent potential.

7.3 Limitations and Future Work

This thesis gives an overview for deploying CP capable roadside ITS-S. Henceforward, the best way to achieve a more accurate result is to perform tests in a real-world environment with an actual vehicle or increase the depth of the HIL system (1). Additionally, standard solutions like traffic lights have proven their value and future solutions should not consider helping with the violation existing traffic rules, e.g., running red lights (2). An in-depth analysis considering the economic aspects of RSI systems should be considered by policymakers (3). V2V systems may substitute for roadside infrastructure at some point in the future when it is assumed that every vehicle has sensors that can see into the blind spot they create (4).

1) Simulations cannot replace real-world testing, as a simulation must be validated by actual tests in the real world. Therefore, a test setup in a city utilizing the actual road network may be conducted to validate the results of this work, as these suggest technical feasibility and benefits.

   Simulating even more detailed scenarios could be done by adding an actual bicycle rider to the simulation system. A bicycle simulator where a human is riding a bicycle in a laboratory could be used to replace the simulated cyclist with a more accurate actual bicyclist. Other extensions of the HIL architecture could also contribute to more precise results. Tests utilizing an actual roadside ITS-S system would be a great way to proceed with the findings of this work.

2) Some researchers evaluated human-driven vehicles running red light violations. Therefore, one may suggest warning CAVs or humans about such an event to avoid conflicts. Firstly, if road users violate traffic rules, this is a fault of the human that cannot be avoided entirely. Secondly, the driver assistance system will solve this issue by automatically warning the driver before violating traffic rules. Roadside ITS-S is rather suitable performing CP that increases the FoV of a CAV. As explained in the motivation of this work (Table 1.1), most issues may be solved by driver assistance systems instead. This work highlights that roadside ITS-S should be used for CP.

3) The initial investigation shows the economic standpoint of roadside ITS-S. Considering all external costs of bicycle accidents in Munich and comparing them to deployment expenses needed for ITS-S, it should be discussed further. The cost of roadside ITS-S could pay for itself by protecting cyclists.

4) In the beginning, the market penetration of CAVs will be low. As with most technologies, the maturity of these systems will advance, and market penetration is likely to increase. At some point in the future, a high penetration rate of CAVs in the road network could make roadside based ITS-S CP obsolete. Parked cars could be used as ITS-S to achieve CP that accounts for occlusion in urban areas.

   On the one hand, roadside ITS-S could be an intermediate technology that may be used until the penetration of CAVs in the market is high enough to make up for perception issues. On the other hand, each self-driving vehicle in the road network has the potential to fail. The study on emergency trajectories for CAVs using roadside ITS-S shows how

RSI could be used in the future to help with traffic efficiency. Roadside ITS-S may evolve from an intermediate solution towards a permanent necessity while automated driving systems advance.

Regarding roadside ITS-S CP, state-of-the-art commercially developed AVs may not need such systems for urban automated driving from a safety point of view. Reducing the AVs' velocity to a bare minimum might be an acceptable solution. Nevertheless, cautious behavior on the part of automated vehicles could disrupt the macroscopic traffic flow of a city. Roadside ITS-S can increase traffic efficiency and road safety at the same time. Literature shows that conflicts in urban intersections exist, and previous work suggests using RSI. This work shows the benefit for urban road networks, where having a safe, efficient, and reliable traffic flow is crucial. These are the areas where the benefit of this technology can outweigh its cost.

Roadside ITS-S CP can help to increase traffic efficiency and safety in urban areas, especially in high-demand areas where a proper traffic flow is crucial for the whole transportation network.

Acknowledgment

Firstly, I want to thank my supervisor Professor Carsten Markgraf and his research group Driverless Mobility. Several people from the group, in addition to him, helped me achieve this work. While working on my research questions, I could always rely on Robert Dollinger, who was a great council in vehicle control topics or Kalman filter development. Julian Stähler was the go-to guy regarding all kinds of technical problems, looking for a solution.

I also want to thank Siemens Mobility GmbH for funding most of this research. Working with the whole team and getting valuable insights concerning industry matters was a pleasure. Primarily Guido Schröer provided essential help in developing simulation systems and enabling me to focus on research matters.

I am very thankful for the microscopic traffic simulation software provided by Aimsun Next. It was essential to achieve the goals of this thesis.

Another significant contributor to my scientific work is the TU Munich. I want to thank Professor Klaus Bogenberger for pointing my research in the right direction and for his invitation to the automated traffic research group. One of the research group leaders, Tanja Niels, supported me with scientific and personal guidance throughout my Ph.D. studies.

Developing automated driving algorithms is challenging. Therefore I appreciate the help from Hatem Darweesh from Nagoya University. His development of the open-source AV planning algorithm enabled several simulator studies and leveraged the significance of the results.

I am very grateful for the love and support of my family. I want to thank my parents and my sister for supporting me in everything I do. I also thank Johannes Gilg, a good friend and advisor in personal and scientific questions. Lastly, there is no way I can thank my girlfriend enough for her personal council and support in keeping me on track in writing this thesis.

List of Figures

List of Tables

List of Terms and Abbreviations

ACC Adaptive Cruise Control 49
ADS Automated Driving System 12, 13, 15, 27, 28, 31, 33, 41, 44–46, 60, 61, 63, 133
AEB Automatic Emergency Braking 12
ASAM Association for Standardization of Automation and Measuring Systems 33
AV Automated Vehicle iii, v, 1–3, 5–7, 11, 16, 17, 21–24, 29–34, 36, 37, 41, 43–45, 51, 54, 55, 58–61, 63, 87, 88, 92–94, 101, 106–109, 111–122, 124, 125, 127, 130, 131, 133

C-ITS Cooperative-ITS 42, 43
CAM Cooperative Awareness Message 37, 43, 44
CAV Connected Automated Vehicle iii, 19, 38, 40, 45, 46, 63, 80–82, 86–90, 92, 97, 99, 101, 103–109, 111, 114, 117, 118, 120–123, 129–131
CNN Convolutional Neural Network 26
CoHIL Co-simulation HIL 16, 54, 90, 93, 97, 121
CP Collective Perception iii, v, 8, 11, 16, 30, 36, 37, 41–46, 62, 97, 100, 101, 120–124
CPM Collective Perception Message 37, 44
CPS Collective Perception Service 37, 44

DDT Dynamic Driving Task 12–15, 127, 133
DESTINy DEcentralized System for Traffic Management 40
DNN Deep Neural Network 25, 26
DRSC Dedicated Short Range Communication 20
DUT Device Under Test 47, 51, 54, 55, 88, 90, 91, 93, 95, 128

ESC Electronic Stability Control 12
ETSI European Telecommunications Standards Insitutute 37

FoV Field of View 4, 5, 22–24, 26, 30, 41, 43, 44, 51, 90, 91, 93, 94, 99, 107, 113–116, 122, 123, 127, 129, 131

GIDAS German In-Depth Accident Study 4
GNSS Global Navigation Satellite System 43, 56, 104

HIL Hardware in the Loop iii, 8, 11, 16, 18–20, 36, 45, 47, 51, 54, 56, 87, 88, 90, 93, 103, 117, 121–123, 127–129

IR infrared 25
ISO International Organization for Standardization 1, 2, 37
ITS Intelligent Transport Systems 16, 37, 39, 40, 42, 44, 45, 92
ITS-C Intelligent Transport Systems Conference 93
ITS-S Intelligent Transport Systems Stations iii, v, 2, 8, 20, 30, 37, 41–46, 48, 62, 63, 65, 82, 85–88, 90, 92, 97, 99, 104–107, 109–111, 117, 118, 120–124, 130, 131
ITSC Intelligent Transport Systems Communications 37
IVICS Intelligent Vehicle Infrastructure Cooperative System 39

KPI Key Performance Indicator 22, 24, 133

LiDAR Light Detection And Ranging 22–26, 56
LKA Lane Keeping Assistance 12
LTC Local Traffic Controller 40

MABX MicroAutoBox 19
MOTA Mutli Object Tracking Accuracy 6, 22, 27, 51, 127
MOTP Mutli Object Tracking Precision 27, 51
MPC Model Predictive Control 34

NDS Navigation Data Standard 32, 33

OBU On Board Unit 19, 37, 39, 43, 56, 88
OD Origin Destination 48, 49, 127
ODD Operational Design Domain 12, 14, 15
OEDR Object and Event Detection and Response 12, 14, 15

P2I Pedestrian to Infrastructure 42
PID Proportional Integral Derivative 61
PPC Pure Pursuit Control 34–36, 61, 127

RADAR Radio Detection And Ranging 22–25
ROS Robot Operation System 54
RQ Research Question 8, 9, 121, 122
RSI Road Side Infrastructure 3, 5, 6, 45, 56, 62, 63, 88, 90, 92, 100, 101, 106–111, 117–121, 123, 124, 130
RSU Road Side Unit 19, 37, 39, 40, 42, 62, 84, 85, 88, 89, 104, 128, 129

SAE Society of Automotive Engineers 2, 11–13, 16, 21, 22, 31, 33, 36
SaFAD Safety First for Automated Driving 2, 17, 21
SIL Software in the Loop 16, 20
SLAM Simultaneous Localization and Mapping 25
SoTIF Safety of The Intended Functionality 2, 16, 17
SPaT Signal Phase and Timing 19
SUMO Simulation of Urban MObility 19, 39, 40
SZ Sueddeutsche Zeitung 69, 70, 72–79

TIS Technology Independent Sensor 58
ToF Time of Flight 26
TraCI Traffic Control Interfrace 40
TRADER TRaffic Light Phases Aware Driving for REduced tRaffic Congestion 40

UAS University of Applied Sciences 56
UN United Nations 1
UWB Ultra Wide Band 43

V2I Vehicle to Infrastructure 37, 38, 40, 41, 44, 84, 85, 129, 134
V2P Vehicle to Pedestrian 43
V2V Vehicle to Vehcile 19, 40, 123
V2VRU Vehicle to VRU 42
V2X Vehicle to Everything 37, 39, 40, 42, 43, 52, 53
V&V Verification and Validation 2, 16–18, 21, 133
VeHIL Vehicle Hardware in the Loop 17, 87, 103, 121, 130
VOSL Value Of Statistical Live 7, 83
VRU Vulnerable Road User iii, v, 3, 5, 11, 41–45, 73, 82, 99, 111, 112, 114, 116, 118, 122, 131, 137
VUT Vehicle Under Test 48, 51, 54, 56, 58, 60, 87, 88, 90, 98–100, 103, 104, 106–110, 112, 113, 115–118, 120, 130, 131

YOLO You Only Look Once 26

Publications

This thesis is based on the publications provided below:

Publication	Description	Section
Mathias, Pechinger; Guido Schroer; Klaus Bogenberger; Carsten Markgraf (10/19/2020 - 11/13/2020). "Hardware in the Loop Test Using Infrastructure Based Emergency Trajectories for Connected Automated Driving". In: 2020 IEEE Intelligent Vehicles Symposium (IV). IEEE, pp. 357–362. isbn: 978-1-7281-6673-5. doi: 10 . 1109 / IV47402.2020.9304757.	**HIL Simulation**, road side support for automated vehicles	Section 5.1, Section 6.1
Mathias, Pechinger; Guido Schroer; Klaus Bogenberger; Carsten Markgraf (7/11/2021 - 7/17/2021). "Benefit of Smart Infrastructure on Urban Automated Driving - Using an AV Testing Framework". In: 2021 IEEE Intelligent Vehicles Symposium (IV). IEEE, pp. 1174–1179. isbn: 978-1-7281-5394-0. doi: 10.1109/IV48863.2021.9575651.	**HIL Simulation**, benefit of road side ITS-S on traffic efficiency	Section 5.2, Section 6.2
Mathias, Pechinger; Guido Schroer; Klaus Bogenberger; Carsten Markgraf (9/19/2021 - 9/22/2021). "Cyclist Safety in Urban Automated Driving - Sub-Microscopic HIL Simulation". In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC). IEEE, pp. 615–620. isbn: 978-1-7281-9142-3. doi: 10.1109/ITSC48978.2021.9565108.	**HIL Simulation**, Cyclist safety analysis inside an urban intersection	Section 5.3, Section 6.3
Mathias, Pechinger;; Guido Schroer; Klaus Bogenberger; Carsten Markgraf (9/19/2021 - 9/22/2021). "Roadside Infrastructure Support for Urban Automated Driving". In: 2023 IEEE Transactions on Intelligent Transportation Systems (T-ITS, Volume: 24, Issue: 10, October 2023), pp. 10643 - 10652. doi: 10.1109/TITS.2023.3277138.	**HIL Simulation**, Utilizing Collective Perception to increase the FoV of a CAV	Section 5.4, Section 6.4

Bibliography

Aimsun Next (2023). *Modelling Vehicle Movement: Car-Following Model*. URL: https://docs.aimsun.com/next/22.0.1/UsersManual/MicrosimulationModellingVehicleMovement.html#lane_changing_model (visited on 2023).

ASAM e.V. (2021). *ASAM OpenDRIVE*. Ed. by ASAM e.V. URL: https://www.asam.net/standards/detail/opendrive/ (visited on 2023).

Berg, Mark de; Marc van Kreveld; Mark Overmars; Otfried Schwarzkopf (1997). "Voronoi Diagrams". In: *Computational Geometry*. Ed. by Mark de Berg; Marc van Kreveld; Mark Overmars; Otfried Schwarzkopf. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 145–161. ISBN: 978-3-662-03429-3. DOI: 10.1007/978-3-662-03427-9{\textunderscore}7.

Bernardin, Keni; Rainer Stiefelhagen (2008). "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics". In: *EURASIP Journal on Image and Video Processing* 2008, pp. 1–10. ISSN: 1687-5176. DOI: 10.1155/2008/246309.

Betts, John T. (1998). "Survey of Numerical Methods for Trajectory Optimization". In: *Journal of Guidance, Control, and Dynamics* 21.2, pp. 193–207. ISSN: 0731-5090. DOI: 10.2514/2.4231.

Blaeij, Arianne de; Raymond J.G.M Florax; Piet Rietveld; Erik Verhoef (2003). "The value of statistical life in road safety: a meta-analysis". In: *Accident Analysis & Prevention* 35.6, pp. 973–986. ISSN: 00014575. DOI: 10.1016/S0001-4575(02)00105-7.

Buch, Thomas Skallebæk; Søren Underlien Jensen (2017). "Incidents between Straight-ahead Cyclists and Right-turning Motor Vehicles at Signalised Junctions". In: *Accident; analysis and prevention* 105, pp. 44–51. DOI: 10.1016/j.aap.2016.07.035.

Cantas, Mustafa Ridvan; Ozgenur Kavas; Santhosh Tamilarasan; Sukru Yaren Gelbal; Levent Guvenc (2019). "Use of Hardware in the Loop (HIL) Simulation for Developing Connected Autonomous Vehicle (CAV) Applications". In: *SAE Technical Paper Series*. SAE Technical Paper Series. SAE International400 Commonwealth Drive, Warrendale, PA, United States. DOI: 10.4271/2019-01-1063.

CARLA: An Open Urban Driving Simulator (2017).

Casademont, Jordi; Anna Calveras; David Quinones; Monica Navarro; Javier Arribas; Miguel Catalan-Cid (2019). "Cooperative-Intelligent Transport Systems for Vulnerable Road Users Safety". In: *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*. IEEE, pp. 141–146. ISBN: 978-1-7281-2888-7. DOI: 10.1109/FiCloud.2019.00027.

Committe on the Rights of Persons with Disabilities (2021). *Convention on the Rights of Persons with Disabilities*. Online. (Visited on 2022).

Dandl, Florian; Gabriel Tilg; Majid Rostami-Shahrbabaki; Klaus Bogenberger (2020). "Network Fundamental Diagram Based Routing of Vehicle Fleets in Dy-

namic Traffic Simulations". In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1–8. ISBN: 978-1-7281-4149-7. DOI: 10.1109/ITSC45102.2020.9294204.

Darweesh, Hatem; Eijiro Takeuchi; Kazuya Takeda; el al (2017). "Open Source Integrated Planner for Autonomous Navigation in Highly Dynamic Environments". In: *Journal of Robotics and Mechatronics* 29.4, pp. 668–684. ISSN: 0915-3942. DOI: 10.20965/jrm.2017.p0668.

Dasanayaka, Nishanthi; Khondokar Fida Hasan; Charles Wang; Yanming Feng (2020). *Enhancing Vulnerable Road User Safety: A Survey of Existing Practices and Consideration for Using Mobile Devices for V2X Connections*. URL: https://arxiv.org/pdf/2010.15502.

Denk, Florian; Pascal Brunner; Werner Huber; Martin Margreiter; Klaus Bogenberger; Ronald Kates (2022). "Assessment of traffic safety interventions using virtual randomized controlled trials: potential of connected and automated driving including V2X for collision reduction at urban intersections". In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1183–1190. ISBN: 978-1-6654-6880-0. DOI: 10.1109/ITSC55140.2022.9921764.

Deutsches Zentrum für Luft- und Raumfahrt (2001). *Simulation of Urban MObility*.

Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". In: *Numerische Mathematik* 1.1, pp. 269–271. ISSN: 0029-599X. DOI: 10.1007/BF01386390.

Dixit, Vinayak V.; Sai Chand; Divya J. Nair (2016). "Autonomous Vehicles: Disengagements, Accidents and Reaction Times". In: *PloS one* 11.12, e0168054. DOI: 10.1371/journal.pone.0168054.

Dong, Changqing; Xiaowei Chen; Haibo Dong; Kaixin Yang; Jinwei Guo; Yunlong Bai (2019). "Research on Intelligent Vehicle Infrastructure Cooperative System Based on Zigbee". In: *ICTIS 2019*. [Piscataway, NJ]: IEEE, pp. 1337–1343. ISBN: 978-1-7281-0489-8. DOI: 10.1109/ICTIS.2019.8883704.

European Telecommunications Standards Institute (2010). *Intelligent Transport Systems (ITS); Communications Architecture*.

European Telecommunications Standards Institute (2019a). *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS);*

European Telecommunications Standards Institute (2019b). *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*.

Federal Office of Statistics Germany (2021). *Bis 2035 wird die Zahl der Menschen ab 67 Jahre um 22 % steigen: Press Release Number 459*.

Gabb, Michael; Holger Digel; Tobias Muller; Rudiger-Walter Henn (2019). "Infrastructure-supported Perception and Track-level Fusion using Edge Computing". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 1739–1745. ISBN: 978-1-7281-0560-4. DOI: 10.1109/IVS.2019.8813886.

Gelbal, Sukru Yaren; Santhosh Tamilarasan; Mustafa Ridvan Cantas; Levent Guvenc; Bilin Aksun-Guvenc (2017). "A connected and autonomous vehicle

hardware-in-the-loop simulator for developing automated driving algorithms". In: *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 3397–3402. ISBN: 978-1-5386-1645-1. DOI: 10.1109/SMC.2017.8123155.

GERSTENBERGER, MARCUS (2015). "Unfallgeschehen an Knotenpunkten". Dissertation. München: Technische Universität München.

GIANNAKIS, ELIAS; DESPINA SERGHIDES; STELLA DIMITRIOU; GEORGE ZITTIS (2020). "Land transport CO 2 emissions and climate change: evidence from Cyprus". In: *International Journal of Sustainable Energy* 39.7, pp. 634–647. ISSN: 1478-6451. DOI: 10.1080/14786451.2020.1743704.

GIPPS, P. G. (1981). "A behavioural car-following model for computer simulation". In: *Transportation Research Part B: Methodological* 15.2, pp. 105–111. ISSN: 01912615. DOI: 10.1016/0191-2615(81)90037-0.

GIPPS, P. G. (1986). "A model for the structure of lane-changing decisions". In: *Transportation Research Part B: Methodological* 20.5, pp. 403–414. ISSN: 01912615. DOI: 10.1016/0191-2615(86)90012-3.

GOLDBERG, ANDREW V. AND HARRELSON, CHRIS, ed. (2005). *Computing the Shortest Path: A Search Meets Graph Theory*. ISBN: 0898715857.

GOOGLE (2021). *GoogleMaps*. URL: https://www.google.com/maps/@48.1486597,11.5692527,137a,35y,355.06h,40.27t/data=!3m1!1e3.

HANNAH RITCHIE AND MAX ROSER (2017). "The past and future of the global age structure". In: URL: https://ourworldindata.org/world-population-growth.

HUBERTUS, PHILIP (2019). *The Benefits of a Common Map Data Standard for Autonomous Driving*. Ed. by NAVIGATION DATA STANDARD E. V. URL: https://nds-association.org/wp-content/uploads/2019/06/NDS-White-Paper__Benefits-of-Map-Data-Standard-for-Autonomous-Driving.pdf (visited on 2023).

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (2018). *Road vehicles – Functional safety*.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (2019). *Road vehicles — Safety of the intended functionality*.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (2020). *Intelligent transport systems — Station and communication architecture*.

J. MA; F. ZHOU; Z. HUANG; R. JAMES (2018). "Hardware-In-The-Loop Testing of Connected and Automated Vehicle Applications: A Use Case For Cooperative Adaptive Cruise Control". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. DOI: 10.1109/ITSC.2018.8569753.

KATO, SHINPEI; EIJIRO TAKEUCHI; YOSHIO ISHIGURO; YOSHIKI NINOMIYA; KAZUYA TAKEDA; TSUYOSHI HAMADA (2015). "An Open Approach to Autonomous Vehicles". In: *IEEE Micro* 35.6, pp. 60–68. ISSN: 0272-1732. DOI: 10.1109/MM.2015.133.

KATO, SHINPEI; SHOTA TOKUNAGA; YUYA MARUYAMA; SEIYA MAEDA; ET AL (2018). "Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems". In: *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, pp. 287–296. ISBN: 978-1-5386-5301-2. DOI: 10.1109/ICCPS.2018.00035.

KONG, JASON; MARK PFEIFFER; GEORG SCHILDBACH; FRANCESCO BORRELLI (2015). "Kinematic and dynamic vehicle models for autonomous driving control design". In: *2015*

IEEE Intelligent Vehicles Symposium (IV). IEEE, pp. 1094–1099. ISBN: 978-1-4673-7266-4. DOI: 10.1109/IVS.2015.7225830.

Krämmer, Annkathrin; Christoph Schöller; Dhiraj Gulati; Venkatnarayanan Lakshminarasimhan; Franz Kurz; Dominik Rosenbaum; Claus Lenz; Alois Knoll (2019). *Providentia – A Large-Scale Sensor System for the Assistance of Autonomous Vehicles and Its Evaluation*. URL: http://arxiv.org/pdf/1906.06789v5.

Krausse, Markus; Rainer Konrad (2014). "IEEE 802.15.4". In: *Drahtlose ZigBee-Netzwerke*. Ed. by Markus Krausse; Rainer Konrad. Wiesbaden: Springer Fachmedien Wiesbaden, pp. 83–90. ISBN: 978-3-658-05820-3. DOI: 10.1007/978-3-658-05821-0{\textunderscore}8.

Lang, Alex H.; Sourabh Vora; Holger Caesar; et al (2019). "PointPillars: Fast Encoders for Object Detection From Point Clouds". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 12689–12697. ISBN: 978-1-7281-3293-8. DOI: 10.1109/CVPR.2019.01298.

Laux, Sven; Gurjashan Singh Pannu; Stefan Schneider; Jan Tiemann; Florian Klingler; Christoph Sommer; Falko Dressler (2016). "Demo: OpenC2X — An open source experimental and prototyping platform supporting ETSI ITS-G5". In: *2016 IEEE Vehicular Networking Conference (VNC)*. IEEE, pp. 1–2. ISBN: 978-1-5090-5197-7. DOI: 10.1109/VNC.2016.7835955.

LaValle, Steven M.; James J. Kuffner (2001). "Randomized Kinodynamic Planning". In: *The International Journal of Robotics Research* 20.5, pp. 378–400. ISSN: 0278-3649. DOI: 10.1177/02783640122067453.

Linus, Freymark; Hutter Dominik (2018). "Das sind die gefährlichsten Stellen im Münchner Stadtverkehr". In: *Sueddeutsche Zeitung* 2018. URL: https://www.sueddeutsche.de/muenchen/unfaelle-unfallatlas-gefaehrliche-strassen-1.4196040.

Lissac, Alexandre; Soufiene Djahel; Jack Hodgkiss (2019). "Infrastructure Assisted Automation of Lane Change Manoeuvre for Connected and Autonomous Vehicles". In: *2019 IEEE International Smart Cities Conference (ISC2)*. IEEE, pp. 173–180. ISBN: 978-1-7281-0846-9. DOI: 10.1109/ISC246665.2019.9071743.

Liu, Weijie; Shintaro Muramatsu; Yoshiyuki Okubo (2018). "Cooperation of V2I/P2I Communication and Roadside Radar Perception for the Safety of Vulnerable Road Users". In: *2018 16th International Conference on Intelligent Transportation Systems Telecommunications (ITST)*. IEEE, pp. 1–7. ISBN: 978-1-5386-5544-3. DOI: 10.1109/ITST.2018.8566704.

Lourenco, Massilon; Fernanda S.H Souza; Cristiano M. Silva; Rodolfo I. Meneguette; Daniel L. Guidoni (2019). "A Hybrid V2I and V2V Approach for Urban Traffic Management in Vehicular Networks". In: *2019 IEEE Latin-American Conference on Communications (LATINCOM)*. Ed. by Carlos A. Gutiérrez. [Piscataway, New Jersey]: IEEE, pp. 1–6. ISBN: 978-1-7281-3955-5. DOI: 10.1109/LATINCOM48065.2019.8937924.

Lu, Xiao-Yun; Han-Shue Tan; Steven Shladover; J. Hedrick (2000). "Implementation of Longitudinal Control Algorithm for Vehicle Merging". In.

Ma, Changxi; Wei Hao; Aobo Wang; Hongxing Zhao (2018). "Developing a Coordinated Signal Control System for Urban Ring Road Under the Vehicle-Infrastructure

Connected Environment". In: *IEEE Access* 6, pp. 52471–52478. DOI: 10.1109/ACCESS.2018.2869890.

MA, JIAQI; FANG ZHOU; ZHITONG HUANG; CHRISTOPHER L. MELSON; RACHEL JAMES; XIAOXIAO ZHANG (2018). "Hardware-in-the-Loop Testing of Connected and Automated Vehicle Applications: A Use Case for Queue-Aware Signalized Intersection Approach and Departure". In: *Transportation Research Record: Journal of the Transportation Research Board* 2672.22, pp. 36–46. ISSN: 0361-1981. DOI: 10.1177/0361198118793001.

MALIK, R. Q.; H. A. ALSATTAR; K. N. RAMLI; B. B. ZAIDAN; A. A. ZAIDAN; Z. H. KAREEM; H. A. AMEEN; SALEM GARFAN; ALI MOHAMMED; R. A. ZAIDAN (2019). "Mapping and Deep Analysis of Vehicle-to-Infrastructure Communication Systems: Coherent Taxonomy, Datasets, Evaluation and Performance Measurements, Motivations, Open Challenges, Recommendations, and Methodological Aspects: IEEE Access, 7, 126753-126772". In: *IEEE Access* 7, pp. 126753–126772. DOI: 10.1109/ACCESS.2019.2927611.

MARC DESPONTIN; ALAIN VERBEKE; K. DE BRUCKER (1997). "The economic evaluation of road safety in the European Union". In: *Proceedings of the European Seminar on Cost-Effectiveness of Road Safety Work and Measures, Luxembourg, November 1997, pp. 13-37*. URL: https://researchportal.vub.be/en/publications/the-economic-evaluation-of-road-safety-in-the-european-union.

MARKGRAF, CARSTEN; MATHIAS PECHINGER; FLORIAN POPRAWA; GUIDO SCHRÖER (2022). "INFRASTRUCTURE-SUPPORTED PROCESS OF ASCERTAINING TRAJECTORIES FOR AUTONOMOUS VEHICLES". WO2022122250 (A1).

MILANES, VICENTE; JORGE VILLAGRA; JORGE GODOY; JAVIER SIMO; JOSHUÉ PEREZ; ENRIQUE ONIEVA (2012). "An Intelligent V2I-Based Traffic Management System". In: *IEEE Transactions on Intelligent Transportation Systems* 13.1, pp. 49–58. ISSN: 1524-9050. DOI: 10.1109/TITS.2011.2178839.

MIURA, KEITA; SHOTA TOKUNAGA; NORIYUKI OTA; YOSHIHARU TANGE; TAKUYA AZUMI (2019). "Autoware Toolbox". In: *Proceedings of the 30th International Workshop on Rapid System Prototyping (RSP'19) - RSP '19*. Ed. by UNKNOWN. New York, New York, USA: ACM Press, pp. 8–14. ISBN: 9781450368476. DOI: 10.1145/3339985.3358494.

NAPOLITANO, A.; G. CECCHETTI; F. GIANNONE; A. L. RUSCELLI; F. CIVERCHIA; K. KONDEPU; L. VALCARENGHI; P. CASTOLDI (2019). "Implementation of a MEC-based Vulnerable Road User Warning System". In: *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*. IEEE, pp. 1–6. ISBN: 978-8-8872-3743-6. DOI: 10.23919/EETA.2019.8804497.

NARKSRI, PATIPHON; EIJIRO TAKEUCHI; YOSHIKI NINOMIYA; KAZUYA TAKEDA (2019). "Crossing Blind Intersections from a Full Stop Using Estimated Visibility of Approaching Vehicles". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, pp. 2427–2434. ISBN: 978-1-5386-7024-8. DOI: 10.1109/ITSC.2019.8917323.

NILSSON, NILS J. (1969). *A Mobile Automaton: An Application of Artificial Intelligence Techniques*. Fort Belvoir, VA. DOI: 10.21236/ada459660.

OPEN SOURCE ROBOTICS FOUNDATION (2018). *Robot Operating System*. URL: www.ros.org.

ORZECHOWSKI, PIOTR F.; ANNIKA MEYER; MARTIN LAUER (2018). "Tackling Occlusions & Limited Sensor Range with Set-based Safety Verification". In: *2018 21st Inter-*

national Conference on Intelligent Transportation Systems (ITSC). IEEE, pp. 1729–1736. ISBN: 978-1-7281-0321-1. DOI: 10.1109/ITSC.2018.8569332.

PADEN, BRIAN; MICHAL CAP; SZE ZHENG YONG; DMITRY YERSHOV; EMILIO FRAZZOLI (2016). "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles". In: *IEEE Transactions on Intelligent Vehicles* 1.1, pp. 33–55. ISSN: 2379-8858. DOI: 10.1109/TIV.2016.2578706.

PECHINGER, MATHIAS; GUIDO SCHROER; KLAUS BOGENBERGER; CARSTEN MARKGRAF (2020). "Hardware in the Loop Test Using Infrastructure Based Emergency Trajectories for Connected Automated Driving". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 357–362. ISBN: 978-1-7281-6673-5. DOI: 10.1109/IV47402.2020.9304757.

PECHINGER, MATHIAS; GUIDO SCHROER; KLAUS BOGENBERGER; CARSTEN MARKGRAF (2021a). "Benefit of Smart Infrastructure on Urban Automated Driving - Using an AV Testing Framework". In: *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 1174–1179. ISBN: 978-1-7281-5394-0. DOI: 10.1109/IV48863.2021.9575651.

PECHINGER, MATHIAS; GUIDO SCHROER; KLAUS BOGENBERGER; CARSTEN MARKGRAF (2021b). "Cyclist Safety in Urban Automated Driving - Sub-Microscopic HIL Simulation". In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, pp. 615–620. ISBN: 978-1-7281-9142-3. DOI: 10.1109/ITSC48978.2021.9565108.

PEREIRA, ANDRE MAIA; HOSSAM ANANY; ONDREJ PRIBYL; JAN PRIKRYL (2017). "Automated vehicles in smart urban environment: A review". In: *2017 Smart Cities Symposium Prague (SCSP)*. Ed. by JIRI RUZICKA. Piscataway, NJ: IEEE, pp. 1–8. ISBN: 978-1-5386-3825-5. DOI: 10.1109/SCSP.2017.7973864.

PILZ, CHRISTOPH; GERALD STEINBAUER; MARKUS SCHRATTER; DANIEL WATZENIG (2019). "Development of a Scenario Simulation Platform to Support Autonomous Driving Verification". In: *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, pp. 1–7. ISBN: 978-1-7281-0142-2. DOI: 10.1109/ICCVE45908.2019.8964914.

POGGENHANS, FABIAN; JAN-HENDRIK PAULS; JOHANNES JANOSOVITS; STEFAN ORF; MAXIMILIAN NAUMANN; FLORIAN KUHNT; MATTHIAS MAYR (2018). "Lanelet2: A high-definition map framework for the future of automated driving". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1672–1679. ISBN: 978-1-7281-0321-1. DOI: 10.1109/ITSC.2018.8569929.

R. CRAID COULTER (1992). *Implementation of the Pure Pursuit Path Tracking Algorithm*. January 1992.

R. PÉREZ; F. SCHUBERT; R. RASSHOFER; E. BIEBL (2019). "Deep Learning Radar Object Detection and Classification for Urban Automotive Scenarios". In: *2019 Kleinheubach Conference*, pp. 1–4.

RAJ, THINAL; FAZIDA HANIM HASHIM; AQILAH BASERI HUDDIN; MOHD FAISAL IBRAHIM; AINI HUSSAIN (2020). "A Survey on LiDAR Scanning Mechanisms". In: *Electronics* 9.5, p. 741. DOI: 10.3390/electronics9050741.

REDMON, JOSEPH; SANTOSH DIVVALA; ROSS GIRSHICK; ALI FARHADI (2016). "You Only Look Once: Unified, Real-Time Object Detection". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 779–788. ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.91.

Rhodes, Cullen; Soufiene Djahel (2017). "TRADER:Traffic Light Phases Aware Driving for Reduced Traffic Congestion in Smart Cities". In: *2017 International Smart Cities Conference (ISC2) proceedings*. Piscataway, NJ: IEEE, pp. 1–8. ISBN: 978-1-5386-2524-8. DOI: 10.1109/ISC2.2017.8090783.

Richard S. Wallace; Anthony Stentz; Charles E. Thorpe; Hans P. Moravec; William Whittaker; Takeo Kanade (1985). "First Results in Robot Road-Following". In: *Proceedings of the 9th International Joint Conference on Artificial Intelligence. Los Angeles, CA, USA, August 1985*. Ed. by Aravind K. Joshi. Morgan Kaufmann, pp. 1089–1095. URL: http://ijcai.org/Proceedings/85-2/Papers/086.pdf.

Robert Bosch GmbH (2019). *MEC-View*. URL: http://www.mec-view.de/.

Rosique, Francisca; Pedro J. Navarro; Carlos Fernández; Antonio Padilla (2019). "A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research". In: *Sensors (Basel, Switzerland)* 19.3. DOI: 10.3390/s19030648.

Ruskin, H. J.; R. Wang (2002). "Modeling Traffic Flow at an Urban Unsignalized Intersection". In: *Computational Science — ICCS 2002*. Ed. by Gerhard Goos; Juris Hartmanis; Jan van Leeuwen; Peter M. A. Sloot; Alfons G. Hoekstra; C. J. Kenneth Tan; Jack J. Dongarra. Vol. 2329. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 381–390. ISBN: 978-3-540-43591-4. DOI: 10.1007/3-540-46043-8{\textunderscore}38.

SAE International (2014). *Levels of Driving Automation*.

SAE International (2021). *SURFACE VEHICLE RECOMMENDED PRACTICE: (R) Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*.

Seif, Heiko G.; Xiaolong Hu (2016). "Autonomous Driving in the iCity—HD Maps as a Key Challenge of the Automotive Industry". In: *Engineering* 2.2, pp. 159–162. ISSN: 20958099. DOI: 10.1016/J.ENG.2016.02.010.

Siemens Digital Industries Software (2020a). *Aimsun Next 20*. URL: www.aimsun.com.

Siemens Digital Industries Software (2020b). *Simcenter Prescan*. URL: https://www.plm.automation.siemens.com/global/de/products/simcenter/prescan.html.

Sinha, Amolika; Sai Chand; Vincent Vu; Huang Chen; Vinayak Dixit (2021). "Crash and disengagement data of autonomous vehicles on public roads in California". In: *Scientific data* 8.1, p. 298. DOI: 10.1038/s41597-021-01083-7.

Sistu, Ganesh; Isabelle Leang; Senthil Yogamani (2019). *Real-time Joint Object Detection and Semantic Segmentation Network for Automated Driving*. URL: http://arxiv.org/pdf/1901.03912v1.

Sören Kohnert; Julian Stähler; Reinhard Stolle; Florian Geissler, eds. (2021). *Cooperative RADAR Sensors for the Digital Test Field A9 (KoRA9) – Algorithmic Recap and Lessons Learned*. ISBN: 10.23919/IEEECONF54431.2021.9598409. (Visited on 2022).

Statista Research Department (2022). *Verteilung der CO2-Emissionen weltweit nach Sektor bis 2021*. URL: https://de.statista.com/statistik/daten/studie/167957/umfrage/verteilung-der-co-emissionen-weltweit-nach-bereich/.

Staubach, Maria (2010). "Identifikation menschlicher Einflüsse auf Verkehrsunfälle als Grundlage zur Beurteilung von Fahrerassistenzsystem-Potentialen". Dresden, Techn. Univ., Diss., 2010.

Sun, Pei et al. (n.d.). "Scalability in Perception for Autonomous Driving: Waymo Open Dataset". In: *2020 IEEE/CVF Conference on Computer 6/13/2020*, pp. 2443–2451. doi: 10.1109/CVPR42600.2020.00252.

Tas, Omer Sahin; Christoph Stiller (2018). "Limited Visibility and Uncertainty Aware Motion Planning for Automated Driving". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 1171–1178. isbn: 978-1-5386-4452-2. doi: 10.1109/IVS.2018.8500369.

Tihanyi, Viktor; András Rövid; Viktor Remeli; Zsolt Vincze; Mihály Csonthó; Zsombor Pethő; Mátyás Szalai; Balázs Varga; Aws Khalil; Zsolt Szalay (2021). "Towards Cooperative Perception Services for ITS: Digital Twin in the Automotive Edge Cloud". In: *Energies* 14.18, p. 5930. doi: 10.3390/en14185930.

Tom Nokes, Ben Baxter, Harry Scammell, Denis Naberezhnykh, Leonardo Provvedi (2020). *Cost Analysis of V2I Deployment: Final Report: Report for 5GAA*. Ed. by Ricardo Energy & Environment. url: https://5gaa.org/wp-content/uploads/2020/09/5GAA_Ricardo-Study-V2I-Cost-Analysis_Final_110820.pdf.

Tong, Kailin; Zlatan Ajanovic; Georg Stettinger (2020). "Overview of Tools Supporting Planning for Automated Driving". In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1–8. isbn: 978-1-7281-4149-7. doi: 10.1109/ITSC45102.2020.9294512.

Treiber, Martin; Arne Kesting (2010). *Verkehrsdynamik und -simulation: Daten, Modelle und Anwendungen der Verkehrsflussdynamik*. Springer-Lehrbuch. Berlin and Heidelberg: Springer. isbn: 978-3-642-05227-9.

Tsukada, Manabu; Takaharu Oi; Masahiro Kitazawa; Hiroshi Esaki (2020). "Networked Roadside Perception Units for Autonomous Driving". In: *Sensors (Basel, Switzerland)* 20.18. doi: 10.3390/s20185320.

van Essen, Huib; Lisanne van Wijngaarden; Arno Schroten; Daniel Sutter; Cuno Bieler; Silvia Maffii; Marco Brambilla; Davide Fiorello; Francesca Fermi; Riccardo Parolin; Kareen El Beyrouty (2019). *Handbook on the external costs of transport*. Version 2019. Luxembourg: Publications Office of the European Union. isbn: 978-92-79-96917-1.

Venon, Arthur; Yohan Dupuis; Pascal Vasseur; Pierre Merriaux (2022). "Millimeter Wave FMCW RADARs for Perception, Recognition and Localization in Automotive Applications: A Survey". In: *IEEE Transactions on Intelligent Vehicles* 7.3, pp. 533–555. issn: 2379-8858. doi: 10.1109/TIV.2022.3167733.

Wood et al. (2019). *Safety first for Automated Drving*. url: https://www.daimler.com/innovation/case/autonomous/safety-first-for-automated-driving.html (visited on 2021).

Yann Macron (2022). *Distance-based clustering of a set of XY coordinates*. url: https://www.mathworks.com/matlabcentral/fileexchange/56150-distance-based-clustering-of-a-set-of-xy-coordinates.

Annex

1 Vehicle Dynamics Model Parameters

Table 1: Overview of the model parameters of the vehicle dynamics model

Type	Parameter	Description	Unit
Miscellaneous	P brake max	Maximum brake pressure	[bar]
	Cw	Air resistance	[-]
Inertias	M	Mass of the vehicle	[kg]
	Jzz/Jyy/Jxx	Moment of Inertia around the z/y/x-axis	[kg m2]
Tire Stiffness	K tires, front/rear	Cornering stiffness of the tires	[N/rad]
Suspensions	C susp. front/rear	Stiffness front/rear suspension	[N/m]
	D susp. front/rear	Damping rate front/rear suspension	[Ns/m]
BB Dimensions	Track Width	Track width of the vehicle	[m]
	Ground clearance	distance from road to vehicle	[m]
	BB Width	Bounding Box Width	[m]
	BB Length	Bounding Box Length	[m]
	BB Height	Bounding Box Height	[m]
	Offset	Vehicle rear offset	[m]
CoG Location	h CoG	Height of the center of gravity	[m]
	b	Distance of the CoG to the rear	[m]
	a	Distance of the CoG to the front axle	[m]
Bumpers	Bumper f/r	Ground distance of the front/rear bumper	[m]
Wheels parameters	Rw	Tire radius	[m]
	l	Wheel base	[m]

2 OpenPlanner Configuration Parameters

This Annex describes several parameters available in OpenPlanner, divided by different subsections.

Common Parameters

Group	Parameter	Description
Behavior	enableStop SignBehavior	Enables/disbales the behavior planners response to a stop sign
	enableTraffic LightBehavior	Enables/disables the behavior planners response to a traffic light
	enableFollowing	Enables/disables the behavior planners response to a leading vehicle ahead
	enableLaneChange	Enables the vehicles capabilities to change its lane to a parallel lane in the same direction (must be properly setup in the HD Map)
	enableSwerving	Enables the generation of multiple rollouts/path in parallel to the center line of our current path.
	goalDiscovery Distance	Distance to the goal point until the finish state is triggered.
Global Planning	smoothingData Weight	Map correction (global and local trajecotry) 0 to 0.5 high = ignore data smooth as much as possible
	smoothingSmooth Weight	Higher value inceases smoothing of the path 0 to 0.5
Local Planning	horizonDistance	This value conducts a circular area filter around the ego vehicle. Every object outside this distance in neglected
	minFollowDistance	If a vehicle is on our path and closer than the set minimum distance to follow, the vehcile will switch to the Follow state
	maxLocalPlan Distance	The maximum distance the local path is planned along the global path
	speedProfileFactor	Curvature response to the calculation of the target velocity. (Low means slow, 1.0 means average, 2.0 means double)
	horizontalSafety Distance	An additional safety distance on top of the vehicles defined width
	verticalSafetyDist.	An additionl safety distance on top of the vehilces length defined in length
	maxAcceleration	Targeted maximum Accleration of the vehicle
	minAccleration	Targeted maximum Deceleration of the vehicle
	maxVelocity	Limits the maximum velocity of the vehicle
	minVelocity	Limits the minimum velocity (Should not be 0.0 due to an issue in the motion predictor)

Group	Parameter	Description
Local Planning	giveUpDistance	The vehicle will give up to stop state if we have passed the giveUpDistance (e.g. "-1" keep driving if the vehicle passed the stopline by one meter in order to avoid blocking a singalized intersection half way during the turn maneuvre)
	additionalBraking Distance	additional distance to stop for the braking calculation
Obstacle Avoidance	minDistanceToAvoid	The minimum distance to start avoiding obstacles (if swerving enabled)
	maxDistanceToAvoid	The maximum distance to stop avoiding obstacles (if swerving enabled)
Control	accelerationPushRatio	Push the velocity to increase even when the velocity feedback signal from sensing is very small or delayed, contribute to final acceleration. Where 1 predefined acceleration value is used. Acts like a feed forward controller
	brakingPushRatio	Same as accelerationPushRatio except for braking
	control_frequency	The control frequency sets the frequency at which the whole system calculates a new local path
	minPursuiteDistance	Minimum distance to forward following target point on the trajectory for the purepursuit controller
	use_internal_acc	Use the internal acc controller developed for OpenPlanner
Vehicle	width	Width of the vehicle
	length	Length of the vehicle
	height	Height of the vehicle
	front_length	Length from the front axle to the front bumper
	back_length	Length from the rear axle to the rear bumper
	wheelBaseLength	Length of the wheel base
	truningRadius	Minimum turning radius of the vehicle
	maxWheelAngle	Maximum wheel angle
	steeringDelay	The time, the steering wheel takes to turn from zero to max left or right

Motion Predictor

Group	Parameter	Description
Motion Predictor	minDistanceToAvoid	The minimum distance to start avoiding obstacles (if swerving enabled)
	max_distance_to_lane	Maximum distance an object can be away from a lane until it gets neglected by the motion predictor
	min_prediction_distance	Minimum distance an object is predict along its assigned lane
	min_prediction_time	Minimum time an object is predict along its assigned lane
	enableGenrateBranches	Generates branch trajectories for detected objects which are assigned to a lane
	enableCurbObstacles	include curba into obstacle detection
	distanceBetweenCurbs	define the distance between curb obstacle
	enableStepByStepSignal	Used for timed simulations. Every time a new object is received, the motion predictor advances one step.
	particles_number	Number of particles used for tracking
	min_particles_num	Minimum number of Particles
Weights	pose_weight_factor	Weight of the position of the detected object for the motion prediction
	dir_weight_factor	Weight of the direction/heading of the detected object for the motion prediction
	vel_weight_factor	Weight of the velocity of the detected object for the motion prediction
	acc_weight_factor	Weight of the acceleration of the detected object for the motion prediction
	ind_weight_factor	Weight of the indicator light of the detected object for the motion prediction

Trajectory Evaluator

In the trajectory evaluator, the decision on which roll out to choose is made.

Group	Parameter	Description
Trajectory Evaluator	enablePrediction	Enable the prediction of detected objects
	collision_time	Based on the prediction of detected object a collision time is evaluated. If the collision time of an predicted object is below the "collision_time" the path in front is set to blocked and a collision point is published in rviz.

Trajectory Generator

This parameters are used to generate the roll out trajectories for more details on the roll outs themselves refer to [Darweesh et al., 2017].

Group	Parameter	Description
Trajectory Generator	samplingTipMargin	The distance between the start of the rolled out trajectory and the actual expansion from center line trajectory.
	samplingOutMargin	The distance from the end of the tip of the rolled out trajectory until the maximum expansion from the centerline is achieved.
	startFromFrontAxel	Start the roll out from the front axle
	rollOutDensity	Spacing between the roll outs
	rollOutsNumber	Number of roll outs generated