Platooning on a string of intersections

A functional combination of platooning and traffic-adaptive intersection control

Kai Timon Busse

Technische Universiteit Delft

Delft University of Technology

Challenge the future

PLATOONING ON A STRING OF INTERSECTIONS

A FUNCTIONAL COMBINATION OF PLATOONING AND TRAFFIC-ADAPTIVE INTERSECTION CONTROL

by

Kai Timon Busse

in fulfillment of the requirements for the degree of

Master of Science
in Systems Engineering, Policy Analysis and Management

at the Delft University of Technology,
to be defended publicly on 23.08.2017 at 10:00 AM (to be determined).

Supervisor:	Dr. R. van Katwijk	TNO
Thesis committee:	Prof. L. Tavasszy,	TU Delft
	S. Cunningham,	TU Delft
	Dr. S. van Cranenburgh,	TU Delft

Preface

The ideas, designs, analyses and results that can be found in this report constitute the last chapter of my Master studies in *Systems Engineering, Policy Analysis and Management* at the Faculty of Technology, Policy and Management, TU Delft. The report is the result of a six-month long graduation project, which was carried out in cooperation between the TU Delft and TNO. The project and subsequent research was conducted between February and August 2017.

During my thesis I experienced plenty of assistance and helpful advice from everyone throughout the 'Sustainable urban mobility solutions'-department at TNO. I especially want to thank Ronald van Katwijk, who has helped me a lot during these six month and was not only a good advisor but also a great support for me. Furthermore, I want to thank my graduation committee at TU Delft, consisting of Scott Cunningham, Sander van Cranenburgh and Lóránt Tavasszy, who guided the research from an academic point of view. Here, special thanks go to Scott, who has always had an ear for me when I needed guidance or a different perspective on my work.

Lastly, I would like to thank my family. They have always been by my side with endless support and warmth. It is due to them, that I could take the path I desired to. They have made it possible for me to obtain this Master degree and become the person I am today.

Kai Timon Busse
Delft, August 2017

Executive summary

In 2013, a study of the European Commission has shown that more than 75% of all passenger-kilometers and more than 50% of tonne-kilometers are made "on-street" in the Netherlands. Considering that this portion is not believed to decrease in the short- or medium-term and that the overall traffic demand will further increase, subsequent congestion issues are nothing but a logical consequence. Together with drastically increasing emission values, these issues have developed into one of the most pressing problems for the Dutch government. Hence, in cooperation with the government, various organizations have targeted their research efforts on exploring potential solutions to conquer these problems. In this context, TNO, being one of these organizations is working on so-called smart traffic solutions.

Two of those smart traffic solutions are platooning and intersection control. They represent interesting alternatives to lengthy and costly road expansions and have individually already proven their potential to tackle congestion and emission issues in certain occasions. Furthermore, it has been suggested that promising benefits can be expected from a combination of both if a functional synergy can be reached. Within the context of this thesis, it is specified that such a synergy can be considered achieved if a successful navigation of a platoon through a string of intersections is ensured while not making the urban traffic environment worse off in terms of the congestion or emission objectives. The reasoning behind this is as follows:

Urban platooning might very well represent a door opener for highway platooning, effectively expanding the latter's applicability range. If a technical solution can be found which allows to safely navigate a platoon through one or several intersections, platoons can either be formed on provincial roads (e.g. at an intersection) before they enter a highway or they do not need to be decomposed before leaving the highway. Both effects enable a longer overall highway platooning period, which subsequently leads to a longer period where the benefits of highway platooning can be harvested. If additionally no negative effects occur on the urban environment, a contribution can be made to the overarching governmental objective. Namely, decreasing overall (urban and highway) congestion and emissions.

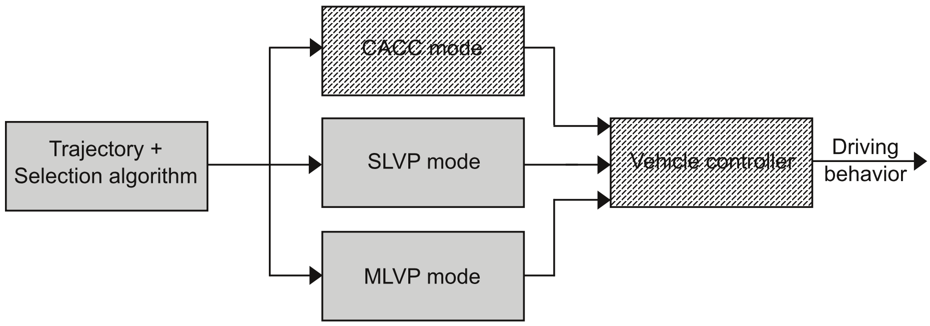
Yet, in order to make a reasoned decision on whether and how to promote urban platooning, the Dutch government requires founded knowledge on its expectable benefits and the associated costs. It is the purpose of this research to support the government in making this decision by producing this knowledge. The thesis attempts to do so by firstly developing a technical solution that enables urban platooning through a functional combination between the two concomitant systems, highway platooning and intersecting control and secondly, by providing a policy-development framework that can be utilized to design a policy which promotes this new technology. These two aspects - the technical solution and the policy-development framework - represent the main contributions of this research. Considering that the latter promotes a deployment of the former, an inside-out research approach is adopted for their development. That is, the design of a technical solution for urban platooning is executed first, pursuing the overarching goal of minimizing congestion and emissions. After the technical design is completed, the second stage is entered where a framework is introduced, aimed at an effective introduction of this technical solution. Henceforth, in the following the two are introduced and explained in this order.

Technical solution

Platooning describes the capability of a set of autonomous vehicles to drive in an array formation, where the intra-vehicle gaps are reduced to a minimum. Using sensor data and sophisticated, mostly decentralized platoon control systems, safety gaps of less than 10 m at 90 km/h are possible. As freeway platooning functionality is mostly established and proven, research has recently shifted its focus on solving remaining burdens, such as legal constraints with driverless cars. Yet, the fact that platoons will sooner or later enter the current road network is hardly contested and therefore assumed in this work. Given this assumption, five major research steps can be identified that guide this work towards a functional urban platooning solution.

The following introduces these 5 steps, of which the latter three constitute the technical main contribution of this research.

The first of those is the implementation of a mathematical vehicle controller model (1), which allows to realistically depict platooning in a simulation environment. As shown in below figure, this controller serves as a basis to (2), the deployment of the so-called CACC platooning mode, where vehicles determine their longitudinal position according to their direct precursor. Yet, as further illustrated in that figure, both, the CACC platooning mode and the vehicle controller are based on an exiting concept and therefore not seen as scientific contributions of this thesis. They rather provide a representation of real-world vehicle systems, accurately modeling the vehicles' functionality and characteristics. Yet, both these models serve as a basis for an urban platooning solution.



Illustrating the low-level technical contributions of this thesis (Gray represents a scientific research contribution; shaded gray represents a model of an already existing real-life system or functionality).

Now, with platooning slowly becoming a serious option for highway trafficking, it is only a matter of time until this possibility will also be discussed for urban roads. When this is the case, one of the major issues will be how platoons of vehicles can successfully be navigated through one or several intersections. Yet, this does not only represent a challenge but potentially also a chance to push towards synergy effects with intersection control. That is, most recent traffic controllers tackle congestion and emissions by pursuing an optimized green light schedule. The green lights are dynamically assigned, according to a traffic model, which is internally established by the intersection controller. The model is based on in-street sensors, which are located around an intersection and it gives valuable insight about the demand on all intersection. Now, sharing this model with a platoon is what constitutes the basis of the proposed urban platooning solution. That is, besides others, the model predicts the time at which each vehicle is able to enter the intersection (i.e. get a green light).

This so-called estimated time of departure is the main element around which the integration with platooning centers. Through a V2I channel, the intersection controller provides a platoon leader with the estimated time of departure. Henceforth, a "Trajectory and Selection"-algorithm is developed (3), which uses this value to calculate an energy-optimal trajectory along which the platoon needs to be navigated. Yet, the sole knowledge about this trajectory does not suffice to enable urban platooning. That is, the conventionally used CACC mode does not allow to process such a trajectory and deploy it by feeding it into the vehicle control.

Hence, besides this algorithm, further technical design is required to enable a deployment of the energy-optimal trajectory through the vehicle-internal control system. For doing so, two alternative platooning modes - SLVP and MLVP - are developed (4)(5), which although making use of the same vehicle controller, follow essentially different purposes than the initially modeled CACC mode. Both of these platooning modes, as well as the C++ code constituting them is, what is considered the technical main contribution of this work. Eventually, the "Trajectory and Selection"-algorithm ensures that the right platooning mode is selected in the right situation. In conjunct functionality, the conventional CACC mode and the newly designed SLVP and MLVP modes do not only allow to model platooning, but especially the latter two also enable the desired functional combination with intersection control and henceforth urban platooning.

Altogether, the three platooning modes are capable of facilitating all the longitudinal and lateral vehicle motions that are necessary to deploy urban platooning on a simulation basis. Furthermore, the "Trajectory and Selection"-algorithm ensures that the right mode is applied at the right time through which overall functionality of the traffic simulation is reached. This simulation is of high representativeness, especially in terms of congestions and emissions. Yet, as for every model there remain certain residual concerns about the informative value that can be gained from these simulation conclusions. These concerns are acknowledged and tackled through the policy-development framework.

THE POLICY-DEVELOPMENT FRAMEWORK

The policy-development framework is to large extents based on the costs and benefits that can be expected from an implementation of urban platooning through above-introduced technical solution. In this sense, yet another contribution of this work is the quantification of the latter. This quantification is done by means of a microscopic, time-step oriented traffic simulation of urban platooning. In order to put the benefits of urban platooning in context, four different scenarios are simulated, whereas each represents a different technological setup. The evaluation is executed, based on a simulation model of the N260, near Tilburg, The Netherlands. The traffic demands of that simulation represent those of an average weekday on that track and they are the same for all possible combinations of the two technologies. Given, a 60 minutes simulation period, the following results are found:

Impact evaluation for the four possible technology setups, concerning the average velocity and the overall emissions for the N260 case.

Scenario	Performance Indicator	Value
No platooning, no intersection control	Average vehicle velocity	70.7 km/h
	Average CO_2 emissions per km	216.68 g/km
Only intersection control	Average vehicle velocity	72.03 km/h
	Average CO_2 emissions per km	210.60 g/km
Only platooning	Average vehicle velocity	71.21 km/h
	Average CO_2 emissions per km	208.84 g/km
Platooning and intersection control	Average vehicle velocity	71.74 km/h
	Average CO_2 emissions per km	208.85 g/km

As it can be seen from above table, the platooning-intersection scenario does not only satisfy the design requirement of not making the urban environment worse off in terms of congestion or emissions, but it even features the overall best performance in terms of emissions and a reasonable improvement in terms of congestion. Given the success of the technical design phase, sufficient motivation exists to follow-up upon this with a set of high-level policy notions on how to implement such a scenario. However, given the novelty of the urban platooning concept and the fact that only little attempts were made on enabling this form of driving behavior, more research is required to establish a scientific sound basis on which a concrete policy can be designed. Hence, due to this disruptive nature of the research, no actual policy is designed. Instead a generically applicable policy-development framework is proposed, which can assist in the formulation of a policy.

At first sight, the fact that a policy is necessary in order to promote the implementation of urban platooning infers two major deductions: (a) The government is not able to single-handedly implement the proposed technical solution. It rather relies on a second party's help to do so and (b) this second party is generally against an implementation, as otherwise no policy would be necessary. Now, through thorough literature research on *complex network theory* and more specifically *decision-making in complex networks*, this second party can be identified to be the traffic participants, which have to be persuaded to partake in platooning. If this is achieved, the government can self-reliantly implement the necessary intersection controllers and hence realize the desired scenario.

Given this understanding, it becomes clear that an effective policy needs to persuade the traffic participants to align their strategy with that of the government, namely to invest and participate in platooning. In order to model the traffic participants' decision making process towards taking this policy-bait, game theory can be employed. Game theory is in this sense a tool that can help to process the numerical congestion and emission savings from above table into valuable political insights for the government. By doing so, a financial threshold is determined, which symbolizes the point after which the traffic participants are willing to align their strategy with that of the government. A set of mathematical conditions is provided, which can be used to check whether such a policy can be established and whether it is worthwhile for the respective policy maker to do so. Hence, if the government manages to find a policy which satisfies these conditions, this policy has proven workability on a game theoretic basis. That is, if the government decides to cover the financial threshold and further invest in innovative traffic controllers, the urban platooning scenario is the sole logical consequence in a game theoretic context.

Yet, one must not forget that these game theoretic considerations represent a simplified model of real-world matters. Modeling the traffic participants' decision-making process infers a trade-off choice, whereas game theory appears to be a reasonable comprise between model simplicity and real-world representativeness. However, the simplifying nature of this choice needs to be kept in mind and handled with attention. Additionally considering the residual concerns about the informative value of the traffic simulation, one major purpose of the policy-development framework is to revisit the made modeling assumptions, the accepted simplifications and the limited data value that is drawn from this. Hence, although being based on game theoretic notions, the policy-development framework attempts to embed these insights in a more far-sighted context, effectively compensating for the (necessarily) made assumptions.

To do so, two policy-development phases are suggested - a theoretical policy verification & validation and a practical policy implementation phase. The former revisits the above-mentioned modeling assumptions and proposes a set of plausibility checks to test whether or not these hold truth for a given scenario. If this is the case, hence the policy is theoretically considered fit, the framework proceeds to the second phase, which is constituted through a policy-deployment roadmap. This roadmap gradually works its way from small field tests up towards a large-scale policy implementation. The benefit of this framework is that a policy can be tested on theoretical basis before any real-world experiments are initiated. It is believed that despite the made modeling assumptions, this allows to acquire a solid evaluation of a broad range of policies without facing major expenses. The government can hence maintain a broader design space when seeking policy solutions that promote urban platooning, effectively increasing the chances for a successful policy.

CONCLUSIONS AND RECOMMENDATIONS

The project itself is conducted in cooperation between the TU Delft and TNO. Although, the basis of this work, the individual technologies are known and well-analyzed in their respective literature, a combination between the two systems represents a significant knowledge gap. In that sense, the research at hand is of disruptive nature. Both, the design process and the subsequent findings should therefore be considered as an exploratory project, which requires in-depth follow-up research. Furthermore, it solely aims at evaluating the congestion and emission impacts of the technologies. It therefore needs to be understood that this thesis does not aspire to present a new fully-functional system but only to quantify its potential benefits and provide a first basis upon which further technical improvements can be made.

Due to the disruptive nature of this research, a choice is made to intentionally provide no concrete policy. The government is free to find ways on how to close the introduced financial threshold itself, whereas the provided insights and the policy-development framework can serve as a basis for doing so. Consequently, one of the main recommendations of this work is to follow up on this gained insight by designing a policy, which would do so.

Following the technical design, its embedment in the simulation and the traffic model itself are thoroughly verified and validated. One of the main findings of this is that although, the modeling quality suffices for assessing potential congestion and emission savings, some flaws remain. These are mainly concerned with model inaccuracies and occur due to simplifications that were made during the technical development. It is therefore recommended to apply the model solely to the purpose it was designed for. In order to allow e.g. for an evaluation of safety aspects or driving comfort, further modeling effort is necessary. This is mostly concerned with improving the technical design and refining both the technical and the game theoretic model. It is recommended to seek such model improvements and incorporate further relevant impacts into the decision-making process of the government.

Contents

1 INTRODUCTION

Assuming a global point of view, the 21st century appears to be shaped by one major challenge – facilitating human growth. Within the last 50 years the economy and especially the automobile industry have experienced an enormous upswing. The influence this substantial transformation has on the automotive transport sector is twofold. On one hand, the increasing population has entailed an escalating demand for transportation, both in terms of passenger and freight transport. On the other hand, technological advancements have led to a gain of personal mobility and an improved access to transport infrastructure (Becker, 2006). As a consequence of both these effects, the traffic on the current road network has largely increased, leading to accretive emission and congestion problems.

Through the years, the conventional fossil-fueled car has established itself as the most utilized means of transportation in Europe and the Netherlands, substantially contributing to above mentioned problems. According to 2013 statistics of the European Commission, more than 50% of all tonne-kilometres and 75% of all passenger-kilometers were made 'on street' in the Netherlands (EC, 2013). The remainders are divided over airborne, waterborne or rail transport. Furthermore, this portion is not believed to decrease in the short- or medium-term, while overall travel-kilometers will further increase (EEA, 2016). In additional consideration of the ongoing upswing of the automotive sector, it is merely a logical consequence that more and more cars enter the road network (Becker, 2006), amplifying those issues.

Taking into account that road expansion is a long and costly process, this infers a few demands for the Dutch traffic system. One of which is the need for innovative traffic solutions, which are implementable in the current road network and do not require expansions. The Dutch government is seeking such solutions, which is why the high-level goal of this thesis is to establish a framework that can support the government in an implementation of one such smart traffic solution - urban platooning.

Urban platooning is a new, hardly researched interpolation of highway platooning onto urban environments. Given the disruptive, novel nature of this technology, the afore-mentioned high-level goal is complemented through a technical, low-level goal, namely the development of a technical design, which enables urban platooning. The thesis at hand attempts to fulfill this low-level goal through a synthesis of two already existing systems, highway platooning and intersection control.

Although, the low-level technical design is shaped through high-level, governmental needs, it is suggested that the reader should first get a rough idea of urban platooning and its potential benefits. In order to provide a superficial understanding of the intended technical design, Section 1.1 briefly introduces the two concomitant systems and henceforth contours the aimed at novel traffic solution - a functional combination between the concomitant systems.

Given this generic understanding of the to-be-developed technical design, its embedment and role in the high-level considerations can be researched. As mentioned, a governmental point of view is adapted in order to analyze whether and how such a system can be brought in place. For doing so, the aimed at technical solution is embedded in a four-layered problem formulation in Section 1.2. Going from the highest-layer governmental point of view the problem is broken down and the initially outlined technical solution is made subject of a set of design requirements and objectives on the lowest layer. These requirements and objectives are logically derived from high-level needs and broken down into a low-level technical challenge. Subsection 1.2.4, the low-level technical problem formulation hereby sets the scene for the main contribution of this research - the technical design. Given this, Section 1.3 introduces the deliverables, that will be developed to conquer the presented problem formulation.

The remainder of the Chapter then elaborates on how these deliverables are developed and brought into place in the course of this thesis. Section 1.4 proposes a six-step research framework. Together with Section 1.5, which contours the scope of this thesis, this lays the basis for developing the thesis outline in Section 1.6.

1.1. BACKGROUND

Before any elaborations on the high-level governmental needs are made, this section aims at establishing a superficial understanding of the intended design of a functional combination of platooning and intersection control and its potential benefits. For this it needs to be understood that the concomitant systems, platooning and intersection control represent two well-researched concepts. The knowledge gap, which this thesis attempts to fill, can be allocated in the functional combination of these technologies. Yet, about the concomitant systems extensive knowledge is available. Hence, in a first attempt, the two underlying systems are briefly explained in the following, before further elaborations are made on the potential synthesis between them.

When reading this introductory section, the reader needs to remember that although both systems are allocated in the domain of smart traffic solution, their current fields of application are essentially different. While platooning has until now mostly been researched for highway use, intersection control is naturally limited to urban and provincial roads. Yet, both technologies pursue the same aim, namely minimizing congestion and emissions. Given this, Sections 1.1.1 and 1.1.2 firstly introduce how the two technologies approach this goal, individually. From this a line of reasoning can be derived, which motivates why their synthesis represents a potentially interesting technological advancements. The latter is described in Section 1.1.3 and complemented through a first notion on the low-level technical knowledge gap.This will be revisited and comprehensively explained in the problem formulation of Section 1.2

1.1.1. PLATOONING

The earlier mentioned growth of the automotive industry implies two trends that are relevant for this research. As mentioned in the beginning, the first one is the increased need for smart intersection solutions. The second one, originating from the technological advancements, is a strong improvement in vehicle capabilities in terms of autonomy and cooperative driving (Lu et al., 2010).

In an era in which computers are pushing their way into the transport sector and governments are striving towards eco-friendlier automotive solutions, platooning has gained its spot among the most promising approaches on utilizing these new capabilities. Logically, this concept has lately moved into the innovation focus of multiple OEMs, although being constrained to highway application until now. Companies like Scania (Scania, 2017), Volvo (Volvo, 2017) or MAN (MAN, 2017) are gradually working their way towards a realization of highway platooning. More specifically, through the on-going integration of driver assistance systems, cars are gaining autonomy and driverless vehicles are on the edge of becoming reality. With the continuous improvement and integration of sensing technologies, both passenger vehicles and trucks will soon be able to observe their environment self-reliantly. Enabled through innovative vehicle-to-vehicle communication, the shortcomings of a human driver (such as reaction time, negligence or fatigue) can thus be resolved. Braking or steering motions can be triggered simultaneously throughout the platoon and sensed information is shared among the vehicles. Eventually, through the synthesis between sensing and communication technologies vehicles are capable of driving in an array with significantly lower vehicle clearances than conventionally possible.

In this sense, platooning describes the driverless formation of two or more vehicles in a very short range to each other (Chen and Wang, 2005). Within the European Truck Platooning Challenge this is defined as follows: “Platooning comprises a number of cars equipped with state-of-the-art driving support systems – one closely following the other. This forms a platoon with the cars driven by smart technology, and mutually communicating.” (Eckhardt, 2015, p. 16). “These linked vehicles then proceed to travel along the [...] road system acting as one unit” (Kavathekar and Chen, 2011, p. 2). The benefit of this technology is mutually interesting in congestion and emission terms. Due to the absence of human driving characteristics such as reaction delays the vehicles are e.g. capable of accelerating precisely when the precursor does. They can occupying less space in crowded traffic situations or simply trigger a braking motion immediately when it is necessary and not with the typical delay of a human driver. Through all this, the emergence of congestion can effectively be anticipated and hampered. Alternatively measures can be taken to resolve congestion in early stages. Furthermore, reduced safety gaps between multiple vehicles of a platoon can reduce the drag resistance for following vehicles and hence help to save on emissions.

Various research projects were launched and executed in order to analyze these and potential other benefits of platooning. Although, being limited to highway application only, their findings are of high importance for the project at hand nonetheless. The reason for this is twofold. Firstly, they allow for an indication of what the research should aim at and secondly, they represent an incentive for the execution of this research. They do not only hint at potential benefits that can be reached on urban roads but they also constitute a reward that can be harvested on freeways. The implementation of highway platooning might very well amplify the chances of urban platooning and vice versa. Consequently, this research indirectly contributes to the promotion of highway platooning as well as urban platooning.

For some of the potential benefits reasonable quantifications or assessments exist. E.g. Lammert et al (2014), Alam et at. (2014) and Eckhardt (2015) all attempt to quantify the emission savings that can be reached through platooning, by employing latest research insights. Lioris et al. (2016) provide a recent assessment of the potential congestion optimization. This and further, less recent research insights are summarized in Table 1.1. All those cases however, assume a highway or freeway environment. Thus, their conclusions can, on direct effect only be used as reference values for this research, which itself attempts to evaluate a performance optimization on urban streets.

In sense of the indicative function of Table 1.1, it should be noted that fuel savings, energy savings and emission savings often go hand in hand, as "energy and emissions are directly related to the volume of fuel used" (Bakermans, 2016, p. 12). Therefore, findings on emission savings allow for deductions about fuel consumption and previous research allows for implications vice-versa.

Table 1.1: Specification of assessed consequences (adapted from (Bakermans, 2016); based on (Bonnet and Fritz, 2000), (Robinson et al., 2010), (Alam, 2014), (Lammert et al., 2014), (Eckhardt, 2015), (Lioris et al., 2016), (Michael et al., 1998), (Nieuwenhuijze et al., 2012), (Schermers et al., 2004), (Ploeg et al., 2011a) and (Farokhi and Johansson, 2015)).

Researched consequence	Source	Conclusion
Emission savings	Bonnet & Fritz, 2000	4.5-21%
	Robinson et al, 2010	2-13%
	Alam et al, 2015	6%
	Lamert et al, 2014	5.3-9.7%
	Eckhardt, 2015	8-13%
Throughput	Lioris et al, 2016	up to 150% increase on highways under perfect circumstances
	Michael et al, 1998	high increase on freeways and moderate increase on urban intersections
	Nieuwenhuijze et al., 2012	increase
	Schermers & Malone, 2004	increase
	Ploeg et al, 2011	decrease

Although, this research's scope is limited to emission and congestion issues only, platooning does bring along further potential consequences - both benefits and drawbacks. Having understood that platooning vehicles can autonomously drive very close to each other (Chen and Wang, 2005). The most prominent of those are summarized in Table 1.2.

Table 1.2: Potential positive and negative consequences that can be expected for an implementation of urban platooning (adapted from (Bakermans, 2016)). Consequences that are assessed within this research are written in bold.

Consequences	
Expected but not assessed	Assessed and proven in literature
Decreased labour costs	**Throughput optimization**
Higher asset utilization	**Emission savings**
Higher road degradation	Decreased fuel consumption
Increased road safety	Decreased value of time losses
	Higher initial investment costs

Until now this Chapter has attempted to establish a good understanding of the perceivable benefits of highway platooning. In the further course of this chapter it is explained how urban platooning can contribute to harvesting these benefits on highways and further reduce congestions and emissions on urban roads. Firstly however, the second concomitant system is introduced - Intersection control.

1.1.2. INTERSECTION CONTROL SYSTEMS

Given the earlier-mentioned, continuous increase of vehicles on our roads and subsequent congestion and emission issues, research has put its focus on identifying bottlenecks within the current traffic system. In consensus with earlier traffic theory publications (see e.g. Fouladvand et al., 2004; Rouphail et al., 1992) Sun et al. (2015, p. 18) identify “intersections that cause queue spillover and network gridlock” as one of the major reasons for capacity drops of the common street network. They facilitate opposing traffic flows, which naturally leads to delays for all involved road users. Logically, this represents a potential starting point for tackling congestion and emission issues. Self-evidently, a smart traffic solution is most effective where the network performance is lowest. By identifying uncontrolled intersections as this node of low performance, it becomes clear that the installation of intersection control potentially represents a promising consideration, which should be made subject of thorough analysis. And indeed, although it is proven that signalized intersections can lead to an improvement of traffic flows when compared with non-signalized intersections, it is also clear that they nonetheless represent one of the biggest bottlenecks of the common road network (Rouphail et al., 1992).

Furthermore, research has shown that oversaturated intersections of any form can lead to significant delays in their wider area of influence (Knoop et al., 2006). Their connectivity amplifies the intersections’ bottleneck nature throughout the system. In other words, a network system differs from a stand-alone system in that sense, that an action at one point can potentially influence the traffic demands at all other points. The more connected such a system is, the higher is the extent of this bottleneck influence.

Applying this on traffic network terms, congestion issues amplify themselves in settings with numerous, closely-located intersections (Porta et al., 2006). This is typically the case in urban networks, which therefore represent such a critical aspect for the overall network performance. With municipality roads making up 86% of all paved streets in the Netherlands (CBS, 2016), this thesis sets the focus on said urban traffic environments. More specifically, it looks at a string of signalized intersections, which are widely believed to be a main cause of congestion (Fouladvand et al., 2004) and how a platoon of vehicles can navigate through these intersections. These situations are henceforth termed ’urban roads’ or ’urban environment’

The simplest form of intersection control is the so-called ‘fixed-time controller’, which assigns green times to the different intersection entries according to a predetermined green-time schedule. Although these fixed-time controllers naturally have “theoretical limits of optimum” (Sun et al., 2015), the existing intersection control systems usually do not reach this point. Especially, when a network-wide point of view is assumed, this basic control algorithm fails to maximize the system’s performance. Hence, within traffic theory literature it is widely agreed that there is a pressing demand for smart intersection control solutions (see e.g. Van Katwijk, 2008). These smart intersection controllers represent the second concomitant system of this thesis.

Yet, the need for such smart solutions is a long-known problem. The shortcomings of conventional fixed-time intersection signals are a well-understood problem, which is already being tackled. TNO is working on various solutions to this, which are generally consolidated under the umbrella term 'traffic signal control' (TSC) (see e.g. Van Katwijk et al., 2006).TSC essentially describes a controller algorithm, which measures certain intersection parameters (e.g. the traffic demand from the different intersection entries) and assigns green times to the directions accordingly. It utilizes several sensors and processes those signals in order to steer the traffic through its sole actuator – the traffic light. The technology or more precisely the idea of controlling traffic through variable green, yellow and red phases, serves as a basic concept for this research.

JUNO is one of those intersection controllers and the algorithm, which will be analyzed and utilized in this thesis. It represents a specific form of 'look-ahead traffic-adaptive control', which itself is a sub domain of TSC. In Chapter 2 a review of JUNO's functionality is provided and it is illustrated how the algorithm is adopted in the context of this research. For now its potential benefits have to be understood. Various research efforts were made in order to quantify these benefits and have shown that travel time savings of 6%-15% and emission savings between 2.9% and 9.3% are possible (see e.g. van Katwijk and Gabriel, 2015).

1.1.3. SYNTHESIS

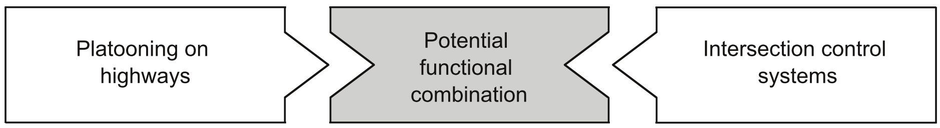


Figure 1.1: Illustrating the low-level technical knowledge gap (Gray represents a research contribution; white represents an already established systems).

Both previously introduced technologies are to a certain extent already understood and analyzed in the respective literature. With platooning slowly becoming a serious option for highway trafficking, it is only a matter of time until this possibility will also be discussed for urban roads (see e.g. Larson et al., 2015). When this is the case, one of the major issues will be how platoons of vehicles can successfully be navigated through one or several intersections. However, the combination of platoons and modern intersection control is a very new consideration.

Using conventional traffic signals (i.e. fixed-time control), platoons would constantly be interrupted through red lights or worse the following vehicles would run over red traffic lights and put crossing cars to danger. Theoretically, this could be tackled e.g. through the Transit Signal Priority (TSP) system (see e.g. Garrow et al., 1997), which is currently in place to allow for bus priorities in the Netherlands. This however would not improve but rather worsen the already existing congestion and emission problems. That is, because the TSP system relies on a simple 'If bus approaching, then bus has priority'-logic. The thesis at hand however, intends to minimize these issues by means of combining the two technologies. It aims at a smart platoon-intersection controller solution which guarantees the successful and safe navigation of a platoon through a string of intersections, while harvesting and increasing the congestion and emission benefits both technologies offer. That is, if the integration of platoons to urban environments does not make the congestion and emission-wise performance on these urban environment worse off, the overall perceivable benefits will be higher. The line of reasoning behind this statement is as follows:

As mentioned, one major aspect, motivating the necessity of this research is that urban platooning might very well represent a door opener for highway platooning, effectively expanding the latter's applicability range. If a technical solution can be found which allows to safely navigate a platoon through one or several intersections, platoons can either be formed on provincial roads (e.g. at an intersection) before they enter a highway or they do not need to be decomposed before leaving the highway. Both effects enable a longer overall highway platooning period, which subsequently leads to a longer period where the benefits of Table 1.1 and 1.2 can be harvested. Hence, additional benefits can be reached if such a solution can be found. Given that no negative consequences are imposed to the urban environment, through this measure an overall improvement can be reached, diminishing the existing congestion and emission issues. Additionally remembering that platooning effectively eliminates the shortcomings of a human driver, the reader will understand that (slight) improvements on the urban roads are perceivable.

Yet, only very limited efforts were made on a combination of the technologies so far. Effectively, only one potential solution, namely utilizing the TSP system has been presented to safely navigate a platoon through a string of intersections. Hence, a clear need for further research and a technical knowledge gap can be identified. The research at hand attempts to fill this low-level knowledge gap through the meaningful exchange and processing of sensor data, both from the platooning vehicles and the intersection controller. For now it needs to be understood, that data is shared through a V2I-channel and then used as such that the above requirement of not making the urban congestion and emission values worse off is fulfilled.

The intersection controller is characterized through its ability to steer traffic through a set of sensors and one actuator - the traffic light. Now, going one imaginary innovation step further from this, the intersection controller could also make use of several actuators instead of only one. Coordinated intersections (see e.g. Niu et al., 2011) for example are capable of communicating speed advice with approaching drivers or vehicles. Within the vehicle control system this can be translated into an acceleration or braking motion, which is then executed by the in-vehicle actuators. The system combination that will be presented at the end of this thesis is loosely based on the idea of these coordinated intersections.

The difference between conventional traffic lights and coordinated intersections is in this sense that the road infrastructure communicates with the vehicle and not with the driver. Similar to TSC, it gives unilateral orders, which however do not necessarily require driver actions anymore. Thus, the information coming from the signal can directly be incorporated in the automated vehicle's control system. The thesis at hand will use this form of I2V communication as a basis and expand upon it by assuming that the vehicle can communicate with the infrastructure vice-versa. This form of intersection is in the following termed a cooperative intersection (CI). In the course of this report, this functionality is incorporated both in the to-be-develop platooning controller and in JUNO.

Within the next sections it is explained why and how traffic can be modeled as a network. Given this, a global view of the traffic situation and a platooning-intersection controller that takes those network interactions into account is required. This high-connectivity characteristic however, also represents a chance to increase network performance. Several CIs can be connected in order to form a smart traffic network, which is capable of steering traffic and traffic demands (see e.g. Lebacque, 2005) and hence make room for a platooning convoy. The idea of network optimization is already incorporated in JUNO and will be elaborated on in Chapter 2.

Concluding this research, a combined controller is presented, which features and utilizes a V2I-channel in order to navigate a platoon through a string of intersections and optimize traffic from a global point of view. This controller integrates the conventional platooning idea with that of intersection control and traffic optimization. This is reached through well-designed data exchange between the two systems and functional extensions mainly within the platooning but also within the intersection controller.

1.2. PROBLEM FORMULATION

By now the reader should have a superficial understanding of the central element of this thesis, urban platooning and how the thesis attempt to bring it into place. In the previous section its potential benefits are explained and sufficient motivation is provided to attempt a subsequent technical design process of a functional combination between platooning and intersection control. This Section centers around the formulation of design requirements and objectives that need to be considered in this process. It does so by taking a step back - from analyzing the actual technical design to examining its context, why it is needed and what it is supposed to do. As mentioned earlier, the initial point of view which is adapted herefor is that of the Dutch government. The underlying goal of the government, to which the proposed technical design represents one possible solution is that of increasing emission and congestion issues on the Dutch road network.

In this Section it is illustrated how these problems are essentially a specific case of complex systems management, which, in a first approach, allows for the formulation of a policy objective. This high-level policy objective is then refined and embedded in a game theoretic context. As part of this Section it will be explained what game theory is and how it can serve as a useful tool for the formulation of a policy. The game theory consideration itself serves as a basis for a simulation objective and eventually for a low-level technical of platooning and intersection control. The underlying so-called 'V-model'-approach, which this logic follows, is illustrated in Figure 1.2.

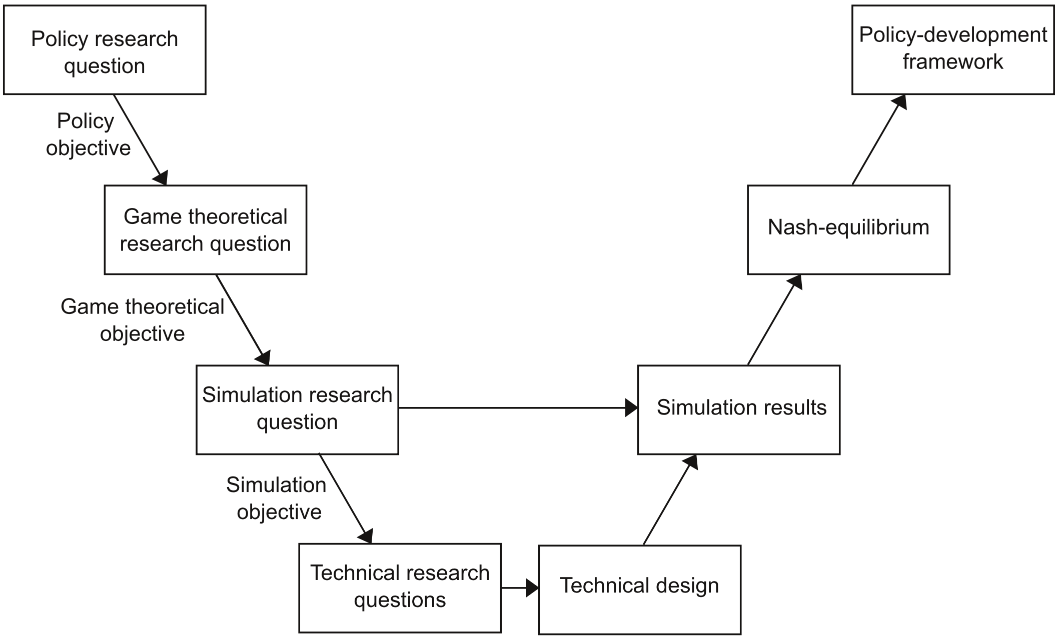


Figure 1.2: High-level illustration of the research approach as a V-model.

Originally, the V-model (Pressman, 2005) originates from the development of IT-systems. Yet, over the last decades it has established itself as one of the major research frameworks in a broad range of engineering fields. It is a common method for breaking down high-level research questions into low-level problem formulations, finally leading to a set of technical design requirement and objectives. As illustrated, this technical design objective is what is eventually taken hold of through a well-structured research approach and answered through the associated technical design. The sections succeeding this one, which are solely concerned with this low-level research, can therefore be allocated to the lowest level. In contrast to that, the purpose of this section is to put the research into a high-level context, which eventually is revisited in Chapter 6, where high-level implications are derived from this low-level design. That is, in converse manner to the left arm of the V-model, the technical design provides knowledge that is required to conquer the medium- and high-level objectives. In the context of the research at hand, it is the technical design of a platooning controller and its integration with an intersection controller, which serve as input for the game theoretical aspects, which are eventually translated into policy considerations.

In this sense, the research willingly refrains from adapting a traditional research question-research subquestion structure. Instead of establishing a tree-like map of research questions, the reader is guided through a cascading refinement of the main research question (see e.g. Hoffman and Beaumont, 1997). The underlying notion that a low-level question contributes to the solution of the overarching question, remains the same however. In order to ensure this right-arm relation between the levels, the left-arm problem formulation has to be made accordingly. That is, the low-level design requirements and objectives need to be derived from high-level needs or in other words: Using the V-model, the needs of the government are translated into low-level technical design requirements. The policy deliverable requires certain game theoretic knowledge. To gather this knowledge, the second level needs certain simulation knowledge and lastly the simulation needs a technical design to provide this knowledge.

Given this, each of the levels has a deliverable, although the overall deliverable is that of the highest level, namely a policy-development framework. The lower level deliverables represent building bricks that are part of the eventual overarching policy deliverable. In order to develop the highest level deliverable, certain research knowledge is necessary from the next lower level. As illustrated in Figure 1.2, the deliverable of this next lower level, hence constitutes this necessary knowledge.

This relation needs to be analyzed and well understood in order to ensure deliverable quality on all levels. In order to do so, the following approach is chosen: The highest level firstly poses a research question. As mentioned, this research question requires further knowledge from the next lower level. This need is formulated into a research objective, which is posed by the higher level and imposed on the lower level. That is, as depicted in Figure 1.2 all of the first three levels impose such an objective to their next lower levels. The research question of the level to which a research objective is imposed can therefore be formulated as such that it provides the necessary research knowledge to the next higher level. Yet, this is not the only knowledge that is required to answer the next higher level research question. It is rather, that on every level a certain value is added. In synthesis with the knowledge that is provided from the lower level, this eventually allows to answer the research question of the level at hand and provide input for the next higher level.

It is characteristic for this approach that the main research effort is concerned with the lowest level, answering the technical objective. This means that not only the main share of this chapter but of the whole report centers around said controller design and the functional combination with an existing intersection controller. However, besides others it is the high-level problem formulation, which this technical design eventually attempts to answer. Therefore, close attention should be drawn to the following subsections, which firstly introduce and motivate a high level policy objective and then explain how this can be translated into a game theoretical consideration to which a simulation of the low-level technical design eventually contributes its share.

Section 1.2.1 guides the reader through the high-level governmental needs. Eventually, a problem formulation and subsequent research question is established, which represents these needs. From this a game theoretic problem formulation and research question is derived in Section 1.2.2. Here it is further illustrated that in order to answer this second-level research question, the game needs to be informed with concrete numbers. Section 1.2.3 introduces simulation as a potential way of doing so and motivates the choice for this evaluation method. It subsequently formulates a simulation research question. This comprises the simulation of a set of traffic situations, of which one includes the introduced concept of urban platooning. Lastly, this demands that such urban platooning system must be designed, which lays the basis for the lowest-level problem formulation. Keeping in mind that this represents a major share of this work, Section 1.2.4 goes into more detail about the formulation of a technical design research question. This final research question is supplemented by a set of design requirements and objectives, which are derived from the cascaded nature of the higher levels.

1.2.1. LEVEL 1: POLICY PROBLEM FORMULATION

In order to establish a high-level problem formulation, further understanding of the point of view which is adapted here for is necessary. It has already been said that the problem of rising congestion and emission levels is analyzed and tackled from a governmental point of view. However, the reasoning behind this choice, the unique position of the government, its characteristics and last but not least its needs have yet to be introduced.

The issue itself, being increased congestion and emissions, represents the outcome of a long range of choices, decisions and actions taken on a daily basis by a variety of stakeholders such as car users, pedestrians, the automotive industry, governments, etc. What unites all of these stakeholders is that they are part of a socio-political network, in which their actions can potentially influence a range of other parts of that network. This setup does not only term the network a complex one but it also makes it difficult to identify the one, always applicable problem owner or problem solution. A strategy that is often applied in such situations is to make a choice as such that the problem owner has sufficient motivation to conquer the problem, yet also network position that locates him at the root of the issue, putting him in the right position to implement a solution (Easley and Kleinberg, 2010).

To identify this particular actor, the underlying network situation requires further consideration. Upon closer inspection, one will notice that the above-mentioned socio-political network is indeed not the network from which congestion and emissions originate. As illustrated in Figure 1.3, it is the traffic system that is responsible for the cause of those problems. However, both networks seem to be interwoven and (especially long-term) interactions exist across networks. To reach a better understanding of this, a quick example of a possible network setup is introduced: Actors 1,2 and 3 are traffic participants. They interact with each other in a given traffic situation and therefore form a traffic system. Besides that, actor 2 has a long-term interaction with the automotive industry (actor 4), because he is considering to buy a car. The automotive industry itself is regulated by the government (actor 0) and therefore has a socio-political interaction with that actor.

Now, the problem owner that seeks initiative should be located at the root of the problem, because this gives him the ability to reach a wide range of problem participants, which is a necessity for an effective deployment of a policy across the issue-causing network. Section 1.2.2 will explain why a policy deployment is the most feasible option to intervene in a complex network. For now it needs to be understood that the problem owner needs to be capable of deploying such a policy. Following this logic, choosing e.g. the automotive industry as a problem owner would be rather ineffective, as only one of the actors seeks to buy a new car. Actor 4 does not possess sufficient range to deploy a policy. Note however, that policies can at times aim at building interactions between certain actors. In this case that could e.g. be to create a demand for new high-technology cars.

Policies are not always but usually introduced by a governmental entity. This is, because governments have access to a variety of channels, which enable them to reach such a broad range of problem participants. And indeed from all involved actors, only the government can potentially impact the described traffic network as a whole, given the introduced setup. That is, because the physical playing field on which those issues have their roots are the roads and streets across the Netherlands - the basis of the traffic network. Furthermore, the government has the ability to deploy policies, which as a side-effect create new interactions across network borders, such as the demand for newer cars. Both effects are potentially valuable in approaching escalating congestion and emissions on Dutch roads. Henceforth, it is understood as a logical consequence that the ownership of the introduced physical playing field is one of the keys to invoking these effects.

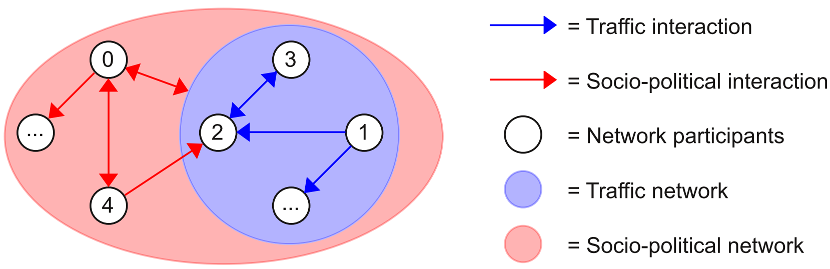


Figure 1.3: Illustration of a socio-political and a traffic network.

The Dutch road network is owned by a set of different governmental bodies. These are either public authorities such as the*Rijkswaterstaat* or one of the provinces of the Netherlands. In this context, it is assumed that all these parties are in some way concerned either with emission issues, congestion issues or both. Besides the central network position of this problem owner, the willingness to tackle the problem at hand is a logical necessity. For governmental bodies this willingness can be assumed, which is why they are consolidated under the generic term 'Government'. It is found that the government brings along both, the willingness to tackle the problems and the network position to do so. Hence, a conjoint 'Government' point of view is adapted. Nevertheless, it is understood that there is a certain heterogeneity in the way the governmental entities address congestion and emission issues and what level of importance they assign to them. This is therefore revisited in Chapter 6, where a heterogeneous policy-development framework is introduced based on the individual values of the governmental bodies at hand.

Having understood why the problem owner of choice is the Dutch government and that it is generally in favor of all measures which potentially decrease emission and congestion issues, a generic policy-level research question can be defined. Given the scope of the thesis, this solely considers the introduction of urban platooning:

Under which conditions can a policy be designed that incentivizes urban platooning, which itself pursues the goal of saving on congestion and emissions without road expansion and how can such a policy be implemented?

In this sense, it has once again to be stressed that the research at hand does not aim at an actual policy design. It solely attempts to answer the question if a policy as introduced is possible and if yes, how it can be implemented. By adapting this problem scope, it leaves the policy design space open and rather aims at identifying and quantifying the so-called "resistance threshold" (see e.g. Sohoni et al., 2011), which this policy needs to overcome. In order to fully grasp the idea of a resistance threshold , one has to first understand that a policy is only needed if there is a certain natural resistance against a change, which is in this case desired by the government. That is, a certain stakeholder or stakeholder group exists, which hinders a desired measure, because it is not worthwhile for that group. Or in other words: If a measure was worthwhile for a certain stakeholder, it would not frustrate it and a policy would hence be obsolete. In this sense, the resistance threshold attempts to quantify the resistance of a certain stakeholder group against the action, which that group would need to make to fulfill the policy goal of the government.

The research question centers around a design, which "incentivizes the use of smart traffic solutions". This implies that the stakeholder group, which is subject of this policy consideration are the traffic participants. The next section will further elaborate on this choice. For now it needs to be understood that although the traffic participants represent only one of a variety of relevant stakeholders, they represent the center of such a potential policy consideration. Additionally keeping in mind that no actual policy design is attempted, this thesis therefore solely aims at quantifying the resistance of the traffic participants against the mentioned smart traffic solution. This is essentially different from attempting an actual policy design. The contribution of this work should therefore only be seen as a basis upon which a policy can be designed to nudge the traffic participants towards this desired scenario. The high-level policy deliverable, a policy-development framework will eventually provide this basis and assist the government in such a policy design. As mentioned before, this deliverable is presented in Chapter 6.

Furthermore, it was mentioned that this framework is based on a resistance threshold. In the presented context this threshold asks the question of how much compensation is necessary in order to make the alternative scenario more desirable than the current situation for the traffic participants (Gibbs et al., 2009). Keeping in mind that the left arm of the earlier introduced V-model is constituted by a set of high- and low-level objectives, the introduced research question can therefore be translated to a policy objective, which the next lower level will attempt to answer. Following the V-model approach, this objective is imposed through the policy level and needs to be answered through the game theory level. It is therefore formulated as such that it contributes to the solution of the high-level policy research question through a research demand to the game theory level. Given the above-introduced line of reasoning, It is concerned with finding the introduced resistance threshold:

Quantify the resistance of the traffic participants towards an introduction of smart traffic solutions, which lower emissions and congestions without requiring road expansion.

In the course of presenting the high-level deliverable, Chapter 6 goes into detail how above objective can be used to answer the policy-level research question. For now it needs to be understood that by quantifying the resistance threshold a basis is established upon which a policy development framework can be build.

1.2.2. LEVEL 2: GAME THEORETIC PROBLEM FORMULATION

Before formulating a game theoretic research problem, which is capable of satisfying the above-introduced policy objective, it first needs to be understood why game theory is employed for this in the first place. Game theory provides a thinking structure which can be used in order to model complex socio-technical problems. It is commonly defined as "the study of mathematical models of conflict and cooperation between intelligent rational decision-makers." (Roger, 1991). What Roger implies when referring to "conflict and cooperation between [...] decision-makers" is the existence of divergent interest, concerned with a potential change. Given the context of the problem at hand, this can be understood as follows: Urban platooning is a traffic solution, which can not be brought in place through the government alone. This player therefore needs to cooperate with another player, namely the traffic participants, in order to deploy the desired scenario. Game theory is a way of mathematically modeling this setup. It is a helpful tool that can be used to map the relations between the players and their interests and communicate the trade-offs, which the government and the traffic participants are faced with. By doing so, it sets the basis for a potential win-win solution. As implied through the policy objective, such a win-win situation can be reached if the earlier mentioned resistance threshold is overcome through an effective policy design. Game theory is the tool that will be used to tackle this objective.

Now in order to convert this policy objective into a scientific, tangible game, it is helpful to adapt a 'Complex systems management'-notion. This paragraph illustrates how the congestion and emission issues at hand essentially represents a specific case of this. As mentioned, the ownership of the Dutch traffic system represents a strategically valuable network position for political, but also game theoretic considerations. However, the traffic system itself is a very complex one, which makes it especially difficult to take the right decisions to fulfill the policy objective, as stated above. This is mainly due to the large amount of problem participants (here: the traffic participants) and their interconnectivity. Easley & Kleinberg describe such high-connectivity systems as 'complex networks' (Easley and Kleinberg, 2010). In the opening chapter of their book, they distinguish two underlying aspects of every complex network - behavior and structure. They elaborate that the dynamics, hence the performance of a network are based on the behavior of its individuals and on the connectivity between these individuals. The latter is what is comprised under the term structure.

Network structure determines which individuals are connected and describes the form of their connection. In other words: It defines whose "individual's behavior have implicit consequences for the outcomes of which other individuals in the same system" (Easley and Kleinberg, 2010, p. 4). An individual that is not connected to another can not have any influence on that other. If two individuals are connected then the structure determines how an action of A has an impact on B. In the context of traffic systems this could e.g. be: *Which other cars are affected through a braking motion of mine and how are they effected?* In Figure 1.3, examplary structural connections in the traffic system are illustrated as blue arrows. Clearly, these arrows depend on the current traffic situation with more traffic leading to higher congestion and emission issues (In Section 1.5 the physical scope is consequently set to urban roads with generally high connectivity). The denser the traffic the more interaction structures arise.

Following the same notion, Easley & Kleinberg's term behavior is the driving behavior of the traffic participants. In Figure 1.3, these are comprised in each of the participants of the traffic system. According to their theory, it is then the conjunction of behavior and structure that determines the performance of the system, concerned with any given goal. For the case at hand two goals were formulated within the policy objective, namely the emission and congestion within the network. Clearly, the individual's driving behavior and the influence of that behavior on other traffic participants influences the emissions and congestion in the network. it can therefore be concluded that a traffic system fits the frame of 'Complex systems' as introduced by Easley & Kleinberg very well. The management of a traffic system towards one or both of those goals is indeed a case of complex systems management and consequently complex systems management measures can be applied.

Within the theory of complex systems management several management methods are introduced to cope with the high complexity of networks. Depending on the level of complexity more or less direct measures are proposed, whereas simple networks should be tackled with direct and complex systems with indirect measures. Easley & Kleinberg motivate this with the unique network position of the problem owner and its limited set of available regulation choices. With the problem owner usually being at the root of the problem it is likely that any direct intervention in the system will lead to forced changes in its structure. This explicitly does not consider the specific structures (i.e. the interaction with other cars) but the general amount and level of connectivity of each individual . An example for that would be if the government was to implement lower speed limits to save on emissions. Such an intervention would likely lead to unforeseen compensation effects, i.e. drivers maintaining shorter safety gaps between vehicles. This would lead to a higher and tighter connectivity of the system, which would allow congestion problems to amplify throughout the network. Although such a modification to the structure of a system is not necessarily always undesired and can in some cases even have positive effects, it is very difficult to predict its effects. Consequently, it is stated in various literature that systems which do not allow for an evaluation through implementation (i.e. understand the effects upon implementation), should be managed with indirect measures (Li et al., 2017). For the case at hand, an implementation without prior impact assessment is no alternative, which is why the choice for an indirect measure is the logically favorable option.

As mentioned earlier, the regulation method of choice is the deployment of a policy that effectively lowers congestion and emissions. This is due to the fact that firstly, it is in the governments range of doability to deploy a policy and secondly, a policy represents a prime example of an indirect measure. One of the main characteristics of an indirect measure is that all participants have a choice to abide by that measure or not. In most cases a policy adopts a very similar approach. Rather than building upon laws or rules, policies do often rely on incentives. The benefit of this is that no unnatural behavior is imposed on the problem participants through vastly changed structures. They are rather nudged into a certain direction, whereas they can still maintain their usual behavior if wanted. In other words: No changes to their behavior are imposed by force. In contrast to direct measures, indirect measures subtly affect the whole network. In the context of complex systems, it is widely agreed upon that this makes them more likely to succeed, as they tend to maintain similar structures as the initial system. This is because such measures do ideally affect all participants equally. The big benefit of this is that it is therefore possible to model the impacts of indirect measures and evaluate their outcomes. In Section 1.2.3 a modeling choice for such an evaluation is introduced and motivated. Firstly however, the general to-be-evaluated policy options of the government will be introduced.

Due to above reasoning, it is at this point agreed that only indirect policies, hence measures that leave the traffic participants a choice are considered. Furthermore, it was already mentioned that platooning and intersection control represent two potential smart traffic solutions. Given the topic and scope of this thesis the subsequent choice of policies is limited to either support or not support those solutions. In Section 1.1 both technologies are properly introduced. For now it only needs to be understood that the government essentially has two available regulation measures. From those two, only the intersection controllers can be effectively implemented through the government. To support and adapt platooning on the other hand is solely in the hands of the traffic participants. Yet, both systems have to be adapted in order to establish urban platooning, as it is researched in this thesis.

In a game theoretic sense, the government therefore only has the choice to 'Do invest' or 'Do not invest' in traffic controllers, which allow for platooning on intersections and a general traffic optimization. However, the effect of this measure largely depends on whether the traffic participants actually partake in platooning. That is, whether they invest in a vehicle that is capable of platooning themselves. From this a game theoretic payoff-matrix can be generated, as illustrated in Table 1.3.

Table 1.3: Potential moves of the government and the traffic participants in the Platooning-Intersection control game.

		Government Y	
		Do invest I	Do not invest $\bar{I}$
Traffic participants X	Do partake P	$x_{P,I}, y_{P,I}$	$x_{P,\bar{I}}, y_{P,\bar{I}}$
	Do not partake $\bar{P}$	$x_{\bar{P},I}, y_{\bar{P},I}$	$x_{\bar{P},\bar{I}}, y_{\bar{P},\bar{I}}$

By now it should be clear that game theory represents a valuable tool that can be used to model the relation between the government and the traffic participants. A large aspect of this modeling process is to numerically determine the payoffs of the proposed game. Yet, before this table is further introduced, a few comments need to be made about the game at hand and its characteristics:

- *Choice of players*: Clearly there are more actors related to the introduction of platooning and intersection control than what is displayed in this table. Yet, game theory is a modeling technique and modeling techniques usually try to translate real-world problems into simplified, understandable representations of reality. The level of detail in modeling largely depends on the scope of the research, whereas a model should always first focus on the main aspects of a problem. Hence, the choice of players is made because those two actors possess the longest lever for the integration of these smart traffic solutions and not because they are the only ones. This contribution can very well be supplemented by further game theoretic considerations, including other actors. This is one of the points that is recommended for further research.
- *Choice of possible moves*: Furthermore, the players do in real-life have more moves than those that are

possible within this game. Yet again, a modeling boundary needs to be found whereas high attention is put on incorporating the most important moves of the involved actors.Iit is out of the scope of this thesis to e.g. include intermediate decisions (i.e. half of the traffic participants partake in platooning). The aim of this game theory consideration is hence not to analyze the heterogeneity of the players and their intrinsic values, but rather to explore the possibilities of what can be possible under complete participation and complete commitment.

- *Game classification*: Although, the government will try to make this a coordinated game (i.e. facilitate a conjunct strategy), it is assumed that this is not entirely possible, neither desirable. It is rather endeavored that the government tries to manipulate the payoffs as such that the only logical choice for the traffic participants is to partake. This is due to the chosen policy approach. As it was shown an indirect policy is the most favorable choice for the given setup. In game theoretic terms such a policy can be deployed by creating such a situation, which indirectly steers the other players decision. This idea is further elaborated on in the latter paragraphs of this section.

Altogether, above game features four different simulation scenarios. The P, I-scenario is the one, which represent an introduction of urban platooning, and the $\bar{P}, \bar{I}$- scenario represents the base case. Before further elaborations are made, some remarks need to be made on the base case of choice. For this scenario it is assumed that a so-called model-predictive (Van Katwijk, 2008) intersection algorithm is applied. To a certain extent, this algorithm already optimizes the performance of the intersection. Without going into too much detail, the reader should however be made aware that this is the best solution, which the government can establish without facing necessary investments. Although, it is found reasonable to take this as the base case, this however means that even the base case performance is already to a certain extent optimized.

Clearly, the introduced game represents a strongly simplified model of the real-world problem. For the following research this is both a blessing and a curse. On one hand, the simplicity and the mathematical nature of the established game allows to draw concrete implications, which can then be used for a potential policy design. On the other hand however, the above mentioned assumptions diminish the representativeness of the model to a certain extent. This is an important aspect, which will be revisited in Chapter 6 and taken into account when establishing a policy-development framework.

Now, given above considerations, the payoffs x and y for different combinations of moves $M \in \{P, \bar{P}, I, \bar{I}\}$ can be formulated in dependence of a set of variables. This can generically be expressed as:

$$
\begin{aligned}
x &= w_{x,C}C + w_{x,E}E - V_x \\
y &= w_{y,C}C + w_{y,E}E - V_y,
\end{aligned}
\tag{1.1}
$$

whereas C and E represent the congestion and emission savings respectively and w represents the weight a player gives to those savings. V represents the investments that either the government or the traffic participant needs to make for a certain move. Following from this a few statements can be made, concerning the values of those variables. These are:

$$
\begin{aligned}
V_{y,\bar{I}} &= 0 \\
V_{x,\bar{P}} &= 0 \\
C_{\bar{P},\bar{I}} &= 0 \\
E_{\bar{P},\bar{I}} &= 0.
\end{aligned}
\tag{1.2}
$$

The reasoning behind these values is as follows. If a player does decide to not invest or to not partake, this is the same as not making any move. It therefore represents the status quo for the investments of that player. Hence, the investment is zero. The same goes for the emission and congestion savings, if both players decide to not invest. In this case, the network remains in the current status, which leads to no changes in emissions and congestions but also no investments. When inserting this into (1.1), the eight payoff values from Table 1.3can be expressed as:

$$
\begin{aligned}
x_{P,I} &= w_{x,C}C_{P,I} + w_{x,E}E_{P,I} - V_x \\
y_{P,I} &= w_{y,C}C_{P,I} + w_{y,E}E_{P,I} - V_y \\
x_{P,\bar{I}} &= w_{x,C}C_{P,\bar{I}} + w_{x,E}E_{P,\bar{I}} - V_x \\
y_{P,\bar{I}} &= w_{y,C}C_{P,\bar{I}} + w_{y,E}E_{P,\bar{I}} \\
x_{\bar{P},I} &= w_{x,C}C_{\bar{P},I} + w_{x,E}E_{\bar{P},I} \\
y_{\bar{P},I} &= w_{y,C}C_{\bar{P},I} + w_{y,E}E_{\bar{P},I} - V_y \\
x_{\bar{P},\bar{I}} &= 0 \\
y_{\bar{P},\bar{I}} &= 0.
\end{aligned}
\tag{1.3}
$$

Now that a generic game is established between the two players X and Y some considerations about the desirable outcome of this game can be made. One major aspect of game theory is to find a so-called 'Nash-equilibrium' to a given payoff matrix. A Nash-equilibrium represents the combination of moves at which no player can be better off by changing his move (for further explanation see Easley and Kleinberg, 2010). This is especially relevant for the game at hand, as the government does not know which of their options P and $\bar{P}$ the traffic participants are going to choose yet. Except from the status quo-situation no impact can be sufficiently assessed, neither for the government nor for the traffic participants. However, as it was mentioned earlier, it is assumed that the government generally supports both technologies. Given this assumption, it becomes clear that $y_{P,I} \geq 0$. Essentially, this means nothing more than that the government would be better off if platooning and intersection control were both adopted. In a game theoretic sense however, this has some implications for the remaining payoff values.

In Chapter 6 a more thorough explanation of game theory and Nash-equilibria is provided. For now it needs to be understood that the government will want to create a Nash-equilibrium in the upper-left corner of the payoff matrix to make this the inevitable choice for the traffic participants. This would mean that the traffic participants would want to partake in platooning technologies, while the government itself will invest in the respective intersection controllers, which would allow both parties to benefit from potential synergy effects between the technologies. If necessary, this can be reached by incentivizing the traffic participants to 'partake'. In mathematical terms this implies the addition of a compensation term Z to the payoffs in the upper row of Table 1.3. As it can be seen in (1.4), the payoffs were supplemented by such a compensation term, whereas this is naturally positive for the beneficiaries of this compensation and negative for the government, which has to finance this measure. Note however, that the compensation itself does neither actually have to be of financial terms, nor does it need to cost the government the same amount, as it benefits the traffic participants. This is due to the earlier introduced nature of a policy.

When remembering that it is the objective of the high-level policy considerations to quantify the resistance threshold of the traffic participants it can now be understood that the previously introduced compensation term does indeed represent this threshold. It was earlier mentioned that the threshold poses the question of how much compensation is necessary in order to nudge the subjected stakeholder towards a desirable action. This action was found to be the traffic participants partaking in platooning and its threshold is therefore constituted by Z_x.

$$
\begin{aligned}
x_{P,I} &= w_{x,C}C_{P,I} + w_{x,E}E_{P,I} - V_x + Z_x \\
y_{P,I} &= w_{y,C}C_{P,I} + w_{y,E}E_{P,I} - V_y - Z_y \\
x_{P,\bar{I}} &= w_{x,C}C_{P,\bar{I}} + w_{x,E}E_{P,\bar{I}} - V_x + Z_x \\
y_{P,\bar{I}} &= w_{y,C}C_{P,\bar{I}} + w_{y,E}E_{P,\bar{I}} - Z_y
\end{aligned}
\tag{1.4}
$$

In order to determine whether a compensation is necessary and what extent that compensation has, more knowledge is required. To establish a policy, which does indeed create such a Nash-equilibrium, the eight payoffs need to be filled with explicit figures, rather than with mathematical functions. Hence, knowledge needs to be gained about the remaining variables from (1.3). Here, the weighing factors $w_{y,C}$ and $w_{y,E}$ represent the political attitude of the individual governmental bodies towards emission and congestion issues.

Similar to the intrinsic values of the traffic participants $w_{x,C}$ and $w_{x,E}$, these attitudes are for now kept variable. This allows for a heterogenous policy-development framework in the concluding chapters, customized on the individual weighting values of the governmental entity at hand. Given this, what is missing are the emission and congestion savings E and C for every scenario. These represent the impacts of platooning and intersection control, as well as the impact of a functional combination of those technologies on the current traffic network. The policy research question can therefore be refined into a game theoretic research question:

Can a Nash-equilibrium be triggered for the P, I-scenario in the platooning-intersection control game?

Note, that the policy research question further asks how a successful policy can be found. Yet, this aspect can not be found in the game theoretic research question. The reason for this is that this part of the research question will be answered by the policy deliverable - a policy-development framework. This framework however, is based on to-be-found knowledge on the afore-mentioned resistance threshold. As described in this section, this threshold can be determined by applying game theory. More precisely, it is the deliverable of the second-level game theoretic research effort. The introduced game theoretic research question allows for direct conclusions about this threshold and therefore for fulfilling the policy objective. Yet, it was also mentioned that in order to answer the game theory research question further, lower-level research about the emission and congestion savings E and C for every scenario is necessary. For this reason, the game theoretic objective that is imposed on the next lower level is:

Determine the emission and congestion savings E and C for every simulation scenario.

1.2.3. LEVEL 3: SIMULATION PROBLEM FORMULATION

As mentioned generating a Nash-equilibrium requires knowledge about the emission and congestion savings E and C for all four scenarios. Furthermore, it was mentioned earlier, that the choice for an indirect system intervention through a policy allows for modeling the impacts of various design choices, as long as these do not force a behavior on the individuals of the traffic system. It will later be shown that for neither of the scenarios this is the case. For now it will be explained why simulation is the modeling method of choice and which simulation software is chosen in order to gain insights on the explicit, numerical values of E and C. For this choice a few requirement can be formulated that need to be satisfied in order to allow for realistic conclusions about the game in Table 1.3. These are:

1. *Scenario implementation*: The evaluation method needs to allow for the integration/deplyoment of platooning and intersection controllers, as well as for a functional combination of those.
2. *Scenario accuracy*: A highly-capable and detailed method needs to be used to deliver quantitatively accurate scenario analyses.
3. *Scenario representativeness*: The scenarios needs to represent real-life matters.
4. *Impact accuracy*: These scenarios further need to be translatable into realistic emission and congestion values.
5. *Scientific standard*: The evaluation method should meet scientific standards and should ideally be widely accepted in the field of traffic management.
6. *Doability*: The evaluation method has to be feasible and doable within the frame of a master thesis.

The first requirement can technically be met with a variety of methods. It is the second in conjunction with the last point, which makes simulation the most logical choice. Clearly, real-life tests are no option within the given scope of this research. An affordable, yet accurate alternative needs to be found. Especially in the context of traffic systems, simulation has proven to deliver high-quality results. Furthermore, within the domain of traffic simulation a choice favoring microscopic simulation is the most reasonable, as those generally possess higher capabilities in terms of accuracy. The simulation software VISSIM is chosen from this subdomain, as it is the best fit for the remaining requirements. It allows for an implementation of all four simulation scenarios and it is a proven software, which is widely adopted in the field of traffic simulation. Furthermore, for the purpose of emission evaluation the EnViver-software is used, which allows for an accurate and realistic translation into emission values, calibrated based on real-world measurements. Lastly, in order to represent real-life matters a case study is chosen to represent an actual traffic setup. The traffic setup of choice is the N260 - Burgmeester Letschertweg in Tilburg, The Netherlands. The subsequent simulation research question is:

Which congestion and emission savings can be reached within each of the four scenarios of the N260-case study?

In the course of this research, especially in Chapter 5, this question is answered through a so-called simulation-based optimization (see e.g. Carson and Maria, 1997). Simulation-based optimization is a subdomain of simulation and incorporates the element of optimization by iteratively changing a set of system parameters in order to enhance system performance. Or applied on the case at hand: The traffic situation on the N260 is taken and mathematically modeled using VISSIM. This model can then be used in order to understand how it performs under different technological setups. These technological setups are the parameters, which are varied. Given the case at hand, they are the simulation scenarios from Table 1.3. Through evaluating the performance of each scenario, knowledge can be gained on how traffic behaves, given these different setups, which eventually allows for insights in the potential congestion and emission savings C and E and therefore holds the capability of satisfying the game theory objective.

A large branch of literature exists, illustrating and motivating the feasibility of simulation optimization in order to model traffic engineering problems (see e.g. Herty and Klar, 2003; Wiering et al., 2004). One aspect that almost all literature agrees upon is that one major strength of simulation-based optimization is the capability to accurately assess different network interventions, despite the high complexity of a network, especially if these measures only intervene in the behavior of the network participants and not in the network structure. In the past sections, it was thoroughly explained why the traffic system at hand is a complex one and that all proposed measures attempt to modify the behavior rather than the structure. Given this it can be understood that simulation optimization using VISSIM (and simulation evaluation using EnViver) is the most logical choice of methodology. Yet, VISSIM does not offer an ad hoc solution to implement all of the proposed interventions. Namely, it does not provide the functionality of platooning, neither of a functional combination of platooning with intersection control. What it does offer however, is an application programming interface (API), which can be used to expand the simulation capabilities of VISSIM with the functionalities of these smart traffic solution. Hence, what needs to be done is to implement the functionality of both these technologies through the API. From this two final objectives can be imposed on the lowest level - the technical design. Namely these are:

1. *Model a platooning controller that realistically represents the driving behavior of real-world platoons.*

2. *Develop functional combination of the modeled platooning controller and an intersection controller, which aims at minimizing congestion and emission issues.*

In particular, this thesis will consider a JUNO deployment on all three intersections of the N260 case study as the lower-left $(I, \bar{P})$ scenario in the game consideration from Table 1.3. In Chapters 2 to 4, it will further be explained how JUNO and the introduced model-predictive control serve as a benchmarks for the calculation of congestion and emission values C and E of the scenario at hand. Firstly however, the last level of this four-layered problem formulation is introduced, preparing the reader for those main research contributions in Chapters 2 to 4.

1.2.4. LEVEL 4: TECHNICAL PROBLEM FORMULATION

Given above-described simulation setup, the scenario in the lower-right corner of Table 1.3 represents the status quo and will therefore serve as a traffic benchmark on the N260. The lower-left scenario represents the introduction of intersection control only. For this, as well as for the benchmark scenario, no technical design will be needed, hence the direct arrow between 'Simulation objective and 'Simulation results'. A solution to that part of the game theoretic objective can be obtained without requiring a low-level technical design. This is, because the to-be-evaluated intersection control algorithm represents an available resource at TNO. The remaining two scenarios however, do require a technical design, whereas the upper-left corner indicates the technical main contribution of this thesis. Yet, from above simulation objectives it can be seen that filling the knowledge gap is based on the development of a platooning model. Hence, in a first approach a technical research question can be formulated that targets the said development of a platooning controller:

How can a group of multiple autonomous vehicles be navigated and actuated as such that they self-reliantly form a string of vehicles, which moves through traffic as one entity, by means of a platooning controller?

Strictly seen, this development of a platooning model is a preparatory step for achieving the major goal of this thesis - developing a functional combination of platooning and intersection control. As mentioned earlier, that is, because the functional combination of the two concomitant systems is based on said platooning model. Logically, one major objective for the design of a platooning model is therefore to maximize the level of representativeness. Within Table 1.4 this is divided into the representativeness of the functionality and the characteristics of platooning. Especially the latter however requires prior literature review, as it first has to be defined what these characteristics constitute. Further information on this is given in 1.4, where a research framework is introduced, which besides others aims to fulfill these design objectives.

Given that above technical research question is answered, the main contribution of this research can be tackled, namely filling the research gap that was outlined in Figure 1.1. In this figure it is illustrated that filling this gap can be reached through a meaningful combination of the two concomitant systems. Considering this, the above simulation objectives and the objective to save on congestion and emission values C and E, the following second technical research question can be deducted:

How can a platoon be navigated through a string of intersections through a combination with intersection control while allowing for a performance optimization of the traffic network in terms of congestion and emissions?

Having understood that the knowledge gap, which this research question attempts to fill is embedded in an overarching policy research question, further conclusions can be deducted about the to-be-designed low-level solution. Namely, this research question has been imposed in a top down approach through three different layers. It is characteristic for such a cascaded problem formulation that the higher levels impose certain design requirements or objectives to the lowest level. Within the beginning of this section the two first design objectives that are targeted at the platooning model have already been introduced. In the following, the same is done for the second technical design as well as for requirements and objectives that hold for both designs. All requirements and objectives are summarized in Table 1.4 at the end of this section.

To start off with, it is hardly contestable that although the government is assumed to pursue an implementation of urban platooning, it will still want to minimize the costs of the policy which will be proposed to do so. One essential part of the costs of the policy are those costs that are directly imposed through e.g. a technical design or its installation. Hence, from the desire to minimize the policy costs, an objective to minimize the necessary implementation costs can be deducted.

Furthermore, within this thesis, the goal of 'filling' the knowledge gap is taken literally, in that sense that the deliverables should minimize their intervention in the two existing systems. As it was mentioned earlier, platooning and intersection control systems are well researched. In this regard, their underlying functionality should not be modified. Rather will the to-be-developed platooning-intersection controller accept these proven concepts as a given, which should be changed as little as possible, while still ensuring a good combined functionality. It will fill the knowledge gap that is allocated between these technologies without redesigning either of the concomitant systems. Hence, the objective to minimize intervention in the concomitant systems in Table 1.4. Besides others, this is motivated in the ihg-level need for an easy implementation and high compatibility.

As mentioned, a major benefit of urban platooning is that it can potentially be a door opener for highway platooning, allowing to harvest more benefits of highway platooning by effectively extending its range. This however, only holds truth under the condition that the urban platooning, preceding or following a highway section does not infer negative effects. From this another two design requirements can be formulated. Firstly, this is, that the urban platooning solution must ensure the safe navigation of a platoon of vehicles through a string of intersections. Secondly, the solution must maintain or decrease the overall congestions and emission values C and E in the urban traffic situation.

Effectively, this design requirement is a must-have for the technical solution to be worthwhile. Its underlying message is: If a platoon plans to enter an urban traffic environment, intersections will not be the problem to keep it from doing so. Up until now, the sole motivation for a platoon to do so is to harvest further benefits in the highway environment. Yet, the technical design should ideally lead to urban benefits as well. From this a follow-up design objective can be formulated as: The urban platooning solution should enable congestion and emission savings.

Additionally, a design requirement can be deducted from the game theory level. Section 1.2.2 and especially the actor behavior from Table 1.3 infer a few assumptions. A major one of those is that the traffic participants are homogenous enough to allow to model their decision cumulatively. Although this assumption can be justified by the fact that game theory is a modeling tool, which maps complex problems in a simplistic, yet reasonably realistic way, this homogeneity should at least not be further driven apart through the integration of urban platooning. By using complex systems notions, this is in the game theoretic section theoretically described as "the intervention should not modify the network structure, but rather the behavior of the traffic participants". For the technical design this mainly infers that all traffic participants should benefit equally much from the new technology. Hence, it is a design requirement that the new system must not lead to distributive effects, whereas certain traffic participants are statistically profiting more than others. However, it is understood that no perfect equality can be reached. In this sense, the requirement is considered reached, if no traffic participant is made worse off than before.

Table 1.4: Design requirements for the technical design level.

Technical design	Requirement/ Objective	Design requirement or objective
Both	Requirement	Both technical designs need to be compatible with VISSIM through the VISSIM API.
Platooning controller	Objective	Maximize the representativeness of real-world platooning in terms of functionality.
	Objective	Maximize the representativeness of real-world platooning in terms of platooning characteristics.
Platooning-intersection controller	Requirement	The urban platooning solution must ensure the safe navigation of a platoon of vehicles through a string of intersections.
	Requirement	The solution must maintain or decrease the overall congestions and emission values C and E in the urban traffic situation.
	Requirement	The solution must not allow for distributive effects, where certain traffic participants benefit more from the new technology than others.
	Objective	Minimize the intervention in the concomitant systems that become necessary when implementing the proposed platooning-intersection controller.
	Objective	The urban platooning solution should enable congestion and emission savings.
	Objective	Minimize the investment costs that are necessary to establish urban platooning.

The above-presented set of design requirements and objectives in conjunction with the two introduced research questions conclude the problem formulation. The following sections centre around the deliverables that are proposed to answer the presented problem formulation.

1.3. DELIVERABLES

Throughout the preceding sections, four levels of problem formulation were introduced and through their interrelatedness, the reader should have by now understood that the deliverables of the lower levels constitute a certain knowledge that is vital to those of the higher levels. Given this understanding, the reader has probably also obtained a rough idea of what the deliverables of each level are. In Figure 1.4, these are specifically named and their relation is illustrated. In the following some final remarks are made on these deliverables and it is shown that their relation originates from the right arm of the V-model.

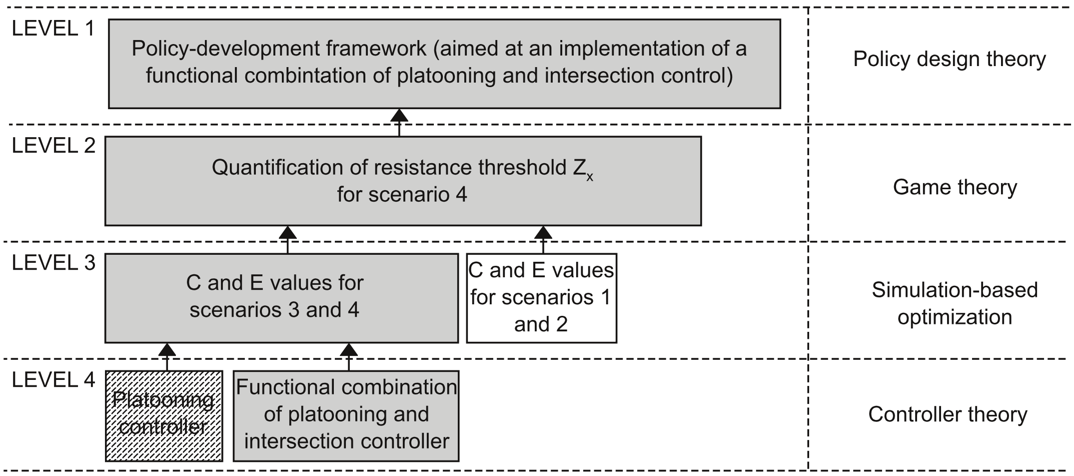


Figure 1.4: Illustration of the deliverables of every level and their relation. Deliverables in gray represent a contribution of this work and those in white represent available resources. Although the development of platooning controller is part of the scope of this thesis it does not represent a contribution in the sense of being new. Its development is rather based in various literature on controller theory in platoons. On the right, the most prominent tool for every research level is illustrated.

Until now, this chapter has mainly centered around the left arm of the V-model from Figure 1.2, guiding the reader from the highest to the lowest level - the technical design. It has been shown how the governmental desire to decrease congestion and emissions can be translated into a policy research question. Henceforth using the V-model approach, this policy research question was broken down over four different levels until finally a technical design was outlined which can help to conquer the high-level problems. Eventually, this design was contoured by means of two technical research questions and a set of design requirements and objectives. Alltogether, it is this package of design specifications which constitutes the lower-left box of the V-model. Before going into detail with how these specifications are brought in place, some final explanatory paragraphs about the right arm of this model are provided. This is mainly concerned with the question of how the technical design will eventually contribute to the formulation of a policy. For this a good understanding of the structured relation between the four deliverables from Figure 1.4 is vital.

That is, the overarching goal of this thesis is to decrease congestion and emission problems. Within this thesis' scope it is analyzed whether a potential policy deployment can trigger this change. In an attempt to design such a policy, it was shown how the policy can be formulated as part of a game, whereas an effective policy would guide involved players towards a desirable combination of moves by means of a so-called compensation policy. In order to determine its necessity and quantify this compensation, knowledge needs to be gained about the potential payoffs for the two players of the game in various scenarios. A mathematical dependence between these payoffs and the congestion and emission savings of the scenarios implies that quantified knowledge need to be gathered about these values. This is done by means of a traffic simulation of the N260, using the VISSIM- and EnViver-software. However, before these simulations can be executed and the associated congestion and emission savings can be quantified, two technical designs are necessary. This is firstly in order to model platooning within the VISSIM environment and secondly to model the functional combination between platooning and intersection control. The technical designs can then be used as part of the simulation, providing knowledge about the congestion- and emission-wise impacts of each scenario.

The right arm of the V-model can be understood in converse manner, whereas the preceding step to any of the boxes does in this case not infer a requirement, but rather a solution input for the following step. This is illustrated in Figure 1.4. More precisely, the figure illustrates the deliverables of each level and how they contribute to the next higher level. However, it should be noted that not only the eventual policy advice is considered a deliverable. Every step provides its own contribution in terms of gained knowledge, concerned with its respective research question. Especially the technical design of a combined platooning-intersection controller in conjunction with its simulation impact assessment is considered a main research contribution. This is firstly due to the fact that a major amount of time is spend on the lowest level of the V-model. Secondly, this is because this deliverable represents the answer to a technical knowledge gap, which can be identified within the interaction of platooning and intersection controllers. Subsequently, the following section proposes a research approach that can be used to further structure and tackle the lower two levels of above problem formulation.

1.4. RESEARCH APPROACH AND FRAMEWORK

At the end of this thesis a functional combination of platooning and intersection control is presented, which eventually fills the technical knowledge gap that was introduced in Section 1.1.3. Accordingly, in this section a brief outlook is provided, outlining the chosen research approach, which leads to this outcome. This is solely concerned with the third and fourth level of the introduced V-model. The reason for only subjecting the lower two levels of the problem formulation to a structured research approach is that they represent the widest part of the overall research effort and the main scope of this thesis.

Concluding the approach section, two frameworks are drawn up. These are, firstly, a controller evaluation framework and secondly, the actual research framework, which will guide the reader through the conducted research and towards the final deliverables. For the former, three mathematical objective functions are found, originating from the above-introduced design requirements.That is, in order to initiate a technical design and a subsequent simulation optimization approach on controller design it needs to be made clear how the objectives from Table 1.4 can be measured. It is vital to have one or several clearly understood mathematical objective functions.

In different literature different measurables, such as 'Intersection performance', 'Throughput' or 'Usage rate' are used to accurately measure the level of resulting 'Congestion'. In order to make this criteria comparable and effectively measurable it is in the following defined as the average traveling time of all vehicles in the network. As all four scenarios will be simulated under the exact same conditions (same traffic demand, same intrinsic driving behavior, same road setup), this represents a sufficient way of comparison. Similar to the emissions in gCO_2, it can be mathematically optimized and is therefore feasible for the purpose at hand. Together with the necessary investments from a governmental point of view, these terms represent the values for C, E and V in the formulated game. The three mathematical objectives, originating from the three objectives of for platooning-intersection controller in Table 1.4, are:

1. Average travel time (minimize) [s]
2. Emissions (minimize) [g CO2 / km]
3. Necessary investments (minimize) [€]

Hence, the proposed solution for urban platooning comprises a functional combination of a platooning and an intersection controller which is adapted to the (partly pre-designed) interfaces, while pursuing a set of design requirements and objectives, whereas the objectives pursue an optimization of the three presented mathematical objective functions.

The evaluation framework is based on thesee three objective functions. On a closer look, one will notice that the three objectives do not necessarily complement each other. Depending on how a certain solution (i.e. a controller) is implemented, it might favor one objective function over another. Different ways of implementation, such as the variation of decision rules within the algorithm, lead to different performances concerned with the objectives. Hence, a standardized means of comparison is required. Within this report the controller performances are evaluated through a three-dimensional evaluation framework, as illustrated in Figure 1.5. More details on the dimensional scaling of this illustration are given during the evaluation of the combined controller design. At this point, it needs to be noted that the evaluation framework assumes a governmental point of view. Hence, the depicted costs represent the costs of the government. Furthermore, these costs must not be confused with the investments from Section 1.2.2, which are solely associated with the financial aspects of a potential intersection controller implementation. In contrast to this, the costs presented in this framework furthermore consider the governmental effort that goes into a potential compensation to overcome the mentioned resistance threshold.

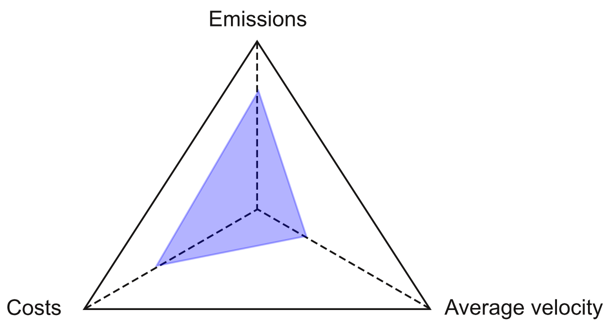


Figure 1.5: Example evaluation framework of a controller that favors the 'Investment' and the 'Emission' objective over the 'Average velocity'.

Now that it is understood how the design results will be made comparable, the following will briefly illustrate how this functionality is incorporated in the overall research process. Firstly however, a few general remarks need to be made on how this research attempts to satisfy the mentioned deliverable requirements:

- Firstly, the research will build upon the '(look-ahead) traffic-adaptive control' of the JUNO intersection controller. This is, it will study the control algorithm and modify it as such that it provides all necessary data that is needed to meet the deliverable requirements and answer the research question. This controller approach is already being researched at TNO, which means that the initial algorithm represent an available resource for this research.
- Secondly, this research is of a purely empirical-simulation nature. This means that the controllers are developed and tested on computer-simulation basis. For testing purposes, TNO provides a VISSIM simulation environment of the N260 (see Fellendorf, 1994, for a description of the program's capabilities and drawbacks). VISSIM is a microscopic traffic simulator (MTS), which has developed to a vast standard, used in various sectors in order to examine the impacts of road technologies on traffic flows (Detering, 2011). It is therefore considered feasible for this application. The simulations serve as a modifiable basis, which can be altered to gain further insights into the performance of the designed controllers. However, the eventual assessment of the combined controller will be done on basis of the N260 case study for the purpose of real-world relatedness and comparability.
- Eventually, this approach leads to a set of quantified, yet completely theoretical results. The designed controller will not be tested in a real-life environment within the context of this research. Yet, the report is concluded by a policy advice chapter, which attempts to draw a set of practical conclusions from these quantified results. Hence, while further validation will be needed to support the outcome of this report, it can at least deliver a first notion on the range of possibilities of platooning on urban roads. The thesis is in this sense considered as an exploratory research.

As previously mentioned, the solution to the given research question should not intervene in either of the internal functionalities of the systems. This being said, the combined controller will facilitate some kind of interaction, which allows the traffic light(s) to let the whole platoon pass through, while optimizing the performance of the traffic system. As depicted in Figure 1.6, it does so by connecting the available interfaces of the platooning and intersection system. While the outer boxes, representing the underlying functionality of platooning and intersection control, are not a subject to the thesis at hand, the definition of the interfaces partly is. Specifically, the technical specifications, which are ensuring compatibility between the platooning and intersection control systems with the deliverable, will be elaborated on. That is, these interfaces, a set of technical specifications, represent guidelines for the design space of the deliverable. They provide a boundary to the design space of the combined controller design, whereas it is the function of this boundary to restrain the deliverable from intervening in the concomitant system functionalities. As a logical consequence of this, the first step of the to-be-made research effort is to analyze and define these interfaces.

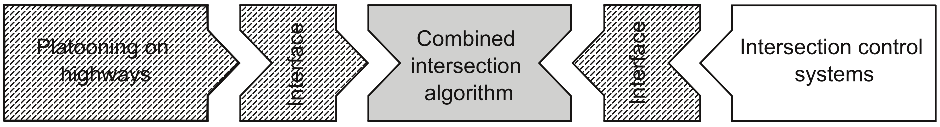


Figure 1.6: Illustrating a qualitatively filled know¬ledge gap (Gray represents a research contribution; shaded gray represents a research contribution, which is based on and therefore constrained by already established systems. These systems are either provided by TNO or based in literature study; white represents such already established systems).

Following this, the research process itself distinguishes between five main essential stages, which are initiated as soon as the mentioned interfaces are understood and established. This and the following steps are depicted in Figure 1.7 and briefly described in the following:

0. Before the actual research is conducted a profound literature study is executed, establishing a solid knowledge an all relevant technical aspects. This lays the basis for all further steps.

1. The first stage focuses on establishing a VISSIM test bed in which the controllers can be tested. In this step the existing VISSIM simulation environment of the N260 is modified as such that they allow for testing the final deliverable as well as the to-be-designed platooning controller in various situations. Furthermore, this stage involves understanding the design requirements for the platooning and the combined controller, which need to be respected during Stage 2 and 3 to make the controllers compatible with the simulation. This is concerned with yet another interface of the controller design, being the functional data exchange between VISSIM and the controller algorithms. As one of the intermediate steps a data exchange basis between the controllers and VISSIM needs to be established here.

2. The second one is concerned with the platooning controller design. This platooning controller will be strictly based on a set of technical expectations that are found in Chapter 2. These expectations are based on the literature research from step 0. They are formulated as such that they resemble real-life platooning as closely as possible. As the platooning controller is a basis for the design of a combined platooning-intersection controller, there is no possibility to iteratively test it in the end of the overall design phase. Therefore, this stage of controller design already incorporates iterative testing and a subsequent controller calibration.

3. Only when the design and calibration of a platooning controller is finished, an interaction of this controller with JUNO can be designed. This design is not guided by technical exceptions but rather by the intent to maximize the performance of the traffic network in terms of the introduced mathematical objectives. This stage represents the initial design of such a combined controller. It is modified and improved further until it meets the first deliverable requirement.

4. The fourth stage centers around the iterative improvement of the controller from Stage 3. The algorithm will be continuously tested in the VISSIM environment, which allows for repetitive improvements of its network-wide performance. The status quo scenario, which does neither include an implementation of platooning nor of traffic-adaptive intersection control, will in this context serve as a performance benchmark. The look-ahead traffic-adaptive controller JUNO on the other hand already adapts a more global view of the traffic situation (Van Katwijk, 2008) and will therefore (to some extent) serve as a leading functionality benchmark for the development of a combined functionality with platooning (a thorough analysis of the controllers can be found in the next chapter). Although the performance was earlier subdivided into the objective functions, this step will consider the congestion objective as leading. It will later on be shown that emissions decrease, as congestion does.

5. Before the technical design can be evaluated and interpreted on a high-level, it first requires verification and validation. Note, that the technical design consists of two major elements. Given this setup, the verification adopts a two-staged approach. That is, the individual designs will at multiple steps already be made subject of a plausibility check, which then serves as a basis for the overall verification. This is supplemented by the validation, which motivates why and for what situations, the presented research is applicable.

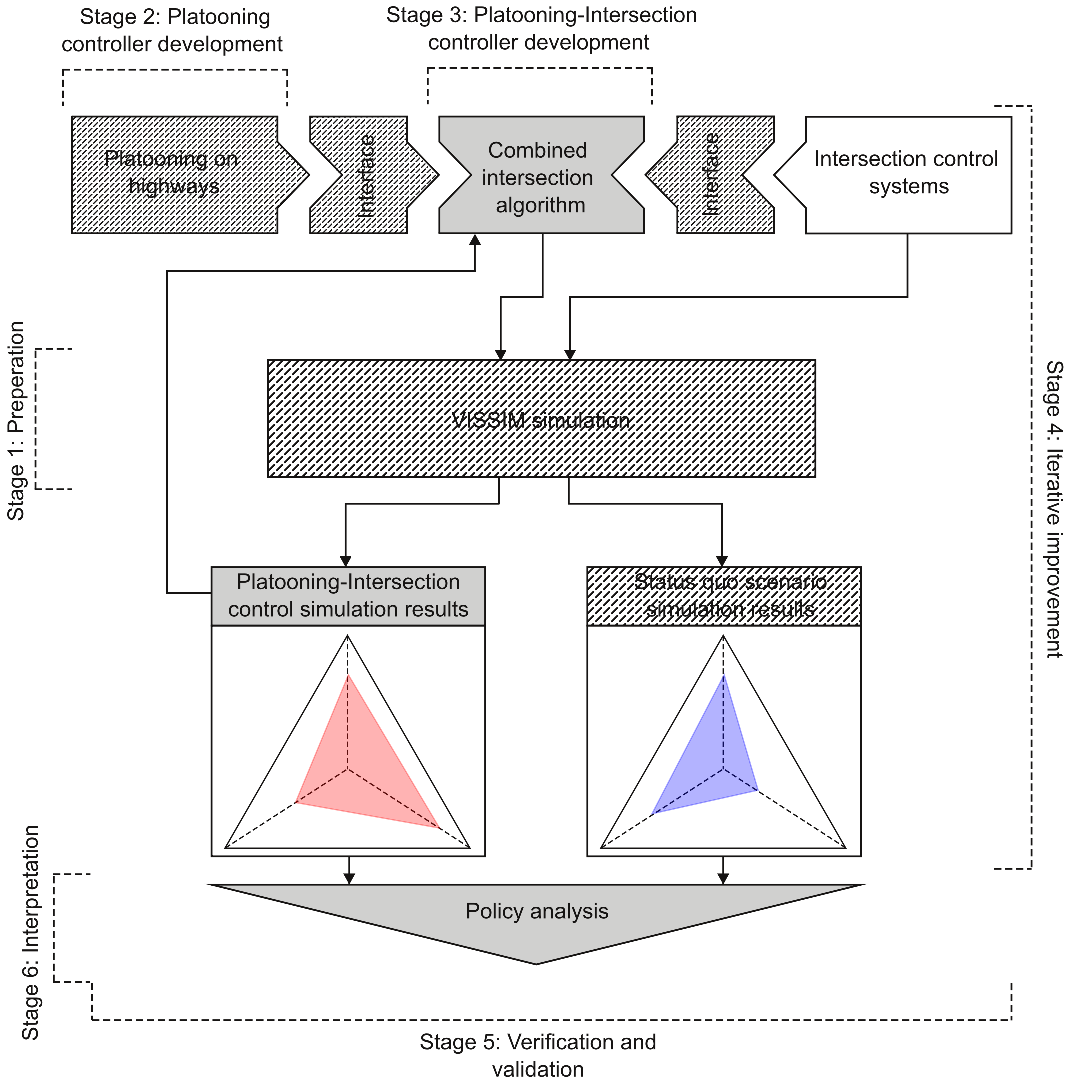


Figure 1.7: Research framework (Gray represents a research contribution; shaded gray represents a research contribution, which is based on and therefore constrained by already established systems. These systems are either provided by TNO or based in literature study; white represents such already established systems).

6. Lastly, the performance results of these alternations can be compared, based on the evaluation framework of Figure 1.5. From this a policy is derived aiming at a quick introduction of the technology that was developed through this research. The policy is based upon the simulation results in combination with a thorough context analysis and centers around the combined platooning-intersection controller.

1.5. SCOPE OF THE THESIS

To both underlying concepts there is a variety of aspects, which could be considered relevant either for a successful implementation or for the design of the deliverable itself. As the following low-level research focuses on the technicalities of platooning on a string of intersections, non-technical aspects will partly be neglected or subjected to assumptions. A scope needs to be applied, to distinguish between relevant and non-relevant aspects. Within the review protocols of the European Truck Platooning challenge five scope dimensions were identified (ETPC, 2017). In additional consideration of the physical scope, these are employed and specified for the research at hand in Table 1.5.

Table 1.5: Defining the scope of the thesis.

Dimension	Explanation	Adoption within this research

Physical scope	In order to effectively test the potential of a combined Platooning-Intersection controller, an environment is necessary, which challenges this controller and represents real-world matters.	With the N260 a case study is chosen as a testbed for the technical design. This is done in order to prove real-world feasibility. The choice of this case study is motivated in the usual, representative traffic demands on this string of three intersections. It is found that the case study at hand challenges the controllers in a similar manner as it can be expected for various other traffic situations and setups across the Netherlands. Choosing a rather 'average' case allows for comparable results.
Legal scope	Driverless cars are in general considered controversial when it comes to safety and their appliance and obedience with street rules.	Ignored. The theoretical nature of the research does not touch upon the legislation that is concerned with this topic. Additionally, there is a vast amount of activity centering around the legalization of platoons on urban roads already. Hence, this does not represent a knowledge gap, worth investigating in this context.
Informal institutional scope	There are a variety of institutional barriers, which might frustrate an implementation of platooning. E.g. the introduction of self-driving platoons might lead to distrust towards the technology and anxiety over the interaction with platooning vehicles (i.e. merge-in maneuvers).	Ignored. The theoretical nature of the research does not allow for quantitative conclusions about the involved informal institutions or how to deal with them. Furthermore, a functional design of informal institutions is usually very difficult or impossible.
Impact scope	Besides the obvious direct impacts of the controller implementation, such as increase/decrease of average speed, emissions or investments, there are a variety of other indirect impacts. These are e.g. impacts on the automotive industry, car owners, etc.	Constrained to direct impacts. While the simulation allows for a quantitative assessment of mentioned measurable, it does not for the indirect impacts.
Human factor scope	The integration of platoons certainly demands for a process design to establish an adoptable form of human-machine interaction. This is concerned with both, car drivers and other traffic participants.	Ignored. This is not part of the scope of this thesis. It is however, recommended to conduct research in this area as a potential follow-up project.

Technical scope	The last scope dimension is concerned with the technical system, which allows for the functionality of urban platooning.	Constrained to the combined platooning-intersection algorithm (the deliverable) and its interfaces. The scope is applied as such that it comprises all relevant, technical factors, which are directly linked to the development of the algorithm. That is, commonly understood subsystems, such as the internal functionalities of intersection control and platooning (e.g. power supply of intersections, driveline of the platoon vehicles, etc.) are not touched upon.

1.6. THESIS OUTLINE

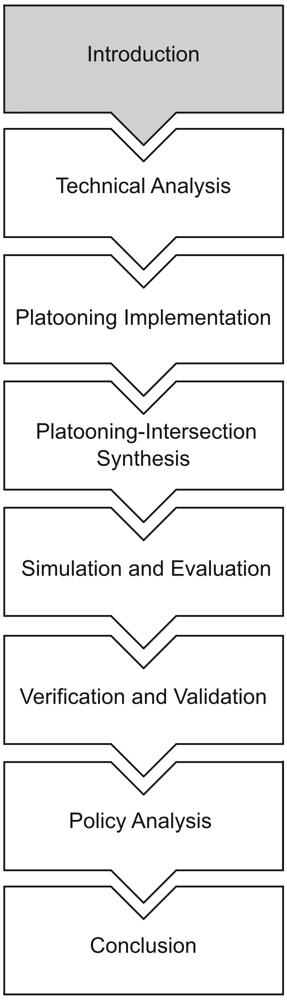


Figure 1.8: Chapter overview

From the six main research steps that were introduced in Section 1.4 a thesis outline can be derived, as illustrated in Figure 1.8. The current chapter is called 'Introduction'. It is followed by a 'Technical Analysis' in Chapter 2, which centers around establishing the interfaces with existing systems and examining the functionalities of the relevant systems. The platooning interface besides others provides a technical specification of a standardized platooning system, which serves as a basis for the following chapter. In Chapter 3 the platooning controller is developed to resemble these specifications. The chapter further more contains a section on the controller calibration and verification. After the reader has understood the platooning controller design it will be shown how this can be linked with the intersection controller JUNO. Chapter 4 shows how a technology called 'Virtual platooning' can be used to facilitate the interaction of a platoon of vehicles and an intersection controller. Again, this chapter includes the calibration and verification of the final deliverable. Chapter 7 presents the verification of this research and outlines in which context it holds validity. Note, that the verification in this chapter is based on a set of plausibility checks that are embedded in Chapters 3 and 4. Based on this, Chapter 5 then uses this both technical designs - the platooning controller from Chapter 3 and the combined controller from Chapter 4 - to generate result data for the N260 case. In Chapter 6 this data is used to formulate policy advice, which aims at an introduction of the combined controller and Chapter 8 concludes this report by summarizing the main findings and giving future research recommendations. Sections 3 and 4 both close with a summary of the presented low-level design steps of the respective chapters. This summary is aimed at readers who are not familiar with the details of controller theory, yet interested in a high-level explanation of the design. Furthermore, Sections 4, 5 and 6 feature a conclusion section. These are directly linked to the four levels of the V-model. The technical conclusion of Section 4 answers the research question of the lowest level, the simulation conclusion that of the third level and the conclusion of Chapter 6 answers to the two highest-level research questions.

A chapter overview can be found in Figure 1.8. It is noteworthy, that the middle chapters are solely concerned with the low-level technical design from Figure 1.2 and its simulation and optimization . Hence, the introduced high-level considerations represent a frame, which wraps around the actual main element of the thesis - the technical design.

2
Technical Analysis

A broad, yet solid knowledge on all relevant concomitant systems is vital when attempting to design an interaction of driving systems and road infrastructure towards a combined platooning-intersection control algorithm. With these concomitant systems being platooning and intersection control, it is the purpose of the second chapter of this report to provide such knowledge. Within the next sections, a common understanding of the underlying systems and their mode of operation is established in order to prepare the reader for stages 2 to 5 of the research framework. As illustrated in Figure 2.1, this contributes to the development of the actual deliverables by analyzing the underlying systems (cf. outer boxes) and subsequently establishing the interfaces between the deliverable and the concomitant systems. Both of these aspects are of vital importance. The latter one, because it guarantees compatibility of the deliverables with its environment and the first one, because the concomitant systems represent a basis for the eventual development of a vehicle controller and its interaction with an intersection controller. Yet, a differentiation needs to be made on how the systems are reviewed in the following. This is due to their different roles, which they assume during the design process of the introduced platooning-intersection controller. While the literature review of intersection control systems directly aims at providing the basic knowledge, which is necessary for the design of an integration with platooning, the platooning review itself also serves as a basis for an intermediate step. As mentioned in the previous Chapter, platooning first needs to be modeled within this research. Only after this is done, a combination with intersection control can be attempted. Consequently the subsequent literature review on the latter technology does not only provide knowledge for the design of the deliverable, but also information that are necessary for establishing a platooning model. Given this, the goals of this Chapter are to: (1) Establish a solid knowledge of currently-in-use and state-of-the-art intersection controllers and their functionality; (2) understand and define the technical interface, which these controllers provide for a potential combination with platooning; (3) establish a solid understanding of state-of-the-art platooning systems; (4) establish a platooning definition as preparation for platoon modeling and (5) understand and define the technical interface, which this platooning definition provides for a potential combination with intersection control.

Figure 2.1: Illustrating the contribution of the 'Technical analysis' chapter.

In a first step, the characteristics of intersection control systems and platooning are explored individually in Section 2.1. This is done by means of a thorough analysis of the leading (implemented) concepts for each of the concomitant systems. In terms of intersection control, this aims at establishing a taxonomy of the most prominent controlling approaches (Setion 2.2.1). This allows for a conclusion on which of those approaches are feasible for a potential integration with platooning. It also illustrates how the final deliverable can in general be combined with a range of already existing intersection controllers, although only being tested in conjunction with JUNO in the context of this research. The Section concludes with some specific elaborations on JUNO, which are relevant for the development for such combination.

Initially, a similar approach is chosen in terms of platooning. First a set of existing, proven platooning concepts is introduced. However, in contrast to the intersection controllers, all of those are considered relevant for the development of the platooning model. That is, no subset of relevant platooning systems is identified here. The Section rather aims at establishing an 'average' understanding of platooning, which considers all of those existing implementations and eventually serves as a modeling goal. This goal is what the VISSIM model tries to represent as closely as possible and the choice to consider all introduced implementations is done in order to make the eventual deliverable design as applicable as possible. That is, the design of a platooning-intersection controller is done on basis of the platooning model. The deliverable will eventually function in conjunction with this model. Hence, the model itself should consider and represent the existing platooning implementations as accurately as possible. Consequently, in terms of platooning, this Section provides two conclusions. One of those is the interface, which the deliverable has to adapt to, the other modeling goal for the platoon implementation in VISSIM.

Eventually, the provided knowledge, a set of interface specifications between the systems and a modeling goal for platooning suffice as a basis for the next Chapters. Firstly, the actual platooning implementation in VISSIM and secondly, the deliverable design. It guarantees compatibility between the systems but it also shows that the system generally has a wide range of applicability, including a variety of feasible intersection controllers and platooning implementations. The knowledge presented in this chapter is mutually founded in expert interviews and a thorough literature study. Mainly the literature study is executed using Google Scholar, Mendeley and Web of Science. The boundaries of this knowledge exploration are limited to those aspects, which are necessary to guarantee functionality of the eventual combined controller design. Additional information that is necessary for the actual implementation of the deliverables, is provided in the respective implementation Chapters.

2.1. SYSTEM FUNCTIONALITY

Before any interfaces can be defined, their embedment in the concomitant systems is elaborated on. This is concerned with understanding the systems, represented as the outer boxes in Figure 2.1. Firstly, the 'Intersection control systems' box is analyzed in detail (Section 2.1.2). Besides JUNO, a set of other intersection controllers is reviewed as well. This has two main reasons. Firstly, the JUNO algorithm solely represents the desirable situation. In the current Dutch road network a set of other control approaches is in place, of which the fixed-time and the traffic-actuated one are the most prominent ones. In that sense, an essential part of understanding the AS-IS situation is to to gain knowledge on their functionality. Besides others, this is concerned with understanding the shortcomings of the current situation, which is needed in order to conquer those by means of the deliverable. In this sense, the functionality of current intersection controllers is motivation for and guidance of the design of the final deliverable. Secondly, the to-be-designed combined controller should be compatible with a broader range of intersection controllers. That is, all intersection setups that generally allow for V2I communication (especially those that feature 'look-ahead traffic-adaptive control) should be considered for the design of the deliverable. This is also the reason why the review of these controllers starts from a rather generic point of view.

Following this, a thorough summary on platooning systems can be found in Section 2.1.2. In contrast to the Section on intersection control, this part of the technical analysis aims at finding a common understanding of platooning. While the technical divergence of intersection controllers is appreciated as inspiration for the deliverable design, the divergence of platooning systems is rather a necessary evil. The thesis attempts to review a set of implementations, which eventually allows for defining an 'average' conception of platooning. This is crucial, as only one platooning system will be modeled by means of a platooning controller. The 'average' platooning conception should therefore cover a major amount of the variety of existing implementations of platooning. Eventually, this allows to create an platooning-intersection control algorithm, which is capable of facilitating said agreed-upon average platooning definition without intervening in its underlying functionality. It is hereby ensured that the designed technical solution is applicable for a rather broad range of in-place concomitant system variations.

2.1.1. INTERSECTION CONTROL SYSTEMS

The essence of signalized intersections is to facilitate traffic in a space, which is common to several opposing traffic flows. Bento et al. (2012) divide this space into sectors, as illustrated in Figure 2.2. It is the traffic light's responsibility to ensure that no sector is accessed by two vehicles at the same time. Although not all literature explicitly adapts the sector notion, the goal of preventing multiple vehicle accesses is consistent. Functionally, this resembles the first deliverable requirement, which aims at preventing the platoon to access a sector, which it does not have the priority for. A variety of ways to do so exist, which can generically be divided into four categories: 'Fixed-time' control, 'Traffic-actuated' control, 'Traffic-adaptive' control and 'Look-ahead traffic-adaptive' control. In the following, these four types of intersection control are introduced and explained.

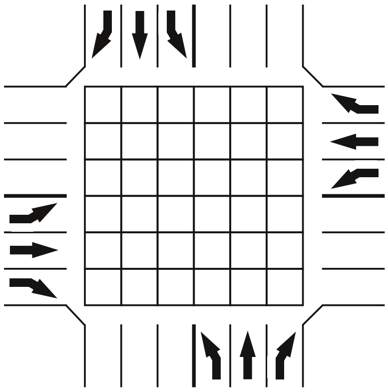


Figure 2.2: Largest possible intersection setup.

Traffic lights split the available space over time, meaning that they prevent collisions by assign time-wise priorities. No two vehicles will have priority to the same sector at the same time. Logically, all in-place TSC approaches fulfill the first deliverable requirement. However, approach and result for the second requirement vary widely. An overview is given over how these tackle the performance optimization of intersections, how they collect and process data and how they communicate their decisions with the vehicles. Each of the approaches is analyzed in detail, pursuing two goals. Firstly, knowledge is gained on the functionality of each approach. This is of great importance, as these functionalities represents the foundation of the deliverable. Especially for the latter one this holds validity. The algorithm developed in Chapter 4 is a slightly modified variant of the JUNO control algorithm, which itself can be allocated in the 'look-ahead traffic-adaptive' control domain. Secondly, a clear understanding of their technical interfaces is reached to enable functioning with the concomitant systems.

Note, that this chapter does only give limited insights to the network effects and area-wide implementation of the individual approaches. Although the connectivity aspect represents one of the challenges for the research at hand, it is of no relevance for the definition of the technical interfaces from Figure 2.1. The elaborations on network functionality are limited to the optimization approach of the look-ahead traffic-adaptive controller, as this is the approach on which the eventual technical design is based. This controller form naturally features a global view of the traffic situation.

FIXED-TIME CONTROLLER

As the name implies, intersection control is initially grounded in controller theory. In this subject area, the so-called 'open-loop' controller comprises the easiest form of intervening in an uncontrolled environment. Its signal processing approach is depicted in Figure 2.3. The intersection itself, with its underlying, non-controlled nature represents the environment (cf. right box in Figure 2.3). This box comprises all the non-negotiable, yet controllable parameters of the intersection. In traffic control this is understood as the setup of the intersection with its intersection entries and outlets (cf. Figure 2.2), as well as the instinctive driver behavior.

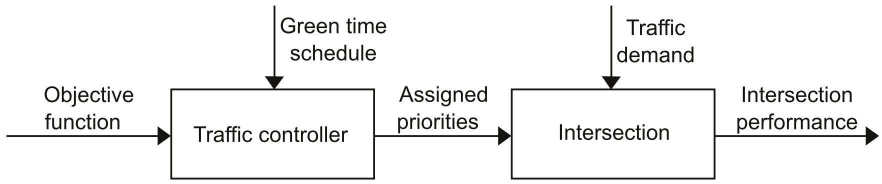


Figure 2.3: Signal processing of a fixed-time controller.

Through an incoming, say 4-way traffic demand (cf. top-right arrow) the environment is triggered to reveal its underlying nature. Assuming that the intersection is non-controlled (i.e. the left part of Figure 2.3 is missing), this nature would be that all vehicles attempt to enter the intersection at the same time, with the drivers following their instinctive behavior. In traffic management terms, the right box would thus be an intersection, exactly configured as depicted in Figure 2.2, lacking any actuators (i.e. traffic lights) or sensors. In context of the research at hand, the intersection environment is represented as a VISSIM simulation. It comprises the underlying nature of the intersection and how it responds to certain control measures. Specifically, this includes a driver model, which prescribes the instinctive behavior of all traffic participants and the intersection infrastructure itself. Now, given sufficiently high traffic demands, it is nothing but a logical consequence that the traffic within the intersection will result in chaos. Hence, the output variable of this environment (i.e. the 'Intersection performance' arrow) will deliver rather bad results, both in terms of congestion and emissions.

In order to bring structure to this chaos, the natural functionality of the environment can be manipulated through an actuator (cf. the arrow between the boxes). In the conventional TSC idea, this actuator is the traffic light. It conveys a priority to one or several of the intersection entries and thereby intervenes in the intersection and driver behavior. In conjunction with the actual traffic controller (cf. left box), this setup is, what is considered as the 'Intersection control system' in Figure 2.1.

The traffic controller itself is an algorithm, steering the environment through assigned priorities, which are conveyed by means of the light signal. It determines which entry gets the right of way at what time and for which duration. An open-loop controller regulates the environment by assigning green, yellow and red lights to the respective directions without incorporating feedback loops. Thus, the controller has no sensors, neither does it follow any decision rules, except from the green time schedule, which it is adapted to. This schedule usually prescribes a priority sequence vector $\vec{S}$ and a priority duration vector $\vec{t_d}$. In that sense, the traffic controller is simply an execution device for a man-made green light sequence, switching between the priority states as determined by $\vec{t_d}$. Although different green time plans can be executed at different times of the day, week or year, the duration and sequence of the individual priority plan is fixed. Hence, in traffic theory terms, this form of intersection control is called a 'fixed-time' controller.

TRAFFIC-ACTUATED CONTROLLER

Green time schedules for fixed-time controllers are usually designed based on historical traffic data in a hands-on process (Van Katwijk, 2008). This approach does therefore not allow for immediate reactions to changing traffic demands (cf. top arrow). However, depending on the situation, traffic demands can change quite severely within short periods of time. Roess et al. (2011) suggest a division into four general types of traffic fluctuation:

- *Gradual traffic fluctuations* describe the constant, yet slow shift of traffic demand due to perpetual environment changes, such as new technologies in the transport industry and consequential changes in driver behavior.
- *Seasonal traffic fluctuations* are associated with alternations in traffic, based on seasonal influences. Examples are e.g. the seasonal climate or the yearly consequences of switching to winter tires.
- *Weekly and daily traffic fluctuations* describe occurrences such as peak demands at specific times of the day or their absence on the weekend.
- *Event-driven traffic fluctuations* are effects that occur as a consequence of events within the traffic network. Such events can e.g. be accidents, road works or infrastructure malfunction.

Using historical traffic data, the second and third category are quite predictable. They describe fluctuations, which have occurred on a timely basis and will do so in the future as well. Hence, a fixed-time controller is capable of coping with these fluctuation characteristics. Additionally, the first category appears to be an acceptable, yet expensive happening. Although the "aging of fixed-time traffic signal plans" is unpredictable (Bell and Bretherton, 1986), it is sufficiently slow to theoretically allow for periodic schedule adaptions e.g. by a human supervisor. The last category however, is something that an open-loop controller can not cope with. Event-driven fluctuations are both, unpredictable and abrupt. Thus, given the defined research scope, the latter type of traffic demand fluctuation is especially relevant. The high connectivity of urban road systems makes the network especially vulnerable for event-driven traffic fluctuations, such as the ones that are exemplified in above itemization. Furthermore, in additional consideration of the bottleneck character of intersections and the potential amplification of traffic through a network, non-event-driven traffic can emerge. Especially during peak times it is of stringent importance to effectively maximize the capacity of an intersection. Due to the unpredictability of these events, a fixed-time controller however, might not always do so.

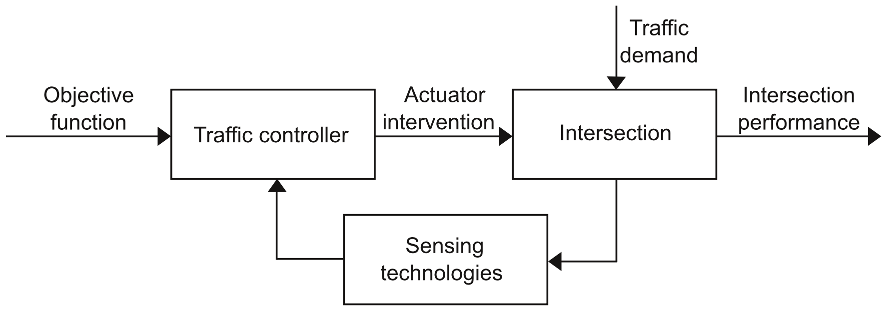


Figure 2.4: Signal processing of traffic-actuated and traffic-adaptive controllers.

The traffic-actuated controller attempts to conquer this drawback with the additional incorporation of a feedback loop. As illustrated in Figure 2.4, the system is equipped with sensing technology, which is capable of extracting supplementary, real-time information Z about the traffic within and around the intersection. This sensing technology usually constitutes an inductive loop detector, which is located on the approaching intersection lanes. It is triggered through a change in its magnetic field, hence the existence of metal in its immediate surrounding. It is therefore only capable of sensing the existence however no other parameters of the vehicle. The algorithm uses the gained information to compile an estimate of the actual traffic situation, yet restricted to the intersection entry, which currently has the right of way. Although, this estimate is limited in terms of accuracy (usually only one sensor per intersection entry), it nevertheless delivers adequate information Z_x about the proximate traffic demand of the intersection entry x.

The right of way for entry x is terminated as soon as no cars were detected by the sensors on that entry for a prescribed time gap d. This duration d usually constitutes a value between 2 and 5 seconds, whereas low values are used for highly congested intersections and higher values for mostly free-running setups. The reasoning behind this is that if no cars have been detected within d, the waiting queue has most likely vanished. Here, the waiting queue is defined as all vehicles, which have accumulated during the preceding red phase. This way, green times are terminated when no demand is remaining from that intersection entry. This is a strong advancement to the earlier introduced situation, where cars might wait, although no other car demands to enter from the crossing direction. The green time duration is now developed internally and continuously updated, according to real-time traffic information. Additionally to the traffic-actuated termination, the green phase might be terminated after a certain pre-defined maximum duration t_{max} or extended to a certain predefined minimum duration t_{min}. The former is done to prevent unreasonably long waiting phases for crossing vehicles, the latter to adapt to low-traffic situations. Without t_{min} an uncongested intersection would switch priorities every d seconds. Approaching vehicles would slow down (and produce more emissions), as they are not able to determine whether they can catch the current green phase or not. The deployment of t_{min} can therefore be seen as a means to make the intersection more predictable.

The latterly introduced t_{max} and t_{min} resemble the functionality of the fixed-time controller. The innovative aspect of the traffic-actuated algorithm is the additional incorporation of a sensor, which provides traffic data through ha feedback loop. Applying this feedback loop on the interface notion from Figure 2.1, it seems vital to define the content and specifications of the sensors and the data that they produce. Consequently, this is one of the aspects, which is treated in Section 2.2. Incorporating this knowledge is a basis for the design phase of the platooning-intersection controller.

TRAFFIC-ADAPTIVE CONTROLLER

The traffic-adaptive approach is essentially an enhancement of the signal processing of the traffic-actuated controller. Instead of only considering traffic demands of the currently prioritized lanes, this algorithm takes sensing data from all intersection entries into account. In contrast to the traffic-actuated controller, the estimate of the traffic situation is not restricted to one lane only, although it does at times use the same sensor setup. Given the additionally incorporated information, it runs an optimization process, which evaluates all possible actions and eventually choses the one, which maximizes its objective function. As an amendment to the traffic-actuated approach, which works on a strict if-then-basis, this algorithm incorporates decision-making (Van Katwijk et al., 2006). Van Katwijk et al. (2006, p. 3) state that "the traffic signal control problem solved by all traffic adaptive systems can be formulated in the form of a general decision problem", which itself can be presented as a decision tree.

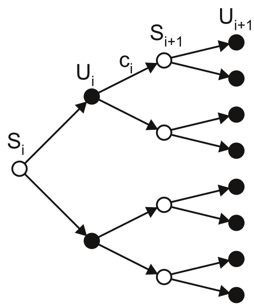


Figure 2.5: Illustration of a decision tree, as utilized in traffic-adaptive control (adapted from Van Katwijk et al. (2006)).

Given, the current state of the intersection S_i the controller has a set of possible actions U_i. These actions lead the system to a next state S_{i+1}, whereas each transition from S_i to S_{i+1} is associated with a cost c_i. In Figure 2.5, the action U_i has two successor states, which represents stochastic outcome variations. The optimization process aims at minimizing the costs c_i. They should therefore be computed as such, that their minimization is associated with an optimization of the objective function. Although the way in which these costs are assigned varies among different implementations of the traffic-adaptive approach, all of them incorporate the sensing data in the costs assignment. The enhancement to the traffic-actuated algorithm is that this approach considers several potential successor states S_{i+1}, whereas the traffic-actuated and the fixed-time controller both follow a prescribed state sequence $\vec{S}$.

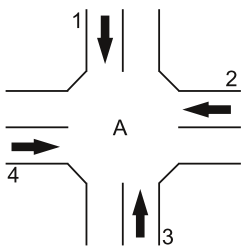


Figure 2.6: Illustration of a four-way, signalized intersection, with one possible route per entry.

As mentioned, the exact decision-making process varies per implementation. Generally however, it is remarkable that traffic-adaptive controllers are capable of considering multiple states and transition stages. While the traffic-actuated controller works according to a simplistic action-response scheme, the traffic-adaptive controller can plan a sequence of actions. It can use predictive future costs and consider not only the immediate but sub-sequential costs as well. To illustrate this, an example is given in Figure 2.6. A four-way, signalized intersection is assumed, where all vehicles intend to go straight at all times. The traffic-adaptive controller, trying to regulate the traffic on that intersection optimizes an objective function, which aims at a reasonable trade-off between emissions and waiting time equality. The objective function does so by allocating the transition costs c_i as follows:

- *Waiting costs* are continually increasing, starting with the beginning of a new phase. The costs are accumulated (mathematically integrated) over the waiting period. Hence, the longer the red phase is maintained, the higher the costs will be. Here, the costs of waiting is increased by c per cycle time t_c. This means that the waiting costs after a priority switch are $1c$, while they increase to $1c + 2c = 3c$ (for both time steps), if that priority is extended.

- *Emission costs* occur when the priority is changed, as cars that previously had green have to brake and therefore consume additional fuel. The costs are fixed to $2c$ per priority switch. However, they are only activated if approaching vehicles are sensed on the intersection entry that is currently assigned a red light.

For reasons of simplicity, infinite traffic demands from all entries are assumed. All intersection entries of Figure 2.6 are filled with cars. Furthermore, presuming that no stochastic outcome variations exist, the controller will construct a decision tree as depicted in Figure 2.7. In every state, the controller has the option to either maintain the current green phase or terminate it and give the right of way to the opposing direction. Given that three future states $S_{i+1}, S_{i+2}, S_{i+3}$ are considered, the eventual priority plan will be: *maintain-switch-maintain*, as illustrated for intersection A in Figure 2.10.

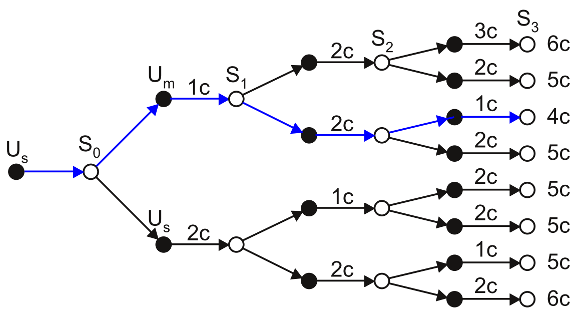


Figure 2.7: Illustration of a decision tree example for a singe, four-way intersection under traffic-adaptive control. The eventual priority plan is depicted in blue.

It now has to be understood, that the tree is continuously expanded when new sensor data is available. Additionally, it is repetitively modified after a pre-defined cycle time t_c. Firstly, this infers that the costs of each branch of the decision tree are reassigned after every decision. Secondly, as time progresses and the controller moves to a new state $S_i + 1$, a new set of branches is added to the tree. Consequentially, in every state a new priority plan is generated.

LOOK-AHEAD TRAFFIC-ADAPTIVE CONTROLLER

While traffic-adaptive controllers have a broader, all around view of the traffic situation than traffic-actuated controllers, they nevertheless have the same vision depth. That is, their estimate of the traffic situation is limited to the detection range of the intersection sensors. As sensors are expensive and construction work is necessary to embed them in the road surface a lot of traffic-adaptive controllers only consider the immediate surrounding of the intersection. Yet, traffic demands are usually neither uniform nor are they predictable. More precisely, demand characteristics differ per vehicle and intersection, and the extent of the demand (i.e. the amount of vehicles demanding to cross the intersection) can not be predicted accurately before arrival with this sensor setup. Therefore, the controllers which have been introduced so far often lack knowledge on approaching vehicles. Their sensors merely conjecture a rough estimate of the actual traffic situation. Given the collected sensing data, assumptions are made about the amount of cars, their speed and sometimes their destination. Look-ahead traffic-adaptive control builds upon this by additionally integrating sensor data from surrounding intersections, as depicted in Figure 2.8 (cf. 'External sensing technologies' box).

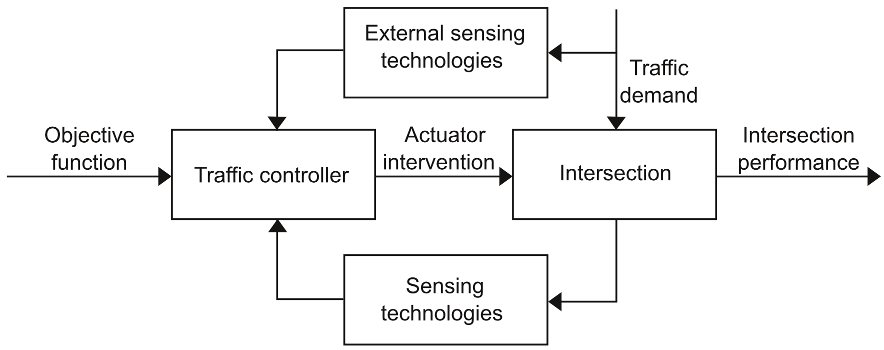


Figure 2.8: Signal processing of look-ahead traffic-adaptive controllers.

The way in which the additional data is integrated in various look-ahead traffic-adaptive traffic controller implementations varies widely. In contrast to the other controller approaches, there is no common standard, neither on the way traffic is modeled, hence how an estimate of the traffic is made, nor on how that estimate is incorporated in the internal decision-making process. A variety of sub domains and respective controllers exist, f which one will later be introduced - the JUNO controller. Nevertheless, a few general remarks that hold relevance for all implementations of look-ahead traffic-adaptive control can be made. Van Katwijk (2008) provides a good summary on the leading competition algorithms, such as SCOOT, MOTION, TUC, UTOPIA or OPAC and their network implementation. Although, their specific functionality is not relevant for the design of the deliverable of this research, a general overview of them can contribute to the definition of the technical interfaces for that deliverable. Hence, some generic remarks on the look-ahead traffic-adaptive control functionality are provided in the following, loosely based on the mentioned competitive systems.

Firstly, the additional data solely provides knowledge on the traffic situation in the intersection's outer environment. Data that is directly associated with the current state of the intersection at hand is qualitatively and quantitatively the same as in the traffic-adaptive case. That is, no additional sensing technology is installed at each intersection. The gained knowledge solely emanates from data sharing amongst a net of closely located traffic controllers. The implementation of look-ahead traffic-adaptive control therefore does nor require more investments than that of a traffic-adaptive control. Nevertheless, the additional data allows for a more far-sighted vision. The extent of the intersection's view has increased beyond the borders of its sensors, which in turn, allows for an extended estimate of the traffic situation in terms of scale.

Secondly, decision-making is executed in the same general manner as it is in traffic-adaptive control. The algorithm establishes a decision tree, which serves as a basis for an optimization process, eventually leading to the best possible decision plan. Again different optimization strategies and algorithms can be used to find the optimal path through the decision tree. What is different however, is the decision tree itself. Given the increased extent of the traffic estimate, predictions can be made about future traffic demands and their respective relevance for future decision nodes within the tree. The tree will not only gain accuracy about future transition costs c_{i+t} but also depth. In other words, the controller "is capable of determining the optimal control decision on basis of a longer term analysis which often incorporates information from further upstream" (Van Katwijk, 2008, p. 53). Having said that the way newly available data is integrated varies widely, can therefore be specified as 'the way in which newly available data is incorporated in the decision tree varies widely'.

Thirdly, almost all look-ahead traffic-adaptive controllers engage in some kind of area optimization. Here, a general differentiation between centralized and decentralized area optimization is made (Lei and Ozguner, 2001). Within the centralized approach all available data is processed by one agent, who henceforth optimizes the traffic for the whole network. However, difficulties with network expansions, the dependency on a central computer, the subsequent system vulnerability and the reliability on the communication network make it a less favorable design choice for most situations (for a detailed explanation of the drawbacks of centralized control see Van Katwijk, 2008, chap. 3.3.1). Thus, the research at hand is based on the decentralized approach, of which JUNO represents one potential implementation. That is, the to-be-designed interaction of a platooning controller with a decentralized look-ahead traffic-adaptive intersection controller is implemented in conjunction with JUNO. However, it is designed as such that it can be applied to a range of look-ahead traffic-adaptive controllers.

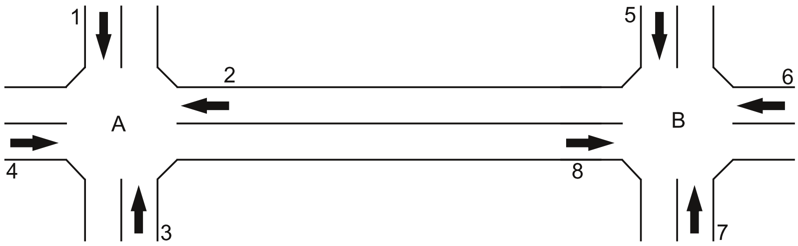


Figure 2.9: Two intersections, which are connected through a single road arc.

Within the field of decentralized look-ahead traffic-adaptive controllers again, various variants of network optimization are distinguished. For the purpose of understandability a minimalistic, two-intersection example, as illustrated in Figure 2.9 is employed. This allows for a rather generic explanation of the controller's functionality, comprising most of those variants. In the following, the reader is guided through an example of intersection communication, which then allows for a deduction of a generic functional summary of look-ahead traffic-adaptive control.

Two intersections, A and B are connected through a road arc, which is reasonably short, so that cars from one intersection will shortly after arrive at the other intersection. Hence, whenever intersection A releases a group of vehicles onto the connecting arc this has a direct impact on intersection B and vice-versa. We now suppose, a similar traffic situation as above. Here, the demand is continuous and equally distributed among all entries except from 2 and 8. Furthermore, it is assumed that the cars take roughly t_c to get from one intersection to the other.

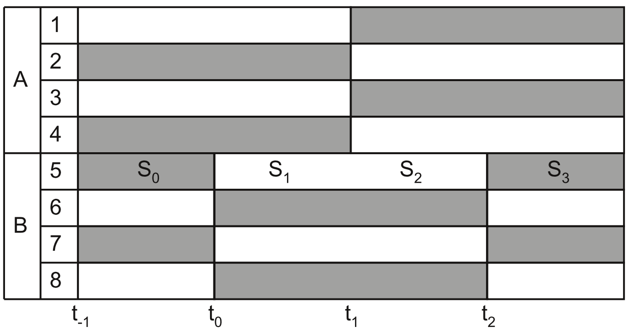


Figure 2.10: Priority plan of intersection A and B. The allocated priority phases are illustrated in gray for each intersection entry.

Conventionally, the traffic-adaptive control attempts to find a priority plan, which maximizes its own objective function. Again, this is a trade-off between throughput and equality of waiting time, which causes the intersections to initially deploy a priority plan S_0, as displayed in Figure 2.10 at t_{-1}. As the intersections are connected within a bigger network, they communicate their priority plans. Assuming that intersection A communicates its current and planned priorities with intersection B, B can integrate these information in its decision tree. Specifically, this concerns the first branch, leading to S_1. As vehicles on the arc are arriving at intersection B at t_0, these vehicles would have to brake if the current priority is maintained, causing additional emission costs. On the other hand, emissions can be saved when they receive priority on arrival. Hence, the costs of switching priority are at this point in time smaller than maintaining it. A 'Pareto' improvement was achieved, where the performance of B is better off, while the performance of A remains the same. Consequentially, the priority plan will be modified as such, that the current green phase is terminated earlier to let the arriving vehicles pass, without forcing them to brake. As it can be seen in Figure 2.10 the initially ideal plan of switching every two t_c, is abandoned in order to allow for a reasonable alignment of the two intersections.

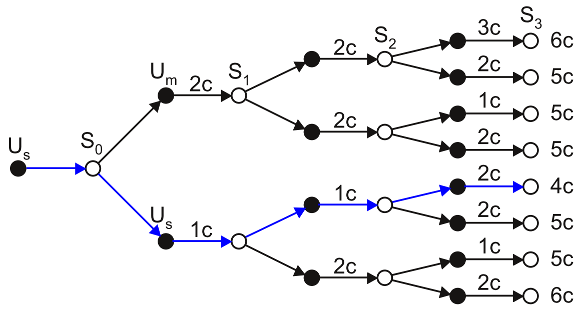


Figure 2.11: Illustration of a decision tree example for a singe, four-way intersection under look-ahead traffic-adaptive control. The eventual priority plan is depicted in blue.

Within the context of this thesis, such Pareto optimization is considered the minimum functionality, which should be reached in terms of network optimization. This describes a performance improvement of one node of the network, which does not make any of the other nodes worse off (Ngatchou et al., 2005). In traffic theory terms: If a certain action can improve the performance of an intersection in terms of its objective function without frustrating the performance of other intersections, the proposed action will be executed. An action in this context, represents a decision U_i within the internal decision tree of that intersection. From this it can be derived, that connected intersections have to participate in a communication process. Here, two interactions are possible. Firstly, internal actions can be announced through the executing intersection. This announcement can then be used by surrounding intersections to update their decision trees. Secondly, external actions can be requested. In this case, the executing intersection receives the request, estimates its local impacts Δc and if applicable processes it as an internal announcement within its network. The latter allows for local improvements by means of signal plan changes in surrounding controllers, the former for local improvements through gained information from surrounding intersections.

JUNO

As mentioned, JUNO represents one possible implementation of a look-ahead traffic-adaptive controller, which possess the characteristic of incorporating traffic data from further upstream. For this thesis, JUNO is the controller of choice, which is being used for the impact analysis of a functional combination with urban platooning. The choice is made as such, because JUNO already adapts a form of V2I communication, which allows for a good combination with platooning. In the following, the algorithm will be embedded in the previously introduced intersection controller taxonomy. It is elaborated on, how JUNO builds upon the general functionality of look-ahead traffic-adaptive control and how this potentially allows for a combined functionality with platooning. Furthermore JUNO engages in a negotiation process with surrounding intersections in order to reach a conjunct state plan, which makes all involved agents (i.e. surrounding intersections) better off. The latter part of this Section goes into detail with how such a Pareto improvement is reached through this. Eventually, it is shown that JUNO's network optimization approach can be maintained and resumed within a functional combination between a platooning controller and JUNO.

JUNO does not only fall into the domain of look-ahead traffic-adaptive controllers but also into the subdomain of green light optimized speed advice (GLOSA) controllers, which itself builds upon the idea of a coordinated intersection. As introduced in Section 1.4 CIs make use of V2I-communication, mutually retrieving data from approaching vehicles and sending information back. Although JUNO does strictly seen not represent a CI, as data from the intersection controller is not directly incorporated in the vehicle control system, it does make use of the same principle. Especially the latter, communicating information with approaching vehicles, is relevant for JUNO's functionality. That is, as any other look-ahead traffic-adaptive controller JUNO executes an optimization process of a given decision tree. A byproduct of that optimization process is the calculated value of a so-called 'estimated time of departure' t_{ed}. This estimated time of departure represents the predictive time until a certain vehicle is able to cross the stopping line. This value is calculated for every approaching vehicle. What distinguishes JUNO from competitive look-ahead traffic-adaptive controllers is that the algorithm makes use of this t_{ed}-parameter. The estimated time of departure serves as a basis for an emission-optimizing approach towards an intersection entry. The technical details of this are explained in the combined controller design chapter. For now, only the potential benefits and current drawbacks of this idea are introduced.

When a car approaches an intersection, the average driving behavior of the driver in charge is to approach it the same way, he would approach a stop sign. Due to the lack of knowledge on when the driver will receive priority, he maintains cruise speed until he reaches his intrinsic braking point. This braking point is usually allocated close to the stop line or the last queuing vehicle. Naturally, the driver then proceeds to decelerate as late as possible, adding his vehicle to the queue of vehicles already standing at the intersection. The potential for improvement which JUNO identifies in this behavior is described as the following: "Although the overall travel time stays the same, fuel is wasted when a vehicle decides to maintain its current speed until it is forced to stop. Maintaining cruise speed only increases the time that is spent waiting in the queue, and does not reduce overall travel time. This time could also be spent decelerating thus minimizing fuel consumption" (van Katwijk and Gabriel, 2015, p.480). The idea behind this is clear: The travel time of the driver at hand will be the same, no matter whether he arrives in the queue at the beginning of the red phase or at its end. Hence, the adopted speed level during that time can be prescribed as such, that the vehicle will arrive exactly when it is possible for it to enter the intersection. This can be done by decelerating early and then maintaining a constant, below-cruise-speed velocity. This approach leads the vehicle to arrive at the intersection with said velocity, say half of the initial cruise speed. Consequently, it then only needs to accelerate half as much, as it would have to in the initial situation. This way, unnecessary acceleration and deceleration can be prevented, which subsequently saves energy and emissions. As mentioned, the precise process of calculating and establishing this motion is further elaborated on in Chapter 4. For now the described potential benefit of this needs to be understood.

What also needs to be understood is that the desirable, constant, below-cruise-speed level of velocity needs to be communicated with the driver somehow. This is where JUNO reveals its current main drawback. Although a theoretical energy-optimal speed trajectory can be calculated by the intersection controller, there are only very limited means of transmitting this knowledge to the drivers that are approaching the intersection. Currently, this communication is mostly facilitated through variable speed limit signs that are situated in the immediate surrounding of the intersection. A sometimes applicable alternative are LED-panels located on-top or next to the traffic light, which display the time until the next green phase. The latter however, is mostly in place in conjunction with fixed-time controllers, while the former allows for all kinds of inaccuracy and impreciseness. That is, drivers firstly do not brake at the exact point they are supposed to brake and secondly they are not be able to maintain the exact prescribed speed level. Both effects amplify each other and lead vehicles to arrive either too late or too early at the intersection. The first one infers additional congestion problems, whereas the latter leads the vehicle to a full-stop, eventually making no difference to the normal intersection situation. Hence, it can be concluded that JUNO strongly relies on the precise deployment of the provided speed advice and indeed, as it will be shown later, the platooning concept offers this possibility.

Firstly, however it is explained how this form of speed advice is embedded first in the internal and then in the network-optimization approach of JUNO. As explained by Van Katwijk & Gabriel (2015, p.480) "The estimated time of departure incorporates current estimates with respect to green times, queue lengths and saturation flow." In other words: The t_{ed}-value of a vehicle constitutes a metric for the current level of congestion on that intersection. Logically, if the average estimated time of departure value is high, this indicates a high level of congestion and vice-versa. Hence, minimizing this average value goes along with savings in terms of congestion C and subsequently in terms of emission savings E, which together form the main objective functions of this thesis and traffic management in general. Given this, it is a logical consequence that the t_{ed}-value can be used for the calculation of the decision costs c_i, which constitutes a major aspect of the construction of the intersection-internal decision tree. Now, in order to make the technical design of the next chapters applicable for a wider range of look-ahead traffic-adaptive controllers it needs to be understood that firstly, all such controllers generally allow for the calculation of t_{ed} and secondly, all controllers attempt to maximize this value in one way or another. For that it is not relevant whether the estimated time of departure is the actual object of maximization or not. Its close relation to alternative congestion metrics, such as the queue length or the throughput, allows for the conclusion that optimizing one of those values, subsequently leads to improvements on the alternative metrics as well. Consequently, even if the decision costs c_i are calculated differently in other look-ahead traffic-adaptive controllers, their optimization processes will lead to a similar result. It can therefore be assumed that the final deliverable, a combination of platooning and look-ahead traffic-adaptive control will hold similar benefits. However, only those of the JUNO controller are quantified within the scope of this thesis. This is due to the fact,that JUNO does not only try to optimize the t_{ed}, but also makes use of it in the process of steering the traffic, which itself can be incorporated in a platooning controller.

Having understood that JUNO constructs a decision tree, which eventually leads to a decrease in average t_{ed}-values, this notion can further be embedded in the network-optimization process of JUNO. In general JUNO participates in network optimization by communicating with immediately surrounding intersections. An agent-based networking approach is chosen, whereas each intersection represents an agent. In the following the applicable communication protocol is exemplified by means of a negotiation between the two agents A and B from Figure 2.9. The basic protocol consists of five steps, which are executed in repetitive order, whereas the execution of each step is done within one time step t_c:

1. Both agents calculate their respective decision costs c_i for their available decisions based on internal sensor data and external information about vehicles that were released to approach their respective intersection. As a result of this step, each agent has its own decision tree, which incorporates both internal and external sensor data.

2. Both agents optimize their internal decision trees individually, given their respective decision costs c_i and come to a subsequent array of decisions which represents the minimum of $\sum c_i$.

3. If that decision array contains an entry which is not 'maintain' ($U \neq U_m$), then that move is filed as a request and send to the surrounding agents. Suppose e.g. that A's state plan includes switching priority within the considered time horizon. If this is the case, then a request to do so is send to B. This request contains the outflow, which A intends to release on the link to B.

4. In this step A calculates the potential cost savings of the requested action. This is done by comparing the overall costs of A's optimal state plan, with the 'all-maintain' strategy, where the decision to switch priorities is neglected. Simultaneously B processes the received request by updating its internal decision tree, given the newly available information and subsequent new costs c_i. The new best state plan is generated and the overall costs of that plan are compared to those of the initial plan of the controller. IT answers the request by sending the calculated additional costs $\Delta \sum c_i$ back.

5. In this step A receives all additional costs, which its action would invoke. The agent uses this information to make a final decision whether to execute the action that was filed as a request or not. This is usually done by comparing the additionally invoked costs to the cost savings the agent at hand makes through the requested move.

If the newly generated plan contains a $U \neq U_m$, then this itself is filed as a request to B's surrounding agents (B enters into step 3).

If all involved intelligent agents execute above protocol, a network evolves, which integrates the consequences of all surrounding agents into the respective local decision making processes. Each agent is connected to its neighbors and negotiates over priority switches over them. This will lead to a network optimization, where all agents are better off in the long-term. The subsequent system behavior maximizes the overall utility. Although agent A might not be better off this time, it will benefit in the long-term when e.g. a request from B can be neglected for the good of A. The agents are coordinating their moves in order to come to a conjunctly improved network performance (In Section 5 of his Ph.D. thesis Van Katwijk () provides an example of how this structured coordination can lead to an improved resolution of a congestion issue). Naturally, this invokes a behavior where decisions to switch priority will on average be delayed by a few time steps t_c in order to allow for a maximization of the overall performance. In order to prevent signals from maintaining their current state for too long, the designer can at times choose to counteract this.

Depending on the number of immediately surrounding agents it is in that case proposed to give a weighting factor $w < 1$ to the accumulated costs of the surrounding agents. Generally, this action is proposed if the agent at hand has a high number of neighbors. In such a high connectivity network it can sometimes be that the agents are blocking each other from switching priorities, which leads to a decision-unfriendly and therefore too static network. In contrast to that, it can also be that the network has a low connectivity, that is, each agent has only few neighbors. In this case, a measure that can be taken to increase network performance is to incorporate second degree neighbors in the decision making. This is done if A files a request to B, B then calculates the local impact of this and its new best strategy, which B then communicates to C. C responds by providing its additional costs back to B, where these are used by B to check if the proposed new plan is still optimal. If this is the case, the accumulated costs of B and C are being send to A, where a final decision is made in a similar manner as before. Which of the two ways of implementation is used depends on the traffic network and varies per intersection setup.

Before the notion of platooning is introduced in the next chapter a few summarizing remarks and a conclusion for a potential combination of the systems are made. Firstly, a taxonomy of intersection controllers was established, consisting of four general groups of controllers. However in order to get a good understanding of the latest technological advancements in this sector, a major amount of time was spend on understanding the latter two, the traffic-adaptive and the look-ahead traffic-adaptive control. What unites them is that they establish a state plan, based on a traffic model, which itself can be represented as a decision tree. In order to generate this model, the controllers require certain data, which allows for conclusions about the current traffic situations around the intersection. The look-ahead controller acquires this data through in-street inductive loop sensors and the look-ahead traffic-adaptive controller further also incorporates data from surrounding intersections and their sensors. JUNO was introduced as one implementation, which can functionally be allocated in the latter class of controllers. It uses a traffic model, which is based on internal and external sensors and it further engages in a network optimization process. Within the last Section it was shown that JUNO's traffic model centers around the estimation of a predictive departure time, which makes it well-feasible for a combination with platooning. However, it was shown that this combined application is not limited to JUNO only. The t_{ed} is a byproduct of a wide range of traffic-adaptive and look-ahead traffic adaptive controllers, which makes the eventual technical design of this thesis applicable for all of those. Although the research at hand only assesses a functional combination between platooning and JUNO it needs to be noted that the system can easily be expanded to cover further controller variations. This is due to the central element of the t_{ed}, which how it will be later shown is the key communication factor between the systems.

The last Section further went into detail with the network optimization within JUNO. By means of an example it was shown how the decentralized optimization strategy can lead be the basis for a network-wide optimization if deployed in conjunction with a communication protocol with the surrounding agents. From the fact that this strategy nonetheless consists of a number of agents that all seek to optimize their own objective, a conclusion can be made for the design of the deliverable of this thesis. That is, if this design does not intervene or frustrate the introduced protocol and it further contributes to the optimization of the individual intersection performances, then it also contributes to the network performance. The technical design which is introduced in the next chapters is therefore designed to only interact with one intersection at a time. This allows for a functional combination of the systems, which requires no further considerations in terms of network optimization.

2.1.2. PLATOONING

By now it should be understood that platooning is a highly promising technology and a potentially valuable addition to our current traffic system, especially when combined with an intersection control system. The technology holds various, conceivable benefits and is in its core a well-understood concept. Furthermore, different prototypes and pilot projects have shown the general feasibility for highway implementation and proven its practical functionality. Only lately, six GMOs have successfully sent truck platoons from their respective headquarters to the port of Rotterdam, within the context of the European Truck Platooning challenge. Scania, covering more than 2000 km and four borders has had the longest route in this (DMI, 2017). Yet, this is only the latest proof of doability in a highly dynamic research field. Much more, partly divergent, investigations have been made on various potential platooning implementations.

Considering that especially highway platooning represents a brown rather than a green field, the chosen approach of this thesis is to model the system as it is, instead of inventing a new or adopted version of platooning. After all, one major aim of this research is to assess the real-world impacts of the technology, as it is likely to spread on urban streets in the future. Consequently, within this research, platooning is assumed to be a proven concept, which is treated in a 'fixed system'-manner, meaning that the functionality of platooning is understood as non-negotiable. Plenty of concepts exist, illustrating how a platoon is supposed to function, communicate and behave in common traffic situations. In order to get a generic, yet reasonably diverse overview of these concepts, four leading platooning approaches are analyzed in the following. These are chosen as such, that they represent a 'proven' concept, which has already shown its workability in road tests. The analyzed concepts are namely: The SARTRE project, PATH, GCDC and Energy ITS. The goal of this is to establish a common understanding of platoons, which can then be modeled. Relevant information and data is collected as such that it allows for a model deployment within the external driver model of VISSIM.

Within the next sections, this Section reviews the four platooning systems and eventually establishes a platooning definition, which represents a reasonable combination of the projects, which is theoretically applicable for urban use. On one hand this provides an increased understanding of platooning and on the other it leads to establishing a technical interface, connecting the platooning system with the deliverable. The former is especially relevant for Chapter 3, in which the theoretical knowledge from this Section is transferred to a VISSIM model. The interface, taking the form of a definition of an 'average' platooning system, sets the design boundaries for this model. Its purpose is to assure that the platooning model comes as close to real-world platooning technologies as possible.

THE PROVEN CONCEPTS

In 2009, the SARTRE project was founded by the European Commission with the aim of integrating existing driver assistance solutions towards a working platooning concept for highways. The final concept (see EC, 2017), which was partly developed by Volvo Trucks, features a manually driven lead truck that is autonomously being followed by both, other trucks and passenger cars. The following vehicles (FVs) are longitudinally and laterally automated, which means that no driver interaction is necessary, once a platoon is formed. Furthermore, following vehicles can enter or leave the platoon during operation.

The concept, which does not require any infrastructure modifications, is based on V2V communication, using the ITS-G5 standard (Bergenhem et al., 2012a), which is a common standard also for the other projects. The signals that are shared comprise all relevant sensor data collected among the vehicles of the platoon and planned, as well as executed, braking or steering actions. The control over following vehicles is partly external, which means that the lead truck gives unilateral orders, specifying a desired trajectory. Yet, also internal orders are possible, such as desired longitudinal movements in order to maintain the prescribed vehicle clearance. The internal orders are solely based on in-vehicle sensors, whereas external orders can originate from sensor data that does not come from the platoon leader (Bergenhem et al., 2012b). Both of those orders are executed through the in-vehicle control system, which uses the car's actuators to realize said orders. In 2012 a road test in Spain has proven the general functionality of the system, given a five vehicle platoon.

Although PATH (see Berkeley, 2017), in contrast to the SARTRE project, aims at up to ten vehicles per highway platoon, only four-vehicle platoons were tested yet. The study assumes that all vehicles of the platoon are fully automated, including the platoon leader. The absence of a human driver is however compensated through the fact that dedicated highway lanes are assumed. Following from this, no solutions are in place to allow for lane changing. Furthermore, no combined platoons exist. Here, the truck platoon is the leading research focus but considerations on car platoons were made as well. However, within the initial research report (Zabat et al., 1995) it is stated that differences in braking and acceleration forces of trucks and cars made a combination practically impossible at that stage. Yet, there are thoughts about considering combined platoons in follow-up projects.

Together with SARTRE, PATH is the project with the highest level of data awareness. All sensor data, as well as intended longitudinal or lateral movements are communicated. Yet, it is also the project with the highest infrastructure requirements. The lateral position of every car is measured against road surface markers, which themselves are located on the dedicated platooning lane.

GCDC stands for the 'Grand Cooperative Driving Challenges', which took place in 2011 and 2016 (see CP, 2017). During the latter event (see GCDC, 2017) research teams from six different nations competed on platooning in both, urban and highway situations. With lateral control being non-automated, all vehicles had a driver, conjunctly, keeping the convoy in lane. Although this is strictly speaking not considered platooning, it is one of the only studies that successfully facilitated longitudinally-automated convoying on urban streets.

The navigation exercise the teams had to master comprised non-signalized, intersections, signalized intersections, ramp meters and roundabouts.

Lastly, in 2008 the Japanese Ministry of Economy initiated the Energy ITS project. The project (see Tsugawa, 2013) initially aimed at quantifying the potential benefits that truck platooning can bring to national streets. Within a test track environment, which resembles a highway, convoys of three trucks were driven at a 10m clearance at 80 km/h. Although slight infrastructure requirements exist (road-side lane markers), it is noteworthy that the project has shown that completely autonomous platoons are indeed possible. It therefore represents the concept, with the highest level of automation among the reviewed projects. Concluding the final report (Tsugawa, 2014), 15% energy savings were found reachable for the given test setup.

An overview of the analyzed platooning systems can be found in Table 2.1. The characteristics that are summarized here comprise all aspects that are relevant for a successful implementation in a VISSIM simulation. Note that some of the characteristics remain 'unknown'. Here, no information could be found in the available literature, which is why those projects are neglected when determining a certain characteristic of the 'average' platooning definition.

Table 2.1: Simulation-relevant characteristics of various platooning systems (based on (Bergenhem et al., 2012a), (Bergenhem et al., 2012b), (Robinson et al., 2010), (Zabat et al., 1995), (EC, 2017), (GCDC, 2017), (Berkeley, 2017), (Tsugawa, 2013), (Tsugawa, 2014), (CP, 2017)).

Characteristic	SARTRE	PATH	GCDC	Energy ITS
Platooning characteristics				
Maximum number of vehicles	5	4	5	3
Platoon composition	Trucks & cars	Trucks or cars, not mixed	Trucks & cars	Trucks only
Application environment	Highway, mixed traffic	Dedicated highway lane	Urban & highway, mixed traffic	Highway-like test track
Vehicle dynamics				
Maximum speed	90 km/h	Highway speed	unknown	80 km/h
Vehicle clearance	≈ 10 m	4 m (± 20 cm)	unknown	10 m
Lane changing	Yes	No	Yes	No
Vehicle communciation				
Planned trajectory	Yes	Yes	Yes	Yes
Sensor data	Yes	Yes	Yes	Yes
Longitudinal motion	Yes	Yes	Yes	Yes
Lateral motion	Yes	Yes	No	Yes
Vehicle control				
Leading vehicle (LV)	Human	Autonomous	Human	Autonomous
Longitudinal control	Yes, through cooperative adaptive cruise control (CACC)	Yes	Yes	Yes
Lateral control	Yes, through lane keeping assistant (LKA) system	Yes, through road surface markers	No	Yes, through road-side lane markers

MODELING BASIS FOR PLATOONING

Given the characteristics of the various platooning implementations from Table 2.1 and the explanations of each concept, provided in Section 2.1.2.1, a common understanding of platooning can now be established. However, this understanding is not yet concerned with finding an average definition of platooning. In order to shed some more light on this, the platooning definition can be seen as the modeling goal. A real-world technology, which the model should represent as accurately as possible. In contrast to that, this Section answers the question of 'what constitutes such a model?' As illustrated in Figure 2.12, this mainly aims at understanding and mapping the signal processing approach of the generalized platooning concept. Providing yet another input for the coming Chapter, the signal processing is a rather technical analysis, concerned with the general structure of the to-be-developed model. The purpose of this is to establish a modeling basis for the next Chapter. In this Section it will be shown that the platooning controller is the main piece of what constitutes the functionality of platooning. It is concluded that designing such a controller is the best way of modeling the later-on provided definition of platooning. However, this controller has to make use and is closely bounded to the technicalities of the signal processing of the vehicle it is controlling. Hence, in order to allow for such a controller design, this Section first elaborates on said signal processing.

For understanding, which importance the control of a vehicle holds, one has to understand that almost every motion or action of a car, which is not triggered through the driver itself is triggered through its internal control system. The degree to which this control system takes over driving tasks can be classified by means of e.g. the SAE framework or the NHTSA framework (SAE et al., 2014). Both propose various levels of autonomy, which are briefly introduced in Table 2.2.

Table 2.2: SAE Framework about the five levels of autonomous driving. Additional remarks of the NHTSA framework are included.

Level	Definition	Requirements and characteristics
Level 0	**Zero automation** - Driving as usual	A human driver is required at all times to operate the vehicle safely.
Level 1	**Driver assisted / Function specific** - Intelligent features add a layer of safety and comfort	A human driver is required for all critical functions. However, the car can alert the driver about critical conditions, environments and obstructions. It can also offer assisted driving functions and is capable of smart driving support.
Level 2	**Partial automation / Combined autonomous function** - Key automated capabilities become standard but driver is still in control	At least one autonomous task is performed without the help of the driver. This however, is limited to a set of use case scenarios.
Level 3	**Conditional automation / Limited self-driving** - The car becomes a co-pilot	The car manages most safety critical driving functions in known (mapped) environmental conditions. Even in those conditions, a human driver is still present and expected to manage and supervise the vehicle operation.
Level 4	**High automation** - Capable of performing all safety critical driving functions while monitoring environments/conditions in defined use case scenarios	Per NHTSA, this level represents full self-driving automation. Per SAE self-driving is fully possible in most conditions without the need for human intervention.

Level 5	**Full automation** - Vehicle operates without driver	Per NHTSA this level does not exist. Per SAE, this level represents full-time automation in all conditions and environments without a human driver.

Given, this classification of autonomy, it can be said that the form of urban platooning, which this thesis attempts to propose can be allocated in the fifth level, the full automation. Although conventional platooning is often categorized as a level-4-automatization, the proposed enhancement of being able to navigate through urban environment elevates it to the highest level. Without being restricted to a certain environment and being capable of mastering a broader range of conditions, what is aimed for is the highest degree of autonomy.

Now in order to reach this degree of autonomy, the vehicle control must fulfill the requirements and possess the characteristics not only of the fifth but of all preceding levels. That is, it must incorporate general driver assistance systems, such as an ABS (level 1), it must be able to accurately observe and assess its environment (level 3 and 4), and it must eventually be able to process this information and translate it into a vehicle actuation, which safely navigates it, or in this case the platoon, through traffic (level 5). From these requirements, it can be derived that the system is in need of sensor data, guaranteeing observability of the environment. This implies a similar signal processing setup as that of an intersection controller (cf. e.g. Figure 2.4). It was already explained in the previous section, how a controller can be used to intervene in an uncontrolled system, which itself is surveyed by said sensors. The signal processing setup of a vehicle is very similar to that. As displayed in the top layer of Figure 2.12, the uncontrolled system is the vehicle itself. It is brought into motion through a set of actuators (i.e. throttle, brake, steering column), which, in case of level-5-automation are controlled through what is here called the 'vehicle controller'. The vehicle controller acquires data from a set of sensors and translates this information into an actuator intervention (i.e. actuating the brake). If the controller is well designed the vehicle is then capable of accurately following a prescribed trajectory plan, which itself is an input for the vehicle controller. In the ideal case, the 'Vehicle movement' on the right side of the Figure equals the prescribed trajectory on its left side.

Yet, it needs to be kept in mind that a platoon consists of multiple such autonomous vehicles. Following this notion, it needs to be understood that a platooning controller is nothing else than the decentralized conjunct functionality of a set of vehicle controllers. Yet, in terms of vehicle control, a general distinction between lead vehicles and following vehicles is made. Although both are constituted through the same controller setup, their system inputs vary. The following vehicles receive a prescribed trajectory, which is based on the movement of the platoon leader. This usually aims at maintaining the same lateral position and a predefined safety gap between the vehicles. The system input of the lead vehicle on the other hand, is either fixed through the driver, who can e.g. define a destination, or in certain situations through an external entity. Especially the latter case is relevant for this thesis, whereas the external entity will later be chosen to be the intersection controller. For now it needs to be understood that the design of a vehicle controller in the next chapter needs to allow for this functionality. It is therefore a logical consequence, that the controller should make use of a signal processing approach, as it was shown in Figure 2.12. That is, if the illustrated signal processing approach is brought into place by means of a good platooning controller, the system will be able to follow both, a human-based trajectory plan and that of an intersection controller.

The introduced signal processing map resembles that of the SARTRE project. The platooning concept, as it is from now on defined in this thesis, therefore builds upon the vehicle control of this project. Full longitudinal and lateral automation is assumed, brought in place through the meaningful embedment of LKA and CACC in each individual vehicle controller (cf. 'Vehicle controller box' in Figure 2.12). Note, that the LKA system allows for lateral control without any infrastructure changes. The following vehicles orient themselves on the position of the leading vehicle (V2V), as well as the lanes on the street (cf. 'Vehicle sensing technologies' boxes). The CACC is responsible for maintaining the inter-vehicle clearance and does so based on internal RADAR data (cf. 'Vehicle sensing technologies' boxes), which measures the current distance to the next vehicle. This setup suggests a vehicle clearance of roughly 10 m at 90 km/h. This represents the distance, which has been proven in some of the projects above. However, this clearance can gradually be decreased for lower velocities.

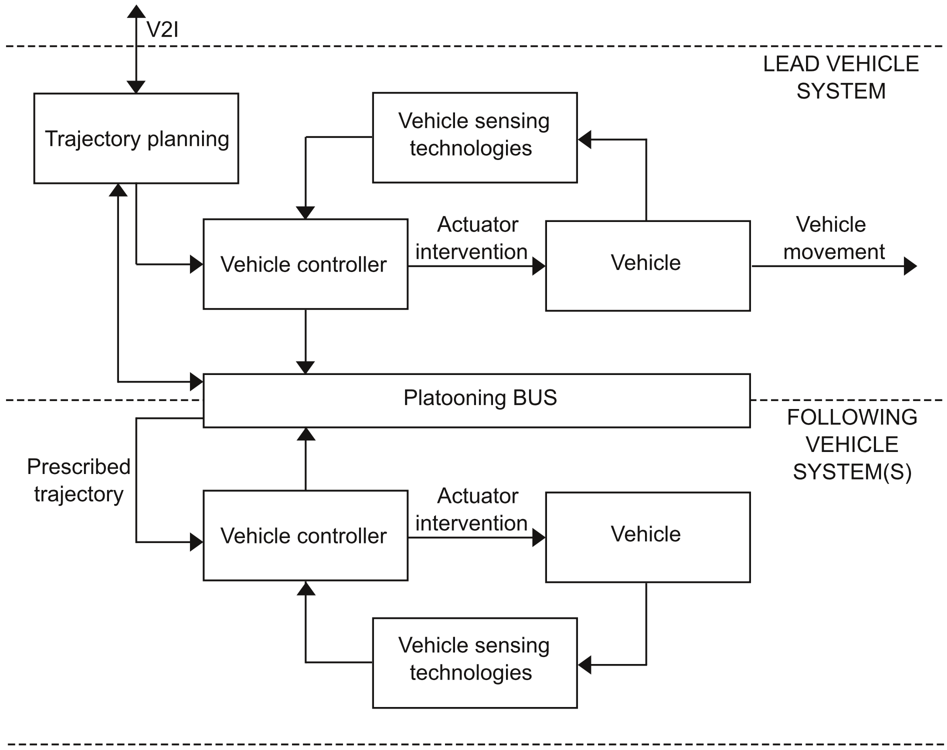


Figure 2.12: Signal processing of the combination of lead and follow platooning vehicles.

Although the applicability of the SARTRE project is yet to be proven on municipality streets, its urban functionality can be presumed. As the GCDC project, featuring a similar setup in terms of data sharing and platoon characteristics, has shown, platoons of similar vehicles can indeed navigate through a municipality environment. In this sense, if the information exchange allows for the sharing of maximum acceleration and deceleration levels, the same should be feasible for combined platoons. This research explicitly aims at a truck-and-car-platoons and in consideration of the proven feasibility for highway speeds, their applicability for significantly lower speed levels is self-understood.

In contrast to the mentioned SARTRE and GCDC, the other two concepts, PATH and Energy ITS feature complete automation of the leading vehicle. Especially the Energy ITS illustrates the functional doability of fully automated lead vehicles without significant infrastructure modifications. Hence, this aspect is also presumed for the research at hand (cf. ’Vehicle controller’ box for LV and FV).

ITS-G5 is defined as the standard V2V communication technology. It has displayed the ability to convey the planned trajectory, all relevant sensor data, as well as the longitudinal and lateral motions of the vehicles. Given the capabilities of this standard, transmitting the earlier mentioned maximum acceleration and deceleration levels does not represent a problem. Within this research, it is therefore assumed that all gathered information is available for transmission via V2V, using this standard. The V2V data sharing is in the following modeled as a platooning BUS, where all data, gathered inside the platoon is made available (cf. ’Platooning BUS’ box). In terms of simulation it will later be shown that this platooning BUS represents the data exchange between different vehicles of the external driver model. This allows for modeling the data exchange as a shared database between the platooning vehicles.

Lastly, the signal processing structure of the SARTRE project is employed once again. For the subsequent work in this thesis, it is presumed that the trajectory planning is executed solely in the leading vehicle (cf. ’Trajectory planning’ box). This resembles a human driver who naturally selects a trajectory, which appears to be right for him. It is therefore found reasonable to allocate the autonomous trajectory planning in the LV. From here the planned trajectory is distributed to all following vehicles through the platooning BUS. Each of the FVs as well as the LV incorporate this trajectory as the objective function that the internal vehicle control is trying to bring into place. A detailed illustration of the whole signal processing setup can be found in Figure 2.12.

Upon correct implementation the delineated setup allows for similar driving characteristics as the Energy ITS project. It is therefore argued that similar benefits in terms of energy savings can be made. Hence, the motivation from Section **??** appears to be reasonable and achievable for the given definition.

2.2. TECHNICAL INTERFACES

Previously, the two concomitant systems intersection control and platooning were analyzed. Firstly, four intersection control approaches were introduced and explained, whereas the latter two, traffic-adaptive and look-ahead traffic-adaptive control were subject of thorough analysis. This was done not only to elaborate the general functionality of the approaches, but also to allow for a formulation of a universally applicable interface. Despite the fact, that this thesis only illustrates an implementation with JUNO, it was shown that a broad range of traffic-adaptive and look-ahead traffic-adaptive controllers are generally feasible. A contrary approach was chosen for the concomitant system platooning. Here, four specific systems were introduced and specified in terms of a set of simulation-relevant characteristics. From this a definition of platooning was derived, which lays the basis for the formulation of the adjacent interface. In other words, the interface has to ensure the compatibility of that exact platooning definition with the design of the deliverable. As illustrated in Figure 2.1, both of the concomitant systems share an interface with the deliverable.

For each of the concomitant systems a signal processing analysis was employed. These can now be integrated to illustrate how the different concomitant systems will function together. This is depicted in Figure 2.13, whereas the chosen intersection controller approach represents the look-ahead traffic-adaptive control JUNO. As previously shown this can be adopted to a traffic-adaptive intersection controller by simply neglecting either the 'External sensing technologies' block. In Section 2.2.1, the embedment in the intersection control is specified by means of defining all connecting arrows and the interactions with the surrounding subsystems. This is followed by the interface definition of the platooning system in Section 2.2.2.

2.2.1. INTERSECTION CONTROL SYSTEMS

As displayed in Figure 2.13, the traffic side of the to-be-designed deliverable receives various kinds of traffic data, either from its own sensors or from sensors that belong to nearby intersections. Both inputs however, can and should be specified in the same format. Assuming that all intersections deliver the same data standard does not only ease the network optimization of a traffic system, it also ensures that the deliverables can function in all considered intersections. In this sense, Favilla et al. (1993) propose that two induction loop detectors, each being capable of detecting up to ten enqueued vehicles are installed in front of every intersection entry. The first one is located directly at the stop line. The second one in a variable distance d_1, which depends on the allowed maximum speed at the intersection. The specific number of sensors and their setup is dependent on the specific road situation, yet it can at this point be said that loop detectors are considered as the sole intersection sensing technology. These loop detectors are activated through the presence of a car. However, they do neither measure the speed, nor the velocity of that car. Van Katwijk (2008, Section 2.3.2) tackles this by proposing a total of five sensors in front of each entry. These are positioned according to the maximum velocity and the yellow time of the traffic light. This allows for calculating an average speed between two sensors. For the research at hand, it is therefore defined that the traffic controller receives two to five sensor signals per intersection entry. The subsequent sensor data is requested through the traffic controller every t_c. However, the sensors themselves provide continuous data. The request cycle time t_c can consequentially be modified at wish.

In case of the look-ahead traffic-adaptive controller, predictions are made based on up-stream information from other intersections. It is assumed that the sensor data from these intersections has the same format as that of the intersection under consideration. Furthermore, the data is assumed to be available upon $t_c + 1$.The same goes for requested actions U_{req}, which are also conveyed between nearby intersections. This guarantees that the network-optimization protocol between the intersections can be executed without delay.

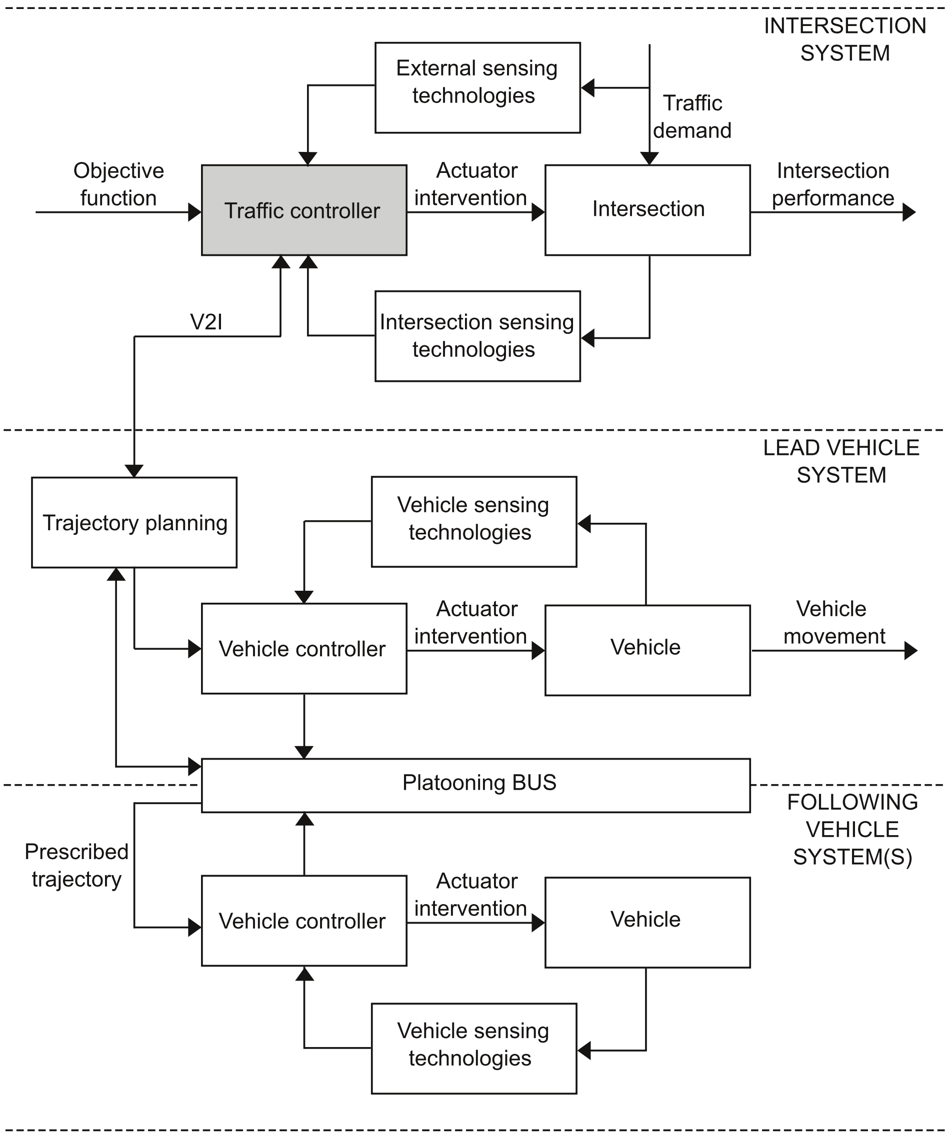


Figure 2.13: Signal processing of the combination of look-ahead traffic-adaptive controllers and platooning systems.

2.2.2. PLATOONING

In Section 2.1.2.2 a signal processing structure was established, which serves as a basis for modeling the platooning concept. This concept is in the following defined by means of synthesizing the gained knowledge about the four platooning projects from Section 2.1.2.1. In this Section a set of global platooning characteristics is derived, which constitute the modeling goal for the next chapter. These are summarized in Table 2.3. The various table entries are relevant, partly for the simulation in VISSIM and partly for the interface with the deliverable design. For the latter, the 'Vehicle communication' and the 'Vehicle control' aspects are especially important. The interface is designed as such, that a V2I communication between the 'trajectory planning' and the 'Traffic controller' subsystems is established (cf. 'V2I' arrow in Figure 2.13). This channel facilitates all communication between the platoon and the intersection, hence the deliverable.

Within the SARTRE study, ITS-G5 has proven to be a feasible standard for V2V communication. As mentioned previously it is the option of choice for establishing a platooning BUS. Yet, the PATH project has demonstrated that the same information could even be sent and processed, based on a lower communication standard, i.e. a "166 MHz communication rate" (Bergenhem et al., 2012b, p. 3). ITS-G5, operating at ~ 5 GHz is therefore found to be sufficient not only to enable V2V but also V2I transmitting and the communication of maximum accelerations and declarations between platooning vehicles. Consequently, all information exchange between platoon and intersection can be facilitated through ITS-G5. The underlying hardware constituting this communication is a set of WIFI devices with an antenna, placed in each truck and at the traffic light. The devices send, receive or forward data between all involved entities, allowing for a range of roughly 150 - 250 meters(Bernais and Lotz, 2014; Li, 2015) between each device. For the design of a vehicle controller and its interaction with an intersection controller, it is therefore assumed that platoons are able to communicate with an intersection as soon as they enter into this range. Furthermore, it will be shown that some of the intersections of the N260 have sensors that are located outside of this range. In that case an additional intersection antenna, which allows for communication up to the sensor location is assumed.

The interface is defined as such that all vehicle communication data, which is made available through the platooning BUS is indirectly also made available for the to be designed intersection controllers. Although the intersection only communicates with the trajectory planning process of the LV, this process is already including the vehicle communication data. That is, the platooning data is pre-processed within the trajectory planning subsystem. Here, potential trajectories are generated, based on e.g. the longitudinal and lateral velocity and acceleration of all vehicles. In other words, the trajectory planning entity is aware of which trajectories are possible and which are not due to physical constraints of following vehicles. If e.g. a platoon is moving at a constant 20 m/s and the maximum deceleration is -5 m/s^{-1}, the minimum braking distance is 40 m. This exemplifies the importance of communicating these maximum values. As cars and trucks are capable of essentially different deceleration levels the LV needs to be aware of those at all time to plan a trajectory appropriately. Furthermore, the trajectory planning can communicate this knowledge with the intersection. In this sense, the velocity, which is measured through in-vehicle sensors is translated in relevant information for and then communicated with the intersection. In a similar manner all information, which is relevant for allowing the platoon to pass the intersection is preprocessed and then made available for the traffic controller. More information on how this communication is structured is provided in Section 3.2.

Yet, before going into the actual modeling a few important remarks still need to be made. Firstly, in contrast to the intersection, where sensor data is processed only every t_c, the in-vehicle sensor data is in real-life available at all times. All data that is collected or generated amongst the platooning vehicles is immediately made available through the platooning BUS. However, as mentioned earlier, transmission delays are not taken into account. To compensate for this assumption, an artificial delay time is introduced. This is set to the time of a simulation step $\tau = 0.1s$. This is a worst case scenario as the delay for ITS-G5 is usually significantly smaller. Yet, this is done not only for practical simulation reasons but also because this represents the cycle time t_c of the traffic controller. It is therefore a reasonable assumption that the system as a whole is working on a tact of $t_c = \tau$. Consequently, both the trajectory planning system and the traffic controller are continuously exchanging information with a delay of 0.1 seconds. Secondly, the whole data distribution is assumed to be flawless. Within this research faulty data or wrongly transmitted data is neglected. However, this could potentially be subject of a follow-up research project.

Table 2.3: Definition characteristics of platooning for the research at hand.

Characteristic	Definition
	Platooning characteristics
Maximum number of vehicles	5
Platoon composition	Combined
Application environment	Urban & highway
	Vehicle dynamics
Maximum speed	90 km/h
Vehicle clearance	~ 10 m at 90 hm/h
Lane changing	Yes

Vehicle communciation	
Planned trajectory	Yes
Sensor data	Yes
Longitudinal motion	Yes
Lateral motion	Yes
Vehicle control	
Leading vehicle (LV)	Autonomous
Longitudinal control	Yes, through cooperative adaptive cruise control (CACC)
Lateral control	Yes, through lane keeping assistant (LKA) system

3

PLATOONING IMPLEMENTATION

In this Chapter the generically introduced concomitant system platooning from Chapter 2 is modeled and embedded within a VISSIM environment. More specifically, the Chapter comprises the implementation of platoons, first in the so-called external driver model (EDM) and then its calibration in the VISSIM N260 simulation. This implementation is based on the signal processing that was mapped in Figure 2.13, whereas the calibration aims at representing the platooning characteristics from Table 2.3. It was already mentioned that the modeling approach of choice is to implement a platooning controller, which essentially is a functional combination of a set of vehicle controllers. Section 3.2 sheds more light on how a deployment of this approach is made possible in VISSIM.

Besides the average platooning characteristics, that were carved out in Chapter 2, there is a set of behavioral characteristics that were not mentioned yet. This comprises the driving behavior of a platoon and its individual vehicles, the platoon dynamics, their formation and their decomposition. As these aspects are not specific for a certain platooning implementation, but hold relevance for all kinds of platooning, they were not mentioned in the previous Chapter. Yet, they are equally important for modeling a platoon and especially its underlying control algorithm. Consequently, this Chapter attempts to synthesis them with the modeling goal and the underlying signal processing that were already elaborated on. This attempt is based on selected research about the dynamics and behavior of a platoon. Similar to the characteristics from Table 2.3, the VISSIM model attempts to imitate these characteristics as closely as possible in order to allow for an accurate evaluation of subsequent simulation results.

Furthermore, the modeling process itself is supplemented through or embedded in literature where necessary. For those that are not familiar with the design of controllers, Section 3.1 provides a brief summary on the chosen design strategy for this kind of challenge. Based on this, the EDM is deployed within a Dynamic Linked Library (DLL), which is coded in C++. Besides the underlying information architecture of this DLL, Section 3.2 also explains how and why this EDM approach allows for an implementation of the proposed signal processing structure. This is followed by an analysis of the VISSIM-internal driver model in Section 3.3. Everything after this Section is a theoretical description of the actual platooning controller algorithm development, which is framed through the earlier established information architecture. The conversion from this theoretical, mathematical or rule-based decision-making process to C++ code, which is implemented in the DLL is not further described. However, the EDM DLL and all other code that is produced during this work can be requested from TNO.

In this sense, Section 3.4 explains how the VISSIM internal driver model needs to be modified in order to allow for conventional platooning. This is essentially, the development of a longitudinal car-following controller. It is supplemented with explanations on the implementation of platoon formation and decomposition in Sections 3.5 and 3.5.3. The platoon formation touches upon the topic of 'Virtual platooning', which is an essential aspect for the combination of the platooning controller with a signal traffic controller. Section 3.5 provides a general explanation of this in order to lay a basis for Chapter 4 which will elaborate on virtual platooning in detail.

The introduced Sections are of rather technical nature. In order to make the controller design understandable and explain its functionality and its purpose, the Chapter closes with a summary in Section 3.6. This comprises a brief explanation of the technical challenge, the subsequent technical design and its contribution to the higher level objective of minimizing emission and congestions. This section is aimed at explaining the controller design process for those who are not familiar with the details of controller theory. Together with Section 3.1, this frames the chapter at hand, adapting a higher-point of view.

3.1. TECHNICAL DESIGN STRATEGY

The design challenge that is associated with this chapter is the development of a platooning model that in its functionality and characteristics resembles that of real-world platoons as closely as possible. For addressing this challenge a so-called "model-based design"-approach is chosen (Ahmadian et al., 2005). Model-based design describes a method that is applied for developing complex control and signal processing systems without actual physical testing. Instead of going through iterative prototype-improvement circles, a mathematical or computational model is generated, which accurately depicts the real-world challenge (Reedy and Lunzman, 2010). The technical solution is then fitted to meet the demands of the challenge-model rather than the actual challenge.

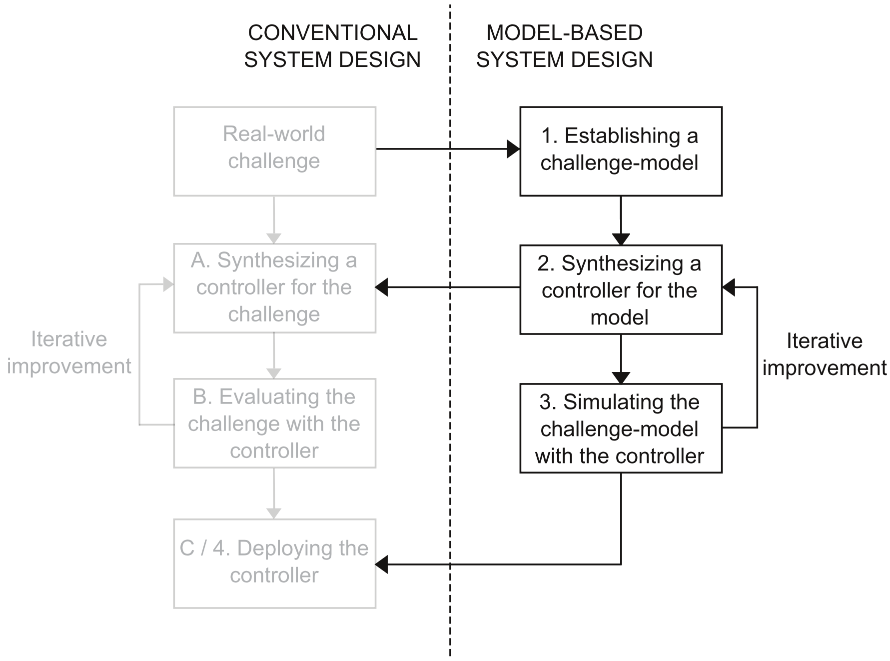


Figure 3.1: Illustration of the major steps in conventional system design and model-based system design.

The design process of model-based design is analyzed and described in various literature (see e.g. Isermann, 1996; Reedy and Lunzman, 2010) and it is commonly agreed-upon that it strongly resembles that of conventional system design. As illustrated, in Figure 3.1, the model-based design steps 2-4 are essentially the same as the conventional design steps A-C. The difference however is, that the developed controller is not meant to solve the actual challenge, but it is rather fit to master the challenge-model. Hence, for the real-world success it is of high importance that this challenge model depicts the real-world challenge as accurately as possible. Only through this it can be ensured that the theoretically developed controller is fit to master the real-world challenge in step 4. Hence, close attention should be drawn to step 1, where this challenge model is generated. In the following, all four steps are subjected to a paragraph which elaborates on their role for the research at hand, with special regard to the fist step.

In the context of the research at hand, the to-be-mastered challenge is the formation of a platoon. More precisely, it is the actuation of several vehicles so that these individual vehicle systems together form a string of vehicles, which then moves as one entity. As mentioned in the previous literature review on platooning systems, this behavior can be reached if every platooning vehicle maintains a certain pre-defined safety gap to the precursor vehicle. Hence, it is the decentralized collaboration of multiple vehicle controllers which enables platooning as a whole. Hence, the platooning challenge of step 1 must be described as such that the development of such a controller is triggered in step 2. Section 2.2 and especially, the signal processing map from Figure 2.13 have already shed plenty of light on how this can be done. It was illustrated that two vehicles with their individual vehicle controllers are the influencing factors behind the vehicle gap. The interworking of these two vehicle systems hence constitutes the controller challenge. This interworking, which is from now on termed 'Intra-vehicle system' must now be described in mathematical terms, as such that a computational model can be based upon it. This model must then accurately depict the origination of a gap based on the dynamics of the vehicle at hand and its precursor. This action, which is at the heart of step 1 from Figure 3.1 is described throughout Sections 3.2 and 3.4 and parts of Section 3.4.1. Effectively, the model is constituted through a specific VISSIM simulation, featuring such an intra-vehicle setup.

Step 2 is described in the later parts of Section 3.4.1. It centers around the development of a controller which is capable of regulating the system it is fitted for, namely above challenge model. Without going too much into detail with the technicalities of controller design, it can be said that is the nature of a controller to always steer a system towards a desired state. With the to-be-controlled system being the intra-vehicle system, this desired state is the constant, pre-defined safety gap to the precursor. Applied on the case at hand, its main strategy to reach this goal is to actuate the throttle and brake of the car at hand. Hence, the second step is aimed at mathematically expressing the longitudinal (negative) acceleration that is required in order to maintain the desired safety gap. Given the nature of the intra-vehicle system this mathematical expression will be dependent on the dynamics of the vehicle at hand and those of the precursor vehicle.

Furthermore, this mathematical expression features so-called controller gains. Controller gains are constant factors, which have to be calibrated in a lengthy simulation-iteration process. For grasping this idea the controller design from step 2 is revisited. Keeping in mind that the controller design is constituted through a mathematical expression that calculates a required acceleration level, based on the system dynamics of the intra-vehicle system, it can be derived that the latter information comes from a set of sensors or V2I-communication. Now in its simplest form, a controller gain can be imagined as a multiplier that is assigned to this input information. To go with an easy example, the required acceleration could be calculated as: If precursor breaks with acceleration force a_{i+1}, then apply $a_i = a_2 a_{i+1}$. In this example, the controller gain is determined to be 2 by the designer of the controller. Obviously, the controller design from step 2 will not be this simple, the underlying idea however, is the same. The difference is that a complex challenge model as that of the intra-vehicle system will lead to a rather complex mathematical controller formulation in step 2. Due to fundamental laws of controller theory this expression will feature $n_d + 1$ controller gains, whereas n_d is the degrees of freedom the system has. The degrees of freedom describe the number of influencing factors that constitute the challenge model, in this case the dynamics of the precursor and of the vehicle at hand. Logically, step 3 is centered around the calibration of $n_d + 1 = 2 + 1 = 3$ controller gains. This is done through a cluster approach, where the various combinations of gains are tested. Whenever a controller improvement is observed, a new iteration circle is started, seeking further improvement. This back-and-forth switching between steps 2 and 3 is continued until a satisfying functionality is established, whereas controller results are generated by using the VISSIM simulation model from step 1. The description of this step can mainly be found in Section 3.4.2 and following.

As it is the overall goal of this Chapter to establish a platooning model through controller theory, the design of this controller (step 2) is to a certain extent based on literature knowledge. After all it is not the goal of this Chapter to design a new platooning controller but to model an already existing one. If literature knowledge is used to ensure this, it is clearly indicated where the knowledge comes from. In this sense, note that the following although largely contributing to the overall research does strictly seen not represent a scientific contribution. It is solely the depiction of an already existing system. For the research at hand, this design strategy offers three major benefits:

1. Strong technical solutions can be developed without requiring real-world testing and hence high development investments. Through the simulation of results on model basis, multiple controller designs can be attempted without leading to increased costs.

2. Both the challenge model and the technical solution can be easily understood and adapted by other researchers. This is especially important for follow-up research that will be introduced in Chapter 8.

3. The challenge model can be quickly adapted to similar problems, settings, etc. This is of high use for the to-be-introduced policy-development framework in Chapter 6, as well as for the development of a platooning-intersection controller towards urban platooning in Chapter 4.

In the following, the first step of above-described research process is initiated. This is started off by elaborating on the information architecture, which the challenge model is based on. The further steps follow in subsequent sections, as described above. Finally, in Section 3.6 a summary of the design process is given. In this Section, the process, which is theoretically described here will be elaborated on, given the taken practical actions.

3.2. INFORMATION ARCHITECTURE OF THE PLATOON SIMULATION

VISSIM itself offers an internal driver model, which calculates the lateral and longitudinal movements of a car, based on its current state and its surrounding environment. Although this model does allow for some generic modifications of the driving behavior, it does not suffice for the simulation of a platoon of vehicles. For the purpose of fulfilling the research requirements of this thesis, it therefore needs to be replaced with a model, which is capable of imitating CACC and LKA characteristics, as determined in Table 2.3. This can be done through the 'External driver model'-API that VISSIM provides. This API offers the possibility to extract certain vehicle-specific information from VISSIM, modify them and pass them back to VISSIM in the same simulation step. This is deployed by means of a Dynamic Linked Library (DLL), which is made accessible for the VISSIM software. It defines modification rules for the vehicle parameters that need to be taken in every simulation step for individual vehicles or classes of vehicles. This means that the penetration rate of this modified driving behavior can be accurately prescribed. A description of all relevant information that are passed back and forth between the DLL and VISSIM can be found in Appendix A.1. These parameters comprise sensor data of the vehicle at hand, as well as environment data, concerned with the direct surrounding of the vehicle.

Figure 3.2 illustrates how vehicle-specific information is shared between VISSIM and the External driver model (EDM). In general, VISSIM provides a set of sensing data, associated with each vehicle. It is noteworthy, that the vehicle data of each vehicle, as well as their unique identification number are stored in a shared database, which is updated after every simulation step. This database is accessible for all vehicles. Yet, as it can be seen from Appendix A.1, each vehicle is only aware of the identifications of closely located, hence adjacent vehicles. This means that only information from adjacent vehicles can be requested by the vehicle at hand. This information architecture depicts a good representation of the 'Platooning BUS', which was introduced in Figure 2.12. That is, it provides the same functionality. Namely, the exchange of information with vehicles that are closely located. Data can be shared among the platoon and this data can be used to either plan a trajectory (vehicle leader) or deploy vehicle-following orders. Furthermore, the architecture allows for the exchange of information between vehicles that are in reach of vehicle Dedicated Short-range Communications (DSRC). This allows for communication with vehicles that strive interaction with the platoon, although not (yet) being part of it. The necessity of DSRC communication with vehicles outside of the platoon is founded in maneuvers like merging of the platoon, merging into the platoon, merging through the platoon or composition and decomposition of the platoon.

Implementing the platooning BUS as a shared database, leads to a communication delay of one simulation step. VISSIM allows for a maximum of ten simulation steps per second, which implies that the minimum communication offset is $\tau = 0.1s$. Given the natural delay of V2V and V2I communication this seems to be a reasonable representation of real-world matters. The identification of adjacent vehicles, hence the key for V2V communication is supplied through the 'VISSIM vehicle environment data', which is immediately available. The combination of both internal data and data from adjacent vehicles can then be synthesized into a set of desired vehicle parameters. This desired vehicle state is then passed back to VISSIM, which internally processes it to move the vehicle, as prescribed through the DLL. In this sense, the 'Trajectory planning'-box and the 'Platooning BUS'-box from Figure 2.13 are implemented by means of the External driver model API. As illustrated in Figure 3.2, the API overwrites the VISSIM-internal driver model, yet it uses its vehicle data

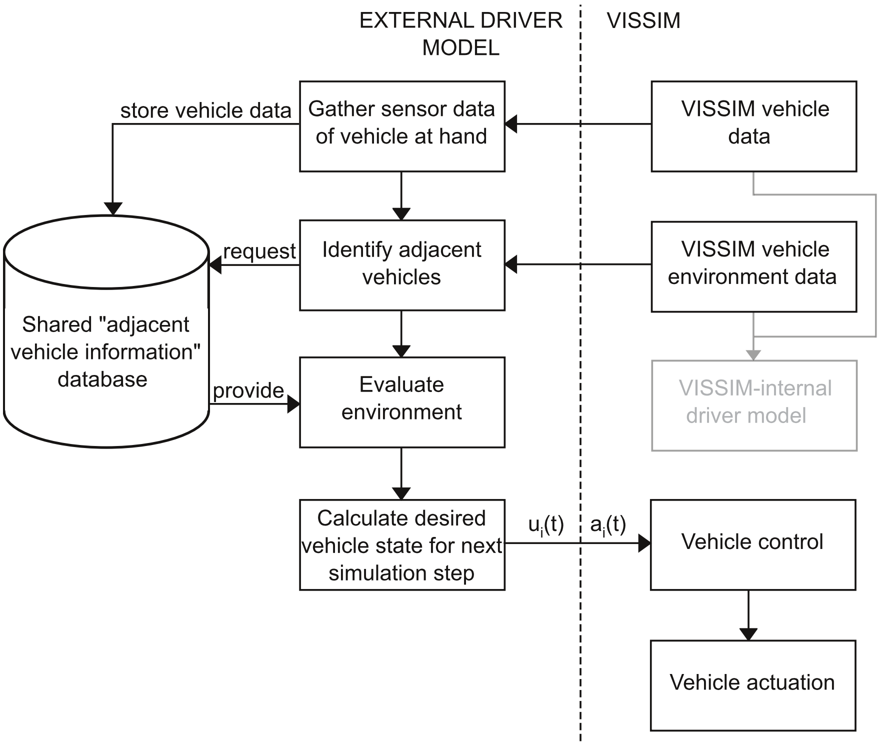


Figure 3.2: Data flow between VISSIM and the external driver model DLL in every simulation step.

and its vehicle control. In the following an accurate description of how a desired vehicle state is calculated is provided (Section 3.4). Furthermore, it is elaborated how the VISSIM-internal vehicle control is reused as an actuator for the values that are generated in the EDM.

3.3. LATERAL AND LONGITUDINAL CONTROL IN VISSIM

In order to allow for designing the EDM (cf. left part of Figure 3.2), it is necessary to understand how exactly VISSIM provides the information on the right side of Figure 3.2. In the following it is explained how the internal VISSIM driver model functions and which data it produces, which can be used for the implementation of an external driver model.

In general, VISSIM vehicles require a longitudinal and a lateral control input. By default, that is when no input is provided by the EDM, VISSIM uses Wiedemann's psycho-physical model (Wiedemann, 1974) for car following and longitudinal control, while the lateral control is founded in rule-based decision-making (Leidos, 2015). Wiedemann's longudinal control distinguishes four driving states, between which the vehicles switch, depending on their current situation. These are (adapted from PTV, 2016):

1. *Free driving*: No influence of preceding vehicles can be observed. Due to imperfect throttle control the actual speed oscillates around the desired speed level.

2. *Approaching*: Process of the driver adapting his speed to the lower speed of a preceding vehicle. The driver increases his level of deceleration until his own speed is reasonably close to the speed of the preceding vehicle. He then precedes to decrease the level of deceleration until both vehicles have the same velocity.

3. *Following*: The driver follows the preceding vehicle without consciously accelerating or decelerating. He attempts to maintain the prescribed safety gap. However, due to imperfect throttle control the safety gap error oscillates around 0. The size of the safety gap depends on the speed of the vehicle. On average itt is big enough to let at least one vehicle cut in from an adjacent lane.

4. *Braking* The driver applies a high level of deceleration if the actual distance to the preceding vehicle falls below the prescribed safety gap. This can happen if the driver of the preceding vehicle abruptly changes his speed or if a third vehicle enters into the safety gap between the first two vehicles.

The model is termed a psycho-physical model due to the fact that it incorporates psychological and physical aspects, which are representative for human drivers. This includes fatigueness, imperfections in longitudinal throttle and brake control and reaction times (PTV, 2016). Furthermore, the accelerations and declarations are limited to a maximum or minimum value. This represents either the maximum engine capability or the maximum braking force. The maximum acceleration varies, depending on the current speed and the slope of the street.

Within the VISSIM-internal lateral control scheme, the vehicles are assumed to be capable of maintaining their lane independently. Every lane is equipped with a strict middle line, around which the driving vehicles oscillate randomly. Although the level of oscillation increases with higher speeds, it is that small that the vehicles will not, at any speed cut into an adjacent lane by accident. Lane cuts are only made in the process of an on-going, desired lane change. Within the lateral decision-making process two lane changes are distinguished (adapted from PTV, 2016):

1. *Necessary lane change in order to reach the next connector to a route*: This lane change is executed at any possibility. That is, if there is a gap which suffices for the vehicle to enter the desired lane and vehicles on the new lane have enough room to decelerate for the incoming vehicle.

2. *Free lane change if there is more space and a higher speed is desired*: This lane change is executed only when a larger gap is available and if the velocity of the trailing vehicle on the new lane will allow it to keep a reasonably big safety gap.

In both cases, the lateral movement is signalized before execution. This allows surrounding vehicles to sense the lane change and prepare for it. In this sense, intersection situations are of special interest. If a vehicle crosses an intersection, it only seemingly appears to be detached from its previous lane. In fact, a VISSIM intersection consists of numerous (overlapping) lanes, which connect all possible entires and exits. It is therefore up to the longitudinal control of VISSIM to prevent vehicles from crashing into each others.

3.4. LATERAL AND LONGITUDINAL CONTROL IN THE EDM

In the following it is first elaborated how a longitudinal control algorithm is established that represents the real-life driving characteristics of a platoon, as identified in Table 2.3. For implementation into VISSIM, this controller is made subject of a code-wise implementation within the EDM DLL, written in C. The conversion from mathematical to coding terms however, is not further described. Yet, everything that is described in this chapter is established as code on the DLL.

Following the description of the longitudinal platoon control, a line of reasoning is presented, explaining why the lateral driving behavior of VISSIM, as described above is sufficient to represent platooning in the context of this research.

3.4.1. COOPERATIVE ADAPTIVE CRUISE CONTROL

As determined in Table 2.3, the longitudinal control of the platoon will be implemented by means of a Cooperative Adaptive Cruise Control system. CACC controls the throttle and brake of a car, based on a variety of V2V and sensor data. In Section 3.4 and Appendix A.1 it has already been explained which data is available for this. In order to design the EDM DLL, it now needs to be understood what exactly the goal of this CACC system is. For this purpose the four states of Wiedemann can be revisited. Clearly, the vehicle requires full automation for the latter three stages, as this allows to nullify the shortcomings of a human driver. By doing so, an automated system can lift the performance of longitudinal control to a superior level. This especially accounts for the aspects of precision (e.g. with maintaining a speed of distance) and reaction time (e.g. with sudden braking or accelerating motions), which subsequently allows for maintaining significantly shorter safety gaps. The benefits of this have already been discussed in Section 1.1.1. It is now explained how a trajectory controller is designed within the EDM layout from Figure 3.2 in order to enable this driving behavior.

The way the CACC system is deployed is largely driven by two systems. Namely, these are the individual dynamics of a car and the platoon dynamics. Each of them needs to be made subject of a controller design, as it is illustrated in Figure 2.13. In this sense, the controller, maintaining the platoon dynamics (cf. 'Trajectory planning' box) serves as an input for the individual vehicle controller, whereas the performance of the individual vehicle controller represents the feedback loop of the trajectory controller. This is illustrated in Figure 3.3. As it can be seen, the vehicle system represents a subsystem of the platoon system. Indeed, the intra-vehicle system, which needs to be controlled through the trajectory controller comprises the vehicle system and its vehicle controller. This setup is generally termed a 'cascaded controller' (Meurer, 2010). In the case at hand, the platoon controller passes a desired trajectory to the vehicle controller, which then actuates an engine (and brake) force $\hat{u}_i(t) = F_e$ in order to keep the vehicle on this desired trajectory. In terms of longitudinal control such a trajectory is provided by means of a desired acceleration level (Antonelli and Chiaverini, 2006). For the EDM from Figure 3.2 this implies that the information, which is passed back to VISSIM (between the 'Desired vehicle information' box and the 'Vehicle control' box) is in fact an acceleration value. This can also be seen in Figure 3.3. While VISSIM internally executes the individual vehicle control, the DLL needs to continuously calculate the level of acceleration, which leads to a constant safety gap for all platooning vehicles. It will do so based on a set of information, which it receives from the vehicle subsystem. This information is called the vehicle state $\hat{x}_i(t)$ and will be thoroughly explained later.

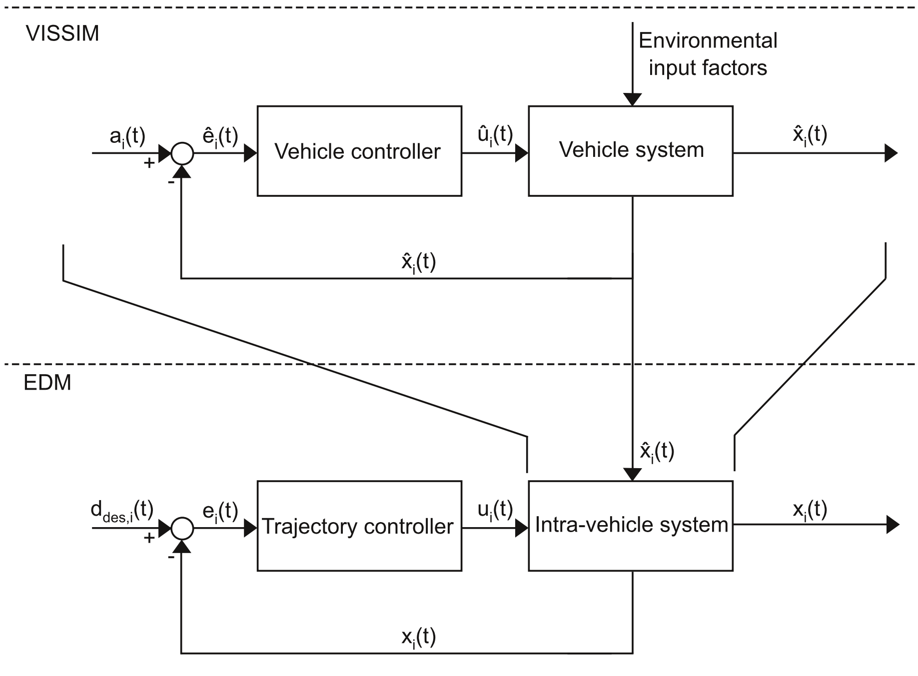


Figure 3.3: Illustration of two interlocked systems, with the individual vehicle system being a subsystem to the platoon system.

As it can be seen from Figure 3.3, the control of the individual vehicle is solely executed through VISSIM. It is only through the data exchange as illustrated in Figure 3.2, that the outer controller receives information about the inner system. In Figure 3.3 this is illustrated through the data vector $\hat{x}_i(t)$, which is passed from the inner to the outer system. Unlike this case, controller theory literature (see e.g. Meurer, 2010) suggests to first design the inside controller (here, the vehicle controller) in order to gain further understanding of the capabilities of that system to react to various inputs (here, desired accelerations). It is in the nature of a physical system that it can not reach every desired input variable and exhibits different performances for different inputs. Nevertheless, passing desired acceleration values to VISSIM has shown that the respective vehicles exhibit said acceleration values shortly after receiving them from the DLL. Although marginal time-delays were surveyed during the experiments, these are neglected in the following. As a reasonable approximation, the transfer function $\hat{G}$ of the inner system can therefore be assumed to be of linear, direct nature:

$$\hat{G} = \frac{a_i(t)}{\hat{x}_i(t)} = \hat{C} * \hat{V} = 1. \tag{3.1}$$

A transfer function describes the relation between the system input and the system output. Mathematically seen this indicates that when a certain acceleration value $a_i(t)$ is put into the system, the combination of vehicle controller and vehicle system will exhibit exactly that acceleration. This implies that the vehicle controller is perfectly designed, such as that it compensates the indirectness of the vehicle system (such as inertia delay). Note, that this does not account when acceleration values exceeding the minimum/maximum values of the respective vehicles are assigned. Consequently, this is prohibited in the design of the outer controller.

Having understood, that the vehicle controller enables the immediate execution of prescribed acceleration values, it is now explained how these accelerations have to be assigned in order to maintain a constant safety gap between vehicles. Here, the main challenge is to enable the so-called string stability. The opposite of this, string instability is a frequently discussed phenomenon in platooning literature (Ploeg et al., 2011b; Swaroop and Hedrick, 1996). It describes how following vehicles of a platoon fall into a longitudinally unstable driving motion, triggered through a (negative) acceleration of the platoon leader. As illustrated in Figure 3.5 and Figure 3.6, the following vehicles respond to this acceleration with a counterproductive level of acceleration or deceleration themselves. Even though the first vehicle has long established a constant speed again, the following (especially the last) vehicles' speed levels oscillate around the intended state, namely exhibiting the same velocity as the leader. This string instability often occurs as a consequence of an infeasible trajectory controller design. Two common mistakes here are to incorporate too few state parameters and to assign wrong controller gains (Meurer, 2010). For the design of the trajectory controller at hand this is counteracted by: Firstly, introducing a three-dimensional state vector $x_i(t)$, instead of the two-dimensional one, which is commonly used for ACC applications. Secondly, following proven procedures for the controller formulation, as described in literature. In the following, the reader is guided through the subsequent trajectory controller design:

In a platoon of N vehicles, where the index $i \in N$ indicates the number of the vehicle in the platoon, $V_{i=0}$ is the platoon leader and $V_{i=N}$ is the last vehicle of the platoon. Every V_i of this platoon eventually follows its own control objective O_i, which however is the same for all vehicles, except from V_0:

$$O_i = min(e_i(t)). \tag{3.2}$$

Here, $e_i(t)$ symbolizes the deviation of one parameter of the to-be-controlled system from a prescribed goal value. In the case of maintaining a prescribed desired safety gap $d_{des,i}(t)$, the goal value is determined to be:

$$d_{des,i}(t) = r + hv_i(t), \tag{3.3}$$

with r being the desired still stand distance, h being the time gap and $v_i(t)$ being the speed of the vehicle at hand. The choice for this goal value is thoroughly motivated in e.g. (Ploeg et al., 2014), (Gehring and Fritz, 1997) and (Rajamani and Zhu, 2002). It represents a commonly agreed upon safety distance, which allows for both, harvesting the benefits of platooning and maintaining a certain safety distance. This safety distance suffices to compensate minor deviations from desired vehicle gap and therefore the shortcomings of technology. Having understood that it is the nature of a controller to minimize e_i, the deviation of $d_{des,i}(t)$ can be described as the difference between the actual safety distance $d_i(t)$ and the desired safety distance $d_{des,i}(t)$:

$$\begin{aligned} e_i(t) &= d_i(t) - d_(des,i)(t) \\ &= (s_{i-1}(t) - s_i(t) - L_{i-1}) - (r + hv_i(t)), \end{aligned} \tag{3.4}$$

with L_{i-1} being the vehicle length of the leading vehicle. $s_i(t)$ and $s_{i-1}(t)$ represent the position of the vehicle at hand and the leading vehicle, respectively. Now if the controller realizes his objective and minimizes $e_i(t)$, this will lead to a constant safety gap $d_{des,i}(t)$. In order to follow its objective O_i, the controller can assign a controller output u_i, which itself has an influence on the system (cf. Figure 3.3). As mentioned, this output takes the form of an acceleration value, which itself is the input to the cascaded subsystem. What controller theory seeks to do is to identify u_i as a function of the current state of the intra-vehicle system x_i. As it will be shown in the following x_i is dependent on the state of the inner system $\hat{x}_i$. Translated into practical matters this means that sensors, providing certain knowledge about the state of the vehicle system can serve as a calculation input for u_i. Furthermore, the presence of V2V communication allows for incorporating the states of adjacent vehicle systems, in this case the preceding vehicles. This motivates e.g. the availability of $s_{i-1}(t)$ data for the vehicle at hand. The controller output can consequently be formulated as:

$$u_i = f(\hat{x}_i, \hat{x}_{i-1}, ..., \hat{x}_0). \tag{3.5}$$

In order to assign a meaningful function to u_i, some considerations about the system itself have to be made. Besides others, this comprises a physical description of the vehicle system. In a platoon of N vehicles, the vehicle dynamics of the lead vehicle can generally be described as:

$$\dot{s}_0(t) = v_0(t), \tag{3.6}$$

where $v_0(t)$ represents the current velocity of the vehicle. Hence, in a similar manner, the individual dynamics of the $N-1$ following vehicles are:

$$\begin{aligned} \dot{s}_i(t) &= v_i(t) \\ \dot{v}_i(t) &= a_i(t), \end{aligned} \tag{3.7}$$

with $a_i(t)$ being the acceleration of the i-th vehicle. Now, as e.g. proposed by Zheng et al. (2014) or Ploeg et al. (2014), the state vector of each individual vehicle system contains its position, velocity and acceleration:

$$\hat{x}_i(t) = \begin{pmatrix} s_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix}. \tag{3.8}$$

As mentioned before, the state vector $\hat{x}_i$ is in reality gathered through sensor data from every vehicle. As a good representation of these real world matters, the state data needs to be made available in every simulation step through the VISSIM software. In Figure 3.2 it is illustrated how the vehicle state parameters are extracted from the VISSIM-internally generated vehicle information. On the left side of Figure 3.2, this data can then be compiled towards the derivative state vector $\dot{\hat{x}}_i$:

$$\dot{\hat{x}}_i(t) = \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau} \end{pmatrix}}_{\hat{A}} \underbrace{\begin{pmatrix} s_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix}}_{\hat{x}_i} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix}}_{\hat{B}} u_i. \tag{3.9}$$

The formation of the B-matrix becomes clear when keeping in mind that the controller output u_i should be assigned as an acceleration value. Furthermore, it was mentioned that the transfer function $\hat{G}$ of the inner system is 1, which implies that u_i has a direct impact on $\dot{\hat{x}}_i$. Following from this, the derivative of the current vehicle acceleration is $\dot{a}_i = (u_i - a_i)/\tau$. In general, the derivative state vector represents the change of state, which hints towards a development of future states. $\dot{\hat{x}}_i$ being the change of state, illustrates how the combination of current state and input influence the change of state. In other words, it is a way of describing the system reaction to a certain input. In a similar manner, the output response towards a change in state can be described. In this case, the output is simply the acceleration entry of the current state vector:

$$\hat{y}_i(t) = \underbrace{\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}}_{\hat{C}} \underbrace{\begin{pmatrix} s_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix}}_{\hat{x}_i} + \underbrace{\begin{pmatrix} 0 \end{pmatrix}}_{\hat{D}} u_i. \tag{3.10}$$

At this point it should be remembered that $\dot{\hat{x}}_i$ represents the derivative state vector of the vehicle system, while indeed the aim of the trajectory controller is to control the outer, inter-vehicle system. Clearly however, the accelerations of each vehicle allow for a calculation of the safety gaps between the vehicles. Subsequently, the consideration of the inner system states, especially the conversion matrices A, B and D are of help when considering the state vector of the intra-vehicle system. Here, the aim of this approach is to identify the relation of a a (outer) controller output u_i and the intra-vehicle gaps. This view of input-output relation needs t bp adapted because the inner system, as described in(3.9) represents one part of the to-be-controlled outer system. As proposed by Zegers et al. (2016) this outer system is defined as:

$$x_i = \begin{pmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{pmatrix} = \begin{pmatrix} (s_{i-1}(t) - s_i(t) - L_{i-1}) - (r + hv_i(t)) \\ v_{i-1}(t) - v_i(t) - ha_i(t) \\ a_{i-1}(t) - a_i(t) - h\frac{u_i - a_i(t)}{\tau} \end{pmatrix}. \tag{3.11}$$

The derivative state vector for the outer system subsequently requires a calculation of $\dddot{e}_i$. Inserting (3.4) (and subsequently (3.9)) into the derivation of $\ddot{e}_i$ leads to:

$$\begin{aligned} \dddot{e}_i(t) &= \frac{1}{\tau}u_{i-1} - a_{i-1}(t) - u_i + a_i(t) - h\dot{u}_i + \frac{u_i + a_i(t)}{\tau} \\ &= -\frac{1}{\tau}\ddot{e}_i(t) + \frac{1}{\tau}(u_{i-1} - u_i - h\dot{u}_i), \end{aligned} \tag{3.12}$$

which implies the outer derivative state vector to be:

$$\dot{x}_i(t) = \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau} \end{pmatrix}}_{A=\hat{A}} \underbrace{\begin{pmatrix} e_i(t) \\ \dot{e}_i(t) \\ \ddot{e}_i(t) \end{pmatrix}}_{x_i(t)} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix}}_{B=\hat{B}} (u_{i-1} - u_i - h\dot{u}_i). \tag{3.13}$$

By mathematically transforming (3.11), the acceleration level can be found to be:

$$y_i(t) = \frac{1}{h}(v_{i-1}(t) - v_i(t)) - \frac{1}{h}\dot{e}_i(t). \tag{3.14}$$

This is because in (3.11), u_i can be set equal to $a_{des,i}$, as it was earlier mentioned that u_i is prescribed as an acceleration level. Understanding that the desired acceleration is immediately brought into place by the inner control system ($\hat{G} = 1$), it can further be concluded that $a_{des,i} = a_i$.

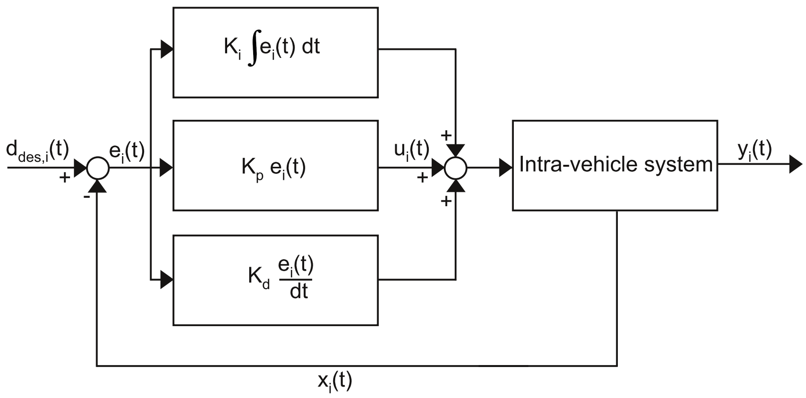


Figure 3.4: Generic structure of a PID controller as proposed by Meurer (2010) .

Similar to (3.9), (3.13) and (3.14) represent the system response to any possible $d_{des,i}$. It is noteworthy that the outer system shows the same behavior as the inner system ($A = \hat{A}$, $B = \hat{B}$). This is a very common system description, where the state of the system is a time-delayed combination of its input and previous state. Fortunately, CACC systems with the same or similar system behaviors have been widely discussed in literature. In the following the implementation and calibration of the trajectory controller at hand is described. Herewith, the controller formulation is adapted by following the methodologies as in general proposed by Meurer (2010). This is validated by comparing the controller descriptions of Zegers et al. (2016), Montanaro et al. (2014)and Swaroop et al. (1994), who all assume a system-subsystem setup as described in (3.9) and (3.13).

The case examples of Montanaro (2014) and Swaroop (1994) both propose a PD-controller, while Meurer (2010) proposes to use a PID controller. Both controllers however, are in its formulation similar. As illustrated in Figure 3.4, a PID controller comprises the sum of the integrative, proportional and derivative controller gains and their respective controller formulation terms. These terms are in general supposed to represent either $e_i(t)$ itself, its integration or its derivation. From (3.14) it can be derived that:

$$a_i = \frac{1}{h}(v_{i-1} - v_i) + \frac{1}{h}\dot{e}(t). \tag{3.15}$$

Swaroop et al. use this expression for the controller formulation, as follows:

$$u_i(t) = \frac{1}{h}(v_{i-1} - v_i)K_{d,i} + \frac{1}{h}e_i K_{p,i}. \tag{3.16}$$

$K_{p,i}$ is in this context the proportional controller gain of the trajectory controller, while $K_{d,i}$ is the derivative controller gain. This consequently represents a PD-controller. The assignment of $K_{d,i}$ becomes clear when one remember that $e_i(t)$ represents a distance. The derivative of a distance is logically a velocity or as Swaroop et al. specify the relative velocity between the vehicle at hand and the direct leader. Montanaro et al. assume a similar system and add the expression

$$u_i(t) = ... + \frac{1}{h}\sum_{j=1}^{N-i} e_{i-j}K_{p1,i}. \tag{3.17}$$

This is a modification of the second term of (3.17) and it represents the incorporation of V2V data. This is a measure that is strictly aimed at preventing string instability. String instability occurs as a consequence of the fact that the vehicle at hand only reacts to the movements, hence the derivative state vector $\dot{x}_{i-1}$ of its directly preceding leader. Equation (3.17) represents a way of leveling out these effects by incorporating not only the error value e_i, representing the distance to the preceding vehicle but also those of the j further preceding vehicles e_{i-j}. $K_{p1,i}$ is yet another proportional controller gain, which however is now applied on the vehicles that precede the direct leader.

To generally elevate the performance of a PD-controller, as illustrated in (3.16) and (3.17), Meurer (2010) proposes an integrative controller gain. As illustrated in Figure 3.4, a controller gain $K_{ff,i}$ needs to be assigned to $\int e(t)dt$. In contrast to the assignment of the derivative controller gain, there is no physical counterpart to the integral of a distance. Therefore, this information can in reality not be drawn from any sensors, which is why it is mathematically calculated within the EDM. This goes along with remarks made in (Meurer, 2010) and (Zegers et al., 2016). The eventual controller description can then be formulated as:

$$u_i(t) = \overbrace{\underbrace{\frac{1}{h}(v_{i-1} - v_i))K_{d,i} + \frac{1}{h}e_i K_{p,i}}_{\text{Swaroop et al.}} + \frac{1}{h}\sum_{j=1}^{N-i} e_{i-j}K_{p1,i}}^{\text{Montanaro et al.}} + \underbrace{\frac{1}{h}\int e_i(t)dtK_{ff,i}}_{\text{Meurer, Zegers et al.}}. \tag{3.18}$$

Equation (3.18) also illustrates, which part of the controller formulation is validated in which literature. Each of these literature pieces provides a proof of controller stability, which is why a theoretical proof of workability is not further provided within this research. However, the functionality can also be tested through a set of experiments. This goes along with the calibration of the controller and is described in the following.

3.4.2. CACC CONTROLLER CALIBRATION

Now that a stable controller formulation is established, which utilizes the assignment of acceleration values to maintain a certain predefined safety distance, this controller has to be calibrated by means of assigning values to the controller gains $K_{ff,i}$, $K_{p,i}$, $K_{p1,i}$ and $K_{d,i}$. If the system reaches a stable state, as determined through the input of the desired safety distance, controller workability can be considered proven. This does not only represent a crucial step within the technical design of the CACC controller, but also a plausibility check. This check is part of the overall verification process, which is summarized in Chapter 7. However, it is already necessary at this point, as preceding design measures will be based on this initial controller design. This section therefore concludes with said plausibility check of the assigned controller gains. In their work, Montanaro et al. (2014) propose that the $K_{p1,i}$ gain is assigned as:

$$K_{p1,i} = \frac{K_{p,i}}{\sum_{j=1}^{N-i} 1}. \tag{3.19}$$

This is adapted for the controller calibration at hand. For the assignment of the remaining controller gains a variety of methods exist. As proposed by Meurer (2010), one of the more applicable ones is the empirical cluster calibration. This describes the iterative evaluation of various combinations of controller gains. Thanks to the computing capabilities of VISSIM, a solution can be found in $K_{ff,i} = 0,7$, $K_{p,i} = 1$ and $K_{d,i} = 0,8$ for this methodology. A good evaluation method for controller performances is to monitor the step response of the system. That is, the controller is faced with a suddenly increased input, e.g. $e_i(t)$ increases from 0 to a certain value in an instant. Although, such an abrupt increase of $e_i(t)$ is not a realistic occasion during normal platooning, it however represents the worst-case scenario for the system. In other words, it is the highest possible challenge for a controller. If the system is capable of adapting to an abrupt change of $e_i(t)$ in a reasonable time, it is also capable of adapting to a gradual change of $e_i(t)$. Note, that an abrupt change of $e_i(t)$ can occur if the vehicle at hand switches its target vehicle. When this is the case, is explained in Section 3.5. For now, it is important to understand that a quick adaption to a new $e_i(t)$ is an essential requirement for the trajectory controller.

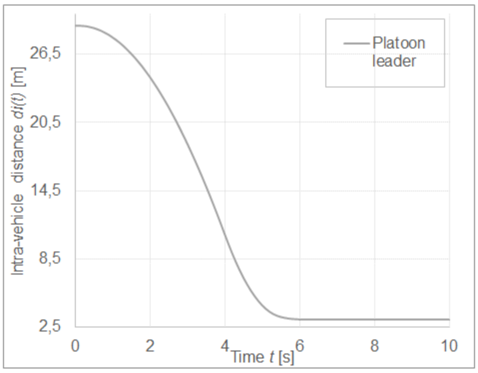


Figure 3.5: Short-term view of the intra-vehicle distance of the lead vehicle step response, initiated through the decentralized, calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 0$.

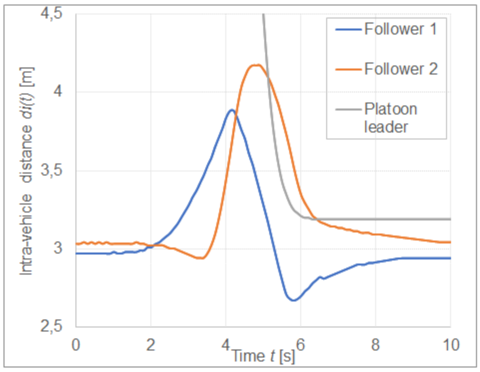


Figure 3.6: Short-term view of the intra-vehicle distance of the platoon step response, initiated through the of the decentralized, calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 0$ for the platoon leader.

The step response of the system was evaluated for values between $K_{p,i}, K_{d,i} = 0,5$ and $K_{p,i}, K_{d,i} = 1$ with a step size of $0,1$. Given this cluster of potential controller gains, the before mentioned values showed to deliver the best solution. The cluster approach was chosen, as the effects of the derivative and proportional controller gains on the step response are deeply interrelated. Determining them in sequential order will not lead to an optimal solution. In contrast to that, the integral controller gain can be determined after the calibration of the proportional and derivative controller gains. Its purpose is to prevent a lasting discrepancy between desired and actual safety distance. Besides this, its effect on the step response is neglectable, which is why it was not considered in the cluster calibration.

CACC PLAUSIBILITY CHECK

As it can be seen in Figure 3.5, the assigned controller gains lead to an adaption time of roughly $6s$. The figure displays how the trajectory controller gets triggered at $t = 0$ with a measured distance of $d_{i=1}(t = 0) = 28m$. In simulation terms this represents the desire of the platoon leader to close the gap to a preceding vehicle, which is located $28m$ away. Both vehicles are standing still at $t = 0$, which implies that the vehicle at hand did not desire to join its preceding vehicle for $t < 0$. The newly introduced desire to join the preceding vehicle then implies that the error function increases from $e_{i=1}(t < 0) = 0$ to $e_{i=1}(t = 0) = d_{i=1}(t = 0) - hv_{i=1}(t = 0) \approx 25m$ with $v_{i=1}(t = 0) = 0$. The step response of the platoon leader (cf. Figure 3.5) to this abrupt increase of $e_{i=1}(t)$ is fairly good. The proportional controller gain is high enough to enable a quick adaption. That is, the vehicle only takes $6s$ to establish the desired safety gap. Furthermore, the combination of the proportional and the derivative gain prevent the car from overshooting. The derivative controller curbs the acceleration when the vehicle gets closer to its goal, which is why the vehicle does not get too close to its precursor. It applies a negative acceleration in time to adapt a smooth step response. Here, a slightly quicker adaption time could have been reached with a lower derivative gain, however this would have caused slight overshooting. As the cars should rather have a too big than a too small safety gap, this is a willingly agreed-upon trade-off.

Figure 3.6 illustrates the driving behavior of the two following vehicles. As these are already in platooning mode, their initial safety gap equals to $d_{i=2,3}(t=0) \approx r + hv_i(t=0)$. Given the acceleration of the platoon leader at $t=0$ the followers adapt to this by trying to maintain a low error function $e_{i=2,3}(t)$. However, while doing so they increase their respective safety gap as a result of the increased velocity $v_{i=2,3}(t=0) > 0$. It can be seen that the first follower exhibits a slight overshoot during the deceleration motion. In the given context, this is an acceptable occurrence. Due to the physics of a car, there exists a trade-off between the overshoot level and the adaption time. To reach a lower overshoot, the adaption time would need to be increased. Given, the low level of the overshoot ($0.28m$ within a $3m$ safety distance), the trade-off as-is seems to be well-assigned.

Given these results, the code-wise implementation of the developed controller in the EDM DLL will allow for a car-following behavior with a safety distance that closely resembles that as expressed in (3.3). This does not only allow for platoons in the conventional sense but, as mentioned earlier, it also allows for virtual platooning. This application of the developed platooning controller is generically explained in Section 3.5, laying the basis for Chapter 4, which describes the combination of the concomitant systems - platooning and intersection control. For now, it can be stated that the controller response and its generic behavior are plausible and represent real-life matters fairly well. This finding is mainly motivated through the fact that the controller reacts well to, what can be seen as a worst-case scenario - a step response, which instantly switches from the desired distance to a value of $28m$. The error derivations $e_i(t)$ that occur once a platoon is established are significantly lower than this, which is why functionality is hereby proven for those as well.Yet, this setups does not only allow for conventional platooning but due to the fact that it can cope with high error derivations, it also oallows for adding vehicles to the end of the platoon.

3.4.3. LANE KEEPING ASSISTANCE

In this research lateral control is executed through VISSIM. VISSIM provides a driver model, which resembles the lateral driving behavior of a human driver. Although automated driving behavior is to some extent different to that of a human, this is neglectable in the context of this research. The reason for this is that the research at hand aims at evaluating the impacts of urban platooning in terms of emissions, intersection efficiency and investments. For none of the three objective functions of the traffic controller design, the lateral driving behavior has an influence. That is, implementing automated lateral control will not lead to any different results than the pre-defined human driving behavior in VISSIM. Subsequently, the EDM DLL leaves the lateral motions, as they were provided by the simulation software. It does not intervene in the VISSIM-internal driver model.

Although they are not modified, lateral movements can in some cases be suppressed. This is the case, if a vehicle is in platooning mode and therefore adapts the lateral movements of its preceding vehicle. Given that the vehicle at hand has the same immediate destination (targeted next link) as its precursor, it copies the lateral movements of that vehicle. Within the DLL this is solved with a set of decision rules, which allow for a realistic decision-making about the suppression of lateral movements. For vehicles whose immediate destination is different from that of their preceding vehicles, there exists an exception to this rule. This is elaborated on in Section 3.5.3 on platoon decomposition.

3.5. PLATOON FORMATION AND DECOMPOSITION IN THE EDM

In the introduction chapter it was explained that one major motivation to establish urban platooning is that it allows to extend the range of highway platooning. In the course of this explanation, it is mentioned that an essential part of this is the composition and decomposition of platoons before they enter or after they leave a highway. From this it becomes clear that this process is shifted towards the urban road network, making it clear why a technical solution is needed to enable this. In the following, this solution is presented. A new platooning mode - SLVP - is introduced the decomposition of platoons is explained.

Within the EDM DLL vehicles are classified in a set of general categories. In the following, these are termed ’type 0’, ’type 1’, ’type 2’ and ’type 3’. All longitudinal formation motions are made based on the classification of the vehicle. More precisely, different types follow different longitudinal behaviors. That is, each type employs a different controller setup for their respective longitudinal driving behavior. In the following these are introduced, before further elaborations re made on the technicalities of type 2 vehicles.

Type 0 vehicles represent vehicles that make use of the longitudinal VISSIM-internal driving behavior. That is, they do not follow any prescribed trajectory and their driving behavior represents that of a human driver. It can only be the case for non-platooning vehicles or platoon leaders. They are able to determine their own trajectory, which is if applicable distributed to all following vehicles. In case of a platoon leader, some actions that are suggested through VISSIM might however be suppressed through the EDM. An example is an intended lane change motion. VISSIM suggests such a motion based on the immediate surrounding of the LV. The EDM DLL additionally considers sensor data from potential FVs and makes decisions about the suppression of lateral movements accordingly. Type 1 indicates the role of a vehicle as a conventional platoon follower. Its driving behavior is automated, hence these vehicles make use of the EDM longitudinal control. Similar accounts for type 2 and type 3 vehicles, whereas the latter will be introduced at a later point. The subject of this Section is type 2. All vehicles can change their type in every simulation step. E.g. a type 0 vehicle can become a type 1vehicle if it joins a platoon (non-platooning vehicle) or if another vehicle becomes the leader. In the following it is explained when a vehicle will switch to type 2.

3.5.1. TWO-LANE FORMATION OF PLATOONS

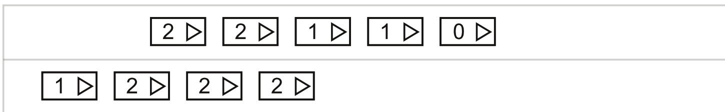


Figure 3.7: Two platoons of vehicles, which detain each other from merging onto the other lane.

The introduced longitudinal CACC controller allows for keeping significantly lower safety gaps between the individual vehicles of the platoon. However, this also brings along a set of problems e.g. in merge-in, merge-through or lane changing situations. Due to the decreased safety gap, vehicles that are not part of the platoon have no chance to merge through or into the platoon. If the platoon reaches a significant length, non-platooning vehicles will therefore not be able to enter the lane of the platoon. An example of this is provided in Figure 3.7, where two platoons detain each others from entering their opposing lanes. Disregarding whether there is a single vehicle attempting to merge into a platooning lane or a whole platoon, this can be tackled through so-called 'virtual platooning' (see e.g. Medina and Nijmeijer, 2017). Virtual platooning (VP) describes the ability of a car to maintain a constant longitudinal distance to a vehicle, which is not on the same lane. In this work two applications of VP are distinguished - 'Multi-link virtual platooning' (MLVP) and 'Single-link virtual platooning (SLVP). The latter one describes the ability to follow a vehicle that is located on an adjacent lane on the same link (i.e. the same street or intersection entry). In contrast to that MLVP describes the adaptation of a longitudinal trajectory according to a vehicle that is located on a different link. This, as well as the vehicle classification 'type 3' will be explained in detail in Chapter 4.

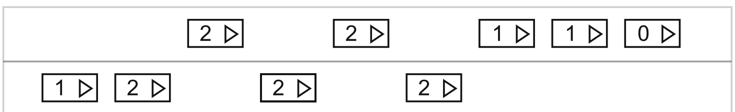


Figure 3.8: Creation of merging gaps through virtual platooning.

SLVP can be used to create merging gaps between the individual platooning vehicles without leaving the automated driving mode. Assuming that all vehicles from Figure 3.7 intend to change lanes, cars that are not following anyone are classified as type 0. Vehicles that follow a preceding car on their own lane are marked as type 1 and cars that use SLVP to follow a vehicle on a different lane are indicated with a 2. The latter classification is only assigned if a merging conflict exists with at least one of the vehicles of the platoon. Only if this is the case, a vehicle can switch from type 1 to type 2, whereas it will switch back as soon as the conflict is resolved. Within the implementation of the EDM DLL, a car will make this switch from normal platooning to

SLVP as soon as a merging conflict is detected through the on-board sensors. This is the case, as soon as the conflict vehicle sets its turning indicator to the respective platooning lane or if a platooning vehicle intends to merge onto a lane on which a conflict vehicle is driving. All vehicles that have been classified as type 2 due to this conflict, will then engage in the creation of a merging gap. Assuming two involved vehicles, these do so by establishing a longitudinal distance of $d_{des,i}(t)$ between them. This is done theoretically mirroring of the longitudinal position of the adjacent-lane vehicle on the respective other lane. Here, a maximum longitudinal detection distance $d_{SLVP,max}$ is employed after which a vehicle is not considered in conflict anymore. For the vehicle which is determined to trail (i.e. the one that is behind at the moment of the conflict detection), a virtual car is created on the own lane, hence the term virtual platooning. In this case, the virtual car represents the necessary merging gap for the actual car to join the lane.

If the two conflicting vehicles on adjacent lanes have the exact same longitudinal position, i.e. they are next to each other, no SLVP actions are taken in that simulation step. The platoon will continue to drive in normal platooning mode. Yet the conflict is registered in the EDM. In the next simulation step it is checked wether one of the two vehicles has assumed a leading position. It is then the trailing vehicle which goes into SLVP mode and follows the leading vehicle on the other lane.

Yet, it is usually more than one car that needs to switch to type 2, to resolve conflicts with a platoon. A more complex example of this is illustrated in Figure 3.7. Through the decentralized implementation of the introduced driving behavior in every car, the two platoons can easily merge. The merging gaps are established as illustrated in Figure 3.8 in automated driving mode. Note, that only vehicles which are marked as type 2 are applying SLVP. The remaining cars are either following their direct leader or they are the platoon leader. As illustrated in Figure 3.7 however, a platoon leader can also be assigned with a type 2 (cf. leader of the lower platoon). The necessary information for facilitating SLVP are V2V-based. In terms of the VISSIM simulation they are therefore drawn from the state parameters of the adjacent vehicles (cf. Figure 3.2).

3.5.2. SLVP CONTROLLER CALIBRATION

As mentioned earlier, SLVP uses the same car-following controller algorithm as introduced in Section 3.4.1 for normal platooning. However, there are two essential differences, which need to be adopted when switching from type 1 to type 2. The first one is the reference vehicle to which a safety distance is established. The moment in which a vehicle switches its target vehicle (e.g. from normal platooning to SLVP) is therefore a good representation of the step response that was introduced during the controller design. Furthermore, as soon as the target is switched to a virtual rather than a real vehicle, different controller gains must be used. This is the second difference.

As introduced, it is the controller gains, which quantify the system response to a certain input. For the conventional platooning controller they were chosen as such that the vehicle accurately maintains a certain safety gap to its precursor. Special attention was paid to keep the vehicle from overshooting, as this means that it could potentially run into the safety gap and reach a point where an accident is possible. The good controller performance and the low overshoot from Figure 3.5 and 3.6 illustrate this behavior. Yet, in order to reach this level of accuracy, rather strong controller gains were chosen. The controller reacts with high levels of acceleration u_i to rather low levels of safety gap deviations $e_i(t)$. Although being appropriate for the conventional platooning situation, this can lead to undesired behavior for the SLVP situation.

When a vehicle switches to platooning type 2, the conflicting vehicles have not yet assumed their desired longitudinal positions. This means that the initial deviation value is at this point higher than it can be found in a conventional platooning situation. The conventional platooning controller would therefore react with over-stringent measures. It could at times happen that the vehicle which is determined to follow, would go into full stop or at least apply unreasonably high deceleration levels. Especially when two platoons attempt to merge, this behavior will amplify throughout the vehicle array, certainly leading to undesired behavior.Consequently, this needs to be conquered with a supplementary controller calibration. This is executed in the same manner it was done within the CACC calibration. The resulting controller gains are: $K_{p,i} = 0,4$, $K_{d,i} = 0,8$ and $K_{ff,i} = 0,1$.

SLVP PLAUSIBILITY CHECK

Using the same controller setup and signal processing as before with these new controller gains leads to a less stringent acceleration behavior. This is possible, as the resulting decreased accuracy is essentially irrelevant. In SLVP, there is no precursor neither a follower, which represents an immediate danger if approached to closely. The vehicle is able to interfere within these safety gaps if this eventually leads it to assume its correct longitudinal position. Yet, this behavior is not only possible but also desired. Lower controller gains do not only lead to a smoother driving behavior with less deceleration, but also to less acceleration. Although, this means that the vehicle will effectively take longer to assume its correct position, the lower acceleration leads to less emissions. Consequently, whenever a vehicle switches to SLVP mode, it applies this less stringent controller strategy, which eventually benefits the objectives that this research attempts to follow.

Within Section 3.4.2.1, it was already shown that the general controller setup is feasible and that the CACC controller gains have shown a sufficiently stringent acceleration behavior to maintain a certain safety gap. As the SLVP mode utilizes the same single processing structure with different controller gains, the general behavior is the same. The only notable difference is that the system takes longer to react. However, additionally considering that no danger for a crash with a precursor exists, this can be understood to be irrelevant. Given, this and the fact that the functionality from Section 3.4.2 is maintained, the SLVP setup is generally considered plausible. The functionality of the CACC controller is supplemented through the possibility to allow for merge-in, merge-out or merge-through behaviors. Self-evidently, this also enables adding vehicles to the platoon from the side, as SLVP creates a merge-in gap for those vehicles.

3.5.3. DECOMPOSITION

In Section 3.4.3 it was mentioned that a platooning vehicle will usually copy the lateral movements of its precursor. However, this does not account for situations where the immediate destination of these two vehicles is different. A specific case of this is if a whole or part of a platoon intends to change lanes. In real life this happens if one or several of the vehicles intend to leave the platoon. In this case, the leaving string of vehicles bases its decision for or against a lateral movement on the sensor data that is provided by all vehicles that intend a merge-out. This implies that the sensor data needs to be send to the first vehicle with a merging intention. That vehicle then processes these information and passes a merging decision back to its followers. Consequently, the decision-making takes two time steps 2τ. That is the time from the intention to switch lanes to a potential decision. This is in contrast to the platooning case without lane changes, where the delay is one simulation step τ. The reason for that is that it takes τ for the first merging vehicle to adapt its new status and switch from being a follower to being the platoon leader. Even if the first merging vehicle is the only vehicle that intends to leave the platoon, it takes one simulation step for it to realize its status. After a decision is made, the first merging vehicle assumes its new role either as a platoon leader or as a non-platooning vehicle (if it is the only merging vehicle). In the latter case, the driving behavior switches from platoon to non-platooning as soon as the car has reached its destination lane. It then stops to make us if the EDM DLL and falls back into human driving mode.

3.6. SUMMARY: PLATOONING CONTROLLER DESIGN

Within the last sections, the reader was guided through a rather technical description of the design of a platooning controller. This Section briefly summarizes these technicalities from a high-level point of view. It does so by revisiting the motivation behind the choice to model platooning by means of a platooning controller. It then proceeds to summarize the design process of this controller, explaining and embedding the main steps within this process. The purpose of this is to make the research at hand understandable for those who are not familiar with the details of controller theory. The section concludes by elaborating on the driving behavior that can be reached through this design and how this differs to the base case situation with a human driver.

In the beginning of this and the end of the previous chapter it was explained why designing a platooning controller is the best choice for modeling platooning within VISSIM. In order to grasp the motivation behind this, one first has to understand that a platooning controller describes the conjunct functionality of a set of decentralized vehicle controllers. A vehicle controller is a central feature, which every car incorporates to some extent in its internal signal processing. It essentially is a device that can take over driving functions for the driver, hence enable a certain level of automation. Yet, it usually only takes over minor tasks, such as automatic parking or lane keeping. Now, the main reason why a vehicle controller can be a means of enabling platooning is the following: Although it does usually not do so in current cars, the control system is potentially capable to control all motions of a car. This can be understood as follows: If a car is able to actuate the steering wheel and the throttle/brake in an automatic parking situation, it is equally able to do so in a driving situation. Now, having understood, that the idea of platooning is basically the coordinated motion of a set of vehicles, it becomes clear that the structured coordination of these vehicle controllers (i.e. a platooning controller) can enable platooning.

According to the SAE framework, the degree of automation through the vehicle controllers, which is necessary to enable platooning is level-5 automation. This means that the vehicle controllers shall be capable of taking full control of the car. In this case, all driving tasks are handled through the control system rather than the driver himself. Besides others, the control system takes over the lateral and longitudinal motion of the car. That is, the control system essentially actuates the throttle, brake, steering wheel, turning signals, etc (i.e. the actuators). When doing this, the system is conditioned to always pursue a certain goal. The formulation of this goal is essential for the representativeness of the model. It has to be formulated as such, that accomplishing the goal infers accomplishing the desired driving behavior. In Chapter 2, a set of platooning characteristics were formulated, which constitute such desired behavior. In order to allow for a complete goal formulation, this is supplemented by some generically applicable behavioral features of a platoon, which are introduced in this chapter.

The main aspects of latterly introduced features lead to a goal definition as follows: Copy the lateral movement of the precursor and maintain a predefined safety gap $d_{des,i}(t)$ to that vehicle. Besides others, the value of this variable was determined as part of the platooning characteristics in Chapter 2. Now, what the control system does is that it uses its actuators in order to reach this formulated goal. Similar to a human driver it should e.g. brake if the vehicle comes to close to its precursor or accelerate if the safety gap gets too big. However, this behavior, which is natural for a human driver first has to be designed for the vehicle control.

The control system lacks the senses of a human driver. Initially, it is not aware of its surrounding environment. However, it can make use of sensors, which are part of the vehicle and deliver certain observation data. The actual design of the control system is then to translate this data into the desired actuator actions, which enable desired lateral and longitudinal movement. In other words: The vehicle control takes sensor information, processes them and gains knowledge on which actuator actions are necessary to maintain its platooning position. Mathematically seen, this process is constituted by a set of mathematical functions and algorithms, connecting inputs (i.e. sensor data) to outputs (i.e. actuator actions). Their explicit form is founded in controller theory, which is why the formulas are not further explained here. What can be understood however, is the effect they have on the driving behavior of the car.

After a general controller definition is brought into place, the vehicle is capable of adapting its lateral and longitudinal driving behavior according to its precursor. Some generic connections between inputs and outputs are established. Without the help of a human driver, it is able to e.g. brake if the sensors detect a braking motion of the leading vehicle. At this point there are two main differences between the driving behavior of a human driver and that of the control system. The first one constitutes the desired positive effect. The control system does not exhibit human-typical shortcomings, such as negligence, false observations or reaction times anymore. Complete and accurate sensor data is collected and processed with minor or no delay. As soon as the vehicle control receives an input, a subsequent output is calculated, whereas no faulty calculations occur. The second difference is however, that the actuator actions of the controller are wrongly quantified. This is concerned with the actual calculations that are made in order to translate sensor data into actuator actions. To bring forward an understandable analogy: The sensing and actuation capabilities of the control system represent the eyes and hands/legs of a human driver, respectively. The difference between the two is that the eyes of the vehicle control deliver vastly better observations and the hands/legs of the system are more accurate than those of the driver. Yet, the brain of the vehicle control system is rather simplistic. It is not capable of precise statements, concerning which hand/leg action is appropriate for which observation. the emphasis here lies on 'precise'. Although, the controller, which is the technical pendant to the brain, is generally able to determine when e.g. a braking or acceleration motion is applicable, it is not able to determine the extent of that motion. To stay with the acceleration/braking example, this means that the vehicle will either accelerate or brake to strong, as a response to a certain observation.

Subsequently, what is needed to conquer this and realistically model platooning is not a simple 'if-then'-relation between inputs and outputs. The vehicle control needs to quantify the actuator extent it applies according to its observation, i.e. the motion of the precursor. That is, if a strong braking motion of the precursor is detected, the vehicle itself also needs to brake harder. What is needed is a controller calibration.

The controller calibration is concerned with finding the correct controller gains. Controller gains are variables, which are part of the mentioned mathematical input-output functions. If the right composition of controller gains is found, the vehicle controller calculates the right output extent that is needed to reach the desired driving behavior. Finding these values is done by means of a cluster approach, where inputs and outputs are experimentally collected for various compositions of controller gains. Eventually, a set of feasible controller gains is selected and implemented.

At this point, the vehicle is capable of conventional platooning. That is, it can maintain $d_{des,i}(t)$ to its precursor and copy its lateral movements. However, the vehicles first have to form a platoon to be able to exhibit their conjunct functionality. When the vehicles enter the simulation, they do so as individual human-driven vehicles. The formation of a platoon is part of the driving behavior, which is designed through the vehicle control. Platoon formation is enabled if another vehicle or platoon is detected through the vehicle sensors. If that is the case, the vehicle control will automatically take actions in order to join that platoon. A specific case of this joining action is if the vehicle does not join the platoon from the back.

In that case, the vehicle does not switch from human to conventional platooning mode but first to single-link virtual platooning mode. This mode features two main differences to the vehicle control of conventional platooning. The first one is that actuator actions are not calculated accord to the direct precursor. The vehicle control now aims at maintaining a longitudinal safety gap to the vehicle, which is the precursor of the potential merge-in position in the platoon. The second difference is that the vehicle, which is trailing the potential merge-in position is also going into virtual platooning mode in order to longitudinally follow the vehicle at hand. Note, that in virtual platooning mode no lateral actions are taken. The respective vehicle controllers prescribe longitudinal actuator actions only. The effect of this is that a merge-in gap is created, which allows the vehicle at hand to join the platoon from the side. This behavior can also be triggered if a vehicle desires to merge though the platoon, e.g. from the left lane to an exit, which is on the right side of the platoon. For this mode of platooning, new controller gains are found. Again, this is done through a cluster approach, whereas the reason for new gains is that the desired driving behavior is different to the driving behavior during conventional platooning. That is, in the merge-in situation, the vehicle is not actually trailing another vehicle on the same lane. Because the vehicle is not in danger to bump into the precursor, smoother accelerations or decelerations are possible. As less stringent accelerations lead to less emissions, this behavior is desired here. Consequently, the numerical values of the controller gains are smaller than those of conventional platooning. Note however, that both use the same signal processing and controller structure. E.g. the way a potential acceleration is calculated is the same. The extent of that acceleration however is not.

4

SYNTHESIS OF PLATOONING AND INTERSECTION CONTROL

In Chapter 3 a conventional platooning controller was introduced, which enables the automation of longitudinal movements in platoons. Within the controller calibration it was shown that this constitutes the functionality of real-world platooning. Furthermore, it was touched upon how SLVP can help to create merging gaps between platooning vehicles, whereas this adapted technology can make use of the same signal processing structure as the conventional platooning controller. In this chapter both, the developed controller and the idea of SLVP will be employed and further refined in order to enable a functional interaction of the developed longitudinal platooning controller with an intersection controller. This is done by means of further modification of the EDM DLL, as illustrated in Figure 4.1. The illustration is based on Figure 3.2 and depicts the addition of a set of intermediate steps in the workflow of the EDM. This centers around the classification of each vehicle into a set of platooning types and the subsequent deployment of different longitudinal control strategies for the respective types 0, 1, 2 & 3.

Similar to Chapter 3, this chapter starts off with some elaborations on the chosen design strategy in Section 4.1. Given this strategy, the mentioned intermediate steps are introduced and explained in Section 4.3. Before this however, some remarks need to be made about the overarching goal of this design and a trade-off between the objective functions that were introduced in Chapter 1. These remarks are made in Section 4.2. Section 4.4 finally elaborates on the combined functionality of the intersection and the platooning controller. It does so by deploying a technical interface between the two concomitant systems and structuring their combined functionality. The technical design of this Section constitutes the Level-4 deliverable of the V-model, which is why Section 4.5, attempts to give a comprehensive answer to the associated Level-4 research question. Similar to Chapter 3, the chapter at hand does again go into controller theory to some extent. Consequently, yet another summary section is provided at the end of the chapter, considering the made design from a high-level point of view. Together with Section 4.1 this frames the low-level technicalities of this chapter from a higher-point of view.

4.1. TECHNICAL DESIGN STRATEGY

Similar to Chapter 3, this chapter once again utilizes the model-based design approach that was introduced in Section 3.1. Given the similarity of the chosen approach, no additional effort will be put into revisiting its characteristics and method. Instead, some remarks are made on what distinguishes the challenge of this chapter from that of modeling a platooning controller in the last chapter.

While the overarching goal of Chapter 3 was to model a platooning controller that resembles an already existing system, this Chapter has creative freedom to propose a solution that enables platooning-intersection functionality. This freedom is solely framed through the underlying functionalities of intersection control and platooning, which should not be modified more than possible. One major aspect that constitutes the latter functionality is the decentralized vehicle controller strategy that was developed in the previous Chapter. In order to satisfy the design requirement of minimizing the system interventions, this functionality is hence

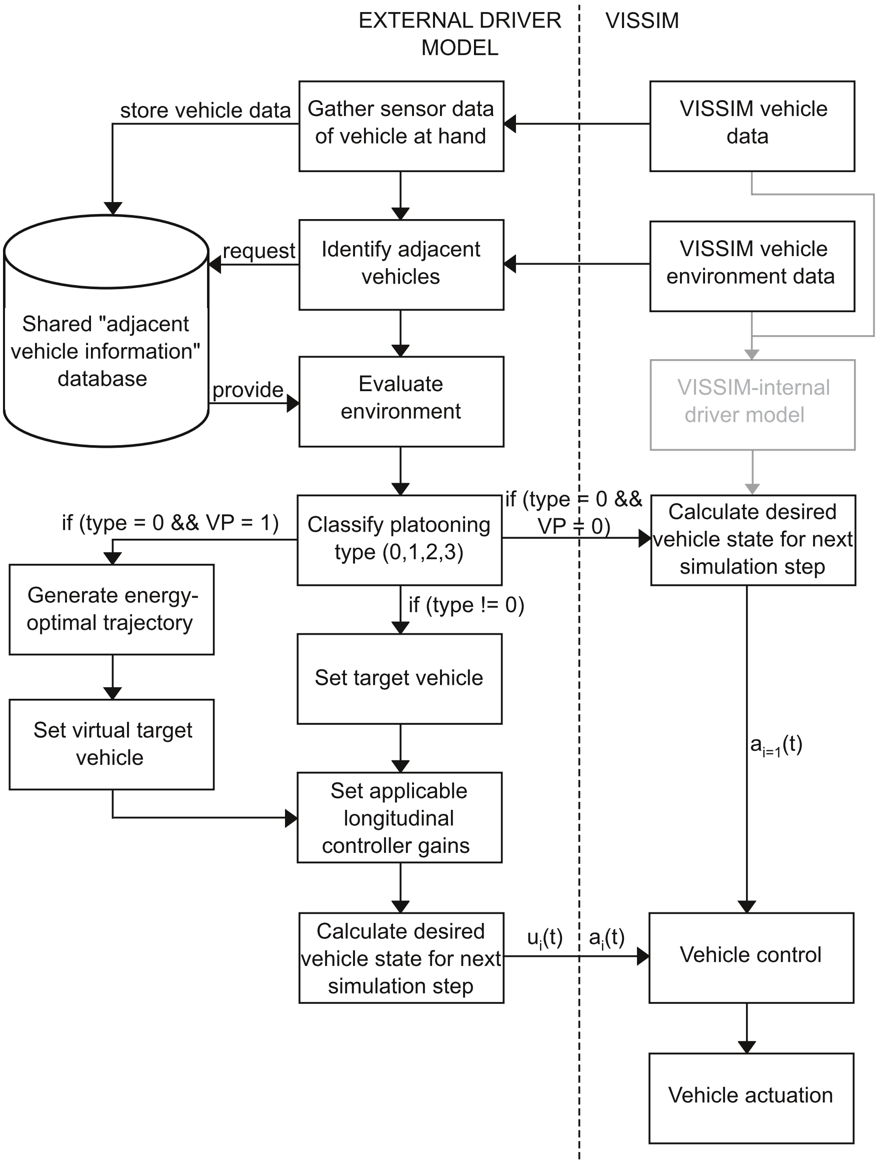


Figure 4.1: Illustration of the data flow between VISSIM and the external driver model DLL in every simulation step for virtual platooning.

reused as a basis for the development of a platooning-intersection controller. Besides the mentioned design requirement another major motivation for doing so is, that this functionality allows to prescribe a vehicle movement at will, by simply changing the challenge, which this controller is fitted to. Hence, by reformulating a mathematical challenge, the vehicle at hand can be forced on a trajectory that allows to coordinate its movements with the green times of a traffic signal. Effectively, this means that the design strategy, which underlies this Chapter is constituted by re-executing steps 1 and 3 from the model-based design approach that was used for the platooning controller.

It is through the thoughtful formulation of a controller challenge that the desired functional combination can be reached. Hence, a major part of this re-execution is to determine what exactly constitutes the desired behavior, which this challenge needs to trigger. For doing so, Section 4.2 provides a detailed reasoning about what driving behavior is desired from a governmental point of view. In Section 4.3, this desired behavior is translated into a mathematical description of a controller challenge and Section 4.5 finally describes how this is brought into place through yet another controller calibration. Effectively, this splits the research strategy in two major parts, whereas the controller calibration represents the second research step (II) of Figure 4.2 and the formulation of a controller challenge represents the fist step (I).

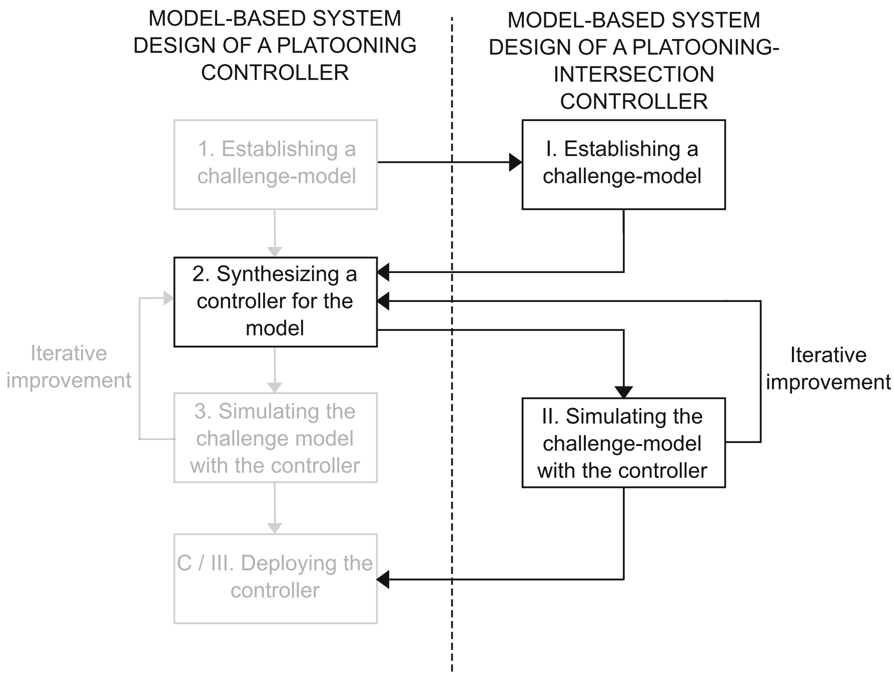


Figure 4.2: Illustration of the major steps in conventional system design and model-based system design.

A major advantage that comes along with choosing to reuse the controller design from Chapter 3 is that the new to-be-developed platooning-intersection controller does not need to be formulated from scratch again. Only by providing it with a different controller goal and different controller gains, a completely new system behavior can be triggered. This makes the second research step from the development of a platooning controller obsolete, hence its omission in above-introduced research strategy. As mentioned, the following starts off the formulation of a controller challenge by describing and motivating a desirable driving behavior. Again, the strategy of this section is revisited in the chapter's summary in Section 4.6, where the taken design choices and the solutions to the individual design steps are presented. Note, that the presented technical design of this chapter represents the main contribution of this research.

4.2. THE PERFORMANCE-EMISSION TRADE-OFF

Before elaborating on the combination of platooning with intersection control it has to be understood, which goal this implementation attempts to reach. Within Chapter 1 it was introduced that this research aims at two main objectives. These are firstly, the minimization of unnecessary congestion and secondly the decrease of emissions. Now, the reader was already guided through the implementation of platooning and it is understood that this infers an elevated level of automation from level 0 to 2 to full automation (level 5). Through this high degree of automation it is possible to intervene in almost all aspects of the driving behavior, simply by changing the goal, which the vehicle control system pursues. One of these aspects which is especially relevant for bot the congestion and emissions of a network is the trajectory control of a platoon.

So far it has only been discussed how intra-vehicle safety gaps are maintained, how string satiability is reached and how platoons compose and decompose themselves. After all, these aspects are mainly concerned with the following vehicles of the platoon. The trajectory control however, is developed and implemented though the platoon leader and it is the trajectory control which offers the highest saving potential for both objectives. The entry sentence of this section can therefore be refined to 'It needs to be understood which goal the trajectory control attempts to reach'.

In order to answer this, it is helpful to understand that the objectives often go hand in hand with each other. Most actions that decrease congestion also decrease emissions and vice-versa. A good example for this is the last chapter. In terms of congestion, the developed platooning controller infers that the vehicles occupy less space, that they have less reaction time, that they do not exhibit faulty behavior and hence, that they generally make traffic more fluid. Yet, platoons do not only contribute to the congestion aspect. Their low safety gaps, e.g. reduce the drag resistance for FVs, which helps to save on emissions. Furthermore, the resolution of traffic problems leads to less waiting times, hence less unnecessary decelerations and once again less emissions. Clearly, the measures that were taken so far have contributed to both, lowering congestion and emission values. It is now important to identify measures where the objectives do not inherit a complementary relation. Here, two potential measures are considered. Both of these are linked to the trajectory control of the platoon leader, which as Section 4.3 shows in detail, can be established through the combination with an intersection controller. This is concerned firstly, with how a platoon approaches an intersection and secondly, with how it leaves that intersection.

Of these two it is only the latter, that features an objective conflict, while the former does indeed have similar effects on both goals. It is therefore the former, which constitutes the main subject of this chapter. For now however, it needs to be understood, how the intersection exit behavior is handled in this research and why it is done like this. Here for, the reader needs to grasp the relation between acceleration a and emissions E. It is in the nature of a combustion engine that less of the potential fuel energy is transformed into kinetic energy when a vehicle accelerates strongly. This means that more heat energy is produced in such case. The ration between heat and kinetic energy gives a good representation of the efficiency of a car, whereas logically high acceleration levels lead to lower efficiencies. This further means that more potential energy is necessary to get to the same level of velocity. This potential energy is in reality stored in the fuel tank. In other words: Higher accelerations consume more fuel, which goes hand in hand with higher emissions.

It is nothing but a logical consequence that leaving an intersection with a slow and stead acceleration is favorable in terms of emissions. This however, contrasts with what is favorable in terms of congestion. Clearly, if vehicles are accelerating very slowly they constrain other vehicles from entering the intersection. Hence, the overall intersection performance decreases, leading to increased rather than reduced C-values. A reasonable solution has to be found to deal with this trade-off. Yet, it was mentioned in the beginning that no objective holds priority over the other. Consequently, the following approach is chosen:

In order to maintain comparability between the different scenarios it is desired to make the driving behavior (especially when leaving an intersection) similar to that of the base case situation. In the $\bar{P},\bar{I}$-scenario, every vehicle follows its natural, intrinsic acceleration behavior, whereas this behavior is different for different vehicle types. I.e. trucks generally accelerate less than passenger cars. Yet, the same is adapted for platoon leaders in the scenarios, which include platooning. As platoons form randomly without restrictions on vehicle types, the distribution of platoon leaders will stochastically represent that of vehicle types in the base case scenario. Now, if the platoon leader follows its intrinsic acceleration behavior and enforces this behavior for the FVs, the statistic overall acceleration behavior remains the same. Comparability is established between the base case and the platooning scenarios. Yet, this choice needs to be made subject to a few constraints. If the natural acceleration of a FV vehicle is less than that of the LV, the vehicle communicates its maximum acceleration with the leader. If this is in the range of $1m/s^2$, the leader will assume the maximum acceleration of this FV, if not then the leader will assume its natural acceleration and the slowest vehicle will become leader of a new platoon. If this rule is implemented for all vehicles, all platoon leaders accept a range of $\pm 1m/s^2$ around their intrinsic acceleration. The stochastic vitiation of this is assumed to be neglect able, while the measure allows for heterogenous platoons.

4.3. OPTIMAL INTERSECTION APPROACHING IN THE EDM

Having understood, how the intersection exit behavior of a platoon is designed to maintain comparability, the focus can now be shifted on designing the intersection approach. As mentioned, this does not feature an objective conflict, which is why the design space of this measure is in the following used to minimize both, congestion and emissions. Similar to the multi-lane virtual platooning that was introduced in Section 3.5, a platoon can also adjust its motion according to a vehicle that is located on another intersection entry. Assuming an intersection, as depicted in Figure 4.3, the platoon that is coming from the left entry can be mirrored

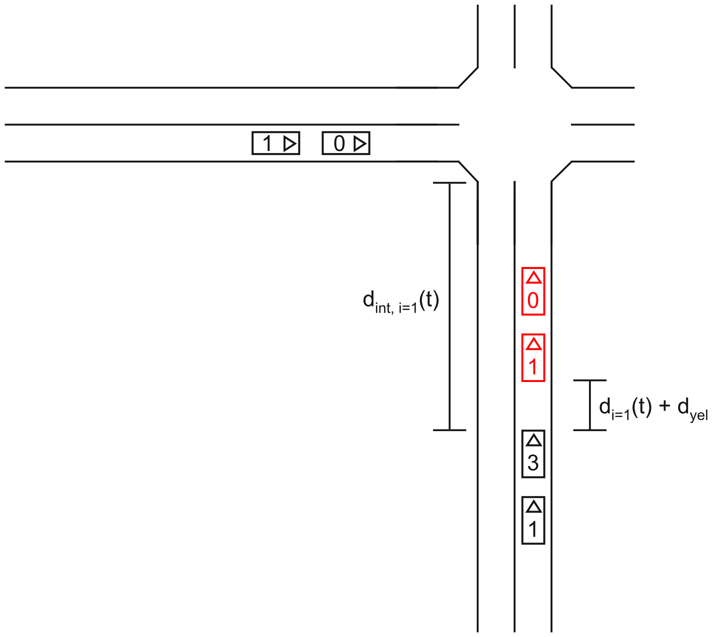


Figure 4.3: Multi-link virtual platooning. Real vehicles are depicted in black, virtual vehicles are depicted in red.

on all other entries. The platoon arriving on the bottom entry can then adjust its speed as such that it will arrive at the intersection shortly after the first platoon has left the conflict zone. This is called multi-link virtual platooning. Although the idea of multi-link virtual platooning originates from uncontrolled intersections, it is equally applicable for signalized intersections. In this case, the platoon could adapt its speed to arrive at the intersection as soon as it receives priority.

In the following it will be shown how this way of artificially decelerating vehicles with subordinated priority allows for a more efficient intersection behavior. This is done by explaining how a human driver approaches both, a conflict with another vehicle on an uncontrolled intersection and a red traffic light. From this, a set of conclusions is made on the way how a vehicle should ideally approach an intersection and how multi-link virtual platooning (MLVP) can help to facilitate this behavior. Eventually this leads to an implementation of MLVP, which enables a functional combination of intersection and platooning controllers. In this sense, it is assumed that the intersection controller provides a so-called 'estimated time of departure' t_{ed} to the platoon. It is later explained how t_{ed} is generated within the intersection controller.

Assuming that the platoon on the left entry of Figure 4.3 was further away from the uncontrolled intersection, yet being in priority, a clear conflict between the two platoons would exist. The natural driving behavior of the LV of the bottom platoon would in that case be to follow a velocity curve, similar to that as illustrated by the black curve in Figure 4.4 (Note that an average velocity curve would also feature a standard deviation, which is neglected here for reasons of simplicity). This represents the intrinsic driving behavior of the VISSIM-internal driver model and is further assumed to be a solid representation of real-life matters (Fellendorf, 1994; PTV, 2016). The platoon's FVs are either in conventional or multi-lane platooning mode, which is why they eventually adjust their speed to the LV. They feature the same velocity curve.

The round shape of the deceleration curve is due to imperfections in throttle control and lacking knowledge on the desired level of deceleration. Even if the human driver of the LV aims for a constant deceleration, hence a straight line, he is not aware of the exact level of deceleration, which will bring the car into the desired state (arriving at the intersection exactly after the conflict has vanished). Therefore, the brake force usually varies throughout the deceleration period. Constant speed adjustments eventually lead the vehicle to arrive at the intersection only roughly when there is a time window which allows for passing the conflict zone. Generally, such a characteristic braking curve features a too high level of deceleration in the beginning of the motion and a too low brake force in the end. This gives the human driver a feeling of safety. This way a relatively big safety gap towards the conflict zone is established. Consequently, the driver will start to resume its initial speed even before he enters the intersection. However, as it will be shown later, this driving pattern is not desired in terms of intersection efficiency and emissions. The enhanced capabilities of automated vehicles allow for clear improvements and a better intersection approach.

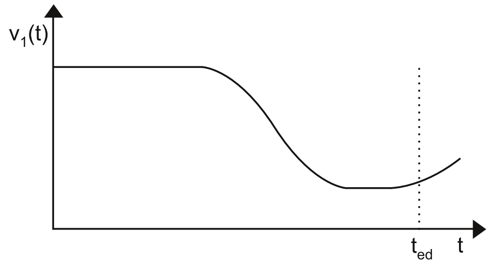


Figure 4.4: Qualitative velocity curve during approach of an uncontrolled intersection for normal driving or conventional platooning.

A second characteristic of the human driver curve from Figure 4.4 is the late initiation of the braking maneuver. The allocation of the start of the deceleration motion largely depends on the intersection setup. For now a best case scenario with clear sight on all intersection entries and a good driver are assumed. Yet, even if the driver was willing to adapt his velocity in such a way that the conflict would have disappeared upon arrival at the intersection, he might not be able to assess the exact time at which a conflict might occur. The time of a conflict depends on the speed of the conflicting vehicles and their respective distances to the conflict zone. Neither of those can be perfectly known by the driver. Hence, he will wait until the situation can be sufficiently evaluated and only react shortly before the conflict time. Logically, this gets worse when assuming a controlled intersection. If the vehicle with subordinated priority approaches a red light, the driver is not aware of the time when the traffic light will turn green. No knowledge is available about the time of conflict, neither when it will be resolved. Naturally, the driver will therefore approach the intersection in such as manner, as he would approach a blockade (in VISSIM this blockade is modeled as a still-standing vehicle), implying that the curve from Figure 4.4 is shifted to the right. Hence, the black curve in Figure 4.5, displaying the human approach to a red traffic light. Note, that in this illustration the speed does not reach 0. This is only to exemplify that the vehicle can receive a priority at any time. The curve does not necessarily need to be completed (However, it could also be that the vehicle has to wait at the intersection for some time).

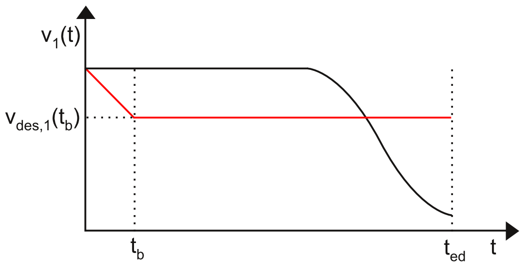


Figure 4.5: Qualitative velocity curve during approach of a controlled intersection for conventional and virtual platooning. Velocity for conventional platooning is depicted in black, velocity for virtual platooning is depicted in red.

What the black curves from Figure 4.4 and Figure 4.5 have in common is that their integral is equal to the distance to the conflict zone. For the sake of simplicity, this is from now on assumed to be the distance $d_{int,i=1}(t)$ of the first platooning vehicle to the intersection entry:

$$d_{int,i=1}(t_0) = \int_{t_0}^{t_{ed}} v_1(t)dt. \tag{4.1}$$

Consequently, both black velocity curves cover the same distance. Yet, the curve where the driver has rough knowledge about the conflict time, features a higher velocity $v_1(t_{ed})$. A late and imprecise reaction towards an unknown conflict time leads the platoon to arrive at the intersection at a lower speed or even with a full stop. In the contrary, both, emissions and intersection performance would be better off if the vehicle would arrive at the intersection at a high speed. From the considerations so far it can be seen, that there is a correlation between the knowledge about the conflict and the maximum velocity level upon arrival that can be reached. Following this trend, the red velocity curve in Figure 4.5 illustrates the ideal conflict approach if the estimated departure time t_{ed} is known. Similar to the previous velocity curves, it features the same initial distance to the intersection entry $d_{int,i=1}(t)$. Yet again, it has a higher eventual velocity. Furthermore, it does not feature the round-shape that is characteristic for human drivers. The high eventual velocity implies that the vehicles have to accelerate less in order to reassume their initial speeds. Knowing that the kinetic energy E_{kin} equals:

$$E_{kin} = \vec{F} \cdot \vec{s} = m \cdot a \cdot d_{int,i=1}, \tag{4.2}$$

it becomes clear that the red velocity curve depicts the energy-optimal approach. It features the lowest necessary acceleration and therefore consumes the least energy. As mentioned in Section 1.1.1, the used energy is directly linked to the fuel use and emissions of a car, which is why this velocity approach is adapted to reach the goal of saving emissions. In Chapter 5 it will further be shown that this additionally helps to increase the efficiency of an intersection.

4.3.1. MULTI-LINK PLATOON COORDINATION

In the following, a communication standard between the platooning controller and the intersection controller is established. Based on this, an algorithm is introduced, which establishes an energy-optimal longitudinal trajectory on intersection entries. Lastly, it is explained how this trajectory is implemented in the EDM DLL from Chapter 3.

In order to allow the platoon to establish and execute a velocity plan, as introduced, some form of communication has to be established between the platoon leader and the intersection controller. In Section 2.1.2, this communication was already characterized and it was stated that only the LV participates in V2I communication. This communication should be as simple and robust, as possible. Transmitting a complete trajectory from the intersection to the platoon is therefore seen as an unfeasible option. Instead, the intersection controller provides the estimated time of departure t_{ed} and the trajectory is generated platoon-internally. The estimated time of departure represents the point in time at which the platoon will be able to cross the intersection or in other words: t_{ed} is the amount of time before the platoon receives priority. As desired in the problem formulation, this design choice minimizes the intervention in the concomitant systems. A single V2I communication channel suffices and the intersection controller only requires minor modification. Furthermore, this allows for a decentralized trajectory controller approach, where each platoon or vehicle can generate its individual trajectory plan, which eventually optimizes its energy usage. This way, the intersection controller does not require awareness of vehicle specifications, such as the maximum level of deceleration.

Now that a V2I communication standard is found, it needs to be understood how the desired trajectory is calculated through the platoon leader. As introduced in (4.1), the distance $d_{int,i=1}$ can be calculated as the integral over the velocity. Consequently, the shape of the desired velocity curve allows for the mathematical formulation of the desired trajectory in terms of the desired distance to the intersection at t:

$$d_{des,int,i=1}(t) = \begin{cases} \int_t^{t_b} v_1(t_0) + a_{min} \cdot t + v_{des,1}(t)\,dt & \text{for } t < t_b \\ \int_t^{t_{ed}} v_{des,1}(t)\,dt & \text{for } t_b \le t < t_{ed}. \end{cases} \tag{4.3}$$

As it can be seen, the desired velocity curve can be modeled as two linear functions. Here, a_{min} represents the maximum deceleration, which logically is a negative value. $v_1(t_0)$ is the initial speed upon reception of a value for t_{ed}. Integration and mathematic transformation leads to:

$$d_{des,int,i=1}(t) = \begin{cases} -\frac{1}{2} a_{min}(t_b - t)^2 + v_{des,1}(t)(t_{ed} - t_b) & \text{for } t < t_b \\ v_{des,1}(t)(t_{ed} - t) & \text{for } t_b \le t < t_{ed}. \end{cases} \tag{4.4}$$

Given, that the LV of the platoon has received a t_{ed}-value and a_{min} is known through the vehicle control, two unknown variables remain. These are $v_{des,1}(t)$ and t_b. Their relation can be described as:

$$v_{des,1}(t) = v_1(t_0) + a_{min} \cdot t_b. \tag{4.5}$$

Furthermore, the first case of (4.4) is known for $t = t_0$. At this point in time the desired distance to the intersection is equal to the current distance to the intersection ($d_{des,int,i=1}(t) = d_{int,i=1}(t)$), which itself represents available sensor data. It is one of the parameters that is passed between the VISSIM simulation and the EDM DLL. Inserting the values leads to:

$$d_{int,i=1}(t_0) = -\frac{1}{2}a_{min}(t_b - t_0)^2 + v_{des,1}(t)(t_{ed} - t_b) \tag{4.6}$$

Through inserting (4.6) into (4.4) and mathematic transformation, using the linear quadratic formula (Kalman et al., 1960) the following term for t_b is found:

$$t_b = t_{ed} + \frac{\sqrt{2a_{min}t_{ed}v_1(t_0) + a_{min}^2 t_{ed}^2 - 2a_{min}d_{int,i=1}(t_0)}}{a_{min}} \tag{4.7}$$

By using this formula and inserting the result in (4.5), the platoon controller can calculate the energy-optimal trajectory, solely based on the estimated time of departure and internal sensor data. As mentioned earlier, this can be done in any platoon (leader) and at any time, as long as the intersection is in transmission reach. Furthermore, this allows for continually updating the desired trajectory in case of a change in t_{ed} or newly available sensor data.

Finally, this trajectory needs to be incorporated in the vehicle control to trigger a vehicle actuation. As mentioned earlier, the actual longitudinal vehicle actuation is reached by sending a desired acceleration value to VISSIM in every simulation step. For this a platoon controller was developed and calibrated in Chapter 3, which calculates these desired acceleration values as a function of the measured distance to a target vehicle. More precisely, this is done as such that the vehicle at hand maintains a certain desired distance $d_{des,i=1}(t)$ to its target. In (3.4) $e_i(t)$ is calculated by subtracting the actual measured distance from the desired distance. Having understood that the platooning controller always strives to minimize $e_i(t)$, the vehicle will therefore strive to maintain the desired distance to the target vehicle. This functionality can now be used as an interface to implement the energy-optimal trajectory control for the platoon leader. Note that all following considerations and subsequent calculations are only made for the LV. The FVs are either in conventional (type 1) or multi-lane platooning mode (type 2). Only LVs can enter multi-link platooning mode and then be followed by FVs. As it can be seen in Figure 4.4, this is indicated through the type number 3 (e.g. LV on bottom intersection entry).

As it was mentioned earlier, VISSIM models red traffic lights as virtual still-standing vehicles. A functional combination between this approach and the platooning controller can be used to make the LV follow the defined trajectory. That is, by creating a moving virtual vehicle, to which the LV will attempt to maintain $d_{des,i=1}(t)$, the platooning controller at hand can be manipulated to adapt its motion accordingly. This is done as such that the vehicle exactly follows the desired, energy-optimal trajectory. This allows for an effective implementation of the desired path without intervening in the underlying platooning controller functionality. Now, as usually, the LV will internally calculate the desired distance deviation between the desired and the actual distance to the virtual vehicle and feed the result into the controller, which then calculates an acceleration accordingly. The only difference to conventional platooning is that the target vehicle does not actually exist. It is merely a tool to enable an automated, energy-efficient intersection approach.

Now, the LV adapts its motion according to the virtual vehicle, which implies that the trajectory of that virtual vehicle must lead the vehicle at hand to follow the path as described in (4.4). In that sense, the desired distance deviation calculated by the LV is:

$$e_{virt,i=1}(t) = d_{virt,i=1}(t) - d_{des,i=1}(t). \tag{4.8}$$

The desired intra-vehicle distance $d_{des,i=1}(t)$ was fixed in (3.3), which is why $d_{virt,i=1}(t)$ needs to be adapted to lead the vehicle at hand on the desired path. Given that the LVs trajectory is described as a desired intersection distance in (4.4), $e_{virt,i=1}(t)$ can alternatively be described as the difference of the actual intersection distance and the desired intersection distance:

$$e_{virt,i=1}(t) = d_{int,i=1}(t) - d_{des,int,i=1}(t). \tag{4.9}$$

Now, if the controller objective is reached, the LV will exactly follow the energy-optimal trajectory. Consequently, $d_{virt,i=1}(t)$ has to be described as:

$$\begin{aligned} d_{virt,i=1}(t) &= d_{int,i=1}(t) - d_{des,int,i=1}(t) + d_{des,i=1}(t) \\ &= d_{int,i=1}(t) + d_{des,i=1}(t) - \begin{cases} -\frac{1}{2}a_{min}(t_b - t)^2 + v_{des,1}(t)(t_{ed} - t_b) & \text{for } t < t_b \\ v_{des,1}(t)(t_{ed} - t) & \text{for } t_b \geq t < t_{ed}. \end{cases} \end{aligned} \quad (4.10)$$

Summing up, this expression can be used to overwrite the calculation of the distance deviation $e_{i=1}(t)$ in conventional platooning. This consequently manipulates the driving behavior of the platoon leader to follow the desired path. This measure only takes action if the target vehicle of the LV is set to the virtual vehicle. If done so, a virtual precursor as illustrated in Figure 4.3 is generated, which 'guides' the platoon on this path. In the EDM DDL, switching to type 3 and hence, multi-link virtual platooning, is triggered through the reception of a t_{ed}-value from the intersection controller. As soon as $t > t_{ed}$, the vehicle falls back into its previous driving mode.

4.3.2. MLVP CONTROLLER CALIBRATION

As it was mentioned, by overwriting the distance deviation $e_{i=1}(t)$, the LV vehicle controller can be manipulated to maintain a certain energy-optimal trajectory. The rest of the platoon naturally follows the LV. By maintaining a prescribed safety gap to the LV, they adapt the same trajectory. Furthermore, it has also been explained how this trajectory looks like and how it can be calculated platoon-internally based on V2I communication. This approach was chosen to satisfy the earlier-formulated need for a solution with minimal intervention in the already established concomitant systems. Other approaches exist, which incorporate more (V2V) data to generate their trajectories and the subsequent vehicle control actuations (see e.g. Medina et al., 2015). Especially, the latter is not possible for the solution at hand. As it was elaborated, this is counteracted by using the platooning controller from Chapter 3 to adapt the generated trajectory. However, similar to SLVP this requires a new calibration of the platooning controller with new controller gains K_p, K_{ff} and K_d.

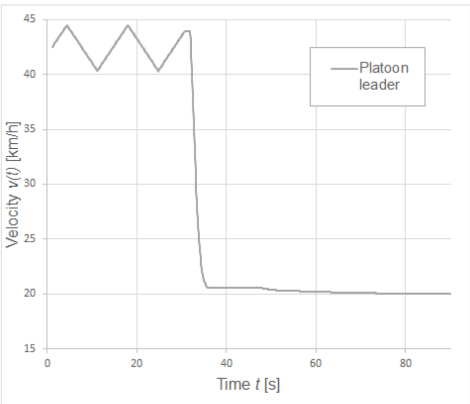


Figure 4.6: Long-term view of the velocity of the lead vehicle step response, initiated through the MLVP-calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 31,7s$.

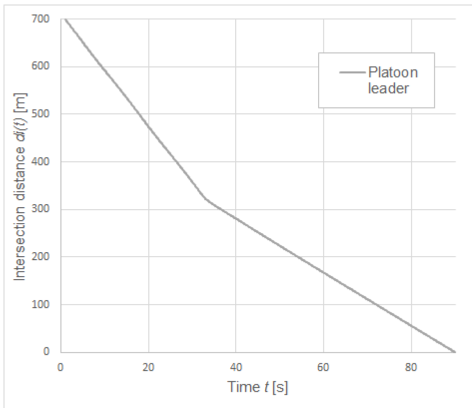


Figure 4.7: Long-term view of the intersection distance of the lead vehicle step response, initiated through the MLVP-calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 31,7s$.

The MLVP situation entails different requirements than the conventional platooning situation. The primary concern of the conventional platooning controller is to prevent a collision with the direct precursor. It therefore strives to precisely maintain a highly stable safety gap. In order to do so the controller is designed to apply very high accelerations or decelerations if required. This is not necessary, neither desired for the MLVP calibration. The nonexistence of a (real) precursor allows for a trajectory adaptation featuring lower acceleration and deceleration levels. Resulting in a smoother driving motion, this does not only save fuel but it also enhances safety for potential FVs or other following vehicles. This is based on two reasons. Firstly, if the FV was to abruptly brake as soon as it receives a t_{ed}-value, it would thereby trigger string instability for

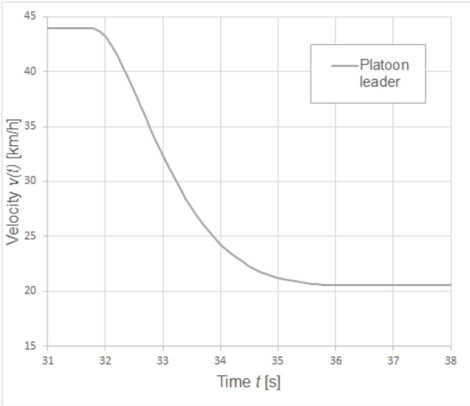


Figure 4.8: Short-term view of the velocity of the lead vehicle step response, initiated through the MLVP-calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 31,7s$.

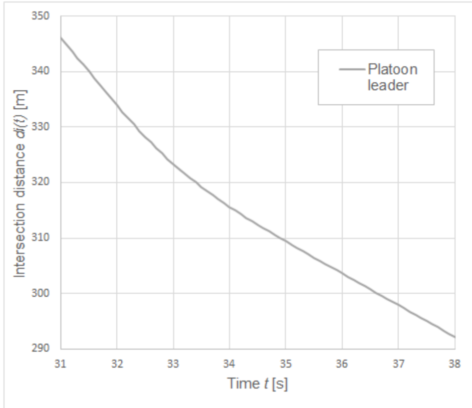


Figure 4.9: Short-term view of the intersection distance of the lead vehicle step response, initiated through the MLVP-calibrated PID controller to an abrupt increase of $e_{i=1}(t)$ at $t = 31,7s$.

the FVs. In Chapter 3 it was explained how an additional controller gain helps to minimize this instability. However, this minimization results in a set of acceleration and deceleration motions through the FVs, which again increases energy usage. Secondly, if the MLVP vehicle was not leader of a platoon but followed by a type-0-vehicle, an abrupt breaking motion would lead the human driver of the second car to 'over-brake', hence apply too much braking force. As illustrated in Figure 4.4, this behavior is characteristic for human drivers. Consequently, the controller gains were chosen as such to represent a good trade-off between the energy-efficient trajectory from Section 4.5 and a driving behavior, which minimizes negative side effects in following vehicles. Similar to Section 3.4.2, a cluster approach is chosen. Eventually, a solid solution was found in the controller gains $K_p = 0,45$, $K_{ff} = 0,6$ and $K_d = 0,8$.

MLVP plausibility check

Figures 4.8 and 4.9 represent a zoom-in of Figures 4.6 and 4.7. Whereas a t_{ed}-value is provided at $t = 31,7s$. The velocity behavior before that value is based on the VISSIM-internal driver model and represents the human impreciseness in throttle control. As it can be seen in Figure 4.8 (after $t = 31,7s$), the FV adapts a reasonably attenuated velocity curve. Although a stringent velocity break, as illustrated in Figure 4.5 would technically be possible (cf. (3.1)), it is argued that the softened acceleration is a better representation of real-world matters and allows FVs to adapt a braking motion themselves. Yet, this also means that the vehicle at hand does at that time have a higher velocity and is therefore further along the trajectory than it should be. Through the integrative controller gain K_{ff}, this offset can be tackled. As mentioned in the calibration of the platooning controller, K_{ff} counteracts remaining $e_i(t)$-offsets. In Figure 4.6, this effect can be seen between $t = 38s$ and $t = 57s$. After the vehicle assumed the desired velocity $v_{des,1}(t)$, this velocity is being constantly decreased to compensate for the trajectory overshoot. All in all, the Figures illustrate the desired behavior. The platoon leader arrives at the stop line when it is supposed to, which reaches the desirable effect of maximizing the residual velocity upon arrival. The functionality of the MLVP controller mode can therefore be considered plausible. The controller gains together with the signal processing structure of the CACC controller allow for the design and implementation of an optimal intersection approach, which is automatically adopted by the LVs that are either in CACC or SLVP mode.

4.3.3. Conjunct functionality

The t_{ed}-values of JUNO are calculated according to the traffic situation, more precisely the demand on other intersection entries. Consequently, the trajectory of the subordinated platoon from Figure 4.3, represents a function of the trajectory of the prioritized platoon. This shows that the MLVP approach is similar to that of virtual platooning on uncontrolled intersections, as it was introduced in the beginning of this chapter. The only difference is that the intersection controller represents an intermediate step in the data processing between the two platoons. it converts the trajectory of the prioritized platoon in an estimated time of departure for the subordinated platoon. Hence, no mathematical description can be found to express the relation between the two platoons and their respective states. Yet, what can be stated is that the intersection-platoons relation is of hierarchical nature.

In the described setup the intersection controller is the leading system. That is, it prescribes a t_{ed}-value to approaching vehicles as such that it maximizes its objective. The platooning controller is then expected to incorporate this value in its trajectory planning. Put in easy words, the intersection poses a requirement, which is then fulfilled through the platooning controller. Seemingly, this limits the decision freedom of the platooning controller in establishing its favorable trajectory. In some setups it can e.g. happen that a platoon does not pass a traffic light even though it would have been possible to do so in the remaining green time. If such situation occurs, the underlying behavior is prescribed though JUNO because that precise move benefits the overall intersection performance. This can at times infer that an individual car or platoon os worse off in that situation. Yet, the possibility of this happening is willingly accepted. This becomes clear when assuming a long-term view.

What the intersection controller reaches through its measures is a network optimization in terms of throughput. Generally it holds relevance, that if the network is optimized the platoon will eventually arrive at its destination faster. That is, a structured traffic flow can prevent jams and allow the traffic community to be better off. Even if the platoon was slower this time it might be on the benefiting end the next time. The intersection controller does not make priorities between vehicle types. Consequently, the selection of who is made worse off for the benefit of all is random. According to the law of big numbers, this implies that in the long-term all traffic participants will be better off. In this sense, the goal of the intersection controller does not only contribute to that of the platoon, but the overall intersection performance will furthermore benefit everyone equally.

4.4. DESIGN CHOICE MOTIVATION

In the previous section one of the main design choices of this thesis was introduced and explained. Namely, this is the combination of the two concomitant systems through a V2I channel, which conveys the estimated time of departure. This value is calculated intersection controller-internally. The calculation does not only consider a local optimization but can also take global effects into account. The V2I channel constitutes a uni-lateral deployment of a desirable velocity curve, whereas this velocity is calculated and deployed through the platoon, rather than the intersection controller. Altogether, this setup generally allows for the combined functionality of platooning and intersection control. Its results in terms of emission and congestion savings are analyzed in the simulation chapter. For now however, the design choices that were made to get to this functional system combination will be motivated briefly.

Surely, there would have been a variety of other design possibilities enabling a similar or the same conjunct behavior of the systems. The choice for the design as presented in the last two chapters was made for two main reasons. Firstly, as already mentioned this is a choice, which minimizes intervention in both of these systems. It was already motivated, that this platooning controller depicts a good representation of real-world matters. Hence, it is especially desirable if the modeled platooning controller can be adapted to the combined functionality as easily as possible. Therefore, the choice to target the already existing vehicle control system with a virtual input appears to be a good design choice. Neither does the internal signal processing of the platoon vehicles require changes, nor does the sensor and actuator setup. Eventually, the changes that are considered are solely of algorithmic, computational nature. The designed functional combination satisfies the requirement to minimize intervention in the concomitant systems.

The first reason is concerned mostly with the structural design of system inputs and outputs. The design aimed at establishing and connecting an interface between the systems. In contrast to that, the second motivation for the introduced design choices is that the system can easily be transferred to other sets of concomitant systems. It was mentioned, how the calculation of an estimated time of departure is a byproduct of range of different traffic-adaptive and look-ahead traffic-adaptive intersection controllers. For all of those, the introduced V2I-channel constitutes the possibility to employ the platoon-intersection controller. Similar accounts for the platooning systems. As it was introduced, the signal processing that is depicted in Figure 2.13 can be found not only in the SARTRE project but it rather illustrates the general functionality of vehicle control. Hence, every vehicle that is capable of level-5 automation (and hence platooning) can therefore participate in the interaction with a traffic controller.

4.5. TECHNICAL DESIGN CONCLUSIONS (LEVEL 4)

In Section 1.1.3 a technical research question was delineated and formulated: *How can a platoon be navigated through a string of intersections through a combination with intersection control while allowing for a performance optimization of the traffic network in terms of congestion and emissions?* This question was answered in this and the previous chapter. An answer was given by pursuing the two technical design goals, which were formulated in Section 1.2.4. Firstly, "a platooning controller that realistically represents the driving behavior of real-world platoons" was designed in Chapter 3. This controller was then integrated with the intersection controller JUNO in order to form a "functional combination [...], that aims at minimizing congestion and emission issues".

The final technical design is constituted through a combination of three platooning modes of which two ensure platooning functionality and one the interaction with JUNO. Their conjunct functionality and code-wise implemetaiton is what is considered the technical deliverable. When revisiting the V-model from Section 1.2, it becomes clear that this lays the foundation for subsequent VISSIM simulations and further high-level considerations. It is the first and the main contribution of this research and it answers the Level-4 research question: A platoon can be navigated through a string of intersections by means of thoughtful combination of the three mentioned platooning modes and their functionality. Yet, due to the nature of this work, being a lengthy C++ implementation of a EDM DLL with a variety of facets, this answer does not live up to the extent of the contribution. Besides others, this is a reason why the following section provides a brief, high-level summary of the made technical design.

4.6. SUMMARY PLATOONING-INTERSECTION CONTROLLER DESIGN

In Chapter 3 a platooning controller is developed, which represents an accurate model of real-life platoons. This controller lays the basis for the evaluation of the actual deliverable of this research, a functional combination between platooning and intersection control. Similar to Chapter 3, the chapter at hand introduces the latterly mentioned combination and similar to the last chapter, this chapter concludes with yet another high-level summary of this technical design. To start with, this revisits the motivation to use the combination of platooning and intersection control in order to optimize the approach towards an intersection. Following this, it is first explained how an intersection should ideally be approached and then how this is deployed in the technical design. The subsequent interaction between the two systems is then described, before some concluding remarks are made on the design choices that were made in this chapter.

Self-evidently, the platooning-intersection controller helps to facilitate how a platoon crosses an intersection. Within the first section of this chapter, this motion is subdivided into two parts - entering and leaving the intersection. Subsequently, it is noted that both can potentially have an impact on the congestion and emissions on that intersection. Yet, only the first of the two situations is made subject of a technical design. This is because for the latter, an objective conflict can be identified. While it would be beneficial for the congestion goal to apply stringent acceleration levels upon intersection exit, this would clearly also lead to higher emissions. No desirable driving behavior of a platoon exists, which could potentially benefit both objectives. This logically excludes the platoon characteristic of maintaining short safety gaps between vehicles. However, as this is implemented within chapter 3 for all driving situations, it is concluded that no intervention is made in the intersection exit driving behavior of a platoon in this chapter. Instead, am intersection exit solution is presented which guarantees comparability with the simulation scenarios that do not incorporate platooning.

That is, the intersection exit is used for platoon formation, whereas each platoon exists of vehicles whose maximum acceleration varies by a deviation of not more than $\pm 1 m/s^2$. Such platoon composition leads to a situation where every car applies roughly the exit acceleration, which is natural for it. Furthermore, through the randomization of platoon leaders, this deviation is considered neglectable, as statistically vehicles will equally often be part of platoons that go faster, as they will be part of platoons that go slower than their intrinsic behavior.

In contrast to the intersection exit, the intersection approach features a set of solutions, which benefits both, the congestion and emissions on an intersection. This set of solutions essentially consists of a variety of velocity curves that are possible, given the remaining distance $d_{int,i=1}(t_0)$ between platoon leader and stop line. In Figures 4.4 and 4.5, the black line illustrate human-typical velocity curves for approaching an uncontrolled and a controlled intersection, respectively. Numerous other velocity courses are possible here. The range of possible speed curves is solely limited through the maximum distance $d_{int,i=1}(t_0)$ that can be covered before before the platoon receives priority. Now, considering that platoons drive autonomously, this allows for an accurate deployment of the most congestion and emission efficient velocity curve. This however, first needs to be defined.

By comparing the two human-typical velocity curves from Figures 4.4 and 4.5, it can be seen that the curve where the driver has more knowledge about the time he will receive priority, features a higher arrival velocity. Naturally, he estimates when he will be able to enter the intersection and adopts a velocity curve which allows him to arrive at that time with a relatively high residual speed. This makes him better off in terms of emission, as he minimizes unnecessary accelerations and in terms of congestion, as he is able to leave the intersection faster. Now taking this knowledge and interpolating it to the improved sensing capabilities of an intersection controller and the improved actuator precision of a platoon, an optimal velocity curve can be found. This curve is displayed in red in Figure 4.5. It is the curve, which features the highest possible residual speed under the given circumstances and a maximum deceleration a_{min}.

Having understood that the introduced curve represents an energy and congestion optimal intersection approach, this now has to be brought in place through the vehicle control of the platoon leader. This is done as follows. The intersection controller communicates its estimate of the time when the platoon will receive priority through the V2I channel between the systems. The platoon leader takes this information in order to internally calculate the optimal velocity curve. Now in order to deploy this velocity curve by means of the internal vehicle controller, one must remember what this vehicle controller is initially designed to do. In Chapter 3, two controller modes (CACC and SLVP) were developed. Both are capable of maintaining a safety distance to a precursor. The difference between the two is that the latter maintains the mentioned distance to a virtual vehicle. The same can be done for the case at hand. In order to manipulate the platoon leader to adopt the desired velocity curve, a virtual precursor is needed which forces it to do so.

This is reached by translating the desired velocity curve into a desired trajectory. Mathematically, this is founded in the derivative relation between position and speed $s = \int v dt$. Knowing the optimal speed trajectory, a respective position trajectory is calculated. Now, this position (+ the length of the virtual vehicle and the safety gap) can be used as an input for the vehicle control of the LV. As usually (see CACC and SLVP), this then proceeds to calculate a deviation $e_i(t)$, according to which it reacts by assigning acceleration values to the actuators. Again, similar to the calibration of the SLVP controller mode, this allows for new controller gains. Yet another cluster approach is adopted, which eventually leads to a controller behavior as displayed in Figures 4.6 to 4.9. The figures exemplify that the designed interaction of platooning and intersection controller can lead to an optimal intersection approach through the platoon.

Through the minimization of unnecessary accelerations, emissions can be saved and through the high intersection entry speed, vehicles can clear the crossing faster. In theory, both objectives mutually benefit of this measure. However, only the next chapter gives insight into the actual simulation impacts. This Chapter concludes with motivating the main design choices that were made so far. It does so by stating that the design firstly minimizes its intervention in the underlying functionality of the concomitant systems. Secondly, the designed interaction is widely applicable, as a range of intersection controllers were shown to be feasible, while it can in general be used with all vehicles that allow for platooning. In other words: The design is implementable on basis of a range of system variations and it is implementable without drastic changes in one of the systems functionalities.

5

Simulation

Chapters 3 and 4 elaborate on the design of a vehicle controller, which is then adopted for conventional, single-link virtual and multi-link virtual platooning. These three platooning modes allow for a safe navigation of a platoon through one or several intersections. Having understood, that especially the latter can potentially also contribute to a network performance-optimization in terms of congestion and emissions, this impact now needs to be evaluated. More precisely, the pay-off values of the four simulation scenarios from Section 1.2.2 need to be quantified. At several occasions throughout this report it is mentioned and motivated why simulation is the method of choice for this. This chapter finally, guides the reader through this simulation and subsequently presents the simulation results.

It does so by briefly summarizing the adopted simulation process in Section 5.1. This is followed by some elaborations on the simulation parameters, the characteristics of VISSIM and the chosen simulation settings (Section 5.2). Besides others, this section contains a description of the N260 VISSIM environment. Finally, in Section 5.3, the simulation results are presented for each of the scenarios.

5.1. Simulation

As mentioned and motivated in Section 1.2.3 the VISSIM software is employed for the simulation of traffic impacts. Besides others this choice is made, because the software possesses the capability of accurately quantifying the state vectors $\hat{x}_i(t)$ of the simulated vehicles. The information that is conveyed through this state vector, is the location, the velocity and the acceleration of each vehicle at each simulation step. Self-evidently, this information allows for an assessment of the congestion status of a network. As introduced in the first chapter, the performance indicator for an evaluation of congestion values is chosen to be the average speed of all vehicles. The motivation behind this choice is as follows: The velocity is calculated as derivative of the location. Therefore

$$t = \frac{\bar{s}}{\bar{v}} \tag{5.1}$$

describes the arithmetic average of the time that is needed to cover a certain distance, given the average velocity. Hence, the average speed is proportional to the average travel time, which itself can be determined an important KPI for e.g. the traffic participants that seek a decision on whether to partake in platooning or not. Furthermore, according to the law of large numbers, the average result of a set of trials will eventually assume or get reasonably close to the expected value for that result. Put into context: In the long-term, the traffic participant will inevitably experience a change in travel time, which is proportional to the change in average velocity $\bar{v}$. Subsequently, the scenario differences in average velocity constitute not only a good representation of the congestion savings, yet also an easily measurable metric, given the simulation setup.

Having understood, that the average velocity holds direct relevance for the traffic participants, it additionally needs to be understood that the law of large numbers can not only be employed in the interpretation of this value but also in its calculation. That is, with 5198 vehicles entering and leaving the N260 traffic network over a duration of 60 minutes, the average of their velocities gives a good representation of the expectable real-world results.

Yet, besides the obvious evaluation of the average velocity, the state vector also allows for an estimation of emissions. This can be achieved by feeding the state vector (and the vehicle type) into the EnViver software. EnViver "combines results of traffic simulation software with emission models. This enables to predict and to study the environmental impact of traffic. Both for existing and modified/future situations" (TNO, 2017, p. 1). The software was calibrated in 2016 through exhaust measurements on more than 3000 cars in a variety of conditions. The measured emissions are linked to a respective state vector for each vehicle type. Consequently, the program is able to look up almost any real-life emission for a given state vector and vehicle type. If the look-up state vector does not exist, the program self-reliantly interpolates an emission value through the combination of similar state vectors and their emission entries. Thus, through the use of EnViver, an accurate and comparable quantification of the emission savings can be reached.

By calculating and storing the state vectors for every vehicle at every time step, both the congestion and the emissions of that scenario can be calculated. Consequently, generating and gathering this data is the main task of VISSIM and the basis of this chapter.

5.2. SIMULATION SETUP

"VISSIM is a microscopic, time step-oriented and behavior-based simulation tool for modeling urban and rural traffic." (PTV, 2016, p. 25). It is constituted by a traffic flow model, which itself relies on an external driver model and a light signal control model (LSC). The latter two are DLL add-ons that replace certain functions of the traffic flow model with refined algorithms and/or a changed system behavior. As illustrated though the colors of the software components in Figure 5.1, the main contribution of this work centers around the external driver model DLL. Furthermore, as mentioned earlier, it connects to the traffic floe model of VISSIM through a pre-defined API interface, which allows for the exchange of vehicle information between the traffic flow model and the EDM. Seen on a higher level, the exchange of data between these two components has allowed for the implementation of platooning in Chapter 3. Similar goes for the LSC DLL. It is the component, which allows for the implementation of an intersection controller. In this case, this is the implementation of JUNO, which interacts with the central traffic-flow model in a similar manner as the EDM DLL.

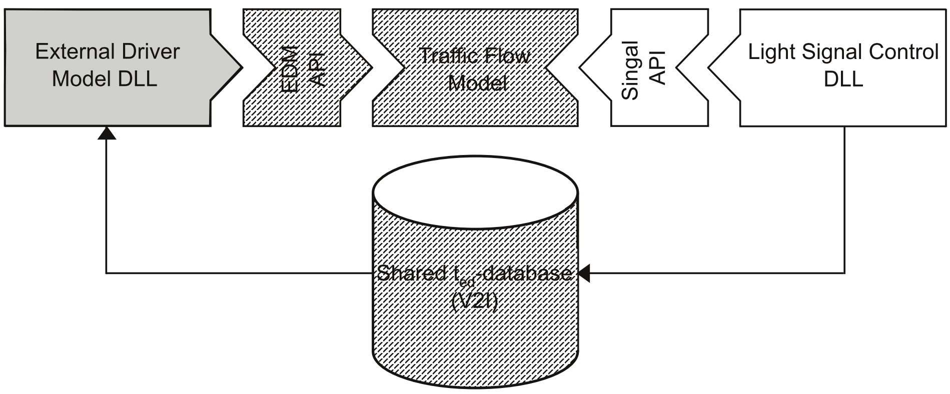


Figure 5.1: Information architecture of the VISSIM traffic model (Gray represents a research contribution; Shaded gray represent a contributions that are based on existing, already established systems or components).

Characteristic for both API interfaces is that they feature a delay of at least 0.1 seconds. This implies that conveying information from one DLL to the other will take twice as much time. The data would have to get from e.g. the LSC control to the traffic flow model where it is saved so that it can be requested through the EDM DLL in the simulation step after this. In Chapters 2 and 3 it is motivated why a delay of $t = \tau = 0.1s$ is a reasonable representation of real-life matters. The fact that the two DLLs can only exchange data with twice that delay is therefore undesired. To conquer this, a shared database is employed, where data can directly be stored and requested. As the estimated time of departure t_{ed} is in fact, the sole interaction between the two systems, the database is called the t_{ed}-database. In contrast to conveying data through the traffic flow model, this solution features the desired time delay of 0.1 seconds. Yet, in order to reach a delay of 0.1 seconds between all the software component it is a requirement that the traffic flow model uses a simulation frequency of 10 Hz, hence a simulation time step length of 0.1 seconds. This is done not only for all subsequent simulations but it was also applied for the controller calibrations in the previous chapters.

The simulation is executed for 60 minutes for each scenario. The traffic data, that is, the demand from all network entries is based on a traffic count which was executed over this duration from 10:00:00 to 14:00:00 during an average week day on the N260. For all scenarios the demand is exactly the same. Vehicles enter the network at the same location and at the same time as they do in the other scenarios. Furthermore, they have the same destination for all simulation runs. Hence, comparability between the results can be guaranteed. The scenarios only differ in the way how they facilitate this demand.

Thus, full comparability can be assumed between the simulations. Yet, especially those that incorporate platooning imply a vastly different driving behavior. As explained throughout the last two chapters, this driving behavior comprises a combination of three different platooning modes, which themselves rely on a set of parameters. These parameters constitute a specification of the driving behavior in the different modes.

Table 5.1: Simulation parameters for the simulation of the four simulation scenarios.

Simulation paramaters	Value
Simulation step length τ	$0.1s$
Maximum deceleration for MLVP mode a_{min}	$-2m/s^2$
Longitudinal detection distance for adjacent cars in SLVP mode $d_{SLVP,max}$	$15m$
Longitudinal detection distance for preceding cars in CACC mode $d_{CACC,max}$	$30m$
Maximum number of vehicles in a platoon (including the platoon leader)	5
Time gap for all platooning modes h	$0.3s$
Standstill distance for all platooning modes r_i	$2.5m$

Additionally, all scenarios can also feature situations where vehicles travel alone. That is, e.g. if no platooning partners can be found in the maximum longitudinal detection zones or if platooning is simply not part of the scenario. In that case, the VISSIM-internal simulation parameters are used.

5.2.1. THE N260 SIMULATION ENVIRONMENT

Figure 5.2: Real-world satellite picture of the *Middeldijkdreef*-intersection (Maps, 2017a).

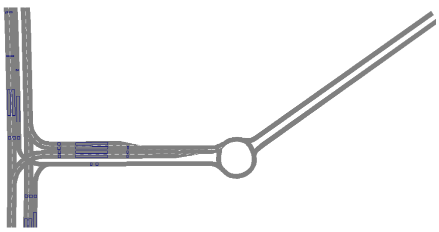


Figure 5.3: Top-view of the *Middeldijkdreef*-intersection street model within the VISSIM simulation environment.

Figure 5.4: Real-world satellite picture of the *Dalemdreef*-intersection (Maps, 2017b).

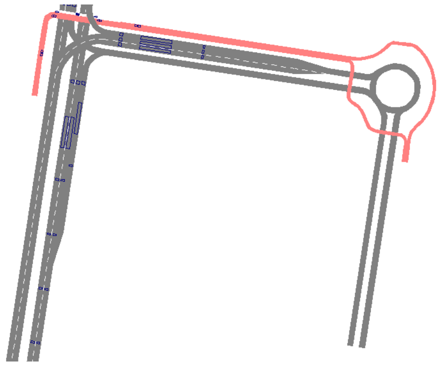


Figure 5.5: Top-view of the *Dalemdreef*-intersection street model within the VISSIM simulation environment.

Figure 5.6: Real-world satellite picture of the *Koolhovenlaan*-intersection (Maps, 2017c).

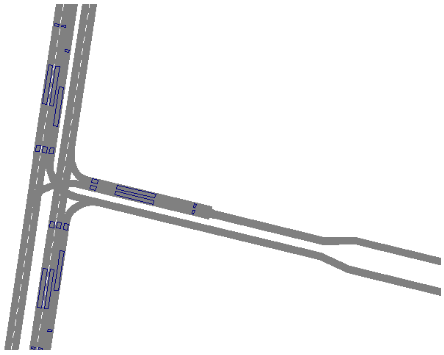


Figure 5.7: Top-view of the *Koolhovenlaan*-intersection street model within the VISSIM simulation environment.

As briefly introduced in the first chapter, the N260 (Noord-Brabant, 2017b) is a provincial road in North Brabant, The Netherlands. The subject of simulation within this chapter is a section of this road, which is part of the Tilburg Ring. Covering roughly $6200m$ in the North-South direction, the Section features three intersections, one with the *Middeldijkdreef*, one with the *Dalemdreef* and one with the *Koolhovenlaan* (going from North to South). A satellite view of the intersections, as well as their respective VISSIM structures are illustrated in Figures 5.2 to 5.7 The maximum speed limit on the N260 is $80km/h$, whereas the three distributor roads are restricted either to 50 or $30km/h$. The N260 has two lanes in both directions over the full length of the considered section. Each of the intersections is equipped with a variety of detectors, with at least three sensor areas located in front of every intersection entry.

Furthermore, the section with the *Dalemdreef* features a set of pedestrian crossings. The intersection entry and exit each feature two crossing areas and additionally, the intersection allows pedestrians to cross the N260 itself. The latter crossing is equipped with pedestrian sensors, which detect the demand of a pedestrian to cross the intersection. In contrast to the vehicle detectors, this demand is only detected upon arrival at the intersection. Currently, all three intersections are controlled by means of a traffic-actuated controller, as described in Section 2.1.1.2. However, for the base case scenario a model-predictive control is assumed, as introduced and motivated in Chapter 1. This represents a slight improvement to the traffic-actuated control. The choice for this base case is founded in the fact that model-predictive control is the best situation the government can achieve without facing necessary investments. It is therefore found reasonable to compare any improvements to this situation. In 2010, a study has shown an average daily demand of 13.500 vehicles per day for this Section of the road (Noord-Brabant, 2017a).

5.3. SIMULATION RESULTS

In the following the simulation results for the four simulation scenarios are introduced. The results in this sense, are not the calculated state vectors that VISSIM exports, but the congestion and emission evaluations of the different setups. The data is already processed through EnViver. The transformation from VISSIM output data to emission and congestion values is not further elaborated on. Self-evidently, only the fourth scenario makes use of the shared t_{ed}-database, as it is the only one, which features an interaction between the systems. Furthermore, the second and the third scenario employ only one of the DLLs and the base case scenario solely relies on the VISSIM-internal driving behavior and the base case signal control, namely the model-predictive controller, that was introduced in Chapter 1. Each VISSIM simulation output is made subject of a comprehensive emission report, which can be found in Appendices B, C, D and E for Scenarios 1, 2, 3 and 4, respectively. In the following, a short summary of the results is provided for each scenario, followed by a concluding section, which compares and differentiates the results.

5.3.1. SCENARIO 1: NO PLATOONING, NO JUNO

Scenario 1 constitutes the base case, solely relying on the model-predictive intersection control system, as described in Section 5.2.1. Neither platooning, nor JUNO are implemented for this simulation run. All vehicles make use of the VISSIM-internal driving behavior, which was explained in Section 3.3 and represents that of a human driver. The subsequent overall CO_2-emissions equal 216.68 g/km. They are split up among the different vehicle types, as illustrated in Table 5.2. Furthermore, the average speed of all vehicles over the simulation period of 60 minutes equals 70.7 km/h.

Table 5.2: Impact evaluation for Scenario 1, concerning the average velocity and the overall emissions for the N260 case study.

Scenario 1: $\bar{P}, \bar{I}$	
Performance indicator	Value
Average vehicle velocity	70.7 km/h
Average CO_2 emissions per km	216.68 g/km
LDV CO_2 emissions per km	189.48 g/km
HDV - heavy CO_2 emissions per km	1246.12 g/km
HDV - medium CO_2 emissions per km	662.81 g/km

72.03 km/h

5.3.2. SCENARIO 2: NO PLATOONING, JUNO

For the JUNO scenario, the average CO_2-emissions per km equal 210.60 g/km. Furthermore, an average speed of 72.03 km/h can be reached.

Table 5.3: Impact evaluation for Scenario 2, concerning the average velocity and the overall emissions for the N260 case study.

Scenario 2: $\bar{P}, I$	
Performance indicator	Value
Average vehicle velocity	72.03 km/h
Average CO_2 emissions per km	210.60 g/km
LDV CO_2 emissions per km	183.39 g/km
HDV - heavy CO_2 emissions per km	1216.54 g/km
HDV - medium CO_2 emissions per km	639.69 g/km

5.3.3. SCENARIO 3: PLATOONING, NO JUNO

For the platooning scenario, the average CO_2-emissions per km equal 208.84 g/km. Furthermore, an average speed of 71.21 km/h can be reached.

Table 5.4: Impact evaluation for Scenario 3, concerning the average velocity and the overall emissions for the N260 case study.

Scenario 3: $P, \bar{I}$	
Performance indicator	Value
Average vehicle velocity	71.21 km/h
Average CO_2 emissions per km	208.84 g/km
LDV CO_2 emissions per km	182.13 g/km
HDV - heavy CO_2 emissions per km	1211.96 g/km
HDV - medium CO_2 emissions per km	620.29 g/km

5.3.4. SCENARIO 4: PLATOONING, JUNO

For the scenario with a functional combination of the technologies, the average CO_2-emissions per km equal 208.85 g/km. The average speed is 71741 km/h.

Table 5.5: Impact evaluation for Scenario 4, concerning the average velocity and the overall emissions for the N260 case study.

Scenario 4: *P, I*	
Performance indicator	Value
Average vehicle velocity	71.74 km/h
Average CO_2 emissions per km	208.85 g/km
LDV CO_2 emissions per km	182.14 g/km
HDV - heavy CO_2 emissions per km	1211.96 g/km
HDV - medium CO_2 emissions per km	621.15 g/km

5.4. SIMULATION CONCLUSION (LEVEL 3)

Table 5.6: Congestion and emission savings *C* and *E* of the four simulation scenario scenarios.

	Performance indicator	Savings value
Scenario 1	Average vehicle velocity	0 km/h
	Average CO_2 emissions	0 g/km
Scenario 2	Average vehicle velocity	1.33 km/h
	Average CO_2 emissions	6.08 g/km
Scenario 3	Average vehicle velocity	0.51 km/h
	Average CO_2 emissions	7.84 g/km
Scenario 4	Average vehicle velocity	1.04 km/h
	Average CO_2 emissions	7.85 g/km

In the previous, simulation results were collected on the four simulation scenarios. For each of the scenarios, the overall CO_2-emissions, as well as the average speed were retrieved. Finally, these values can be translated into emissions and congestion savings *C* and *E*, respectively. As scenario 1 represents the base case for the following evaluation,the savings of the other scenarios are calculated as the difference between their respective values and those of the base case. Doing so, leads to saving-values as illustrated in Table 5.6.

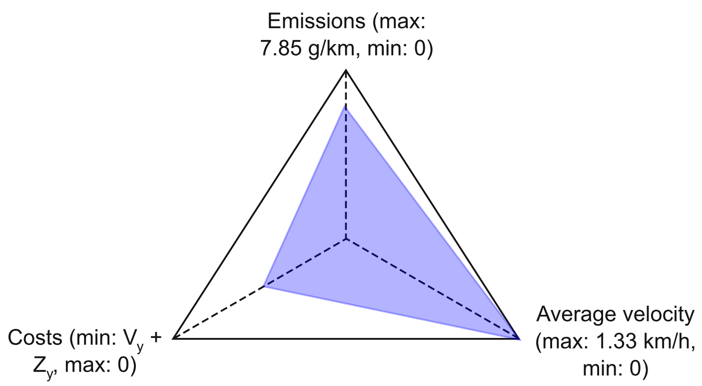


Figure 5.8: Semi-quantified evaluation of the *P, I*-scenario.

Finally, from this an answer to the simulation research question - *Which congestion and emission savings can be reached within each of the four scenarios of the N260-case study?* - can be formulated. The emission savings that can be reached are 6.08 g/km , 7.84 g/km, 7.85g/km for scenarios 2 to 4, respectively. The congestion savings are 1.33 km/h, 0.51 km/h and 1.04 km/h. They represent the emission and congestion savings

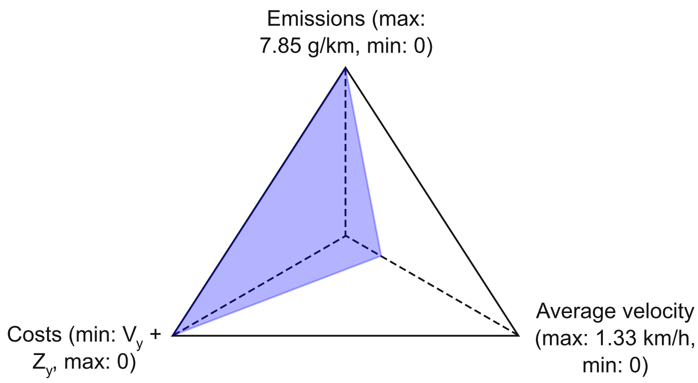


Figure 5.9: Semi-quantified evaluation of the P, I-scenario.

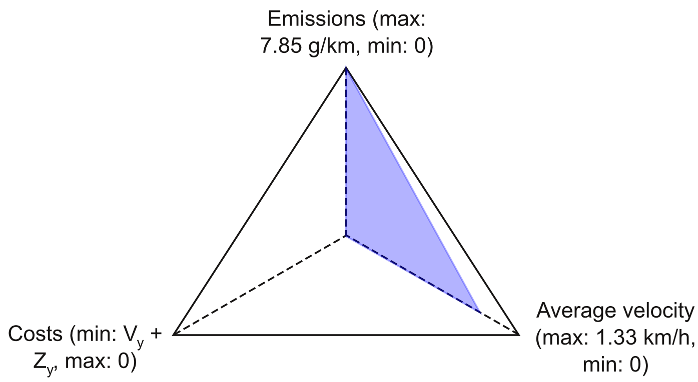


Figure 5.10: Semi-quantified evaluation of the P, I-scenario.

E and C that were introduced for the game theoretic pay-off calculations. The next Chapter will illustrate how these values can be utilized in order to gain further game theoretic and hence policy knowledge. Firstly however, some final remarks are made, motivating why scenario 4 is the most desirable outcome for these considerations.

That is, from the savings values no clearly dominant scenario can be identified. Furthermore, as illustrated in Figures 5.8 to 5.10, there is no one scenario that exhibits a strictly worse performance on all three objectives than another one. Note, that this is only true if both expenditures V_y and Z_y are positive. This will be confirmed in the next chapter. Now, while the JUNO scenario delivers the best congestion savings, the best emission savings can be found in the both platooning scenarios. The only thing that can directly be seen is that the base case scenario is clearly dominated in both objective categories. Yet, one scenario has to be declared the policy goal for the next chapter. In the context of this thesis, scenario 4 (P, I) will from now on be considered the policy goal. This choice is based on the assumption that a governmental body will likely have mutually strong interest in saving on congestion and emissions. However, it is acknowledged that this is not always the case. It is due to reasons of scope that the following high-level considerations are solely concerned with bringing the last scenario in place. However, if possible similar research should be made about the alternative scenarios 2 and 3 as well. Here, it has to be noted that scenario 2 does not require a policy. The government can simply decide to invest in innovative intersection control. In that sense, scenario 4 is the situation which requires the most effort to implement. Hence, the following chapter considers a worst case policy scenario.

6

HIGH-LEVEL IMPLICATIONS

The overarching goal of this report is to assess the congestion and emission impacts of a combined platooning-intersection controller and provide a policy consideration, which helps to promote the real-world integration of this system. In this sense, Chapter 1 gives two insights. Firstly, that a game theoretic approach can be used to gain strategically useful knowledge for finding a feasible policy and secondly, that this game theoretic approach requires quantified data on the potential congestion and emission savings of the proposed technology. For the latter insight a technical design is necessary, which is developed and explained in Chapters 3 and 4 and eventually simulated in Chapter 5. Finally, this VISSIM simulation in conjunction with the EnViver software provides the required quantified data. It does so for four different scenarios, which can now be incorporated into the game theoretic considerations from Chapter 1. This Chapter takes the simulation output of the low-level technical design and uses it as an input for a high-level game theory and policy consideration.

In this sense, Section 6.1 uses those inputs to follow the game theoretic (Level-2) objective and answer its research question. The section concludes with a conditional statement on when a Nash equilibrium can be triggered for the P, I-scenario. This condition serves as an input for Section 6.2, which itself checks whether and when a policy exists, which can trigger such a Nash equilibrium. By doing so it answers the first part of the the Level-1 research question. Furthermore, a policy-development framework is provided, answering the second part of the research question. Finally, Section 6.3 concludes this Chapter, summarizing its main findings and linking them to the objectives of the V-model.

6.1. GAME THEORY AND POLICY IMPLICATIONS

Chapter 5 has provided important data on the impacts of all four simulation scenarios in terms of congestion and emissions. In order to conclude policy knowledge from this, this data can now be incorporated in a two-player game between the government and the traffic participants. However, before this is done some remarks are necessary to elaborate on the choice of game theory as a method.

In Chapter 1, it was explained that game theory is a useful tool in order to structure a complex decision-making problem. Henceforth, it was applied in order to cope with the complexity of policy design and its added value was in this sense, that its application gave structure to the formerly unstructured complex network. This way, a common understanding of the problem could be established. Yet, it needs to be understood that game theory is not only useful to communicate a problem with the reader. Indeed, its main purpose is not only to map existing actor-interest relations but to also to predict the effects of certain measures on these relations. It can be used to manifest a simple, yet effective model of how different interventions will change the overall outcome of a multi-actor problem. In this sense, it can be used to predict, which of the four simulation scenarios is the most likely under a given policy, which subsequently provides useful information for a potential policy design. Despite, this capability it nevertheless exhibits certain drawbacks as well. These will be made subject of Section 6.2.2. For now its benefit, the capability of predicting system effects through certain measures needs to be understood.

More precisely, the applicable pay-offs for each player and each scenario from Formula (1.3) can be calculated. The congestion and emission savings from Section 5.4 can now be inserted in the pay-offs, which were introduced in Section 1.2.2:

$$\begin{aligned}
x_{P,I} &= w_{x,C}C_{P,I} + w_{x,E}E_{P,I} - V_x + Z_x && = w_{x,C}1.04km/h + w_{x,E}7.85g/km - V_x + Z_x \\
y_{P,I} &= w_{y,C}C_{P,I} + w_{y,E}E_{P,I} - V_y - Z_y && = w_{y,C}1.04km/h + w_{y,E}7.85g/km - V_y - Z_y \\
x_{P,\bar{I}} &= w_{x,C}C_{P,\bar{I}} + w_{x,E}E_{P,\bar{I}} - V_x + Z_x && = w_{x,C}0.51km/h + w_{x,E}7.84g/km - V_x + Z_x \\
y_{P,\bar{I}} &= w_{y,C}C_{P,\bar{I}} + w_{y,E}E_{P,\bar{I}} - Z_y && = w_{y,C}0.51km/h + w_{y,E}7.84g/km - Z_y \\
x_{\bar{P},I} &= w_{x,C}C_{\bar{P},I} + w_{x,E}E_{\bar{P},I} && = w_{x,C}1.33km/h + w_{x,E}6.08g/km \\
y_{\bar{P},I} &= w_{y,C}C_{\bar{P},I} + w_{y,E}E_{\bar{P},I} - V_y && = w_{y,C}1.33km/h + w_{y,E}6.08g/km - V_y \\
x_{\bar{P},\bar{I}} &= 0 && = 0 \\
y_{\bar{P},\bar{I}} &= 0 && = 0.
\end{aligned} \tag{6.1}$$

In Section 1.2.2 it was introduced that the second-level objective of this thesis is the creation of a Nash equilibrium for the P,I-scenario. This can be achieved if the desired scenario becomes the only logical solution to the established game. Easley et al. (Easley and Kleinberg, 2010, p. 166) define such a Nash equilibrium as follows: "Suppose that Player 1 chooses a strategy S and Player 2 chooses a strategy T. We say that this pair of strategies (S,T) is a Nash equilibrium if S is a best response to T, and T is a best response to S". In other words: If none of the players can be better of by changing his strategy, the subsequent scenario depicts a Nash equilibrium. The strategies in this sense are the possibility of the government to "Do invest" or "Do not invest" in intersection control and of the traffic participants to "Do partake" or "Do not partake" in platooning.

From the assumption of Chapter 1 that the government is in general supporting the integration of smart traffic solutions and willing to invest in those, it is for now followed that the government does not consider the "Do not invest" strategy. This effectively makes the 4-option game a 2-option game, where only the traffic participants have a choice. Although, this is usually not the case, it is a necessary follow-up assumption in order to answer both the game theoretic and the policy research question. In order to properly understand this, Section 1.2.2 is briefly revisited. In this section, the following research question was posed: *Can a Nash-equilibrium be triggered for the P,I-scenario in the platooning-intersection control game?* Accordingly, the goal of creating the introduced Nash equilibrium was set in the latter section. This however, does not imply that this is in every case the best solution for the government. Obviously, even the government has a certain limit to what it is willing to spend to reach congestion and emission goals. Assuming that the government will "Do invest" is therefore only a mind experiment to test which consequences, such a policy would have and if it is in general possible to create such a Nash equilibrium. This way an answer can be given to the policy research question by intermediately answering to the question: When is "Do partake" the best answer to the "Do invest strategy"?

Given above assumption, it can be followed that in order to create a Nash equilibrium where also the traffic participants invest in platooning, this scenario must make them better off then the $\bar{P},I$-scenario. In order to reach such a situation, the following must be achieved:

$$\begin{aligned}
x_{P,I} &> x_{\bar{P},I} \\
w_{x,C}C_{P,I} + w_{x,E}E_{P,I} - V_x + Z_x &> w_{x,C}C_{\bar{P},I} + w_{x,E}E_{\bar{P},I} \\
w_{x,C}C_{P,I} + w_{x,E}E_{P,I} - V_x + Z_x &> w_{x,C}C_{\bar{P},I} + w_{x,E}E_{\bar{P},I}.
\end{aligned} \tag{6.2}$$

The latter two lines of above equation can be found by inserting (6.1) into the first line of (6.2). Finally, this allows to quantify the resistance threshold of the traffic participants Z_x, which needs to be overcome to nudge them to chose the strategy, which the government desires - Do partake. From (6.2), this resistance follows to be:

$$\begin{aligned}
Z_x &= V_x - w_{x,C}(C_{P,I} - C_{\bar{P},I}) - w_{x,E}(E_{P,I} - E_{\bar{P},I}) \\
&= V_x - w_{x,C}(C_{P,I} - C_{\bar{P},I}) - w_{x,E}(E_{P,I} - E_{\bar{P},I}).
\end{aligned} \tag{6.3}$$

This is the resistance threshold, which has to be overcome through a governmental policy in order to nudge the traffic participants towards the desired move. If a policy is found, which is capable of doing so, the traffic participants will naturally rather chose to "Do partake" in platooning. Consequently, what has been proven at this point is that if the government is willing to cover the necessary costs (investments V_y and compensation Z_x), a Nash equilibrium can be established for the P, I-scenario. In a worst case scenario, where the traffic participants perceive no benefit from congestion or emission savings, Z_x would therefore fully compensate V_x. The answer to the game theoretic research question is therefore: Yes, a Nash equilibrium can be triggered for the P, I-scenario if the condition from formula (6.2) is fulfilled. Fulfilling this mathematical condition essentially means ensuring the correct level of compensation. In this sense, equation (6.3), which quantifies the right level of compensation concludes this section. Yet, it can be seen that a few parameters in this expression remain unknown. These are the respective weights $w_{x,C}$ and $w_{x,E}$, which the traffic participants assign to congestion and emission savings respectively, as well as the investment V_x that they associate with the choice to partake in platooning. These values vary quite significantly, based on a variety of factors.

6.2. POLICY ADVICE

The main purpose of this section is to provide an answer to the policy research question of Section 1.2.1. This is concerned with determining whether a policy can be designed, which triggers a Nash equilibrium, such as in the previous section and with providing a framework that can assist with this doability check. In other words: This section shows if and when a policy leads to a situation where formula (6.2) is satisfied and incorporates this knowledge in a policy-development framework. The motivation behind this second step, namely providing a framework which incorporates this formula, is as follows:

Although game theory has proven to be a useful tool to translate the simulation results from Chapter 5 into hands-on policy knowledge, it must not be forgotten that this brought along a set of modeling choices. That is, game theory is a technique that can be used to mathematically depict the decision-making process of the traffic participants and hence allow for strategic implications on how the government should behave. However, due to the simplifying nature of this modeling technique a set of simplifications are assumed in the process of deriving these implications. Given this, the section at hand is divided into two parts. Section 6.2.1 adopts the game theoretic knowledge from the previous section and derives a set of policy implications from this. Section 6.2.2 then reflects on the theoretical nature of these considerations and attempts to compensate made modeling simplifications through embedding the policy implications in a policy-development framework. Besides the technical design of Chapter 5, this framework represents the second major research contribution of this work.

6.2.1. THEORETICAL POLICY IMPLICATIONS

In order to answer the first part of the policy research question, namely under which conditions a policy is possible, the assumption from the previous section that the government will always "Do invest" needs to be annulled. This effectively makes the former 2-option game a 4-option game again. It should by now be understood that this action was initially taken in order to prove that "Do partake" can be a best response strategy to "Do invest". Given, that a policy exists, which provides the necessary compensation (i.e. (6.3) is fulfilled), the following is known:

- If the government decides to invest and deploy the policy, the logical consequence will be the P, I-scenario.
- If the government decides to not invest and not deploy the policy, the logical consequence will be the $\bar{P}, \bar{I}$-scenario.

The latter is based on the assumption that a resistance threshold exists in the first place (see Section 1.2.2 for a motivation). If this is the case, the traffic participants would rather "Do not partake" as a response to "Do not invest". From this it can be followed that in order to justify a compensation policy, the governmental pay-offs for the P, I-scenario must be higher than those for the $\bar{P}, \bar{I}$-scenario. As the latter represents the base case ($y_{\bar{P},\bar{I}} = 0$), the following statement must be true for the policy to be worthwhile:

$$y_{P,I} > 0. \tag{6.4}$$

From this a direct implication for the policy research question can be made. That is, only if the above formula is satisfied, it is possible to design a policy, which does trigger the desired Nash equilibrium. Hence, this represents the condition under which a policy design is theoretically feasible. To determine when this is the case, the governmental P, I-pay-off requires further analysis.

Similar to the traffic participants, it is from now on understood that the government has a certain threshold, which determines which option is more favorable from a governmental point of view. In order to grasp this threshold, the introduction chapter is revisited. In Section 1.2.2 it was mentioned that the governmental policy expenditures Z_y do not necessarily need to equal the perceived compensation of the traffic participants. That is, the compensation measure does not need to be of financial nature. The policy design space is willingly left open, which is why their relation is only generally described as:

$$Z_y = \frac{1}{k_{eff}} \cdot Z_x. \tag{6.5}$$

k_{eff} represents an effectiveness coefficient. In a worst case scenario k_{eff} is equal to 1. This is the case if the government decides to simply transfer money towards the traffic participants. Usually however, the government has the possibilities to take other measures, whereas the perceived compensation Z_x is considerably higher than the expenditures of the government. This relation allows to delineate the governmental expenditures as a function of the necessary compensation from formula (6.3). The expenditures equal to:

$$\begin{aligned} Z_y &= \frac{1}{k_{eff}}(V_x - w_{x,C}(C_{P,I} - C_{\bar{P},I}) - w_{x,E}(E_{P,I} - E_{\bar{P},I})) \\ &= \frac{1}{k_{eff}}(V_x - w_{x,C}(1.04km/h - 1.33km/h) - w_{x,E}(7.85g/km - 6.08g/km)). \end{aligned} \tag{6.6}$$

Now these expenditures lay the basis for the two next reseach steps. Firstly, they allow to revisit the evaluation framework that was applied in Section 5.4. It was here shown that none of the scenarios is strictly dominant under he condition that the expenditures V_y and especially Z_y are positive. From the above, this can now be confirmed, as no policy would be needed if the resistance threshold was negative in the first place. Hence, the underlying statement of Figures 5.8 to 5.10 are from now on considered confirmed. Secondly, the above can now be inserted into the governmental P, I-pay-off value calculation from equation (6.1). In a similar manner to the line of reasoning of the previous section, it is found that the following needs to hold in order to make make the P, I-scenario worthwhile for the government.

$$\begin{aligned} y_{P,I} &> 0 \\ w_{y,C}C_{P,I} + w_{y,E}E_{P,I} - V_y - Z_y &> 0 \end{aligned} \tag{6.7}$$

Inserting (6.6) into (6.7) leads to:

$$\begin{aligned} w_{y,C}C_{P,I} + w_{y,E}E_{P,I} - V_y + \frac{1}{k_{eff}}(-V_x + w_{x,C}(C_{P,I} - C_{\bar{P},I}) + w_{x,E}(E_{P,I} - E_{\bar{P},I})) > 0 \\ w_{y,C}1.04km/h + w_{y,E}7.85g/km - V_y \\ -\frac{1}{k_{eff}}(-V_x + w_{x,C}(1.04km/h - 1.33km/h) + w_{x,E}(7.85g/km - 6.08g/km)) > 0 \end{aligned} \tag{6.8}$$

This equation is the last contribution in terms of theoretical policy implications. It illustrates the condition under which it is possible to trigger a Nash equilibrium for the P, I-scenario and for which this scenario also makes the government better off. If this mathematical condition is fulfilled, the policy is considered feasible for implementation (on a theoretical level). The answer to the first part of the policy research question can therefore be formulated as: A policy that incentivizes the use of traffic solutions, which themselves pursue the goal of saving on congestion and emissions without road expansion is theoretically possible if the condition of formula (6.8) is fulfilled.

Yet, as it was mentioned within Chapter 1 and the introduction of this section, the presented implications should be handled with care. The reason for this is that they represent scientifically-based theories, rather than real-world proven insights. On one hand, it is the purpose of traffic research to inform policy-makers with scientifically sound numbers. On the other hand, however, these numbers do not depict a one-to-one model of reality. Hence, although above equation represents valuable inputs for policy design, further, high-level considerations need to be made. In the following, a policy-development framework is presented, which attempts to map these considerations and by doing so embeds the theoretical insights from this section in a higher-level, more far-sighted context.

6.2.2. PRACTICAL POLICY IMPLICATIONS

In the previous section a set of game theoretic considerations were made, from which a policy-level insight was deducted. Coming along with that, a set of mathematical conditions was provided, which can be used to check whether a policy can be established and whether it is worthwhile for the respective policy maker to do so. Hence, if the government manages to find a policy which satisfies these conditions, this policy has proven workability on a game theoretic basis. That is, if the government decides to cover the financial threshold and further invest in innovative traffic controllers, the urban platooning scenario is the sole logical consequence in a game theoretic context.

Yet, one must not forget that these game theoretic considerations only represent a simplified model of real-world matters. Modeling the traffic participants' decision-making process brings along a trade-off choice, whereas game theory appears to be a reasonable compromise between model simplicity and real-world representativeness. In this sense, the following needs to be understood: It is impossible to reach both, perfect model representatives and easy model functionality. Hence, rather then depicting a short-coming of the model, the made game theory simplifications represent a willingly agreed-upon modeling choice. This choice was aimed at providing a reasonably simple traffic and decision-making model, which can be reproduced for other settings, while also allowing for an accurate level of implications. And indeed, this aimed-at compromise was reached. The previous section has presented a set of implications, which depict valuable governmental insight. Especially, equation (6.8) can be utilized as a quick way to check the theoretical doability of a policy. However, the simplifying nature of this choice needs to be kept in mind and handled with attention. Given this, one major purpose of the policy-development framework is to revisit the made modeling assumptions, the accepted simplifications and the limited data value that is drawn from this.

When revisiting the low-level design requirements of Section 1.2.4 and the subsequent design process, one will remember that the platooning-intersection controller was designed as such that it is compatible with a wide range of intersection controllers and platooning systems. This is ensured through the design objective that the controller should minimize ins intervention in the underlying functionalities of the concomitant systems. Hence, it can be assumed that a widely applicable system solution was found within the low-level technical design. Subsequently, only those model simplifications that are either associated with the VISSIM traffic simulation or with the game theoretic decision-making model require further consideration through the framework and are thus taken into account. In the following, a list of the most stringent of those simplifications is provided in Table 6.1.

Table 6.1: A non-exhaustive list of the made modeling assumptions that have an influence on the level of representativeness and hence, on the theoretical policy implications.

Modeling level	Simplification	Explanation
Traffic simulation	Traffic density	For the evaluation of the technical design and the calculation of the expectable benefits C and E a scenario was chosen, which represents an average traffic situation. Yet, it can not be assumed that this situation, especially the traffic density is always a good representation of real-life matters.

	Physical setup	Furthermore, a certain physical setup was assumed for the above. This comprises the sensor configuration, as well as the intersection setup and the street layout itself. Again, for the evaluation of C and E a scenario was chosen, which is believed to represent an average physical setup, yet it can not be ensured that this is always representative.
Game theory	Perfect information	Within the game theory section it is assumed that both players have perfect information for their decision-making. This is especially concerned with the expectable benefits and the associated costs of each move of which the traffic participants might not be well informed.
	Full player rationality	Further, it is assumed that especially the traffic participants act rationally. That is, in real-life someone might still decide for an alternative which leaves him worse off, even if he has perfect information on the decision and hence knowledge about its implications.
	Player homogeneity	Lastly, it is assumed that all traffic participants have the same values and hence act the same. Although, a certain homogeneity exists among the traffic participants, this surely is not absolute.

For all these assumptions, reasoned motivation exists. As mentioned the process of modeling a decision-making process essentially represents a trade-off between model simplicity and accuracy. On one hand it is the purpose of a model to make certain simplifications in order to allow for model functionality and usability. On the other hand however, these assumptions go along with a decrease in model representativeness.

The made assumptions, which are outlined in above list are knowingly accepted in order to allow for the former - enable easy model functionality and usability. In this sense, the simplifications have served their purpose. Yet, what needs to be taken hold of is that there is an uncertainty that is associated with this deducted knowledge. This is due to the latter - the decreased model representativeness.

Surely, in order to cope with these model inaccuracies, the policy-maker needs to be made aware about their existence. Hence, the first logical step is to inform the problem owner about the made modeling assumptions and simplifications. Yet, this research aims to further build upon this and provide a generically applicable way of dealing with these issues. For this, a policy-development framework is provided. However, before this is thoroughly introduced, two remarks on the requirement of "generic applicability" need to be made:

Firstly, this means that a framework which can be used in other settings as well, is aimed at. that is, even if the government decides to change the physical scope, the presented traffic model can still hold a certain value for the policy-design process. Although, certain modeling parameters clearly need to be renewed, the framework desires to carve out those insights that still hold relevance for such a situation and make use of them. Secondly, it needs to be noted that above list of simplifications is non-exhaustive. It is due to the nature of a complex network that it is impossible to map out all the aspects in which the model differs from the real world. Although being bothersome, this fact needs to be accepted and taken into account when attempting a real-world deployment of the proposed model amendments. In this sense, the framework attempts to offer a set of tests that are capable of detecting these unnoticed inaccuracies and coping with them.

THE POLICY-DEVELOPMENT FRAMEWORK

Figure 6.1 presents the aimed at framework and herewith the answer to the second part of the policy research question "[...] how can such a policy be implemented?". Those that are familiar with the theories of policy design and deployment might notice that it is loosely based on the CDC policy process-circle (CDC, 2017) and indeed its underlying steps and its iterative nature are similar to the CDC framework. The choice for basing the policy-development framework in the CDC circle is made willingly. The CDC framework is an established and proven assistance in the design and deployment of policies. Hence, the policy-development framework of this research attempts to adapt the underlying strategy of the CDC while expanding it by the insights that were made through the low-level design.

The CDC is a generic policy-development framework that attempts to cover a broad scope and a high-level point of view. The main difference between the two is that the CDC circle attempts to assist the design of all forms of policies, while the framework at hand is solely made for the purpose of promoting urban platooning through a policy design. This allows for the formulation of more concrete steps towards the goal of an effective policy design. However, through its universal validity, the CDC can very well be used as a guideline, towards a policy-development framework that is centered around the introduction of urban platooning only. In this sense, the framework at hand is a specific amelioration of the generic CDC circle.

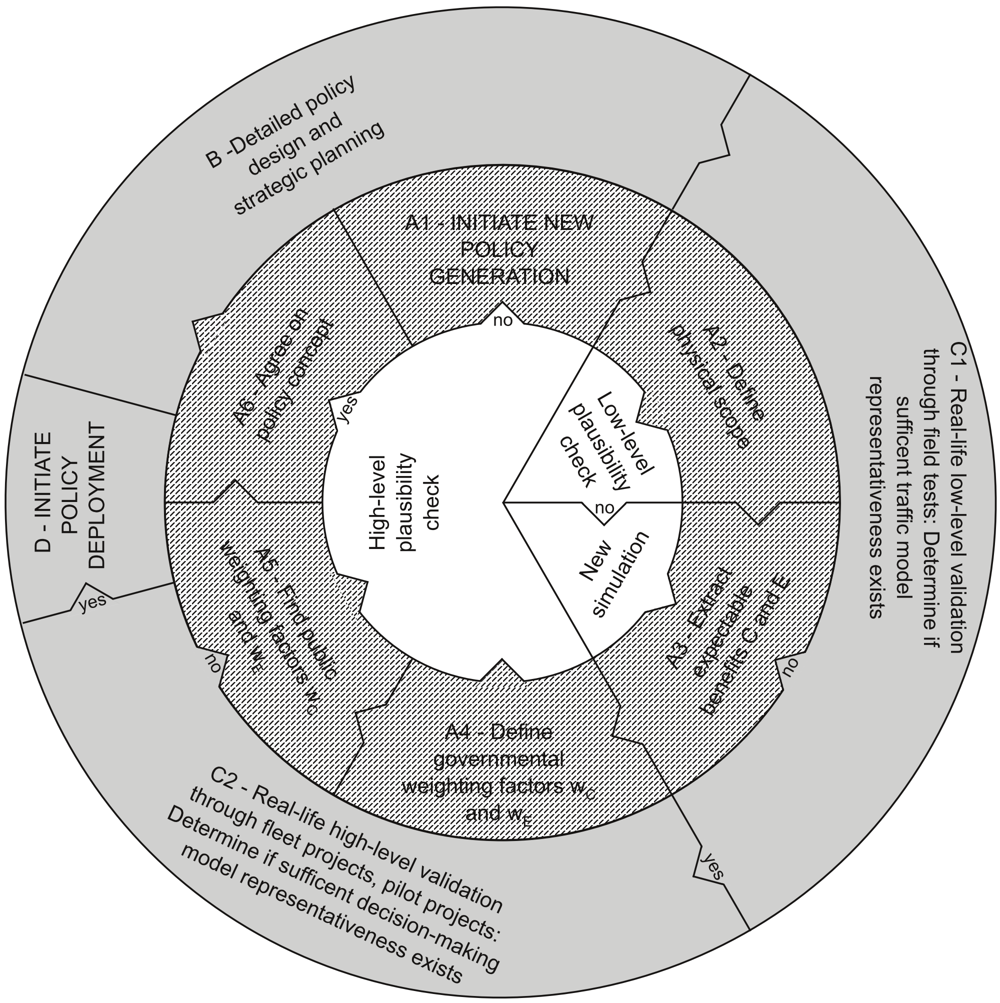


Figure 6.1: Generic policy-development framework, promoting the introduction of urban platooning on Dutch roads.

As it can be seen, the newly developed framework is divided into three circles. These circles represent the three main stages of a policy design - the policy generation (shaded gray), the theoretical policy validation (white) and the practical policy testing & deployment (gray). Especially, the theoretical policy validation and the practical policy testing represent expansions to the CDC framework.

The CDC circle distinguishes five policy development steps, of which four can be found in the presented framework of Figure 6.1. The reason why the initial "Problem identification" is not reproduced in the urban platooning framework is that the latter's scope is already pre-defined. Through the context of this research it has been made clear that the subject of interest is solely that of a quick and effective introduction of urban platooning in order to tackle congestion and emission problems. Hence, there is no need for a problem definition as proposed by the CDC.

Instead, the urban platooning framework starts off with what the CDC describes as the "Policy generation and analysis". In both framework this phase centers around the evaluation of a potential policy solution. The urban platooning framework further divides this phase into five sub blocks. The first of those is the "Initiate new policy generation"-block (A1). The sole purpose of this is to conceptualize a policy solution. Once, such a concept is found, its evaluation begins. Firstly, a physical scope is defined (A2). In order to decide whether the presented traffic model of the N260 is sufficiently representative for the chosen physical scope, a low-level plausibility check can be used. As illustrated in Figure 6.2, this check is concerned with three major validity questions and only if all of those are answered with yes, it can be assumed that the expectable benefits C and E are representative for the chosen physical scope. The reason for the choice of questions is that the three parameters, the traffic density, the sensor setup and the street layout were determined most crucial for the performance of an intersection (see e.g. Bowen and Eubank, 2014; Zhou et al., 2010). Hence, if all three are within an allowed range of the N260 case, it can be assumed that they will have similar benefits for an implementation of urban platooning. If this is not the case however, it is proposed to initiate a new traffic simulation with a simulation environment that resembles the physical scope more accurately. Either of the cases allows to eventually extract the expectable benefits C and E in step A3. The advantage of step A2 and the low-level plausibility check however is that for the average case (for which the N260 simulation environment was chosen) no new simulation is necessary. Hence, a very quick and easy way to roughly estimate the expectable benefits is provided.

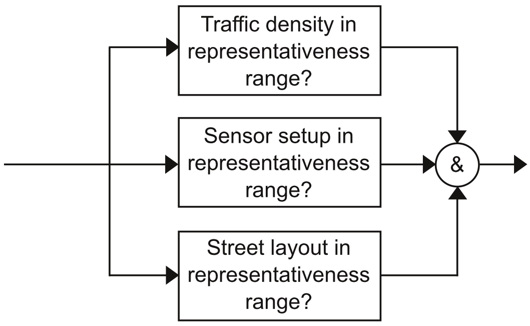


Figure 6.2: Low-level plausibility check to validate the representativeness of the traffic scenario simplifications.

In A4, the government internally determines what weight it assigns to either of the benefits. Given the units of E and C, these weights should be expressed as $[w_E] = € \frac{km}{gCO_2}$ and $[w_C] = € \frac{km}{s}$. Put in words these ask the questions: How much are a saving of 1 gramm CO_2 per km and of 1 second travel time per km worth to the government? Together with the public weighting factors that are determined in A5 (e.g. through questionnaires or a discussion rounds), these values serve as a basis for the high-level plausibility check. This check utilizes equation (6.8) from the previous section to determine whether the conceptual policy is generally worthwhile. If this is not the case, the policy generation is terminated and the process starts over with a new policy concept. If the test is passed however, the policy can proceed to the next steps.

The next steps are focused on testing whether the theoretically feasible policy can also prove practical feasibility. For this, the policy concept is firstly developed into a complete policy and strategy plan in step B. After this is done, the real-life testing begins. Step C1 is centered around the question if the theoretical quantifications of C and E resemble those of the real-world. It is proposed to execute a field test that is capable of determine real-world effects of urban platooning. Although the government is free to decide how exactly such a field test should be set up, it is recommended to use testing facilities, as e.g. that of PAVIN in Clermont-Ferrand (Avanzini et al., 2011). Doing so will not only minimize the costs for a field test but also allow for further expert input through on-site engineers. Eventually, the field test will either show that the expected benefits resemble real-life benefits or the opposite. If the latter is the case, the policy will move back to step A3 where its theoretical feasibility is checked again, given the new real-life insights. If the benefits are proven to be representative however, the policy can proceed to step C2.

Step C2, proposes a similar method as C1 with the aim of validating the public weighting factors w_C and w_E. The only difference is that the opinion of a policy can not be tested through field tests. It is rather suggested to proof the values (that were initially collected through questionnaires and discussion rounds) by means of fleet or pilot projects. This way, further insights on the public decision-making process can be gained while the possibility for a major misinvestment through deployment of an inefficient policy is further decreased. For the case that the test is unsuccessful, it is suggested to once again check the theoretical feasibility of the policy, given this new knowledge on w_C and w_E. If the test is successful and it is proven that the assumed public decision-making process is representative, the policy can finally be deployed on a large scale.

One of the major strength of the presented policy-development framework is that it allows to quickly test the theoretic doability of a a policy concept. Only by using the inner two circles (and hence without any investments) an approximate evaluation can be conducted, hinting at the success possibilities of the concept. Through expanding the CDC circle with this theoretic assessment, it is possible to test a bigger set of concepts in a shorter time, as by utilizing the CDC only.

Furthermore, it is suggested that the process as at multiple times return to the inner circles if one of the real-life tests of the outer circles has shown unexpected results. Through this measure, a policy whose development would need to be canceled according to the CDC circle can again be checked for its theoretic doability. Hereby, a more efficient policy evaluation can be reached.

Yet another benefit of the inner-outer circle relation is that through the proposed real-life tests further simplifications or model shortcomings that were not expected can be identified. Although the outer circle tests are meant to validate a set of critical modeling parameters, this is not their sole purpose. Surely, testing a new technology in a field test or pilot project will unveil further not-expected implications, if those exist.

6.3. GAME THEORY AND POLICY CONCLUSION (LEVEL 1 & LEVEL 2)

In the previous two sections answers were given to the policy and the game theoretic research question. In both cases, the question was answered with a conditional yes, whereas the conditions are expressed in formulas (6.2) and (6.8) for the game theoretic and the policy research question, respectively. In this sense, a concluding remark about these formulas needs to be made. At this point it should be understood that the latter condition is in mathematical terms based on the former. The game theory condition can be found inside the policy condition. Revisiting the research process, it becomes clear that this is due to the cascaded relation of the research question. The two equations represent the two upper-level deliverables of the right arm of the V-model. The objectives leading to these deliverables were purposefully designed as such that the Level-2 deliverable would serve as a input for the Level-1 deliverable. This goes along well with the research paradigm of the lower-level deliverables, which have shown a similar relation. The deliverables of the lowest level were used as an input for the third level and those of the third level represent the starting point of this chapter.

Yet, it is not only logical but also desired that such a relation exists. Due to the cascaded relation of the research questions and their deliverables, the final policy deliverable comprises elements of all lower levels. Following this notion, it is not necessary anymore to check if a Nash equilibrium exists, if formula (6.8) is already fulfilled. Hence, the policy condition with which the previous section has concluded is a powerful tool. It does not only answer the question whether a certain policy is possible but also the question whether the expectable benefits justify such a policy and whether the proposed technologies actually contribute to an improvement of the situation. If the government can find a policy setup, which satisfies this condition, all above questions can be answered positively.

Yet, it has also been reflected that the interrelatedness of this cascaded research approach is not always beneficial. At multiple occasions throughout the research process, model simplifications or assumptions were made which henceforth are conveyed from the lowest to the highest level and eventually to the policy deliverable. Although, this characteristic is a knowingly accepted evil, it requires further consideration. For this, the second part of the policy research question is answered by means of a policy-development framework, which revisits these simplifications and assumptions and proposes a set of tests and checks in order to prevent any wrong decisions as a consequence of blindly accepting the model's predictions. The model outcomes are validated in a two-stepped approach. An initial theoretical validation is followed by various real-life test, which altogether ensure the right use if the model insights.

7

Validation and verification

Within this chapter, two considerations are made, namely, the verification and the validation of the VISSIM simulation from the previous chapter. This choice of subject is founded in the V-model, which was determined the research approach of choice for this thesis. Within the introduction chapter, it was elaborated on, how a high-level objective is broken down into a low-level technical design goal. This left arm of the V-model is then supplemented through the actual design, a simulation of that design and subsequent game theory and policy considerations. The latter represent the right-arm pendant to the left-arm objectives and goals, hence the contribution of this thesis. It is therefore the right arm, which requires validation and verification, asking the question whether, this contribution fulfills the demands that were made in the formulation of the left-arm objectives. Having understood that the design process took a bottom-up approach, where e.g. the game theoretic considerations are based on the functionality of the low-level technical design, this chapter adopts a similar line of reasoning.

Only if low-level functionality is verified and the simulation is validated, a statement about the high-level considerations can be made. Additionally considering that the main share of this research is concerned with this low-level design, the chapter at hand, concentrates on the question whether the low-level design complies with said objectives (Section 7.1), before briefly extrapolating this to the high level (Section 7.2). This comprises both, the objectives that were made explicit and those, which are implicitly assumed through those explicit objectives. In the context of a combination of platooning and intersection control, the former is mostly concerned with the implicit requirement of general functionality, while the latter considers the representativeness of the simulation.

7.1. Low-level verification and validation

Traffic simulations are approximations of real-world traffic situations, pursuing a best-fit imitation of reality. This implies, that such a traffic model requires both validation and verification to ensure sufficient representativeness for the purpose, it is made for (Sargent, 2005). Consequently, this section aims at verifying and validating the VISSIM simulation, which was introduced in the previous chapter. Seeking a methodology for this, plenty of literature exists, suggesting various ways of testing the real-world representativeness of a model. Two publications that are especially relevant for this research are the approaches of Daamen et at. (2014) and Dowling et al (2004). Dowling et al. propose a generic validation structure for microscopic simulation models, upon which Daamen et al. build by adding some notions on the validity of traffic simulations. The nine-step validation plan, which is employed for this research is displayed in Figure 7.1. It constitutes a synthesis of both previously-mentioned approaches, combining those aspects that are relevant for the problem at hand.

As it can be seen, the verification represents an intermediate step within the overall validation process. This notion is supported not only by Daamen et al. and Dowling et al., but throughout a broad range of validation and verification literature (see e.g. Carson et al., 2005; Macal, 2005). Given, this agreed-upon validation process, its structure can now be fitted to meet the validation demands of this thesis and used to guide

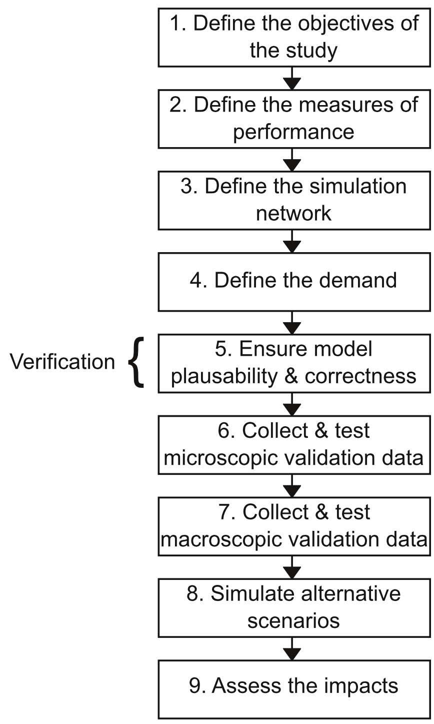


Figure 7.1: Nine-step validation plan for the simulation of traffic networks (adapted from Daamen et al., 2014) and (?).

the reader through some of its steps. Within Table 7.1, a more detailed explanation of these steps and their execution within this research is provided. Besides others, this serves the purpose of differentiating the steps, which have already been treated previously and those which need to be executed within this section.

Table 7.1: Nine-step validation plan for the simulation if traffic networks (adapted from Daamen et al., 2014) and (Dowling et al., 2004)

Task	Execution within this research
1. Define the objectives of the study and the alternative scenarios to be tested	Within Chapter 1 the objective of the simulation is defined to assess the potential emission and congestion savings of all four simulation scenarios that are defined in Chapter 1.
2. Define the measures if performance that will be used to compare the current situation with the alternative	For the sake of comparing the four scenarios, the total CO_2 emissions and the average travel time are employed (Chapter 1).
3. Define the network to simulate by (characterize links and nodes)	Chapter 1, sets the simulation context, by determining that the subject of simulation is a VISSIM network, which is directly adapted from the N260.
4. Define the traffic demand for the simulation network	Chapter 1 elaborates on the The traffic demand was determined through a collection of real-world traffic data, which is then incorporated within the simulation.
5. Verification: Run the simulation and and check whether the model performs as expected	*Subject of this section*
6. Collect & test microscopic data for validation of the simulation model	*Subject of this section*

7. Collect & test macroscopic data for validation of the simulation model	*Subject of this section*
8. Simulate the alternative scenarios	All simulations can be found in Chapter 5.
9. Evaluate the impacts of the alternative technology	The evaluation and interpretation of gained E and C-values is treated in Chapter 6.

As it can be seen, steps five to seven represent the major challenges that need to be made subject of this section. Step five is concerned with proving the general functionality of the system, whereas the following two tasks test whether the designed functional system is a good representation of real-world matters. As it can be seen from Table 7.1, this is concerned with checking whether the base case simulation results are representative. Now, in order to apply this to the research at hand, one has to understand that the VISSIM simulation of Chapter 5 essentially constitutes a model, which consists of a set of subsystems. Validation and verification of the whole system can therefore to some extent be achieved through checking the respective subsystems and their interaction. In his paper "Verification and validation of simulation models", Sargent (Sargent, 2005), suggests that this form of preparatory subsystem testing can strongly contribute to verification and validation quality. The subsequently adopted approach is illustrated in Figure 7.2.

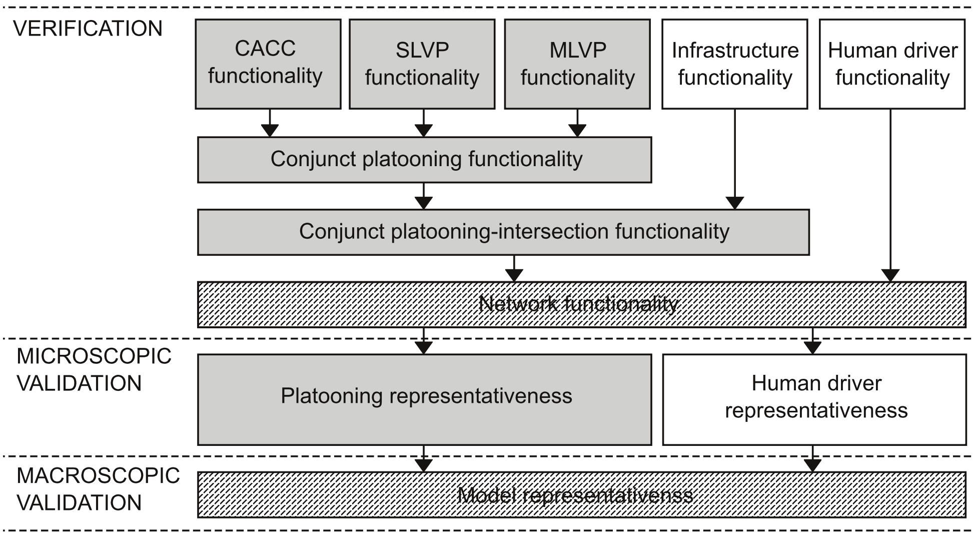


Figure 7.2: Verification and validation in system-subsystem structure (Boxes in gray represent subsystems which require testing, whereas busy stems whose functionality or validity can be inferred are shaded in gray).

As it can be seen, this approach essentially represents a refinement of steps five to seven from the validation plan of Daamen et at. (2014) and Dowling et al (2004). It furthermore illustrates how some of the subsystems do not require any testing. Originating from the fact that certain parts of the simulation represent external, already-proven work, the testing effort can be scoped down to those systems which are new and their interafaces with the existing systems.

Figure 7.2 illustrates how a verification of the three introduced platooning modes and additional testing of their combination allows to prove conjunct platooning functionality. If this and its interface functionality with JUNO is proven, the overall network does not require further verification. Network functionality can be assumed, as all subsystems, as well as their interactions have proven their functionality individually. This is founded in the fact that both the human driving behavior and the infrastructure have already shown functionality in other work (for verification and validation of the VISSIM-internal driving behavior and JUNO see e.g. PTV, 2016; Van Katwijk, 2008).

This sets the scene for the microscopic validation of platooning. In contrast to the verification this is not concerned with the question whether the subsystem shows the desired functionality but with whether that functionality accurately represents the real-world. Similar to the previous step, macroscopic validity can be assumed if platooning representativeness is ensured. Again, this is due to the fact that the initial base case scenario (without platooning) has already been calibrated and validated. This accounts for both, the JUNO and the N260 VISSIM environment(van Katwijk and Gabriel, 2015). Therefore, if a component is added, which has been priorly validated, overall representativeness can be assumed. Consquently, the steps that have to be conducted in this section are those that are colored in gray in Figure 7.2. Besides proving functionality and validity, these steps also serve as a structure for the following subsections. Section 7.1.1 is concerned with testing the three platooning modes and proving their conjunct functionality. Section 7.1.2 illustrates the validity of the introduced VISSIM model and Section 7.1.3 provides some concluding remarks on the applicability of this model.

7.1.1. TECHNICAL VERIFICATION

The low-level verification is solely concerned with the design of a platooning model and its interaction with the intersection controller. This is because the employed intersection controller JUNO is a proven and functioning technology, which does not require verification (cf. gray and white boxes in Figure 7.2). In his work Van Katwijk has already presented comprehensive functionality testing of the system. As no interventions were made in the underlying functioning of JUNO, this testing does not require iteration. The work of Van Katwijk holds continuous relevance for this research. It is therefore not considered here. In general one could argue that the same goes for platooning. In a similar manner, platooning also represents a proven and functioning technology. Yet, one of the main contributions of this research is the deployment of a platooning model within VISSIM. Although, the technology does already exist, its embedment in VISSIM is new and in contrast to the intersection controller part, this component of the research has not been tested before. As the over-arching challenge here is considered with implementing a accurate representation of platooning, it is the functional quality of this model, which will be tested in the following. This Section firstly, verifies the functionality of the introduced low-level platooning controller modes. Based on this, it secondly verifies the platooning VISSIM model, which these controller modes constitute. Lastly, the interaction with JUNO is tested, which eventually proves overall network functionality.

As mentioned in Chapter 1, the verification process distinguishes two steps. The necessity of this becomes clear when considering that the design of a functional combination of the intersection control and platooning is largely based on modeling the latter. In other words, if the platooning controller modes do not exhibit plausible functionality, the subsequent design of Chapter 4 is worthless. In this sense, the individual plausibility checks, which can be found in Sections 3.4.2.1, 3.5.2.1 and 4.3.2.1 have already made a major contribution to prove overall functionality.

Assuming a technical point of view, this dependency can be found to be the functionality of the platoon to design its own optimal intersection approach trajectory (cf. ’Conjunct platooning-intersection functionality’-box in Figure 7.2). Even if the design of this trajectory was perfect, it is upon the vehicle controller(s) to implement the trajectory. The already-presented plausibility tests of the three different controller modes are therefore not only an intermediate step within the controller design, but also the basis for the verification of the platooning model, as they ensure this general functionality of the vehicle controller to implement a certain trajectory.

The individual plausibility check’s purpose was to prove the general functionality of each of the controller modes. That is, the controller modes function well in the specific situation that they are designed for and indeed it was shown that a) The CACC mode allows for conventional platooning and adding vehicles to that platoon from the end, b) The SLVP mode allows for merge-in, merge-out and merge-through motions once a platoon is established as well as adding vehicles to the platoon from the side, c) The MLVP mode allows the LV to design and implement an optimal intersection approach, which is automatically adopted by the LVs that are either in CACC or SLVP mode.

Within these plausibility tests, each of the setups were verified through a test, where the controller was confronted with an abrupt change of its calculation input $e_i(t)$. Especially, the testing of the CACC controller setup was done as such that the controller is able to cope with a worst case scenario (here: a drastic change of $e_i(t)$). Furthermore, the SLVP and MLVP controllers, which are based on this initial design have proven to be able to cope with similar inputs. General functionality of all three platooning modes was proven and the individual modes are therefore considered verified. Given this, what needs to be tested is whether the individual controller modes exhibit a combined functionality. Here, the subject of testing is mainly the EDM logic, which is responsible for choosing the right platooning mode at the right time.

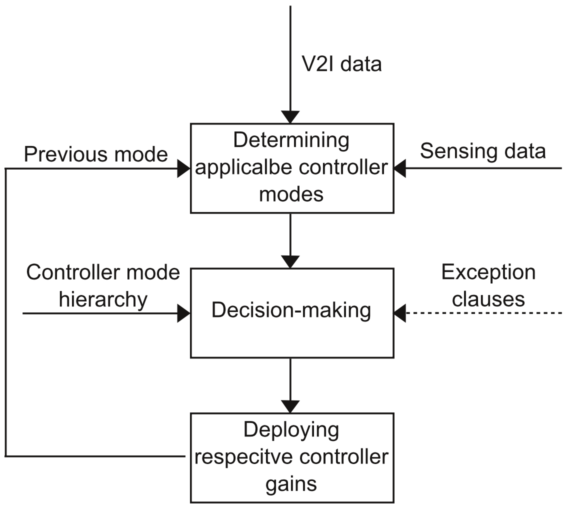


Figure 7.3: Controller mode decision-making process of the EDM logic.

Generally, this choice is made as illustrated in Figure 7.3. First, it is determined which of the three modes are applicable for the current situation. This is done on basis vehicle-internal sensing data and data from other vehicles in the platoon, V2I data and knowledge about the mode selected in the previous time step. At times, situations can occur where more than one of the modes are technically applicable. In this case, a decision is made based on the controller mode hierarchy. For LVs e.g., this prescribes that MLVP platooning is to be chosen over all other options. SLVP platooning is the secondary choice if MLVP is not applicable in that situation and only if none of the others are applicable, conventional platooning is chosen. In contrast to that FVs chose SLVP over conventional platooning and MLVP.

Now in order to test the functionality of this decision making process, a two-stepped process is adapted. Firstly, as proposed by Sargent (2005) animation is chosen as the first verification measure. The vehicles' driving behavior is graphically displayed both within the N260 VISSIM environment and a set of VISSIM testbeds. As the model moves through time, it can be surveyed if the vehicles switch between the modes correctly. Secondly, a set of simulations is executed, whereas an entry is written to a log file, whenever a vehicle exhibits undesired behavior. The difference to the first verification test, is that this method can survey longer simulations (a 60 minutes period is chosen) and do so more thoroughly. The difficulty with this however, is to determine which behavior is considered undesirable. This is why it is argued that both steps are needed for a quality verification. Within this research, the following happenings are logged and tracked within the latter verification step:

Table 7.2: Simulation occurrences that lead to a verification log entry for the platooning controller.

Occurrence	Explanation
$e_i(t) < -2$	*Undesired*: If the error function $e_i(t)$ is smaller than zero, this means that the vehicle at hand is too close to its precursor. Depending on the desired safety distance (which depends on the speed) this can represent a safety critical distance
$d_i(t) \leq 0$	*Severe*: If the distance between two vehicles is zero, this essentially means that the vehicles have crashed.

t_{ed} and $a(t) > 0$	*Undesired*: In the contest of this research there can be no case where the receipt of an estimated time of departure leads to an acceleration motion. A log entry is made if this is the case.
$v_i(t) < 0$	*Undesired*: Vehicles can at times move backward as an effect of a controller action. This can e.g. be if a platoon comes to a full stop and an FV reacts to an overshoot by applying a negative force with a certain time delay. This however, is undesired here as no danger occurs when vehicles are standing. Hence, the derivative safety gap is irrelevant.
$v_i(t) = 0$ and $e_{i=1}(t) \gg 0$	*Severe*: Although the error derivative is not used for the platoon leader, it can be used to check whether a vehicle comes to halt without a necessity for this.

Essentially, each of these occurrences implies that a wrong decision was made within the EDM logic. As the different controller modes have all proven individual functionality, this means that the occurrence was triggered through a situation where a wrong mode was selected. Within the EDM logic this is conquered through a set of exception clauses. These are simple if-then conditions, which overwrite the prior decision. To only name one example it can happen that an FV plans to leave the platoon at the coming intersection, hence take a different intersection exit. Say, the platoon intends to go straight, while the vehicle wants to go right. In many cases intersections have an additional lane and signal for vehicles that go right, which implies that the vehicle at hand might have a different estimated time of departure than the platoon. In this case, an exception clause is triggered, which facilitates first the merge-out (SLVP) motion of the vehicle and then brings it into MLVP mode, as soon as it has reached the target lane.

The final verification iteration has shown no severe and only few undesired occurrences. Considering that the high-level purpose of this technical design is to assess the impacts and not the performance of platooning, this is considered sufficient for the research at hand. However, refining the platooning model to enable flawless functionality is one of the recommendations, that is proposed as follow-up research in the concluding chapter. Yet, the final EDM DLL with all exception clauses does provide the desired functionality that is needed to effectively assess the impacts of platooning. That is, the generic functionality of the platooning controller can be considered proven and verified.

This however only lays the basis for verifying the interaction with JUNO. Once again, a mixture between visual checks and log file verification is employed. This second check uses the trigger functions as introduced in Table 7.3.

Table 7.3: Simulation occurrences that lead to a verification log entry for the platooning-intersection controller.

Occurrence	Explanation
$d_{int,i}(t) \approx 0$ and $S_{int} = red$	*Severe*: If the distance to a stop line is zero, this means that the vehicle is entering the intersection. If additionally the traffic signal is in a red state, this means that the vehicle is crossing a red light.
$t(d_{int,i=1} = 0) \not\approx t_{ed}$	*Undesired*: If the arrival time of the simulation is not approximately the estimated departure time, this implies a malfunction in either the communication between the systems or the processing of the t_{ed} information in the EDM logic.

$a(d_{int,i=1} \approx 5) \not\approx 0$	*Undesired*: If the vehicle features a (negative) acceleration upon arrival at the intersection, this again implies one of the above mentioned problems.

After iterative improvements, no severe and few undesired occurrences were detected, which allows to consider the final design of a platooning-intersection controller verified. In this section it was explained that the low-level design of a set of vehicle controller modes and their combination enables the general functionality of platooning, which suffices for the aimed-at impact assessment of this research. Additionally, considering that JUNO represents an already established system with proven functionality and the fact that the interaction between the two systems functions as desired, this lays the basis for a validation effort.

7.1.2. TECHNICAL VALIDATION

While the previous paragraphs were concerned with whether the technical design of Chapters 3 and 4 allows for platooning functionality, it is now time to question whether the subsequent VISSIM model is actually representative for real-world traffic. For this it needs to be kept in mind that the base scenario of the N260 VISSIM simulation was already verified. That is, without platooning and advanced intersection control, the simulation has delivered results, which closely resemble those that were collected through a traffic survey on the road at hand. The question that needs to be answered can therefore be refined to whether the implemented platooning model is representative for real-world platooning. As illustrated in Figure 7.2, if platooning representativeness is provided, then overall model representativeness can be assumed.

Validating the platooning model is concerned with testing whether the platooning characteristics that were conceptually introduced in Table 2.3 are brought into place within the model with sufficient accuracy. Clearly, some of these characteristics are binary. They are either implemented or not. For others however, a differentiation is possible. Yet, another log file simulation is employed in order to verify these characteristics. The trigger functions are implemented as such that it surveys the performance of each vehicle on the characteristic at hand.

As it can be seen in Table 7.4, there are a few occurrences of platoon vehicles going over 90 km/h. This happens when an individual vehicle or a platoon leader, which is already going just below 90km/h tries to catch up with a platoon in front. Additionally, although an average vehicle clearance of 10 m at top speed is reached, this features high deviations at times. Minimum safety gap values of 3.5m and maximum values of 17 m are observed during the verification cycle.

Table 7.4: Verification of platooning model.

Characteristic	Definition	Implementation
	Platooning characteristics	
Maximum number of vehicles	5	Yes
Platoon composition	Combined	Yes, with restriction to vehicles with similar acceleration capabilities
Application environment	Urban & highway	Only urban
	Vehicle dynamics	
Maximum speed	90 km/h	Few, insignificant occurrences of speeds + 90 km/h
Vehicle clearance	~ 10 m at 90 km/h	$d_i(v) = r_i = 2.5m + 0.3sv \Rightarrow d_i(v = 25m/s) = 10m \pm 6m$
Lane changing	Yes	Yes
	Vehicle communciation	

Planned trajectory	Yes	Yes, unilateral
Sensor data	Yes	Yes
Longitudinal motion	Yes	Yes
Lateral motion	Yes	Yes
	Vehicle control	
Leading vehicle (LV)	Autonomous	Partly
Longitudinal control	Yes, through cooperative adaptive cruise control (CACC)	Yes
Lateral control	Yes, through lane keeping assistant (LKA) system	Yes

Furthermore, for the characteristics: platoon composition, application environment, communication of a planned trajectory and vehicle control of the LV, restrictions exist. The reasoning behind the design decisions, which lead to these restrictions were introduced and explained in the previous chapters. In this context, it has also been explained that the respective measures do not have an effect on the evaluation of emission or congestion values. Similar goes for the deviations of non-binary characteristics, which is why both are considered acceptable. The designed platooning model, its functionality and its simulation characteristics all fulfill their respective requirements. The platooning model as such can therefore be considered validated.

7.1.3. APPLICABILITY OF THE TECHNICAL DESIGN AND THE SIMULATION IMPACTS

Finally, some remarks need to be made about the applicability of the validated traffic model. Simulation is a vast standard in the field of traffic management and especially for impact evaluations of technology advancements it has proven to be a cheap but good method. However, this choice of method also restricts the transferability of its findings. That is, although the model has proven validity, this validity only accounts for a certain purpose (for a comprehensive lists of feasible use cases of VISSIM simulations see PTV, 2016, p. 25 - 27). Applying the model outside of this scope is not or only to a limited extent possible.

Firstly, as it was shown the platoon and its intersection with an intersection controller do exhibit certain undesired behavior. Although, this is not relevant for assessing the impacts on emissions and congestion, it might be for other factors. An example here could be a safety assessment of the system. Clearly, to allow for this the developed model needs to be refined in order to represent platooning motions more accurately. While small inaccuracies in vehicle movements are not relevant for evaluating emissions or congestion, they might very well be for the assessment of other impacts.

Secondly, the VISSIM simulation software does make some physical simplifications. While the software clearly is one of the most sophisticated traffic simulators, it self-evidently still does not perfectly represent real-world matters. As a consequence of this, the developed model is further limited in its applicability. Although, the level of simulation accuracy does suffice as an input for policy considerations, it might not do so for other purposes. A real-world application of the developed platooning controller e.g. would require recalibration and further refinement.

Thirdly, as it was mentioned earlier, the designed traffic model assumes full and perfect information. Although the data transmission is made subject of a certain modeled delay, the data which both systems use and communicate can not be faulty within the scope of this research. Consequently, the designed model can not be used for assessing the robustness of the system. Additional studies on the real-world falseness of data would be necessary to model this and subsequently find a way to cope with the wrong information inside either the vehicle or the intersection control system.

Lastly, the introduced traffic model was fitted and calibrated to the N260 environment only. Although similar effects can be expected for similar settings, each simulation setting would require recalibration. In order to accurately assess average, universally-applicable impacts further testing in other settings is required. This could potentially be a follow-up project of this research.

Having understood, that the technical design fulfills the implicit and explicit requirements of the technical objective, this allows for a subsequent high-level consideration. As the name says, this is concerned with checking the research, which builds upon the pervasively verified and validated low-level technical design.

7.2. HIGH-LEVEL VERIFICATION AND APPLICABILITY OF POLICY CONSIDERATIONS

The following testing of the high-level considerations are solely concerned with verifying the correctness found mathematical condition of formula (6.8). Further verification could be done by means of an actual hypothetical policy design. Yet, this option is not taken for reasons of scope.

As it was mentioned in the introductory chapter, the governmental entity that was featured in the establishment of a policy is an umbrella term for a set of political bodies. From the fact that these bodies are in fact heterogenous, it can be derived that each entity has to decide for itself whether the made considerations are applicable for them or not. This is concerned with two main questions. Firstly, the party must decide whether they consider the traffic participants the most important player for the deployment of such a policy. Secondly, they need to agree with the available choice of moves. It is upon the governmental bodies themselves to decide whether the presented high-level policy holds validity under the given assumptions of players and choices. In this sense, the presented high-level considerations should be seen as an initial approach, which can modified as needed. The calculated emission and congestion impacts hold generic relevance, which is why the presented game can be expanded by other players or other moves if needed.

What needs to be done however, is to test the functionality of the policy consideration, which serves as a basis for a potential compensation measure. For this, once again, a causality as in the low-level section can be applied. If all subsystems and their interactions are validated, no validation is necessary for the game theoretic system. Consequently, the sole subject of this section is the verification, hence proving the functionality of a compensation measure in general. This falls into the domain of formal verification (for a comprehensive explanation on formal verification see Bjesse, 2005), which features game theory as one of its main verification methods. For this it first needs to be understood that a VISSIM simulation in general satisfies the input requirements to fill the pay-of matrix of the introduced game. One of the VISSIM use cases that is mentioned by the developer of the software is the "impact examination of various technology advancements [...] on traffic flows" (PTV, 2016, p. 26). This is exactly what this research is aimed at, making the selected method generally feasible for the game theoretic purpose of this thesis. The pay-off inputs from the previous chapter can be considered verified and validated, hence feasible for policy making.

Consequently, what needs to be shown is that the considered players do indeed favor the P, I-scenario, given that a policy is found, which satisfies equation (6.7). In order to generically prove this, a worst case scenario must be assumed. If workability os proven for this case, it can also be assumed for all other cases. The parameters are chosen as specified in Table 7.5.

Table 7.5: Policy-making parameters in a worst case scenario.

Parameter choice	Explanation
$w_{x,C} = 0$ and $w_{x,E} = 1$	In a worst case scenario at least one of the governmental weight factors is zero. The other can not be there, as this would make a policy obsolete in the first place. No policy is necessary if no improvement is targeted.
$w_{y,C} = 0$ and $w_{y,E} = 0$	In a worst case scenario, the traffic participants perceive no benefit from saving on congestion or emission. According to formula (6.6), this leads to the highest possible need for compensation.
$k_{eff} = 1$	If the effectiveness coefficient is 1, this means that the government has no other policy option than directly and financially compensating the traffic participants for their investments.

$E_{P,I} = 1$, $E_{\bar{P},I} = 0$, $E_{P,\bar{I}} = 0$, $E_{\bar{P},\bar{I}} = 0$	Lastly, in a worst case scenario, savings can only be achieved under the highest possible technological level. The congestion savings are not relevant, as both players value them with 0.

Now, if equation (6.7) is satisfied, hence the policy is generally worthwhile, this implies that:

$$1 > V_y + Z_y. \tag{7.1}$$

Furthermore, taking (6.6), inserting k_{eff} and inserting the result in (7.1) leads to:

$$1 > V_y + V_x. \tag{7.2}$$

According to (6.8), the policy would be approved if the above was the case. If that is true, then the potential pay-offs for each scenario can be calculated by using (6.1):

Table 7.6: Potential moves of the government and the traffic participants in the Platooning-Intersection control game, assuming a worst case scenario.

		Government Y	
		Do invest I	Do not invest $\bar{I}$
Traffic participants X	Do partake P	$0, 1 - V_y - V_x$	$-V_x, 0$
	Do not partake $\bar{P}$	$0, -V_y$	$0, 0$

Finally, considering that equation (7.2) is true and that each of the investments can not be smaller than 0 it is derived that each of the investments must be smaller than 1. Given this, the established game from Table7.6 exhibits the following characteristic: P is the best response to I, as the traffic participants will not be better off, if they switch strategies. Furthermore, I is the best response to P, as the government would be worse off if it decided to switch strategies. From this it can be concluded that the P, I-scenario represents a Nash-equilibrium. Consequently, it is proven than even for a worst case scenario, a policy, which satisfied (6.8) leads to the desired game. Hence, the functionality of this policy tool is hereby verified. Do to the fact that a worst case scenario was assumed, this also holds relevance for all other scenarios, which feature less governmental policy expenditures.

8
Conclusion

In the introductory chapter of this report, it was explained that the V-model serves as an underlying paradigm for the structure of this research. Following this approach, the high-level objective of finding a policy, which promotes congestion and emission savings through smart traffic solutions without considering road expansions was broken down into four research questions, one for each of the levels of the V-model. The solutions to these questions feature a bottom-up causality, whereas e.g. the solution to the second-level question provides valuable knowledge for the solution of the first-level question. Subsequently, not only the start but also a major part of the overall research effort are allocated on the lowest level. That is, the lowest level attempts to develop one potential smart traffic solution. Namely, this is a functional combination of platooning and intersection control. The reasoning behind the choice for this solution is as follows: Platoons have shown to be an efficient mean of saving on congestion and emissions. Similar goes for innovative intersection control strategies. Consequently, promising benefits can be expected from a combination of both. Given this, the thesis then proceeds to develop a technical design, which allows for such a functional combination, which eventually serves as input for the higher-level research considerations. Figure 8.1, illustrates this relation and specifies what the individual lower-level solutions contribute to their respective higher levels.

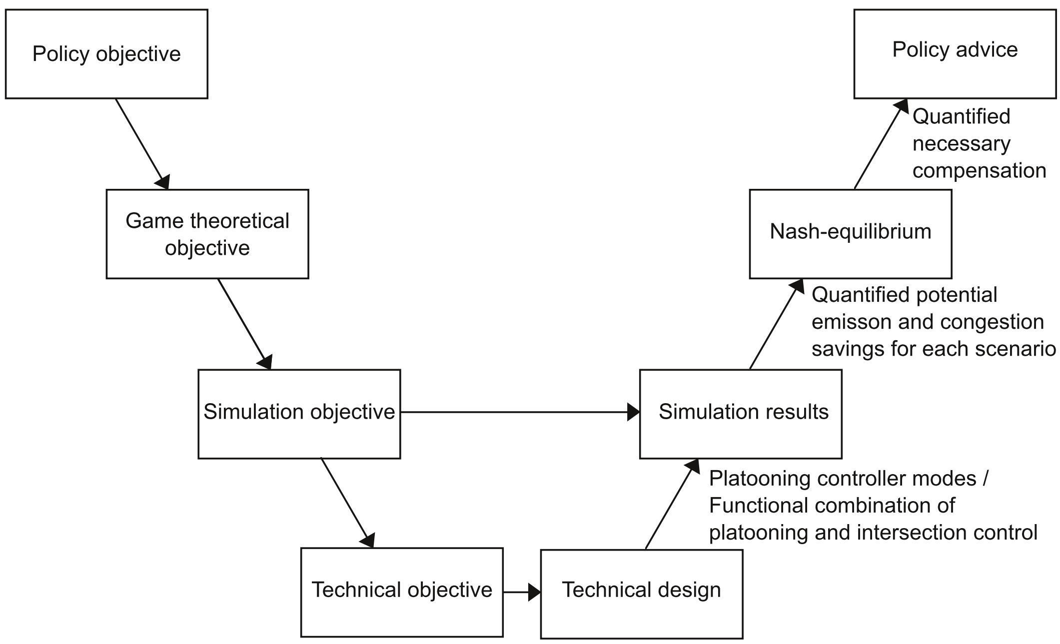


Figure 8.1: High-level illustration of the research approach as a V-model with an illustration of how each lower-level solution contributes to its respective higher-level solution.

In the following, the reader is provided with a brief summary of the different-level design steps. In a first attempt, this is done by means of a chapter summary in Section 8.1. Preceding the research process summary, a summary of the conclusions and main findings is provided in Section 8.2. Besides others, this comprises the answers to all four research questions. Section 8.4 provides some remarks on the societal and scientific relevance of these findings. Finally, Section 8.5 presents a set of recommendations, concerned with both - advice on how to employ the findings and contributions of this work and potential future research.

8.1. CHAPTER SUMMARY

As mentioned in the introduction of this chapter, the start of the research effort can be allocated in the technical design of a smart traffic solution. Chapter-wise however, this is preceded by the introduction and the technical analysis. The introduction chapter (Chapter 1) is mainly concerned with a comprehensive problem statement and a subsequent research framework. Both were touched upon in the introduction of this chapter already and additionally are not part of the actual research effort. This chapter is therefore not further elaborated on.

In Chapter 2, the reader is prepared for the technical design by means of a thorough literature review on the two concomitant systems. The chapter however, does not only pursue the goal of providing a broader understanding of the two systems, but it also illustrates the wide range of applicability of the aimed-at technical design. It is shown that a variety of intersection controllers and platooning concepts can generally be combined. Of those applicable intersection controllers, JUNO is selected for the actual technical design. The chapter concludes with a motivation for this choice and a definition of an average platooning system. Consequently, JUNO and this average platooning system represent the basis for the actual technical design.

That is, Chapter 3, guides the reader through the low-level implementation of platooning in the external driver model of VISSIM. It does so by explaining the different steps of a controller design process. The essential functionality of the final vehicle controller design is to take certain sensor data and process it as such, that the vehicle takes the necessary actions to align its motion according to another vehicle. Through the decentralized integration of this controller in a set of cars, the vehicles can then conjunctly drive as a platoon. For the platooning controller, two controller modes (CACC and SLVP) exist, which together depict a good representation of the average platooning system, which was defined in the previous chapter. If simulated, the developed controller makes the individual vehicles behave, as it was prescribed in Chapter 2.

Chapter 4 builds upon the controller, which was developed previously. It supplements the controller functionality with yet another platooning mode (MLVP). This last platooning mode incorporates V2I communication between the platoon leader and an intersection controller. Functionally, it is based on SLVP platooning, however it adds the possibility of adopting a congestion- and emission-optimal intersection approach trajectory. The V2I communication between the two systems is of hierarchical nature. That is, whenever JUNO identifies optimization potential it communicates this with the respective platoon leader. The leader then decides to abide by the proposal or not. If it does so, it internally calculates its optimal trajectory and deploys it by switching to MLVP mode. In converse manner however, the platoon has no possibility of making a trajectory request to the intersection controller. The end of this chapter represents the end of the low-level technical design process.

Chapter 5 is able to take the developed technical design as input for the simulation level. More precisely, this is done by means of four simulations, which quantify the congestion and emission values for the four scenarios (No platooning, no JUNO); (No platooning, JUNO); (Platooning, No JUNO); (Platooning, JUNO). The simulation is executed in VISSIM on the N260 case study and the eventual quantifications are reached through re-processing the simulated vehicle motions in EnViver. It can generally be seen that both technologies lead to certain improvements. On an imaginary Pareto frontier (i.e. all scenarios on the frontier are desirable scenarios, see e.g. Roger, 1991), the urban platooning scenario shares its place as the front-runner technology with the JUNO-only scenario. However, as argued in Section **??**, this does not hold truth for all kinds of physical scope. Founded reasoning exists, to assume that on a wider simulation ground only the urban platooning scenario is determined most desirable.

Chapter 6 utilizes the simulated emission and congestion values by inserting them into a game between the government and the traffic participants, whereas each of the outcomes represents one of the simulation scenarios from the previous chapter. Being more precisely, the simulation values are used in order to calculate the pay-offs for each of the players and each of the scenarios. From the initial calculations it can be seen that no Pareto optimum can be found in the urban platooning scenario. This infers that a policy is necessary to enforce the (Platooning, JUNO)-situation. Although no advice is given about the exact form of the policy, the found knowledge can be very helpful for the development of one. In the latter part of this section, a policy-development framework is introduced, which makes use of the simulation values and proposes a work plan towards a successful policy, promoting urban platooning. The framework distinguishes between a theoretical testing phase and a practical validation phase. Their combination allow for a quicker and more thorough policy analysis and hence for a better policy design.

Eventually, Chapter 7 verifies and validates the findings of the previous chapters. This is mainly done through animation and log file computing. It is found that the technical design, as well as the higher-level considerations generally suffice for the purpose of this research. However, especially in the simulation a few underside occurrences were observed. Although, they are considered acceptable within the context of this thesis, the findings of this Chapter could potentially serve as a basis for follow-up research.

8.2. CONCLUSION SUMMARY

By now, the reader should be well-aware of the taken research process. In the following, the main findings of this are summarized. This is done by revisiting the four research questions, which originate from the V-model of Chapter 1. Table 8.3 lists these questions and the solutions, which were found in the course of this research for each of them. A brief summary of the research and design decisions that led to their solution, as well as the actual research answer are provided.

Table 8.1: Research questions and answers.

Research question	Research solution
Level 4: How can a platoon be navigated through a string of intersections through a combination with intersection control while allowing for a performance optimization of the traffic network in terms of congestion and emissions?	The answer to this question can be found mainly within Chapters 3, 4 and partly in Chapter 6. The former two present the development of three platooning modes, while the latter explains the underlying logic, which determines the mode of choice for each situation. One of the modes, MLVP, incorporates platoon-intersection communication, which subsequently allows the platoon to adapt an emission- and congestion-optimal trajectory, leading the platoon through one or several intersections. Hence, a platoon can be navigated through a string of intersections by adopting the MLVP platooning mode whenever a junction is approached. The platooning design in conjunction with the V2I communication constitute a solution to the given technical research question.
Level 3: Which congestion and emission savings can be reached within each of the four scenarios of the N260-case study?	Employing the technical design from level 4 and incorporating it in the N260 VISSIM simulation by means of an external driver model DLL, allows to simulate the traffic flow of all four scenarios. This and the process of taking the traffic data and processing it in the EnViver software, is elaborated on in Section 5. The section concludes with what is the answer to the simulation research question, namely the emission and congestion savings for each scenario. The potential benefit for each of the technology setups is calculated and summarized in Table 5.6.
Level 2: Can a policy be designed, which triggers a Nash-equilibrium in the upper-left corner of the Platooning- Intersection control game?	For this question a rather generic mathematical approach is chosen. By doing so, it is possible to gain insights that are applicable to a range of situations and scenarios. Section 6.1 describes under which mathematical conditions "Do partake" is the best response strategy to "Do invest". This solves the research objective as such that it shows for which cases a Nash-equilibrium is generally possible, given that the government is willing to do the necessary investments. The mathematical equation, which can be used for testing this is (6.2). (6.3) can be used to calculate the resistance threshold, which a potential policy has to overcome.

Level 1: Under which conditions can a policy be designed that incentivizes urban platooning, which itself pursues the goal of saving on congestion and emissions without road expansion and how can such a policy be implemented?

On a theoretical basis, the first part of this question can be answered through testing when a policy is justified in terms of costs and benefits. As this largely depends on the importance which a government assigns to emission and congestion goals, a mathematical answer (6.8) is provided. This takes the form of a numeric condition, which has to be fulfilled for the policy to be worthwhile. A governmental body can use this to see whether a potential policy will be successful and worthwhile. This mathematical condition can constitute a valuable tool in policy making.

The second part of the question is answered by means of a policy-development framework. This framework attempts to countervail the theoretical nature of the answer to the first part of this research question. Given this, one major purpose of the policy-development framework is to revisit the made modeling assumptions, the accepted simplifications and the limited data value that is drawn from this. Hence, although being based on game theoretic notions, the policy-development framework attempts to embed these insights in a more far-sighted context, effectively compensating for the (necessarily) made assumptions. To do so, two policy-development phases are suggested - a theoretical policy verification & validation and a practical policy implementation phase. The former revisits the above-mentioned modeling assumptions and proposes a set of plausibility checks to test whether or not these hold truth for a given scenario. If this is the case, hence the policy is theoretically considered fit, the framework proceeds to the second phase, which is constituted through a policy-deployment roadmap. This roadmap gradually works its way from small field tests up towards a large-scale policy implementation. The benefit of this framework is that a policy can be tested on theoretical basis before any real-world experiments are initiated. It is believed that despite the made modeling assumptions, this allows to acquire a solid evaluation of a broad range of policies without facing major expenses. The government can hence maintain a broader design space when seeking policy solutions that promote urban platooning, effectively increasing the chances for such a policy.

8.3. TECHNICAL DESIGN, RESULT AND LIMITATIONS SUMMARY

At several occasions within this report it was stressed that the low-level technical design is the overall main contribution of this research, mainly because a major amount of time was spent on this level of the 4-layered problem formulation. Thus, this section is dedicated to the reached results of this low-level design, which are mainly but not solely constituted by the congestion and emission savings C and E from Table 5.6. Firstly however, it once again needs to be stressed that these results solely depict the outcome of an although realistic, yet theoretic traffic simulation. As throughly explained in Section 6.2, these results and their informative value have to be handled with care. Section 6.2.2 provides a framework that can assist in coping with the drawbacks of simulation optimization. In the following it will be discussed how these results come about and to what extent they fulfill the expectations.

In a first approach, it is analyzed how the results compare to their expectations. For this, the low-level design requirements and objectives from Table 1.4 are revisited. The first entry in that list is the aspect of safety. It is stated that the design must ensure the safe navigation of a platoon through one or several intersections. The question whether this requirement is fulfilled should be answered with a conditional yes. In Chapter 7 it was shown that no *severe* occurrences happened during a 1-hour simulation period. This comprises, that no cars ran over red lights and no crashes between platooning vehicles occurred. Nevertheless, a few *undesired* occurrences remained even after the last improvement iteration. Vehicles were at times driving too close/far to/from each other or were running yellow lights. Although, this has not led to severe incidents during the simulation period, it does provide reason to doubt if it will be the same for e.g. a different physical setting or a longer simulation period. The *undesired* occurrences are termed like this for a reason, namely that they can potentially lead to *severe* incidents. Hence, although no sufficient testing was possible within the scope of this thesis, the aspect of safety should be further investigated.

The second design requirement demanded that the technical solution must maintain or decrease the overall congestion and emission values C and E. From Table 5.6 it can be seen that this is indeed the case. This is directly linked to the second objective which aims at maximizing the congestion and emission savings. In order to analyze whether these results can be interpolated or transferred to other physical settings, simulation types, car type etc. some further investigations were initiated. In this context, one major finding is that the platooning period of a vehicle can be split into two phases - the platoon composition and the platoon-following period, as illustrated in Figure 8.2. Their technical difference is clear: One describes the time until the vehicle has joined or formed a platoon and the other is associated with the remaining time until the platoon is decomposed. The differentiation between these two is noteworthy in terms of their effects on emissions. Namely, it can be seen that the former is associated with emissions that are higher than those of conventional driving, whereas the latter's emissions are significantly lower. The technical reason behind this is that the formation of a platoon involvers rather stringent acceleration motions (cf. the velocity peak in the platoon formation phase), hence the increased emissions. Logically, after the platoon is formed its benefits can be harvested, thus the lower emissions of this phase. Although the context of this research did not suffice to quantify these differences, sufficient proof was found to at least proof their existence and discuss the implications thereof.

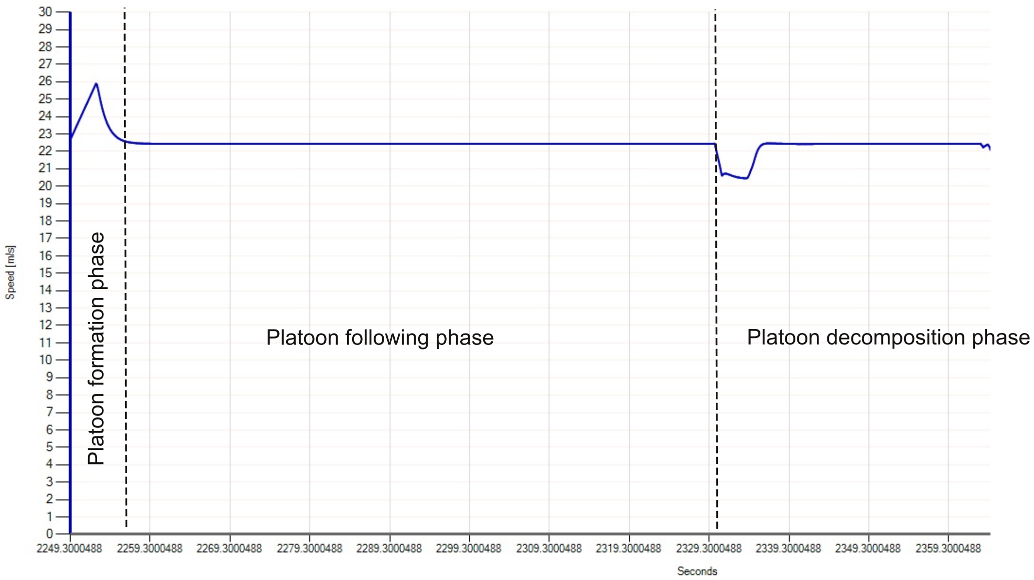


Figure 8.2: Examplatory EnViver speed profile of a vehicle, illustrating its platoon formation, the actual platooning phase and the platoon decomposition.

Keeping in mind that the ration of the two phases is different depending on the route of every car, it can be deducted that there must be a point where the drawbacks of the formation phase outweigh the benefits of the platooning phase. That is, if the platoon following phase is small enough to produce less emission savings than the formation of a platoon costs. Given that the overall emission savings of the presented scenario were positive, it can be assumed that this ratio is rather small, however no quantifications can be attempted in the context of this work.

The existence of this point however, should not be seen as a negative result of the technical design. Given the same reasoning as above, it can also de deducted that the emission-wise benefits increase, the longer a platoon is maintained. In this sense, it has been shown that even for a relative short platooning distance of max. 6500m (usually significantly less), slight emission savings can be reached. Reasoned hope exists that these results can be increased for longer platooning periods. When additionally considering the motivation that was introduced in Chapter 1 for the execution of this work, that urban platooning can help to extent the applicability range of highway platooning, it becomes clear that the presented results represent a success, rather than a failure. Especially for the latter, the presented technical design is of high value. it not only allows to gain further highway platooning benefits, but additionally urban benefits can be harvested when an already-composed platoon enters the urban network from a highway. The platoon formation phase is omitted, implying that the overall emission savings would be reasonably higher than those of Table 5.6.

Another requirement that is listed is that no or only little distributive effects should be allowed. This is concerned with the fact that no group of traffic participants should benefit more from the new technologies more than any another. The success of the technical design with this requirement can be derived from the congestion and emission savings from Table 5.5. By calculating the percent-wise emission and congestion savings for the different vehicle types, through the new technology a good measurable for the benefit level of each group can be reached. The results are presented in Table 8.2.

Table 8.2: Percent-wise impact evaluation, concerning the average velocity and the average emissions per km for the N260 case study for the three different vehicle types.

Scenario 4: *P*, *I*	
Performance indicator	Value
Average vehicle velocity savings in %	1.430 %
LDV vehicle velocity savings in %	1.329 %
HDV - heavy vehicle velocity savings in %	2.038 %
HDV - medium vehicle velocity savings in %	5.650 %
Average CO_2 emission savings per km in %	3.618 %
LDV CO_2 emission savings per km in %	3.873 %
HDV - heavy CO_2 emission savings per km in %	2.741 %
HDV - medium CO_2 emission savings per km in %	6.285 %

As it can be seen, no vehicle type is made worse off through urban platooning in terms of emissions or average velocity. Hence, the minimum requirement for distributive effects is reached. Yet, it can also be seen that the vehicle class "HDV - medium" benefits more from the new technologies than the other classes. This effect is undesired and to a certain extent it represents a short-coming of the technical design.

Furthermore, two additional objectives were formulated, targeting the minimization of necessary costs and the minimization of intervention in the systems, respectively. Although being of equal importance, no quantified implications can be made from the simulation results for those aspects. However, during the whole design process the latter was kept in mind and taken care of. The presented solution is to large extents software-based, which enables high compatibility and easy adaptability, without requiring significant changes in the concomitant systems. It is argued that this mutually leads to lower costs, as an introduction of urban platooning does hereby require less physical intervention. Hence, although no measurables exist to quantify the degree of system intervention, it is found that these requirements were sufficiently considered and hence satisfied.

Lastly, to put the results in context it should further be analyzed which effects drive the simulation values and what characteristics can be derived from them. For the emissions, his is concerned with the way the CO_2 calculations are conducted through the EnViver software. Namely, as mentioned in Chapter 5, vehicle state vectors are translated into speed-acceleration profiles and subsequently emission results. A major benefit of platooning, which is neglected in this method are the emission savings that are associated with the deceased drag resistance of platooning vehicles. in Table 1.2, these are quantified somewhere between 2% and 21% for highway application. No matter the eventual value on urban roads, significant further savings can be expected when incorporating this aspect.

For the average speed calculations, this is concerned with relating the average speeds before and after an implementation of urban platooning to the maximum allowed speed on the N260 track. Namely, the maximum allowed speed is 80 km/h, with the distributor roads being limited to either 50 or 30 km/h. Given this, it is nothing but a logical consequence that the optimization of average speed will at a certain point reach its theoretical maximum. Although no numerical value can be provided for this maximum it can be deducted that it must be somewhere under 80 and above 72 km/h. Keeping this in mind, one will understand that the initial average speed of 70.7 km/h already represents a rather high value, consequently leaving limited room for optimization. Hence, the improvement by 1 km/h can be seen as a solid step towards this theoretical optimum. Realistically seen, one can never reach this optimum. It can hence be assumed that the reached results represent a good overall improvement, given the physical possibilities.

Yet, also a list of limitations can be found. While the overall simulation results were satisfying, these limitations are rather concerned with the representatives of these results. As Section 6.2.2 has already elaborated on these, only a brief summary of the main limitations is given in the following. The most important limitation is probably that of the limited applicability of the found results. The results represent a very specific simulation outcome for a specific physical and technological setup. Major concerns were expressed in the formulation of a policy-development framework about whether and how these outcomes can be interpolated to other settings. Furthermore, the technical design itself exhibits certain shortcomings. Sections 7.1.1 and 7.1.2 were dedicated to find and investigate these. The major drawbacks that were found are that the simulated vehicle behavior does at times differ from those of real platoons. Although, these differences are not important for the evaluation of congestion and emission savings, they might very well be for the investigation of other performance indicators such as safety or driving comfort. Hence, another major limitation is that the model as-is is limited to the evaluation of congestion and emissions parameters.

Yet, altogether a good and satisfying set of results was reached. It was shown that emission and congestion benefits are possible with minimum modification in either of the concomitant systems. Furthermore, the minimum of effectively expanding the range of highway platooning was reached. Hence, the benefits of platooning can be harvested across highway and urban roads without interruption. Furthermore, these benefits are conceivable for all traffic participants, although certain groups appear to benefit more than others. However, despite the satisfying outcome of this research it should be kept in mind that the study does not attempt to present a working system. It solely constitutes a study of the potentially perceivable benefits and hints at one out of many possible technical solution. Until, a real-world working system can be established a lot of work and further research needs to be done. In the following, the relevance of these findings for science and society are summarized.

8.4. SOCIETAL AND SCIENTIFIC RELEVANCE

At this point the reader should have understood that the research at hand distinguishes between high- and low-level contributions. This structure has not only been proven helpful for structuring the research process, but the same terminology can also be applied on the societal and scientific relevance. That is, the low-level technical design is that contribution aspect, which is scientifically relevant and the high-level findings exhibit societal relevance.

The author hopes that the made policy considerations are of value for an integration of the two technologies in everyday traffic situations. It is believed that the gained knowledge can be helpful for a structured policy design, which effectively supports the development of platooning and intersection control and eventually a quick deployment. Furthermore, it is hoped that the research process can serve as a guideline for other, similar work, which is affiliated with alternative smart traffic solutions. Yet, the system which is proposed in this report is of rather disruptive nature. It is for this reason, that the made considerations do only constitute one out of many steps that need to be taken in order to eventually formulate such a policy. Further work and research effort is necessary, for which this thesis can serve as a basis. The research at hand is one out of many puzzle pieces, which are necessary in order to push towards a more eco-friendly and smart urban traffic system. Only if all these pieces can be obtained and put together, the societal impacts of this work can unfold.

From a scientific aspect it is clear that the presented low-level design is of high interest in order to gain technical knowledge and understand possibilities and impossibilities in the field of platooning in urban environments. Yet again, this work needs to be put in context. Due to the disruptive nature of the research, it should solely be regarded as one possible solution out of many. For further research it can be used either as a performance benchmark (for alternative solutions towards the presented problem) or as a basis for a refined technical design. In sense of the latter, it is hoped for that this should ideally not remain a theoretical work but lead to a design that is actually applicable for the real world. Besides others, this involves refinement and improvement of both, the traffic model and the controller design, which were presented. Therefore, this represents one of the main recommendations, which is given in the next section.

8.5. RECOMMENDATIONS

As especially illustrated in the validation of Section 7.1, the traffic model, which is developed in this research is solely designed for the purpose of evaluating the impacts of platooning and intersection control on congestion and emissions. This should be kept in mind by those, utilizing the model. That is, for utilization of the VISSIM simulation, one has to be well-aware of its short comings in order to understand when the model can be applied and when it might lead to false results. Here, it should be noted that especially the platooning controller is not ready for implementation or field tests. Within the log file verification, a residual set of undesired occurrences was observed and the validation has shown that the platoon does not exactly exhibit the behavior of real-world platoons. Although, this is irrelevant for the evaluation of congestion and emissions, it might very well be so for other purposes. It is recommended that users of the provided model first read Chapter 7 before applying the tool to use cases, other than what it is designed for. Furthermore, it is recommended to further refine and expand the model in order to make it applicable to a broader range of use cases. Table 8.3 comprises a set of measures that could be taken to do so. This not only involves the low-level technical design but also its application to higher-level objectives.

Table 8.3: A list of research recommendations and potential follow-up projects, which would supplement this work or remove its quality.

Recommendation	Explanation
Refining the simulation model in terms of data delay and controller design	At multiple occasions it was mentioned that the technical design as well as the simulation itself do at times exhibit (acceptable) flaws. In order to allow for a wider model applicability (e.g. for a safety evaluation), these shortcomings would need to be tackled first. This point is especially affiliated with those flaws, which follow from the modeled delay in data exchange and the controller design itself. Both aspects could be modeled in more detail, taking less simplifications. Part of this would be to modify the VISSIM-internal vehicle model, which is actuated through the EDM DLL. Although no proof for this can be provided, reasoned motivation exists that such model improvements could further enhance the emission and congestion savings. from Table 5.6. Furthermore, when improving the controller design, additional attention could be drawn to minimizing distributive effects. Although the minimum goal of not making anyone worse off was reached, there are certain vehicle types that benefit more than others. In a secondary, improved controller design this could be taken into account.
Incorporating drag resistance benefits in the model	The platooning model and subsequently the presented traffic model have so far solely been concerned with traffic optimization. No effort was made to model the emission benefits that can be harvested from the reduced wind resistance among platooning vehicles. However, as described in Section 8.3, it can be expected that these would further reduce the expectable emissions and hence provide more motivation for an implementation of the technology. Although, these effects will be lower than those that can be expected for highway planning, quantifying them would surely be interesting e.g. for the government, which is concerned with reducing emissions.

Consider faulty data	In multiple occasions throughout this report it is mentioned that faulty data is not considered within its scope. Yet, faulty data is something that platoons and intersection controllers are faced with in real-life. Incorporating this into the controller schemes of this research would make the work more interesting for e.g. the automotive industry or other entities which do not only seek impact evaluation but actual implementation.
Expanding the physical scope	Although the research at hand gives a generic insight into the general effects of platooning and intersection control, it would certainly add value to test and verify these insights on other physical environments. Within the policy-development framework high attention is drawn to the validation of the expectable benefits on other physical scopes. Hence, running the simulation on other environments could help to simplify this validation step. Furthermore, considering a bigger simulation environment can be expected to lead to better emission results for the planning scenarios. That is, because the formation of a platoon leads to a considerable increase of emissions. This process takes up tp 500 m of traveling distance and over this distance the platooning cars usually exhibit higher emissions than they do during the non-platooning scenarios. The formation of a platoon goes along with considerably higher accelerations (e.g. to close the gap to a precursor), as described in Section 8.3. Hence, if a bigger physical scope is applied, the period, where platooning benefits can be harvested can be extended, further compensating for the additional emissions of the platoon formation.
Testing the different technological setups under different circumstances	From seeing how the technologies behave e.g. under higher average speeds or denser traffic could allow for further insight into their capabilities. Such tests should necessarily be made before real-world application can be considered. They could for example lead to gained knowledge on the point where the benefits of platooning doe not outweigh the costs of platoon formation anymore, as described in Section 8.3.
Consider human factors in policy making	The policy considerations of this research assume that the traffic participants always act according to what maximizes their monetary status (including the value of time). This could be supplemented by taking into account preference factors (towards a certain type of vehicle or driving mode) and/or distrust/misbelief in the new technology.
Consider transaction costs in policy making	The integration of a new technology often goes along with so-called transition costs. These costs describe the monetary effort that needs to be made to integrate the new technology in the new system. An example for those are the costs that are associated with the field tests and pilot projects that are proposed in the context of the policy-development framework. Quantifying these costs and adding them to the introduced game would further improve the quality of this work. Taking this one step further, one could also think about a potential process design in order to promote the integration of the technologies.

Consider legal aspects	One major aspect of such a process design would clearly be the legal issues, which follow from the introduction of such a disruptive technology.
Consider further/secondary impacts	The research at hand was solely focused on the simulation of congestion and emission impacts. However, the complexity and high accuracy of the model would allow for assessing further/secondary impacts as well. Quantifying e.g. the impacts on the automotive industry would further allow to consider this as an additional player in the policy making process. In Table 1.2 some of those not evaluated consequences are listed.

A

EXTERNAL DRIVER MODEL API

A.1. VEHICLE-SPECIFIC PARAMETERS

In the following table, the vehicle-specific parameters that are passed back and forth between the external driver model DLL and VISSIM are elaborated on. Note that some of the data might be overlapping with data from the shared database (see e.g. DRIVER_DATA_NVEH_WEIGHT and the associated entry of DRIVER_DATA_VEH_WEIGHT of the next vehicle). Although this data is available twice, only the entries from the shared database is used in order to feature a coherent V2V delay for all data sets.

Table A.1: Vehicle-specific parameters, as defined in the 'External driver model'-API.

Vehicle parpameters	Description
DRIVER_DATA_TIMESTEP	simulation time step length [s]
DRIVER_DATA_TIME	current simulation time [s]
DRIVER_DATA_WANTS_SUGGESTION	flag: does driver model want suggestion? (1 = yes, 0 = no)
DRIVER_DATA_SIMPLE_LANECHANGE	flag: does driver model want VISSIM to control the lateral movement during the lane change (i.e. start lane change when ACTIVE_LANE_CHANGE != 0 but ignore DESIRED_LANE_ANGLE) and stop the lane change after the vehicle has reached the middle of the new lane? (1 = yes, 0 = no)
DRIVER_DATA_VEH_ID	vehicle number
DRIVER_DATA_VEH_LANE	current lane number (rightmost = 1)
DRIVER_DATA_VEH_LANE_ANGLE	angle relative to the middle of the lane [rad]
DRIVER_DATA_VEH_LATERAL_POSITION	distance of the front end from the middle of the lane [m] (positive = left of the middle, negative = right)
DRIVER_DATA_VEH_VELOCITY	current speed [m/s]
DRIVER_DATA_VEH_ACCELERATION	current acceleration [m/s²]
DRIVER_DATA_VEH_LENGTH	vehicle length [m]
DRIVER_DATA_VEH_WEIGHT	vehicle weight [kg]
DRIVER_DATA_VEH_MAX_ACCELERATION	maximum possible acceleration [m/s²] (depending on current speed)
DRIVER_DATA_VEH_TURNING_INDICATOR	left = 1, right = -1, none = 0, both = 2 (also used by DriverModelGetValue()!)
DRIVER_DATA_VEH_CATEGORY	car = 1, truck = 2, bus = 3, tram = 4, pedestrian = 5, bike = 6
DRIVER_DATA_VEH_PREFERRED_REL_LANE	positive = left, 0 = current lane, negative = right
DRIVER_DATA_VEH_USE_PREFERRED_LANE	0 = only preferable (e.g. European highway) 1 = necessary (e.g. before a connector)
DRIVER_DATA_VEH_DESIRED_VELOCITY	desired speed [m/s] (also used by DriverModelGetValue()!)

<table>
<tr><td>DRIVER_DATA_VEH_TYPE</td><td>vehicle type number (user defined)</td></tr>
<tr><td>DRIVER_DATA_VEH_COLOR</td><td>vehicle color (24 bit RGB value) (also used by GetDriverValue()!)</td></tr>
<tr><td>DRIVER_DATA_VEH_CURRENT_LINK</td><td>current link number</td></tr>
<tr><td>DRIVER_DATA_VEH_NEXT_LINKS</td><td>following link number(s) of the vehicle's route This message is sent from VISSIM only if DriverModelSetValue() returned 1 for DRIVER_DATA_VEH_CURRENT_LINK. It is sent once for each link in the route.</td></tr>
<tr><td>DRIVER_DATA_VEH_ACTIVE_LANE_CHANGE</td><td>direction of an active lane change movement (+1 = to the left, 0 = none, -1 = to the right)</td></tr>
<tr><td>DRIVER_DATA_VEH_REL_TARGET_LANE</td><td>target lange (+1 = next one left, 0 = current lane, -1 = next one right)</td></tr>
<tr><td>DRIVER_DATA_VEH_DESTINATION_LINK</td><td>number of destination link (only sent before CREATE_DRIVER if DriverModelGetValue (DRIVER_DATA_SETS_XY_COORDINATES, ...) returned 1)</td></tr>
<tr><td>DRIVER_DATA_NVEH_ID</td><td>vehicle number (negative = no vehicle at this lane/position)</td></tr>
<tr><td>DRIVER_DATA_NVEH_LANE_ANGLE</td><td>angle relative to the middle of the lane [rad] (positive = turning left)</td></tr>
<tr><td>DRIVER_DATA_NVEH_LATERAL_POSITION</td><td>distance of the front end from the middle of the lane [m] (positive = left of the middle, negative = right)</td></tr>
<tr><td>DRIVER_DATA_NVEH_DISTANCE</td><td>gross distance [m] (front end to front end)</td></tr>
<tr><td>DRIVER_DATA_NVEH_REL_VELOCITY</td><td>speed difference [m/s] (veh. speed - nveh. speed)</td></tr>
<tr><td>DRIVER_DATA_NVEH_ACCELERATION</td><td>current acceleration [m/s²]</td></tr>
<tr><td>DRIVER_DATA_NVEH_LENGTH</td><td>vehicle length [m]</td></tr>
<tr><td>DRIVER_DATA_NVEH_WEIGHT</td><td>vehicle weight [kg]</td></tr>
<tr><td>DRIVER_DATA_NVEH_TURNING_INDICATOR</td><td>left = 1, right = -1, none = 0, both = 2</td></tr>
<tr><td>DRIVER_DATA_NVEH_CATEGORY</td><td>car = 1, truck = 2, bus = 3, tram = 4, pedestrian = 5, bike = 6</td></tr>
<tr><td>DRIVER_DATA_NVEH_LANE_CHANGE</td><td>direction of a current lane change (+1 = to the left, 0 = none, -1 = to the right)</td></tr>
<tr><td>DRIVER_DATA_NO_OF_LANES</td><td>number of lanes on the link</td></tr>
<tr><td>DRIVER_DATA_LANE_END_DISTANCE</td><td>distance to end of lane [m] (can be emergency stop position before connector) (negative = no end of lane in visibility range)</td></tr>
<tr><td>DRIVER_DATA_SIGNAL_DISTANCE</td><td>distance [m] to next signal head (negative = no signal head visible)</td></tr>
<tr><td>DRIVER_DATA_SIGNAL_STATE</td><td>red = 1, amber = 2, green = 3, red/amber = 4, amber flashing = 5, off = 6, green arrow = 7</td></tr>
<tr><td>DRIVER_DATA_SIGNAL_STATE_START</td><td>simulation time [s] when signal changed to current state</td></tr>
<tr><td>DRIVER_DATA_DESIRED_ACCELERATION</td><td>desired acceleration [m/s²] in next time step</td></tr>
<tr><td>DRIVER_DATA_DESIRED_LANE_ANGLE</td><td>desired angle relative to the middle of the lane [rad] (positive = turning left)</td></tr>
<tr><td>DRIVER_DATA_ACTIVE_LANE_CHANGE</td><td>direction of an active lane change movement (+1 = to the left, 0 = none, -1 = to the right) (must be != 0 while lane change is not completed)</td></tr>
<tr><td>DRIVER_DATA_REL_TARGET_LANE</td><td>target lange (+ || = next one left, 0 = current lane, -1 = next one right)</td></tr>
</table>

B

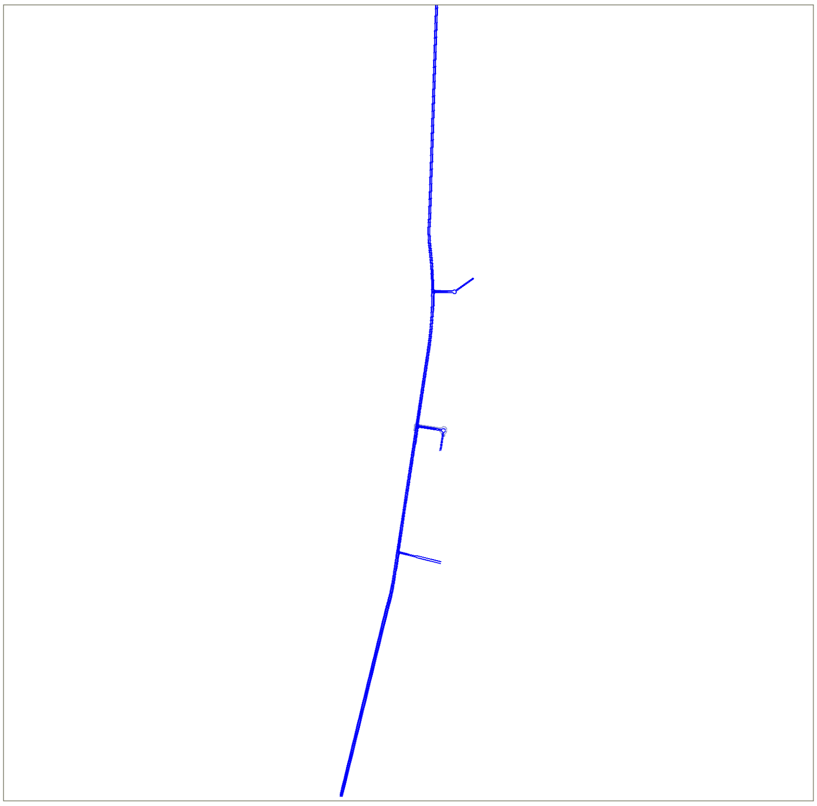
EnViver emission report: Scenario 1

The following, contains the full EnViver emission report for the first simulation scenario with no technological additions.

Emission

CO_2	NO_X	PM_{10}
1048.037 kg	1484.264 g	180.480 g
1000.486 kg/h	1416.921 g/h	172.291 g/h
216.683 g/km	306.874 mg/km	37.314 mg/km

Era: 2016
Time span: 12:00:08-01:02:59



EnViVer uses the VERSIT+micro models developed by TNO.

Traffic data

	Total			Calculated			Excluded		
	Trips	Samples	Distance	Trips	Samples	Distance	Trips	Samples	Distance
Light_Duty_City_2016	1196	2369906	4654.6 km	1196	2369906	4657.6 km	0	0	0.0 km
HD_Heavy_City_2016	16	39945	77.7 km	16	39945	77.7 km	0	0	0.0 km
HD_Medium_City_2016	20	53500	104.4 km	20	53500	104.4 km	0	0	0.0 km
Unassigned	131	118476	41.4 km	0	0	0.0 km	131	118476	41.4 km
Total	1363	2581827	4878.1 km	1232	2463351	4839.7 km	131	118476	41.4 km

Emission totals per class

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	881.973 kg	1152.225 g	153.440 g
HD_Heavy_City_2016	96.882 kg	111.677 g	11.812 g
HD_Medium_City_2016	69.182 kg	220.362 g	15.228 g
	6.6%	14.8%	8.4%
Total	1048.037 kg	1484.264 g	180.480 g

Emission per class per hour

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	841.957 kg/h	1099.947 g/h	146.478 g/h
HD_Heavy_City_2016	92.487 kg/h	106.610 g/h	11.276 g/h
HD_Medium_City_2016	66.043 kg/h	210.364 g/h	14.537 g/h

Emission per class per km.

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	189.484 g/km	247.545 mg/km	32.965 mg/km
HD_Heavy_City_2016	1246.118 g/km	1.436 g/km	151.925 mg/km
HD_Medium_City_2016	662.805 g/km	2.111 g/km	145.893 mg/km

Assignments

	Versit+ emission class	Vehicle type	
Light_Duty_City_2016	All Euro Light Urban 2016	luxe	
		sport	
		cabrio	
		compact2	
		compact1	
		SUV	
		midden1	
		American	
		midden2	
		Jeep	
		MPV	
		motor	
HD_Medium_City_2016	All Euro Middle Heavy Urban 2016	HGV_L2_vol	
		HGV_L1_leeg	
		HGV_L2_vol	
		HGV_L1_vol	
		HGV_L2_80%	
		HGV_L2_80%	
		HGV_L1_80%	
		HGV_L1_80%	
		HGV_L2_leeg	
HD_Heavy_City_2016	All Euro Heavy Urban 2016	HGV_L3_vol	
		HGV_L3_vol	
		HGV_L3_leeg	
		HGV_L3_80%	
Unassigned	Emission class not assigned	Bike	Excluded
		Pedestrian	Excluded

C

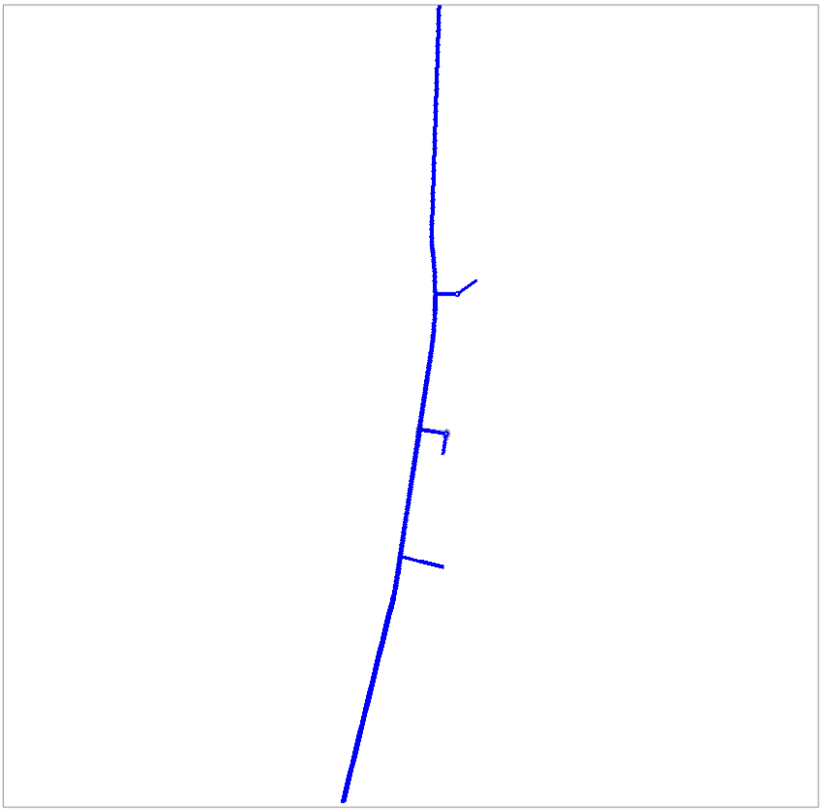
EnViver emission report: Scenario 2

The following, contains the full EnViver emission report for the second simulation scenario, comprising the integration of JUNO.

Emission

CO_2	NO_X	PM_{10}
967.849 kg	**1329.430 g**	**170.928 g**
967.849 kg/h	**1329.430 g/h**	**170.928 g/h**
210.604 g/km	**289.284 mg/km**	**37.194 mg/km**

Era: 2016
Time span selection: 12:00:08-01:00:08 (total 12:00:08-01:14:57)



EnViVer uses the VERSIT+micro models developed by TNO.

Enviver (Enterprise)(Online)

Traffic data

	Total			Calculated			Excluded		
	Trips	Samples	Distance	Trips	Samples	Distance	Trips	Samples	Distance
Light_Duty_City_2016	1429	2784873	5563.6 km	1139	2209809	4421.6 km	290	575064	1143.9 km
HD_Heavy_City_2016	20	48990	96.1 km	16	39014	77.7 km	4	9976	18.3 km
HD_Medium_City_2016	27	63170	125.9 km	20	48976	98.1 km	7	14194	27.8 km
Unassigned	155	142315	49.0 km	0	0	0.0 km	155	142315	49.0 km
Total	1631	3039348	5834.6 km	1175	2297799	4597.5 km	456	741549	1239.0 km

Emission totals per class

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	810.527 kg	1021.148 g	144.924 g
HD_Heavy_City_2016	94.580 kg	108.548 g	11.774 g
HD_Medium_City_2016	62.742 kg	199.734 g	14.230 g
	6.5%	15.0%	8.3%
Total	967.849 kg	1329.430 g	170.928 g

Emission per class per hour

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	810.527 kg/h	1021.148 g/h	144.924 g/h
HD_Heavy_City_2016	94.580 kg/h	108.548 g/h	11.774 g/h
HD_Medium_City_2016	62.742 kg/h	199.734 g/h	14.230 g/h

Emission per class per km.

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	183.387 g/km	231.041 mg/km	32.790 mg/km
HD_Heavy_City_2016	1216.540 g/km	1.396 g/km	151.439 mg/km
HD_Medium_City_2016	639.686 g/km	2.036 g/km	145.086 mg/km

Assignments

	Versit+ emission class	Vehicle type	
Light_Duty_City_2016	All Euro Light Urban 2016	luxe sport cabrio compact2 compact1 SUV midden1 American midden2 Jeep MPV motor	
HD_Medium_City_2016	All Euro Middle Heavy Urban 2016	HGV_L2_vol HGV_L1_leeg HGV_L2_vol HGV_L1_vol HGV_L2_80% HGV_L2_80% HGV_L1_80% HGV_L1_80% HGV_L2_leeg HGV_L2_leeg HGV_L1_leeg	
HD_Heavy_City_2016	All Euro Heavy Urban 2016	HGV_L3_vol HGV_L3_vol HGV_L3_leeg HGV_L3_80%	
Unassigned	Emission class not assigned	Bike Pedestrian	Excluded Excluded

D

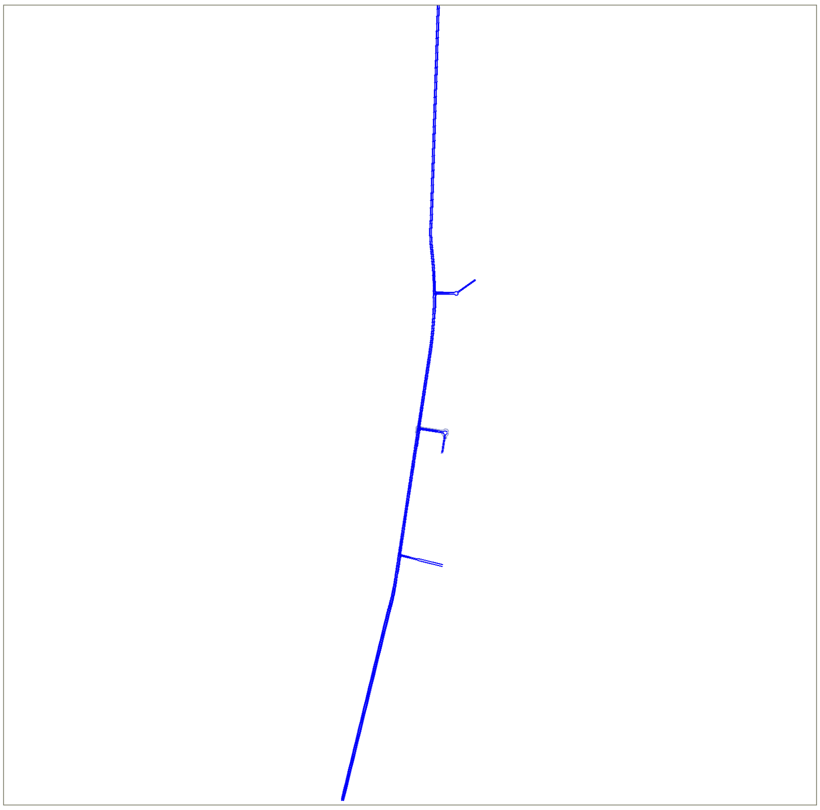
EnViver emission report: Scenario 3

The following, contains the full EnViver emission report for the third simulation scenario, comprising the integration of platooning.

Emission

CO_2	NO_X	PM_{10}
959.095 kg	**1311.768 g**	**170.034 g**
959.095 kg/h	**1311.768 g/h**	**170.034 g/h**
208.836 g/km	**285.628 mg/km**	**37.024 mg/km**

Era: 2016
Time span selection: 12:00:08-01:00:08 (total 12:00:08-01:00:12)



EnViVer uses the VERSIT+micro models developed by TNO.

Enviver (Enterprise)(Online)

Traffic data

	Total			Calculated			Excluded		
	Trips	Samples	Distance	Trips	Samples	Distance	Trips	Samples	Distance
Light_Duty_City_2016	1141	2219150	4423.6 km	1139	2215787	4412.9 km	2	3363	6.5 km
HD_Heavy_City_2016	16	38996	77.4 km	16	38996	77.4 km	0	0	0.0 km
HD_Medium_City_2016	20	47680	98.3 km	20	47533	98.1 km	0	147	0.2 km
Unassigned	125	116664	39.4 km	0	0	0.0 km	125	116664	39.4 km
Total	1302	2422490	4638.6 km	1175	2302316	4588.3 km	127	120174	46.1 km

Emission totals per class

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	804.470 kg	1014.143 g	144.243 g
HD_Heavy_City_2016	93.773 kg	105.328 g	11.661 g
HD_Medium_City_2016	60.852 kg	192.298 g	14.131 g
	6.3%	14.7%	8.3%
Total	959.095 kg	1311.768 g	170.034 g

Emission per class per hour

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	804.470 kg/h	1014.143 g/h	144.243 g/h
HD_Heavy_City_2016	93.773 kg/h	105.328 g/h	11.661 g/h
HD_Medium_City_2016	60.852 kg/h	192.298 g/h	14.131 g/h

Emission per class per km.

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	182.127 g/km	229.595 mg/km	32.656 mg/km
HD_Heavy_City_2016	1211.959 g/km	1.361 g/km	150.709 mg/km
HD_Medium_City_2016	620.290 g/km	1.960 g/km	144.038 mg/km

Assignments

	Versit+ emission class	Vehicle type	
Light_Duty_City_2016	All Euro Light Urban 2016	luxe sport cabrio compact2 compact1 SUV midden1 American midden2 Jeep MPV motor	
HD_Medium_City_2016	All Euro Middle Heavy Urban 2016	HGV_L2_vol HGV_L1_leeg HGV_L2_vol HGV_L1_vol HGV_L2_80% HGV_L2_80% HGV_L1_80% HGV_L1_80% HGV_L2_leeg	
HD_Heavy_City_2016	All Euro Heavy Urban 2016	HGV_L3_vol HGV_L3_vol HGV_L3_leeg HGV_L3_80%	
Unassigned	Emission class not assigned	Bike Pedestrian	Excluded Excluded

E

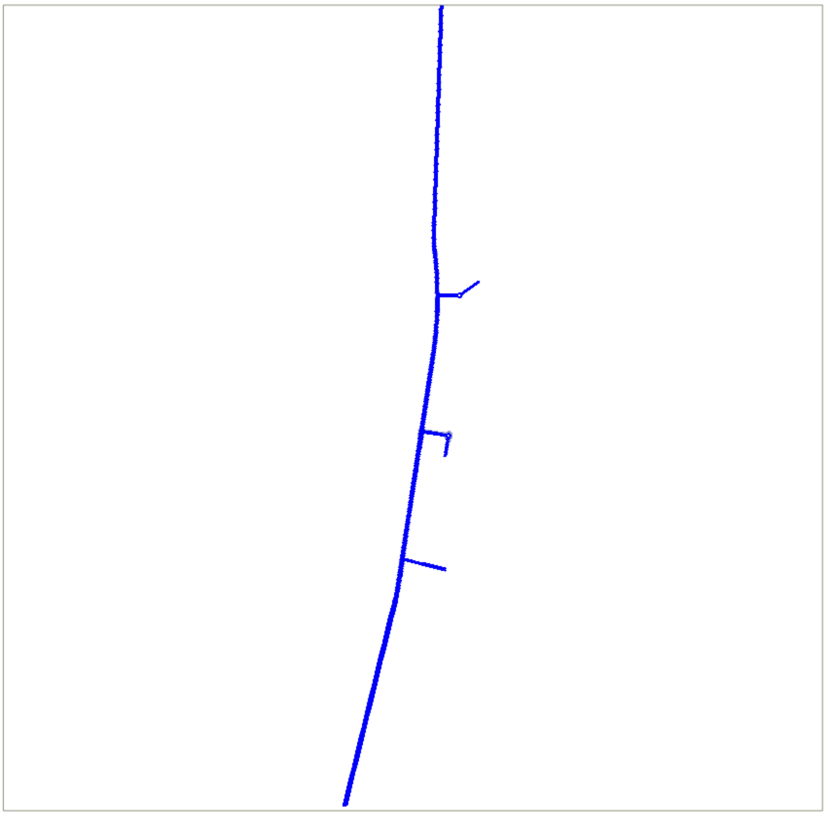
EnViver emission report: Scenario 4

The following, contains the full EnViver emission report for the fourth simulation scenario, comprising the integration of a functional combination between platooning and JUNO.

Emission

CO_2	NO_X	PM_{10}
960.554 kg	**1314.114 g**	**170.278 g**
959.248 kg/h	**1312.328 g/h**	**170.047 g/h**
208.850 g/km	**285.724 mg/km**	**37.023 mg/km**

Era: 2016
Time span: 12:00:08-01:00:12



EnViVer uses the VERSIT+micro models developed by TNO.

Enviver (Enterprise)(Online)

Traffic data

	Total			Calculated			Excluded		
	Trips	Samples	Distance	Trips	Samples	Distance	Trips	Samples	Distance
Light_Duty_City_2016	1141	2219150	4423.6 km	1141	2219150	4419.3 km	0	0	0.0 km
HD_Heavy_City_2016	16	38996	77.4 km	16	38996	77.4 km	0	0	0.0 km
HD_Medium_City_2016	20	47680	98.3 km	20	47680	98.3 km	0	0	0.0 km
Unassigned	125	116664	39.4 km	0	0	0.0 km	125	116664	39.4 km
Total	1302	2422490	4638.6 km	1177	2305826	4595.0 km	125	116664	39.4 km

Emission totals per class

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	805.713 kg	1015.762 g	144.454 g
HD_Heavy_City_2016	93.773 kg	105.328 g	11.661 g
HD_Medium_City_2016	61.068 kg	193.024 g	14.164 g
	6.4%	14.7%	8.3%
Total	960.554 kg	1314.114 g	170.278 g

Emission per class per hour

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	804.618 kg/h	1014.382 g/h	144.257 g/h
HD_Heavy_City_2016	93.646 kg/h	105.185 g/h	11.645 g/h
HD_Medium_City_2016	60.985 kg/h	192.761 g/h	14.144 g/h

Emission per class per km.

	CO_2	NO_X	PM_{10}
Light_Duty_City_2016	182.141 g/km	229.625 mg/km	32.656 mg/km
HD_Heavy_City_2016	1211.959 g/km	1.361 g/km	150.709 mg/km
HD_Medium_City_2016	621.146 g/km	1.963 g/km	144.064 mg/km

Assignments

	Versit+ emission class	Vehicle type	
Light_Duty_City_2016	All Euro Light Urban 2016	luxe	
		sport	
		cabrio	
		compact2	
		compact1	
		SUV	
		midden1	
		American	
		midden2	
		Jeep	
		MPV	
		motor	
HD_Medium_City_2016	All Euro Middle Heavy Urban 2016	HGV_L2_vol	
		HGV_L1_leeg	
		HGV_L2_vol	
		HGV_L1_vol	
		HGV_L2_80%	
		HGV_L2_80%	
		HGV_L1_80%	
		HGV_L1_80%	
		HGV_L2_leeg	
HD_Heavy_City_2016	All Euro Heavy Urban 2016	HGV_L3_vol	
		HGV_L3_vol	
		HGV_L3_leeg	
		HGV_L3_80%	
Unassigned	Emission class not assigned	Bike	Excluded
		Pedestrian	Excluded

Bibliography

M. Ahmadian, Z. Nazari, N. Nakhaee, and Z. Kostic. Model based design and sdr. 2005.

A. Alam. *Fuel-efficient heavy-duty vehicle platooning*. PhD thesis, KTH Royal Institute of Technology, 2014.

G. Antonelli and S. Chiaverini. Kinematic control of platoons of autonomous vehicles. *IEEE Transactions on Robotics*, 22(6):1285–1292, 2006.

P. Avanzini, B. Thuilot, and P. Martinet. A control strategy taking advantage of inter-vehicle communication for platooning navigation in urban environment. In *IROS11 International workshop on Perception and Navigation for Autonomous Vehicles in Human Environment*, 2011.

B. Bakermans. Truck platooning: Enablers, barriers, potential and impacts. Master's thesis, TU Delft, 2016.

H. Becker. *High noon in the automotive industry*. Springer Science & Business Media, 2006.

M. Bell and R. Bretherton. Ageing of fixed-time traffic signal plans. In *International conference on road traffic control*, 1986.

L. C. Bento, R. Parafita, and U. Nunes. Intelligent traffic management at intersections supported by v2v and v2i communications. pages 1495–1502. 15th International IEEE Conference on Intelligent Transportation Systems (ITSC) 2012, IEEE, 2012.

C. Bergenhem, E. Hedin, and D. Skarin. Vehicle-to-vehicle communication for a platooning system. *Procedia-Social and Behavioral Sciences*, 48:1222–1233, 2012a.

C. Bergenhem, S. Shladover, E. Coelingh, C. Englund, and S. Tsugawa. Overview of platooning systems. In *Proceedings of the 19th ITS World Congress, Oct 22-26, Vienna, Austria (2012)*, 2012b.

U. Berkeley. Truck platooning, 2017. URL http://www.path.berkeley.edu/research/automated-and-connected-vehicles/truck-platooning.

B. Bernais and A. Lotz. Analyse von car-2-x-nachrichten für fahrerassistenzsysteme im kreuzungsbereich. *GMM-Fachbericht-AmE 2014–Automotive meets Electronics*, 2014.

P. Bjesse. What is formal verification? *ACM SIGDA Newsletter*, 35(24):1, 2005.

C. Bonnet and H. Fritz. Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing. Technical report, SAE Technical Paper, 2000.

A. Bowen and M. Eubank. *Intersection Decision Guide*. Indiana Department of Transportation (INDOT), 2014. URL https://www.in.gov/indot/files/ROP_IntersectionDecisionGuide.pdf.

I. Carson, D. M. Nicol, B. L. Nelson, J. Banks, et al. Discrete-event system simulation, 2005.

Y. Carson and A. Maria. Simulation optimization: methods and applications. In *Proceedings of the 29th conference on Winter simulation*, pages 118–126. IEEE Computer Society, 1997.

C. B. v. d. S. CBS. *Transport and Mobility 2016*. Statistics Netherlands, 2016.

CDC. Cdc policy process, 2017. URL https://www.cdc.gov/policy/analysis/process/index.html.

Y. Q. Chen and Z. Wang. Formation control: a review and a new consideration. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3181–3186. IEEE, 2005.

C. CP. Road safety - towards zero fatalities ? *Research*eu results magazine*, 42, 2017.

W. Daamen, C. Buisson, and S. P. Hoogendoorn. *Traffic Simulation and Data: Validation Methods and Applications*. CRC Press, 2014.

Detering. *Kalibrierung und Validierung von Verkehrssimulationsmodellen zur Untersuchung von Verkehrsassistenzsystemen*. PhD thesis, Technische Universität Braunschweig, 2011.

D. DMI. Truck platooning challenge 2016, 2017. URL https://www.eutruckplatooning.com/About/Corridors+to+drive+MAP/default.aspx.

R. Dowling, A. Skabardonis, J. Halkias, G. McHale, and G. Zammit. Guidelines for calibration of microsimulation models: framework and applications. *Transportation Research Record: Journal of the Transportation Research Board*, (1876):1–9, 2004.

D. Easley and J. Kleinberg. *Networks, crowds, and markets*, volume 6. 2010.

E. C. EC. Modal split of inland passenger transport, 2013. URL http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=tran_hv_psmod&lang=en.

E. C. EC, 2017. URL http://www.sartre-project.eu/en/Sidor/default.aspx.

J. Eckhardt. European truck platooning challenge 2016. *The Hague, Delta3*, 2015.

E. E. A. EEA. *Transitions towards a more sustainable mobility system*, volume EEA Report. Publications Office of the European Union, 2016.

E. T. P. C. ETPC. Creating next generation mobility - themes, 2017. URL https://www.eutruckplatooning.com/Themes/default.aspx.

F. Farokhi and K. H. Johansson. A study of truck platooning incentives using a congestion game. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):581–595, 2015.

J. Favilla, A. Machion, and F. Gomide. Fuzzy traffic control: adaptive strategies. In *Fuzzy Systems, 1993., Second IEEE International Conference on*, pages 506–511. IEEE, 1993.

M. Fellendorf. *VISSIM: A microscopic simulation tool to evaluate actuated signal control including bus priority*. Springer, 1994.

M. E. Fouladvand, Z. Sadjadi, and M. R. Shaebani. Optimized traffic flow at a single intersection: traffic responsive signalization. *Journal of Physics A: Mathematical and General*, 37(3):561, 2004.

M. Garrow, R. B. Machemehl, et al. Development and evaluation of transit signal priority strategies. Technical report, University of Texas at Austin, 1997.

GCDC. Grand cooperative driving challange, 2017. URL http://www.gcdc.net/en/.

O. Gehring and H. Fritz. Practical results of a longitudinal control concept for truck platooning with vehicle to vehicle communication. *IEEE Conference on Intelligent Transportation System, 1997. ITSC'97*, pages 117–122, 1997.

M. J. Gibbs, K. A. Merchant, W. A. Van der Stede, and M. E. Vargus. Performance measure properties and incentive system design. *Industrial relations: a journal of economy and society*, 48(2):237–264, 2009.

M. Herty and A. Klar. Modeling, simulation, and optimization of traffic flow networks. *SIAM Journal on Scientific Computing*, 25(3):1066–1087, 2003.

M. Hoffman and T. Beaumont. *Application Development: Managing the Project's Life Cycle*. MC Press, LLC, 1997.

R. Isermann. Modeling and design methodology for mechatronic systems. *IEEE/ASME Transactions on Mechatronics*, 1(1):16–28, 1996.

R. E. Kalman et al. Contributions to the theory of optimal control. *Bol. Soc. Mat. Mexicana*, 5(2):102–119, 1960.

P. Kavathekar and Y. Q. Chen. Vehicle platooning: A brief survey and categorization. In *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 829–845. American Society of Mechanical Engineers, 2011.

V. L. Knoop, S. P. Hoogendoorn, and H. J. van Zuylen. Importance of spillback modeling in assessing its. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pages 909–914. IEEE, 2006.

M. P. Lammert, A. Duran, J. Diez, K. Burton, and A. Nicholson. Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, following distances, and mass. *SAE International Journal of Commercial Vehicles*, 7(2014-01-2438):626–639, 2014.

J. Larson, K.-Y. Liang, and K. H. Johansson. A distributed framework for coordinated heavy-duty vehicle platooning. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):419–429, 2015.

J. Lebacque. Intersection modeling, application to macroscopic network traffic flow models and traffic management. In *Traffic and Granular Flow'03*, pages 261–278. Springer, 2005.

J. Lei and U. Ozguner. Decentralized hybrid intersection control. In *Proceedings of the 40th IEEE Conference on decision and Control 2001*, volume 2, pages 1237–1242. IEEE, 2001.

Leidos. Vissim external driver model add-on for the simulation of connected automated vehicle applications. Technical report, Saxton Transportation Operations Laboratory, 2015.

S. Li. *Analyse und Optimierung der Sendeeigenschaften einer ITS G5 Roadside Station.* PhD thesis, Technische Universität Braunschweig, 2015.

S. Li, L. Tong, J. Xing, and Y. Zhou. The market for electric vehicles: indirect network effects and policy design. *Journal of the Association of Environmental and Resource Economists*, 4(1):89–133, 2017.

J. Lioris, R. Pedarsani, F. Y. Tascikaraoglu, and P. Varaiya. Doubling throughput in urban roads by platooning. *IFAC-PapersOnLine*, 49(3):49–54, 2016.

X.-Y. Lu, T. Z. Qiu, P. Varaiya, R. Horowitz, and S. E. Shladover. Combining variable speed limits with ramp metering for freeway traffic control. pages 2266–2271. American Control Conference (ACC) 2010, IEEE, 2010.

C. M. Macal. Model verification and validation. *University of Chicago*, pages 1–21, 2005.

MAN. Man in deutschland, 2017. URL http://www.truck.man.eu/de/de/man-welt/man-in-deutschland/Erste-Transporte-mit-vernetzten-Lkw_-DB-Schenker-und-MAN-vereinbaren-Platooning-Projekt-273088.html.

G. Maps. Top view: Middeldijkdreef, 2017a. URL https://www.google.nl/maps/place/Tilburg/@51.5919429,4.9741884,602m/data=!3m1!1e3!4m5!3m4!1s0x47c69585518d5c67:0x636befab4fb5a182!8m2!3d51.5852528!4d5.056375.

G. Maps. Top view: Dalemdreef, 2017b. URL https://www.google.nl/maps/place/Tilburg/@51.5827052,4.9727319,600m/data=!3m1!1e3!4m5!3m4!1s0x47c69585518d5c67:0x636befab4fb5a182!8m2!3d51.5852528!4d5.056375.

G. Maps. Top view: Koolhovenlaan, 2017c. URL https://www.google.nl/maps/place/Tilburg/@51.5737123,4.97087,599m/data=!3m1!1e3!4m5!3m4!1s0x47c69585518d5c67:0x636befab4fb5a182!8m2!3d51.5852528!4d5.056375.

A. I. M. Medina, N. Van De Wouw, and H. Nijmeijer. Automation of a t-intersection using virtual platoons of cooperative autonomous vehicles. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 1696–1701. IEEE, 2015.

N. Medina, Alejandro Ivan Morales van de Wouw and H. Nijmeijer. Cooperative intersection control based on virtual platooning. *IEEE Intelligent Transport Systems Transactions*, 2017.

T. Meurer. *Reglungstechnik I.* Kiel University, 2010. URL https://www.control.tf.uni-kiel.de/en/teaching/winter-term/control-systems-i-etit-109/fileadmin/rt_ws1617_full.

J. B. Michael, D. N. Godbole, J. Lygeros, and R. Sengupta. Capacity analysis of traffic flow over a single-lane automated highway system. *Journal of Intelligent Transportation System*, 4(1-2):49–80, 1998.

U. Montanaro, M. Tufo, G. Fiengo, M. di Bernardo, A. Salvi, and S. Santini. Extended cooperative adaptive cruise control. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 605–610. IEEE, 2014.

P. Ngatchou, A. Zarei, and A. El-Sharkawi. Pareto multi objective optimization. In *Proceedings of the 13th international conference on Intelligent systems application to power systems, 2005*, pages 84–91. IEEE, 2005.

M. R. Nieuwenhuijze, T. van Keulen, S. Öncü, B. Bonsen, and H. Nijmeijer. Cooperative driving with a heavy-duty truck in mixed traffic: Experimental results. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1026–1032, 2012.

X. Niu, Z. Zhao, B. Liu, and T. Zhang. Analysis of total passager delay at signal coordinated intersection based on green extension strategy. pages 2548–2551. International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE) 2011, IEEE, 2011.

P. Noord-Brabant. N260: Update, 2017a. URL http://www.brabant.nl/kaarten/verkeer-en-vervoer-kaarten/verkeersintensiteitenkaart.aspx.

P. Noord-Brabant. N260: nieuwe weg om kom baarle heen, 2017b. URL http://www.brabant.nl/dossiers/dossiers-op-thema/verkeer-en-vervoer/wegen/wegenprojecten-in-brabant/n260-omlegging-baarle.aspx.

J. Ploeg, B. T. Scheepers, E. Van Nunen, N. Van de Wouw, and H. Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. pages 260–265. 14th International IEEE Conference on Intelligent Transportation Systems (ITSC) 2011, IEEE, 2011a.

J. Ploeg, N. van de Wouw, and H. Nijmeijer. String stability of vehicle platoons. *30th Benelux Meeting on Systems and Control, Lommel, Belgium, March 15-17, 2011*, 2011b.

J. Ploeg, N. Van De Wouw, and H. Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2014.

S. Porta, P. Crucitti, and V. Latora. The network analysis of urban streets: a dual approach. *Physica A: Statistical Mechanics and its Applications*, 369(2):853–866, 2006.

R. S. Pressman. *Software engineering: a practitioner's approach*. Palgrave Macmillan, 2005.

PTV Vissim User Manual. PTV AG, 2016.

R. Rajamani and C. Zhu. Semi-autonomous adaptive cruise control systems. *IEEE Transactions on Vehicular Technology*, 51(5):1186–1192, 2002.

J. Reedy and S. Lunzman. Model based design accelerates the development of mechanical locomotive controls. Technical report, SAE Technical Paper, 2010.

T. Robinson, E. Chan, and E. Coelingh. Operating platoons on public motorways: An introduction to the sartre platooning programme. volume 1, page 12. 17th world congress on intelligent transport systems, 2010.

R. P. Roess, E. S. Prassas, and W. R. McShane. *Traffic engineering*. Prentice Hall, 2011.

B. M. Roger. Game theory: analysis of conflict, 1991.

N. Rouphail, A. Tarko, and J. Li. Traffic flow at signalized intersections. 1992.

O.-R. A. V. S. C. SAE et al. Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems. *SAE Standard J3016*, pages 01–16, 2014.

R. G. Sargent. Verification and validation of simulation models. In *Proceedings of the 37th conference on Winter simulation*, pages 130–143. winter simulation conference, 2005.

Scania, 2017. URL http://www.scania.com/group/en/scania-lines-up-for-platooning-trials/.

G. Schermers, K. Malone, and B. Van Arem. Dutch evaluation of chauffeur assistant (deca): Traffic flow effects of implementation in the heavy goods vehicles sector. In *ITS World Congress, Nagoya, Aichi*, 2004.

M. G. Sohoni, S. Chopra, U. Mohan, and N. Sendil. Threshold incentives and sales variance. *Production and Operations Management*, 20(4):571–586, 2011.

W. Sun, X. Wu, Y. Wang, and G. Yu. A continuous-flow-intersection-lite design and traffic control for oversaturated bottleneck intersections. *Transportation Research Part C: Emerging Technologies*, 56:18–33, 2015.

D. Swaroop and J. Hedrick. String stability of interconnected systems. *IEEE transactions on automatic control*, 41(3):349–357, 1996.

D. Swaroop, J. Hedrick, C. Chien, and P. Ioannou. A comparision of spacing and headway control laws for automatically controlled vehicles. *Vehicle System Dynamics*, 23(1):597–625, 1994.

TNO. Enviver manual, 2017. URL https://www.tno.nl/en/focus-areas/urbanisation/mobility-logistics/clean-mobility/enviver-model-traffic-flow-and-emissions/.

S. Tsugawa. Final report on an automated truck platoon within energy its project. In *International Task Force on Vehicle-Highway Automation 2013 Meeting*, 2013.

S. Tsugawa. Results and issues of an automated truck platoon within the energy its project. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 642–647. IEEE, 2014.

R. Van Katwijk, B. De Schutter, and J. Hellendoorn. Traffic adaptive control of a single intersection: A taxonomy of approaches. pages 227–232. Proc. 11th IFAC Symposium on Control in Transportation Systems, 2006.

R. T. Van Katwijk. *Multi-agent look-ahead traffic-adaptive control*. PhD thesis, TU Delft, 2008.

R. T. van Katwijk and S. Gabriel. Optimising a vehicle's approach towards an adaptively controlled intersection. *IET Intelligent Transport Systems*, 9(5):479–487, 2015.

Volvo, 2017. URL http://www.volvotrucks.us/about-volvo/news-and-events/volvo-trucks-successfully-demonstrates-on-highway-truck-platooning-in-california/.

R. Wiedemann. Simulation des straßenverkehrsflusses. *Schriftreihen des Instituts für Verkehrswesen der Universität Karlsruhe*, Heft 8, 1974.

M. Wiering, J. Vreeken, J. Van Veenen, and A. Koopman. Simulation and optimization of traffic in a city. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 453–458. IEEE, 2004.

M. Zabat, N. Stabile, S. Farascaroli, and F. Browand. The aerodynamic performance of platoons: A final report. *California Partners for Advanced Transit and Highways (PATH)*, 1995.

J. C. Zegers, E. Semsar-Kazerooni, J. Ploeg, N. van de Wouw, and H. Nijmeijer. Consensus-based bi-directional cacc for vehicular platooning. In *American Control Conference (ACC), 2016*, pages 2578–2584. IEEE, 2016.

Y. Zheng, S. E. Li, J. Wang, L. Y. Wang, and K. Li. Influence of information flow topology on closed-loop stability of vehicle platoon with rigid formation. *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2094–2100, 2014.

B. Zhou, J. Cao, X. Zeng, and H. Wu. Adaptive traffic light control in wireless sensor network-based intelligent transportation system. In *Vehicular technology conference fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5. IEEE, 2010.