COBRA

A Cooperative Breakdown Prevention Algorithm

MSc. Thesis by: R.P.L. Overvoorde

TU Delft

TNO innovation for life

COBRA
A COOPERATIVE BREAKDOWN PREVENTION ALGORITHM

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of:

Master of Science in Transport, Infrastructure and Logistics

by

Richard Patrick Leendert Overvoorde

May 2020

Richard Patrick Leendert Overvoorde:
COBRA: cooperative breakdown prevention algorithm (2020)

This thesis was made in collaboration with:

Delft University of Technology

Department of Transport & Planning
Faculty of Civil Engineering and Geosciences
Delft University of Technology

TNO innovation for life

Department of Sustainable Urban Mobility and Safety
Focus Area of Traffic and Transport
TNO

Supervisors:	Prof. dr. ir. Serge Hoogendoorn	TU Delft
	Dr. ir. Andreas Hegyi	TU Delft
	Dr. ir. Riender Happee	TU Delft
	Dr. ir. Yusen Chen	TU Delft & TNO

SUMMARY

Increased traffic loads on highway networks lead to more congestion. Often, congestion occurs due to higher demand than capacity at a bottleneck. When this occurs, the capacity drop may activate and limit the flow upstream of the bottleneck segment where the jam occurs. Traffic management strategies are developed to combat these jams.

Recent developments in communication technology promise to increase the effectiveness of traffic management in the future. These systems are called Cooperative Intelligent Transportation Systems (C-ITS). C-ITS complements these strategies in two ways. Firstly, communication with low latency can be used to transfer sensory data of Connected Vehicles (CVs) to the infrastructure. Secondly, this data can be used to support traffic management strategies with more accurate and individual control of vehicles.

In this thesis, a traffic controller is developed that capitalizes on the low latency and increased span of control of C-ITS. The controller, called ”a COoperative BReakdown prevention Algorithm” (COBRA), manipulates traffic near bottlenecks on highways through individualized control signals.

In the literature review, knowledge is gathered on several aspects of traffic control with C-ITS. Firstly, the mechanism behind traffic breakdown is elaborated on by reviewing literature on breakdown probability and oscillations in traffic. Secondly, an overview is made of the currently available traffic control strategies using CVs. Thirdly, since Connected Vehicles (CVs) are not necessarily automated, the controller is dependant on compliance to control signals and thus knowledge was gathered on what influences driver compliance to in-car control signals.

COBRA is based on reducing the flow of clusters of vehicles with a level of flow that is not expected to pass a bottleneck without causing instabilities in traffic. When measurements from CVs indicate that the flow is too high in a cluster, the algorithm computes target trajectories for all vehicles in the cluster. These trajectories are sent to the vehicles, after which each CV either follows up on the control signal by either decelerating to the target speed or by presenting a speed limit to the driver if the vehicle is not automated. When vehicles comply with the control signals and decelerate to the target speed, traffic within the controlled area becomes more homogenized and can pass the bottleneck safely.

The evaluation of COBRA is performed through a simulation study in which COBRA was applied to a synthetic base case, consisting of a single-lane highway with a bottleneck and a pre-determined demand pattern. During the evaluation, the following factors were varied:

- Penetration rate of Connected Vehicles;
- Compliance of drivers to the speed-control signals;
- Driver behaviour in terms of deceleration.

The evaluation criteria consist of two quantitative measurements, which are network performance and traffic safety. Furthermore, trajectory plots of the vehicles in the network were used to evaluate the stability of traffic during the simulation runs.

Varying the penetration rate of CVs showed that the algorithm is effective for penetration rates of 25% and above. In these scenarios, the majority of jams could be prevented while improving network performance and traffic safety. For a penetration rate of 10%, the controller keeps traffic stable. However, the flow has to be severely reduced, resulting in a decrease in network performance.

Reducing compliance of drivers to speed-control signals resulted in a reduction in each of the three evaluation criteria. When drivers decelerate to different speeds due to different levels of speed-compliance, small clusters of vehicles form, which can become unstable when entering the bottleneck. Due to the oscillations that occur, safety is reduced.

Other behaviour of drivers that may influence the performance of the controller is a mismatch between the deceleration expected by the controller and the actual deceleration. Differences in deceleration of vehicles were evaluated and did not result in large reductions

in performance on either of the evaluation criteria, even when combining the variation in deceleration with a reduction in either penetration rate or compliance.

Even though COBRA is found to be effective in the synthetic case used for the evaluation, many steps are required to prepare COBRA for real-world application. Most importantly, the algorithm has to be extended to multi-lane traffic, with a challenge of finding a solution for overtaking and merging into voids created by the algorithm to dampen oscillations. Furthermore, data sources such as video-based monitoring and ITS-G5 should be linked to the computation and actuation of control signals, which may be challenging as vehicles receive personalized control signals. Many other challenges still lie ahead before COBRA will be ready for the real world. When these challenges are overcome, the implementation of COBRA offers great potential to increase bottleneck throughput.

CONTENTS

ACRONYMS

CAV Connected Automated Vehicle

C-ITS Cooperative Intelligent Transportation Systems

CTM Cell-Transmission-Model

CV Connected Vehicle

FCD Floating Car Data

FD Fundamental Diagram

iTTC inverse Time-To-Collision

MPC Model Predictive Control

MTFC Mainstream Traffic Flow Control

PB Probability of traffic Breakdown

RG Route Guidance

RM Ramp Metering

TTS Total Time Spent

TTT Total Travel Time

VBM Video-Based Monitoring

VMS Variable Message Sign

VSL Variable Speed Limit

1 INTRODUCTION

In this chapter, the design problem for this thesis will be motivated and presented. Chapter 2 contains a literature review that answers the sub-questions from this chapter and builds a knowledge base which can be used to tackle the proposed design problem. Then, in Chapter 3 the theory of the controller is discussed in detail. Following from the theory, the algorithm for the controller is presented in Chapter 4. Furthermore, the set-up of the simulation study on the controller and the tuning of the controller are presented in Chapter 5. The results from the simulation study are then presented in Chapter 6. Finally, conclusions from this thesis and a discussion on improvements to the algorithm, as well as recommendations for real-world application are given in Chapter 7.

1.1 INTRODUCTION TO THE PROBLEM

Increased traffic load on freeway networks results in a higher probability of congestion when it is not controlled. A schematic view of the occurrence of congestion is shown in Figure 1.1. When traffic levels on a freeway approach the capacity of the road, the chance of a breakdown due to a disturbance increases. These disturbances (e.g. a lane change or sudden braking) result in shock waves, which can evolve into a jam wave moving upstream through traffic or activate a standing queue near a bottleneck. A jam temporarily reduces the capacity of the freeway by 3% to 30%, this is called the capacity drop (Hall and Agyemang-Duah, 1991; Cassidy and Bertini, 1999; Srivastava and Geroliminis, 2013). In Figure 1.2 empirical measurements of freeway flow from both congested and free flow conditions are presented, showing the phenomena of capacity drop when traffic transitions to a congested state. However, reduced flow is not the only effect of jam waves. As speed differences between road users increase due to sudden braking, the chance of a collision increases as well. Other effects of increased braking and accelerating are higher energy consumption for cars and increased wear on the road, leading to higher fuel and maintenance costs respectively.

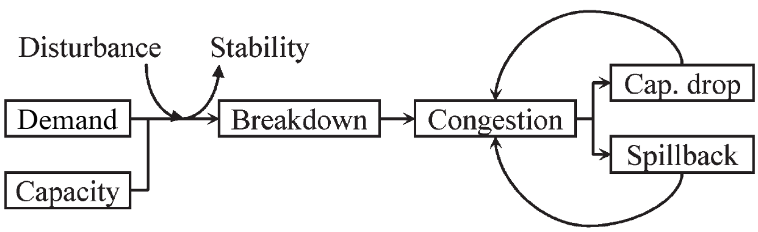


Figure 1.1: Schematic process of traffic breakdown (Schakel and Van Arem, 2014)

Researchers have put much effort into innovative technologies and concepts to resolve jams, or reduce the probability of a traffic breakdown. There are multiple methods for reducing the occurrence of or resolving jams, such as Variable Speed Limits (VSLs), Route Guidance (RG) and Ramp Metering (RM). A powerful line of research which influences the future of traffic management is research on C-ITS, which complements the aforementioned methods.

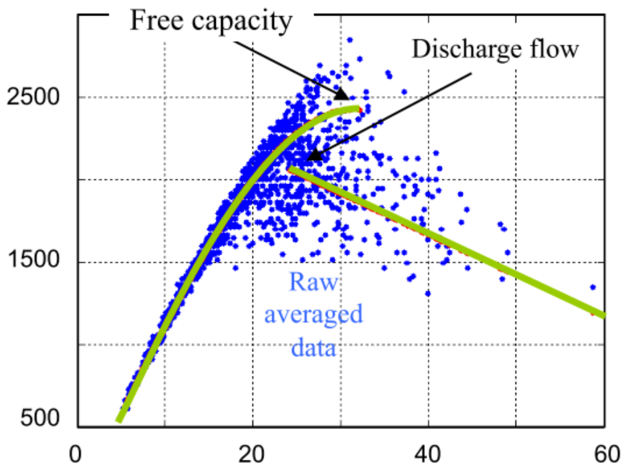


Figure 1.2: A Fundamental Diagram with capacity drop based on real data (Knoop et al., 2019)

1.2 IMPACT OF C-ITS ON TRAFFIC MANAGEMENT

C-ITS contributes to two main components of traffic control through its low latency. Firstly, traffic state estimation, which is used for detection of disturbances or for other important measurements, is important for traffic controllers to know where, when and how to act. Secondly, it adds the possibility of using individual vehicles as actuators for traffic control by giving personalized control signals to CVs, widening the span of control.

1.2.1 Traffic state measurement with C-ITS

One of the main contributions of CV technology to traffic management, is the possibility of sharing sensory data with the infrastructure and using these measurements to detect disturbances or anomalies (Netten et al., 2013; Yuan et al., 2012). This data can be used to complement conventional traffic state estimation tools, such as loop-detectors, in order to enhance accuracy and detection speed greatly.

Due to the low latency and high accuracy of new C-ITS communication technology (such as G5 NR or LTE-V2X (Bazzi et al., 2019)), speed and density measurements on the road are found to be accurate (Bekiaris-Liberis et al., 2016). Recent work has also shown that, even with a low penetration rate of 2%, reliable state estimations per lane can be made using data from CVs (Bekiaris-Liberis et al., 2017; Papadopoulou et al., 2018).

1.2.2 Traffic control with C-ITS

Control methods have shown their value for improvements to the performance of the infrastructure by making use of traffic management techniques (Khondaker and Kattan, 2015; Middelham and Taale, 2006). Due to faster and more accurate estimation with C-ITS, more accurate and directed control can be computed and actuated on equipped roads.

Especially in the VSL line of research, a lot of studies were performed into the use of CVs as actuators for control (Wang et al., 2016). Wang et al. (2016) concluded that using VSL for stop-and-go wave resolvement resulted in both positive and negative effects. Using the SPECIALIST algorithm developed by Hegyi et al. (2008), it was found that the total travel time could be reduced with the introduction of Connected Automated Vehicles (CAVs). However, since automated vehicles show different behaviour than conventional vehicles, lower flow and higher instability in the resolving traffic state were found. In order to solve this, a new algorithm with the same steps, but different computation of control, was developed (Hegyi, 2013). When a shock wave is detected with the support of CVs, the system activates and reduces flow into the tail of the jam by inducing individual speed limits that guide vehicles to a desired trajectory, which results in a higher but stable density.

While speed is an important factor, traffic breakdown is found to be caused by oscillations in traffic that do not fade out. The probability of traffic breakdown is strongly affected by the density, headway distribution and speed of traffic (Chow et al., 2009; Han and Ahn, 2018). Influencing these traffic characteristics in order to reduce the probability of a traffic breakdown can therefore be a solution (Han and Ahn, 2018).

1.3 PROPOSED DESIGN

There are situations where the capacity of the freeway is sufficient to facilitate the current flow, but disturbances in dense clusters of vehicles in traffic can still result in a capacity drop. Since headways are not always evenly distributed, there are clusters of vehicles with high density which have a higher probability of breaking down traffic. This situation can occur downstream of merging areas, but also near other types of bottlenecks. VSL algorithms have been developed in order to reduce the effects of jam waves. However, these algorithms are reactive to the occurrence of jam waves and thus do not prevent the temporary capacity drop, which results in relatively long control periods.

1.3.1 Design objective

CV technology gives the possibility of measuring and estimating traffic states on a high spatial-temporal resolution with low latency. These factors make it possible to measure and respond to possibly unstable traffic in real-time. Also, it is possible to give individual speed advice or send control signals to individual vehicles, which widens the span of control.

This research aims to capitalize on this new technology and develop a traffic control algorithm that aims to reduce traffic breakdown probability near bottlenecks by sending personalized operational advice to drivers of cars equipped with CV technology. Due to CVs not being directly controllable, driver compliance will have to be taken into account when deciding on the control measures, as well as the nature of the advice.

Concludingly, this research aims to design a traffic controller with the following properties:

- Aims to prevent traffic breakdown;
- Uses C-ITS for in-car messaging with low latency and high accuracy;
- Can be applied on freeways;
- Can be applied for real-time control;
- Can be dynamically applied to a wide variety of bottlenecks with some tuning or adjustments.

Before the design can be developed, currently available literature should be explored in order to get a strong base knowledge on how to fulfill these design objectives. In order to guide the research to the aforementioned objective and prepare for the design of the controller, the following main research question is formulated:

In what way can throughput of bottlenecks on freeways be improved, while using C-ITS technology to give in-car operational advice to drivers of CVs, without causing traffic instability in the network?

To answer the main question, knowledge has to be gathered on the different parts that the question contains. Therefore, in order to have a strong knowledge base for development of the algorithm, the following questions will be answered based on existing literature:

1. What are the mechanisms behind traffic breakdown on freeways?

2. What are important factors for predicting traffic breakdown on freeways and which of these factors can be influenced through control of CVs?

3. What are currently available mainline traffic control algorithms that use CVs and what mechanisms do they use?
4. What factors are found to influence compliance of drivers to advice given through C-ITS communication channels and which of these can be taken into account when designing a controller?

Answers to these questions from literature will be used as input for the controller design with the aforementioned properties.

2 LITERATURE REVIEW

In this chapter, information that is required for the controller design is gathered and presented. In the following paragraphs, the method of the literature is discussed and each of the topics and the desired information for these topics are described. Furthermore, the following sections present the literature study for each of these topics.

During the literature study, multiple types of sources will be used. The majority of the literature is published in journals. However, course material, as well as web articles can be used. Papers will be searched through search-engines like Scopus, Scholar and Science Direct by using the keywords which are relevant for the topic. These keywords consist of, but are certainly not limited to:

- "Traffic breakdown" + "Bottleneck";
- "Traffic control" + "Connected Vehicles";
- Traffic breakdown probability;
- "C-ITS" + "Compliance".

Other search methods to find important papers are forward and backward snowballing. Google scholar provides a strong platform to support this method as it provides direct links in both directions of referencing. Furthermore, to increase external validity, peer reviewed papers are highly preferred.

The application which is proposed in Chapter 1, aims to reduce the Probability of traffic Breakdown (PB) when road capacity should be sufficient to facilitate the demand. In Section 2.1, the mechanism of breakdown is studied, with a focus on recurrent bottlenecks. Furthermore, control algorithms require measurements on which they respond. In response to these measurements, control is applied to improve the system performance. As it is unknown when a disturbance in traffic will happen, the PB and which factors are important for increases in this probability will be studied. This knowledge will be used to decide on which factors should be controlled, as well as in which conditions control may be effective or not.

Mainstream Traffic Flow Control (MTFC) is a large field of research in which many control algorithms are developed. In Section 2.2, a short description of each type of control method using Connected Vehicles (CVs) within this field will be given, including some examples of the developed algorithms with and without CVs. This will be used to get an overview of which controllers are applied for which types of traffic management and how CVs are used as actuators for traffic management in the literature. A wide range of characteristics of each algorithm will be described. The knowledge from this part of the review will also be a basis for deciding on the goals and methods of the controller that will be developed.

Another important aspect of this research is driver compliance. As the proposed application uses manually driven CVs as actuators, drivers are still in the loop. Therefore, it is important to study compliance rates to personalized advice given by traffic management through relevant communication methods. However, the operational response pattern of drivers is just as important for the proposed application. Operational response time to the given advice, as well as deceleration magnitude, can have impact on the effectivity of the control and is therefore important to take into account. Available literature on these factors is presented in Section 2.3.

Finally, Section 2.4 will combine the knowledge gathered in the literature review into an overview of building blocks for the controller design. Each block will be discussed along the conclusions from each section of the literature review. Gaps in the literature will be identified, which will be used as a basis for design concept choices made in Chapter 3: THEORETICAL DEVELOPMENT.

2.1 TRAFFIC BREAKDOWN

There are two classes of jams found on freeways. Firstly, stop-and-go-waves (or jam waves) are jams that move upstream over the freeway. Secondly, there are standing queues, which have a fixed head at an active bottleneck. In order for a standing queue jam to occur, there are three ingredients required (Treiber and Kesting, 2013):

- High traffic load
- A bottleneck
- A disturbance in traffic

The high traffic load is the most important factor, as there can be no queue without a sufficient number of cars to propagate the instability. The bottleneck is a local reduction in capacity, a weak point in the road. A bottleneck is required for a standing queue, but not necessary for a jam wave. Activation of a jam is caused by a disturbance in the traffic stream, which results in large speed differences. These disturbances can have many causes, such as a lane change, sudden braking, an accident or a merging vehicle.

2.1.1 Oscillations and hysteresis

A disturbance in traffic often results in a change in speed of a vehicle. Any car that follows the decelerating vehicle closely, will be required to decelerate as well to avoid a collision. Since the 1960's, it has been observed that the acceleration pattern after such a disturbance is asymmetric to the deceleration and is slower (Forbes, 1963; Newell, 1962; Treiterer and Myers, 1974). This delayed recovery of speed is called traffic hysteresis. Furthermore, this phenomenon can result in oscillations when traffic is dense enough. These oscillations propagate through traffic and can cause a capacity drop at a bottleneck. If the demand at the bottleneck remains higher than the queue discharge rate, the capacity of the bottleneck will not recover (Chen et al., 2014).

Oscillations are divided into four stages of development (Chen et al., 2014; Zheng et al., 2011):

- **Precursor:** Characterized by a low wave speed, as it is the first phase of development. Accelerations and decelerations are still close to zero result in local slow-and-go motions;
- **Growth:** This stage is also known as a jam wave, propagating through traffic in upstream direction with a wave speed of -18km/h. Minimum speed of vehicles in the wave (also called amplitude) slowly reduces as the wave propagates;
- **Stable:** This stage has the same characteristics as the previous stage. However, the amplitude is stable instead of growing;
- **Decay:** In this stage the amplitude diminishes as the wave propagates and traffic returns to free-flow.

As the goal in this research is preventing breakdown, the precursor stage and preventing the growth stage are important. Each cycle of hysteresis was found to be caused by local high density and flow, resulting in shock waves propagating upstream. During the precursor stage, drivers were found to generally show "clockwise" acceleration patterns, which are patterns with slower acceleration than deceleration. An example of this pattern is shown in Figure 2.1, where it can be seen that the acceleration is performed slower than deceleration. This often results in larger headways than prior to the disturbance, possibly resulting in upstream propagation of shock waves and thus creating jams. These findings show that it would be beneficial to prevent these local high flow and density clusters of vehicles from occurring or from passing through bottlenecks.

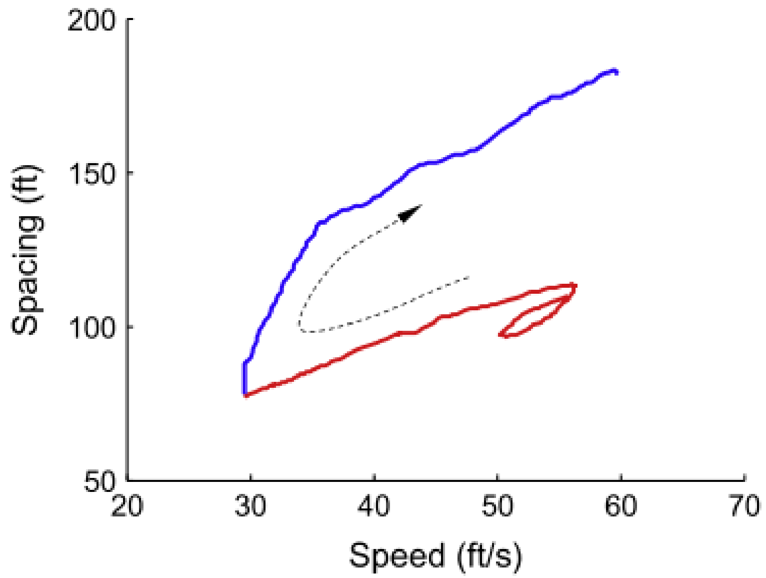


Figure 2.1: A clockwise, or a positive hysteresis loop. The red line indicates deceleration, blue indicates acceleration (Chen et al., 2014)

Effect of driver behaviour on hysteresis

There are two types of microscopic traffic hysteresis: positive and negative (Laval, 2011). Positive hysteresis has a slower acceleration after deceleration, while negative hysteresis has slower deceleration than acceleration within a hysteresis loop. In the study by Laval (2011), 66% of the loops were positive, against 14% being negative loops. The remaining percentage of curves showed no difference in acceleration and deceleration behaviour.

In the literature, positive and negative loops are explained by two types of driver behaviour: aggressive and timid driving respectively (Laval, 2011). Aggressive drivers are less willing to reduce their speed and therefore have a relaxation in their response. However, their response time is a lot faster when accelerating. When looking at timid drivers, the response is the direct opposite. Research by Chen et al. (2014) (with limited data) concluded that the transition from the precursor stage to the growth stage occurred when aggressive driver behaviour induced large hysteresis due to higher response times and shorter headways. It has been suggested by Deng and Zhang (2015) that this knowledge on differences in driving behaviour can be used to suppress hysteresis by balancing responses to oscillations.

2.1.2 Probability of breakdown

It is not possible to make predictions about the exact location and time of individual flow breakdowns (Treiber and Kesting, 2013). However, it is possible to give a probability that a breakdown will occur. Traffic breakdown is a stochastic phenomenon and can happen at a range of flow rates. Instead of having a fixed capacity, the maximum flow on a road is variable and dependent on many different factors. In early research by Brilon et al. (2005), it was investigated how the PB is distributed. In this research, the Product Limit Method was used to estimate a "survival function", which is based on statistical analysis of lifetime data (which was flow), where traffic breakdown counts as a failure event. Furthermore, current flow and the capacity of the road were used for calculation of the PB. When using aggregated data with intervals of 5 minutes, a Weibull-distribution is found for the PB. The same technique was applied to traffic density data. However, this resulted in larger variances, which can be attributed to the fact that the density data was artificially estimated with flow and speed data.

In response to this study, Chow et al. (2009) proposed an extended data-driven method to estimate PB. An important difference is the addition of occupancy to the probability calculations, since the same flow can have multiple densities when looking at a fundamental diagram. From this study, contour plots of the PB, as dependent variable of speed and occupancy, as shown in Figure 2.2. The resulting contour plots of the PB from the case study are however counter-intuitive, since they show a constant PB with increasing density and constant speed or the other way around, where you would expect it to still increase.

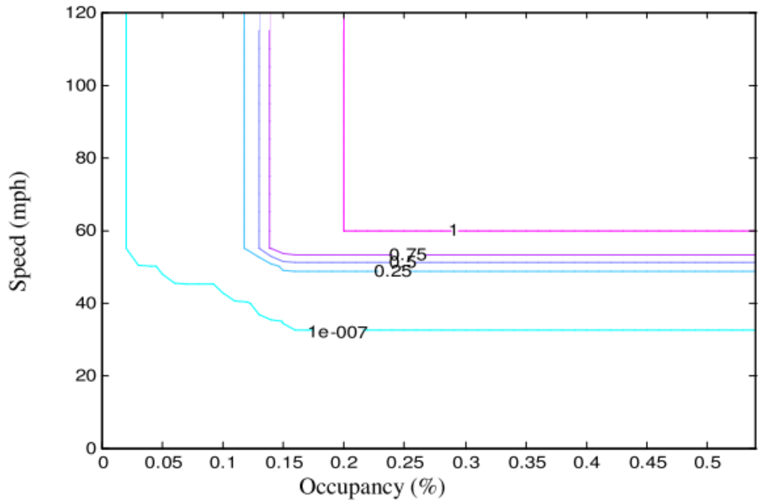


Figure 2.2: Contour plot of Breakdown Probability, dependant of average speed and occupancy, from Chow et al. (2009).

In a more recent study, which looked at a merging bottleneck, a probability function was estimated based on a range of critical factors, such as flow, headway distribution and the share of aggressive and timid drivers (Xu et al., 2013). From this study, an S-shaped distribution was found, where the PB increased with flow. Furthermore, the time headway distribution was shown to have a strong decreasing effect on PB. Also, the PB was lower for higher rates of aggressive drivers, which is in contradiction with the conclusions made by Chen et al. (2014) (Found in Section 2.1.1). This may relate to the definition of aggressive drivers, as Xu et al. defines aggressive driving as drivers with smaller preferred headway.

In research by Han and Ahn (2018), an approach to estimate PB based on observed headway was developed. The goal of the approach was to use the estimation for proactive control to prevent breakdown. Traffic data of a merging area was used. During the research, the same conclusion was drawn on the effect of time headway distribution, where more distribution in headways resulted in a higher PB. However, a more harmonized headway distribution resulted in a higher PB at near-capacity flows. This can be explained by the higher probability of an oscillation fading out due to a random large headway. Also, the same conclusion was drawn on the effect of aggressive acceleration and response time of drivers. Faster acceleration resulted in smaller disturbances in traffic and therefore a lower PB. Also, high merging speed was important in reducing the PB, as reduced speed difference between mainline flow and merging flow results in less severe disturbances.

2.1.3 Summary and discussion

Breakdown of traffic is a phenomenon that requires multiple ingredients to occur: high traffic load, a disturbance and often a bottleneck. When a disturbance (i.e. a lane change, a merging vehicle or sudden braking) occurs, shock waves move through traffic. Due to asymmetric acceleration and deceleration of drivers, local flow is temporarily reduced and a shock wave starts to propagate. When these waves start propagating upstream in traffic, while the density upstream of the disturbance is sufficiently high, a jam can occur. In order to prevent breakdown, early development stages of oscillations in traffic should be prevented or suppressed. Two approaches for this have been identified.

Firstly, Deng and Zhang (2015) concluded that balancing drivers responses to disturbances in traffic on an operational level could be beneficial, which would mean that all drivers would have to behave the same on a vehicle level. This is an unrealistic goal, since vehicle heterogeneity is hard to solve.

Secondly, reducing or preventing effects of disturbances by manipulating traffic characteristics such as flow, density and headway distribution, could be a solution for influencing the PB. However, it is hard, or even impossible, to predict disturbances in traffic. Although, a probability of traffic breakdown can be predicted, based on multiple traffic characteristics.

Traffic breakdown can occur at a wide range of flows and can be seen as stochastic. In the literature, it is found that the Probability of traffic Breakdown (PB) follows an S-shaped distribution in the probability-flow plane. In each of the studies mentioned, both flow as well as density on the road were found to be important for estimating the PB. However, the factor that seemed most important was the headway distribution. Clusters of vehicles with short headways have a larger probability of breaking down traffic. Therefore, it may be beneficial to harmonize headways of vehicles upstream of bottlenecks, to increase the probability that oscillations fade out. Furthermore, the PB can be used as a value that a controller reacts to (i.e. as a threshold or as an objective value), when it can be estimated in real-time.

2.2 TRAFFIC CONTROL ALGORITHMS

In the previous section, it was concluded that flow, density and headway distribution are important traffic characteristics that influence the probability of a breakdown. Traffic control is often based on measuring and influencing one or multiple of these factors. In this section, traffic control methods that try to manipulate these factors are described in order to get an overview of what is currently available for a CV environment.

Multiple types of controllers were found in the literature. In the following paragraphs, a distinction will be made between flow-based, density-based and breakdown probability-based controllers. Within this distinction, controllers are discussed within three categories (when available in the literature): feed-forward, feedback and Model Predictive Control (MPC) controllers. The controller types will be described briefly below.

Feed-forward

Feed-forward control is also called open-loop control. This type of control uses a measurement as input for control, the effect of the applied control can not be measured by the controller and therefore it cannot respond to its own mistakes, which is a major disadvantage.

Feedback

Another common control methodology is feedback control (or closed-loop control). This type of control uses measurements of the system state to determine the control output. In this case, control is applied upstream of the location of measurement. Subsequent measurements of the system state then fed back to the controller, after which the controller computes a new control signal.

Model Predictive Control

MPC is a process control method, which uses a mathematical prediction model of the system to compute the optimal control over a chosen (rolling) time horizon. A schematic view of MPC is shown in Figure 2.3. As can be seen, the model requires state measurements, such as current traffic demand, as input, which it uses as input of the model to predict future states. An objective function has to be determined, of which the goal is to minimize or maximize the expected value over the chosen time horizon. This objective function often contains factors such as Total Travel Time (TTT) of the system.

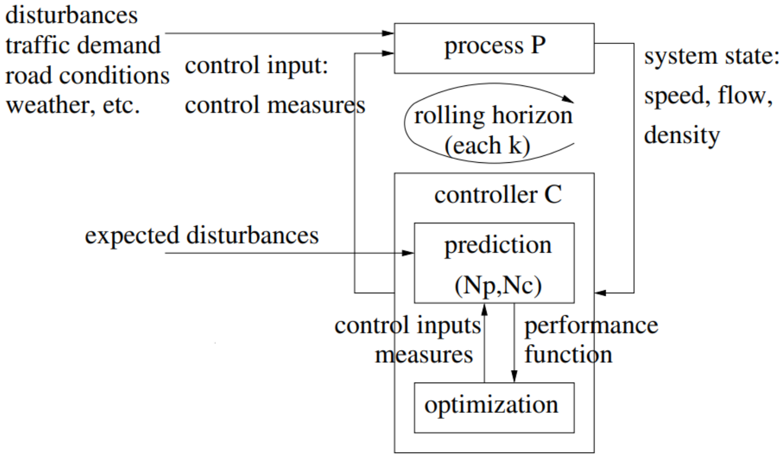


Figure 2.3: Conceptual scheme of an MPC controller, from (Knoop et al., 2019)

In the following sections, control measures that use the aforementioned traffic characteristics are described. This is used as an overview of already available techniques that may be of use in the final controller design. In order to get an overview, a general description of the concept is given, as well as the corresponding controller type, input requirements, actuators and computation methods. Also, driver types and the number of lanes the controller is designed for, will be mentioned.

In the conclusion of this section, an overview will be given of all these characteristics of the available controllers and conclusions will be drawn on what is missing and what can be used for the controller design.

2.2.1 Flow based controllers

In this section, controllers from the literature which are based on flow measurements are described. The focus of the description is on the aforementioned characteristics. A distinction is made based on the controller type.

Feed-forward

Feed-forward control has been used successfully in combination with shock wave theory. The SPECIALIST algorithm developed by Hegyi et al. (2008) is used to resolve jam waves on freeways faster than they do naturally. When a jam wave is detected through a combination of flow and speed measurements, required actuation is computed using shock wave theory. Then, Variable Speed Limit (VSL) is used as an actuator and communicated through overhead gantries. By reducing speed limits upstream, inflow into the jam is reduced and the jam is resolved earlier. The algorithm was later tested in a real-world pilot (Hegyi and Hoogendoorn, 2010). The evaluation concluded that 80% of the resolvable jam waves were resolved in practice, which resulted in a large reduction of delays.

Based on the same theory, an algorithm within the CV environment was developed by Hegyi et al. (2013), which used connected vehicles, as well as Video-Based Monitoring and Floating Car Data to measure jams faster. This resulted in improvements for jam resolvement of the algorithm in space, as well as in time.

SPECIALIST was developed even further for use in a CV environment by Wang et al. (2016). While the algorithm was kept the same, CVs were used as actuators for the speed limits, which were communicated from the controller to the vehicle as an addition to the communication through Variable Message Signs (VMSs). However, this algorithm used automated CVs, which were controlled by an MPC controller, which optimized the vehicles behaviour to the local surroundings. Simulations showed that using equipped vehicles resulted in faster resolvement of jam waves than with human drivers. However, results showed that without control and high penetration of CVs, the total time spent was reduced to similar levels, scenarios with control were slightly better. Adjusting the algorithm to use of CVs might have more beneficial effects.

In research by Han et al. (2017), a CV environment was used for VSL in order to enhance traffic stability. Three strategies were developed:

1. a single CV (per lane) slowing down to regulate flow to the bottleneck capacity when a queue is detected (by either a CV or other measurements);
2. a single CV (per lane) in combination with VMSs upstream of the bottleneck;
3. multiple CVs to create voids that suppress multiple shock waves.

For these applications, shock wave theory is used to calculate control. CVs are used for both queue detection and control actuation. In each of the three scenarios, the CVs are used to create voids, giving the queue time to recover from the capacity drop. Since voids are created, the methods are not fit for multi-lane scenarios, unless there are enough CVs to cover each lane, as overtaking and lane changing may cause problems for the effectiveness of the algorithm. The three strategies were extended to minimize expected delay. To do this, a relationship between the probability of success of control (based on control speed and number of vehicles in congested state) and the expected maximum delay saving was used. Expected delay saving is estimated for both control success and failure, of which the sum is optimized. Both strategy 2 and 3 were found to have strong positive impact on throughput. However, strategy 3 is very reliant on penetration rate of CVs.

This paper offers some ideas for using CVs as actuators for control. However, the measures are still reactive and reliant on fast queue detection. Also, a lot of parameters, such as the probability of instability, are hard to estimate, but are important for the effectivity of the algorithm.

Feedback

In a paper by Hegyi (2013), COSCAL was developed, based on the same theory as the aforementioned SPECIALIST algorithm. COSCAL is based on shock wave theory and uses CVs as actuators for the VSL control for resolution of jam waves. A big difference with the SPECIALIST algorithm is the increase in the span of control. Instead of actuating VSLs through only gantries, the speed limits are communicated to the vehicles, which gives the possibility of actuating the limits dynamically in space and time. This can be seen by looking at Figure 2.4, where the borders between areas are not straight lines. Also, speed measurements of CVs are used for jam wave detection. In COSCAL, the jam resolution area is used to resolve the jam wave by inducing speed limits, while the stabilization area uses the expanded span of control to guide vehicles in the area to a stable target density, after which a speed release is performed to transition back to a free-flow state. The line between the two areas is determined by calculating the position of the jam-resolving vehicle, based on the number of vehicles on the road and the outflow of the queue. The algorithm updates its control signals when a vehicle decelerates faster than expected. This leads to a larger headway than expected, which is a stability risk, as following vehicles will take a very short headway. Therefore, the algorithm computes new target lines for the following vehicles.

COSCAL was later extended to take merging flow from a Ramp Metering (RM) system into account as well (Mahajan, 2014). When the jam resolution or stabilization area passes a merging area, incoming flow from the ramp is used for computing the required space for resolution and stabilization as well. This extension improved the resolving of jams in both space and time. Disadvantages of COSCAL are that it is designed for only one lane and requires a high penetration rate of CVs.

MPC

In more recent work by Roncoli et al. (2015a,b), MPC is applied to a CV environment. A central decision maker is assumed to know the full traffic state due to these CVs. Furthermore, the CVs are used as actuators to optimize the distribution of vehicles over lanes by giving

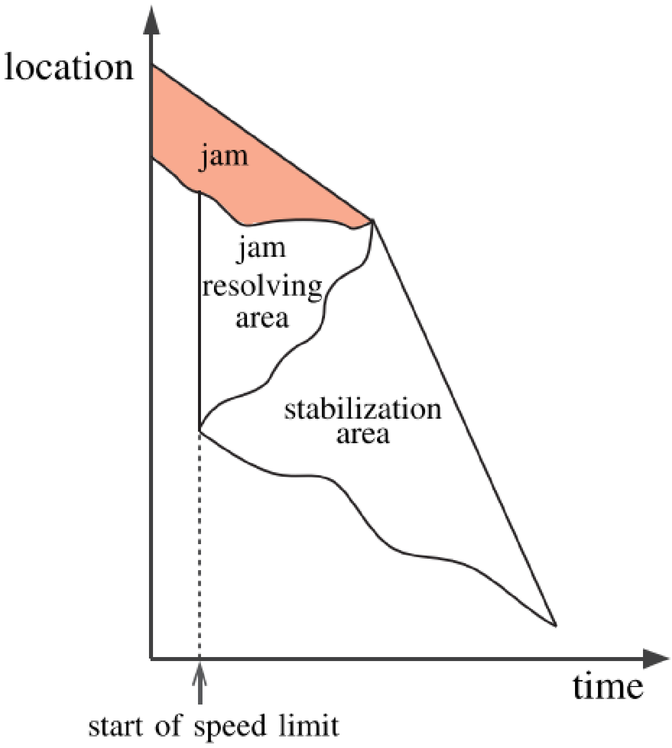


Figure 2.4: Schematic view of different space-time areas of the COSCAL algorithm, from Mahajan et al. (2015)

lane changing advice, as well as for actuating VSLs. A Cell-Transmission-Model (CTM) was extended to account for multiple lanes and lateral and longitudinal flows in order to make it possible to apply it in an optimal control scheme. Later, this system was subdivided into different layers, where the computed optimal control by the central decision maker is used as input for a local control layer, which in turn optimizes VSL, RM and lane changes for a smaller space and time-scale (Roncoli et al., 2016). This control methodology was later evaluated with a microscopic simulation (Perraki et al., 2018). Different penetration rates of CVs were simulated. The simulations revealed that with current demand patterns, the control methodology could reduce or even completely remove congestion in the network for current demand patterns, depending on the penetration rate of CVs.

2.2.2 Density based controllers

In this section, controllers from the literature which are based on density measurements are described. The focus of the description is on the aforementioned characteristics. A distinction is made based on the controller type.

Feed-forward

Feed-forward control was also proposed as control for a cooperative merging assistant by (Scarinci et al., 2017). CVs are used to create gaps on upstream of the merging area, while coordinating with green-times of a RM system. This makes it easier for merging vehicles to find gaps and therefore reduces the occurrence of late-merging and other disturbances. Furthermore, traffic density is taken into account as well, as higher densities result in lower speed differences when creating gaps, as well as shorter gaps as the mainline capacity is required. Through simulation, potential benefits were shown for the capacity of merging areas.

Feedback

Another example of (proportional) feedback control is the control methodology developed by Goñi-Ros et al. (2014). A density-based feedback mechanism with VSL was applied near a sag bottleneck in order to prevent the capacity drop by reducing inflow into the bottleneck when the vehicle density at the sag reaches a determined critical value. When the density at the transition point of the sag became higher than a threshold density, speed limits, which were proportional to the difference between the target density and the current measured density, were actuated through overhead gantries at the locations shown in Figure 2.5. A simulation study concluded that the proposed control resulted in a reduction in delays in the system.

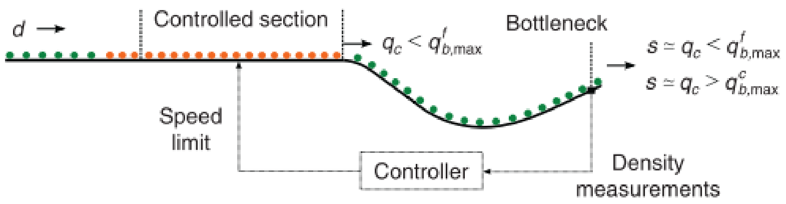


Figure 2.5: Using VSL at sags to reduce bottleneck inflow, from Goñi-Ros et al. (2014).

The idea of VSL upstream of sags to improve throughput was taken into a CV environment by Nezafat et al. (2018). Instead of communicating the speed limits through gantries, they were communicated to CVs in the control section only. The algorithm computes a speed limit for CVs depending on the difference between a calibrated target density on the uphill section and the current density in that section, times a scale parameter, which is very similar to the previously mentioned work by Goñi-Ros et al. (2014). Simulations showed strong reductions in delay of up to near 50%, depending on penetration rate. However, low penetration rates were found to be effective as well. For example, 10% CVs resulted in an average of 40% reduction of delays. The driver compliance was assumed to be 100%, meaning that applying this system in a real situation would be less effective. Also, using this system in reality leads to intentional speed differences at a sag, while the driver model used assumes homogeneous drivers. When driver heterogeneity is taken into account, these speed differences can result in more oscillations and thus more congestion.

Another paper by Grumert and Tapani (2018) suggests to calculate VSLs, based on density measurements at the bottleneck. The speed limits are proportional to the difference between the current and the critical density at the bottleneck. The study uses CVs, as well as VMS for actuation of the speed limits. The algorithm tries to induce controlled congestion upstream of the jam wave and releases the speed limit upstream of the jam wave as well. This limits the resulting density when traffic reaches the bottleneck. Simulations of the algorithm resulted in a 40% decrease in Total Time Spent (TTS). The study criticises itself, since it induces congestion to reduce delays downstream. Drivers may not comply to the speed limits for this reason.

MPC

Another application of MPC was developed by Goñi-Ros et al. (2016), as a continuation of the feedback control algorithm from Goñi-Ros et al. (2014). However, instead of slowing down vehicles upstream of the sag, CVs are used as actuators in an optimal control algorithm which optimizes longitudinal control of the CVs. The behaviour of the optimized control was analyzed and this resulted in the conclusion that CVs should be used to dampen the shock waves that are formed at the sag bottleneck by gradually slowing down and creating voids. These voids are then used to fade out any oscillations created by the vehicles at the sag. When this is not possible, CVs upstream of the sag are used in the same way to dampen the shock waves.

2.2.3 Probability of breakdown based controllers

As mentioned in Section 2.1, researchers have attempted to measure the probability of breakdown. However, only few studies use it for traffic management, since it is a complex concept. One example, using CVs was found and described below.

In a paper by Han and Ahn (2019), a concept based on increases in speed at bottlenecks is developed. The concept uses Connected Automated Vehicles (CAVs) for headway measurements and as actuators. The model estimates the intensity of a disturbance due to a merging vehicle and uses headway measurements of the CAVs to calculate a breakdown probability. This probability is used as input for control, which uses the Variable Speed Release (VSR) concept. Speeds near a bottleneck are increased in order to increase maximum throughput. Even though the authors simulate multiple penetration rates of CAVs, increasing speeds near bottlenecks may cause serious safety issues when manual drivers are involved. Especially at merging areas, since merging vehicles can not always reach high speeds. Therefore, this may work in theory, but probably not in practice.

2.2.4 Summary and discussion

In this section, multiple types of controllers were described, including some examples. An overview of the controllers which use CVs is given in Table 2.1. In the table, the characteristics of each controller are described in short. Each of the three mentioned controller types is represented in the CV environment. Also, multiple types of bottlenecks are represented: regular, sag and merge bottlenecks. This overview can be used as a reference grid when designing a controller, as it gives a short summary of the developed techniques which can be used for the controller that will be designed later.

From the literature it can be found that some applications of CVs for traffic management are based on creating voids to suppress shock waves in downstream traffic, or create a gap for merging vehicles (Goñi-Ros et al., 2016; Han et al., 2017; Scarinci et al., 2017). While this method may seem promising, it is still based on the assumption that no lane changing into the void will occur. When a lane change into the created void does occur, while the algorithm does not account for this, the shock wave does not have enough space to stabilize. This may result in the jam wave not resolving, but propagating to the next platoon. In busy traffic, which is often the condition for which these methods are designed, the assumption that no lane-changing will occur can not be guaranteed. This is a problem that has to be solved when the creation of voids is used in the design of the controller.

Another drawback of most of the applications, is the limitation of using only CVs for actuation of control, completely discarding the current infrastructure. Since the majority of the applications are aimed at the transition period to CVs, current infrastructure, such as VMSs will still be present and can be used for traffic control. In Perraki et al. (2018), an argument is made that control with small penetration rates of CVs will already be effective. However, the results do not directly support this argument, as only three levels of penetration (20%, 50% and 100%, which are not low) are simulated and compared to a no-control case (which is not a strong comparison, as their research is on integration of multiple control techniques). Besides, lower penetration rates seem to be very effective in the research by Wang et al. (2016), as a penetration level of 5% of CVs already showed a significant impact on TTT in simulations. However, it is mentioned that development of an algorithm that is based on using both types of actuators is expected to be more beneficial.

As mentioned in Section 2.1.1, variations in acceleration and deceleration patterns of drivers are one of the causes of growing hysteresis. However, all studies assume homogeneous driver behaviour. Especially in the cases where CVs are used to create voids, a mix of aggressive and timid patterns in speed adaptation of cars that follow this CV can be triggers of jam waves. Especially the control developed by Goñi-Ros et al. (2016), which is based on multiple deceleration and acceleration steps, might have negative effects in real-world application.

Only few studies use the PB for traffic management, even though studies into the breakdown probability all have this application in mind. This may indicate that traffic management developers do not think the estimations are relevant enough yet and thus this has to be developed further. The algorithm from Han and Ahn (2019), which is discussed in this section, uses the headways of all vehicles for a probability, while the papers mentioned in Section 2.1 mainly use flow and density. If the designed controller will be based on this probability, a combination of these techniques could be made.

All the described algorithms have a broad variety of ways to calculate control. The overview will be helpful during the design of the controller, as the acquired knowledge can be used as a reference when developing the mathematical formulation of the controller. If similarities in goals are found between the papers and the design, similar calculations could be used, preventing the need for reinventing the wheel.

2.3 DRIVER COMPLIANCE TO IN-CAR ITS MESSAGES

Since CVs are still operated by human drivers, advice given through in-car messages can be partly followed or even completely ignored. Low compliance rates reduce the effectiveness of Cooperative Intelligent Transportation Systems (C-ITS), or can even result in negative effects on flow (Schakel and Van Arem, 2014). Therefore, it is important to take this factor into account. Research into compliance of drivers can be divided into two groups: driver simulator studies and real-life tests. These groups are both described in the following sections.

2.3.1 Driver simulator studies

Early research by Risto (2014) used a driving simulator, as well as a real-life test with cooperative in-vehicle advice to explore responses of drivers. During the driver simulator study, compliance was measured as dependent of amount of information and estimated penetration rate of the driver, for three situations: near a lane-drop, near an on-ramp and during normal driving. The estimated penetration rate of the driver, is the percentage of other vehicles near him on the road, that a driver thinks are equipped with CV technology as well. Results of these experiments found that the amount of information had a large impact on compliance rates (from an average of 40% for low to 22% for high amount of information). Also, it was found that the perceived compliance rate of other drivers affected the compliance of drivers in the simulation, which was worse for the group with high levels of information. This was assumed to be a result of higher expectations in visibility of compliance due to higher information levels.

In more recent research by Sharma et al. (2019), a driver simulator was performed on a car-following scenario in a connected environment. In this experiment, drivers were presented with additional information streams. A distinction is made between continuous information (leading vehicle speed and spacing) and event-triggered information (leader braking warning). During the test, drivers would receive a notification telling that the leading vehicle would brake. This notification was given 3 seconds in advance of the braking. It was concluded that in this scenario, compliance increased for shorter headways as perceived risk is higher. After calibration of the developed car-following model, response time to the message was found to be 0.2 seconds.

The continuous information stream informed the driver of the current spacing with the leading vehicle, as well as the speed of that vehicle. The study found that this information resulted in stable acceleration and deceleration, low fluctuations in speed, and larger, but more less varying headways. Another driver simulator study by Saffarian et al. (2013), which used a similar system, resulted in the same conclusions, showing that these kind of systems can support traffic management.

2.3.2 Real-life tests

During the real-life test by Risto (2014), a think aloud protocol was used to understand the thought process of drivers. In many cases, a serious distrust in the system was shown, as the drivers often did not agree with the advice. Therefore, the advice was often followed by non-compliance. The system gave multiple types of notifications, such as headway advice or oncoming jam warnings. Since the system used Floating Car Data (FCD), the information delay and underdevelopment of the system might have been the reason for non-compliance of drivers. Another important remark made by Risto was that drivers were unable to estimate their gap size during the experiments which gave gap advice, which resulted in lower compliance rates.

In an extensive study by Van der Pas et al. (2017), compliance to ITS messages was analysed by looking into vehicle data. The data was a result of an experiment with two mobility applications: FlowPatrol and ZOOF. These are two mobile phone applications with the goal of damping jam waves on the A58 in the Netherlands. Both apps use the same loop data, as well as FCD to estimate traffic states on the freeway. These traffic states are used to compute required VSLs to reduce the effects of shock waves on the road. However, the difference is that the apps give speed advice in different ways. Both systems were tested on the A58 in the Netherlands and used messages through an app on the phone of the driver. Both ZOOF and FlowPatrol used FCD and loop-data from third parties to compute their speed advice. The difference between the two services was the type of advice. While FlowPatrol gave ”soft advice”, consisting of messages like ”slow down” or ”speed up”, ZOOF used ”hard advice” by giving exact speed advice in km/h. The penetration rate during the pilot was between 1% and 2.5%.

Compliance of drivers was defined as adapting speed into the desired direction in the case of FlowPatrol and adapting speed to the advised number in the case of ZOOF. A time window of 20 seconds after advice was given was analysed. Time windows that (1) changed speed before the advice came in, (2) showed a much larger speed difference than required, or (3) were not during rush hours, were left out of the data analysis.

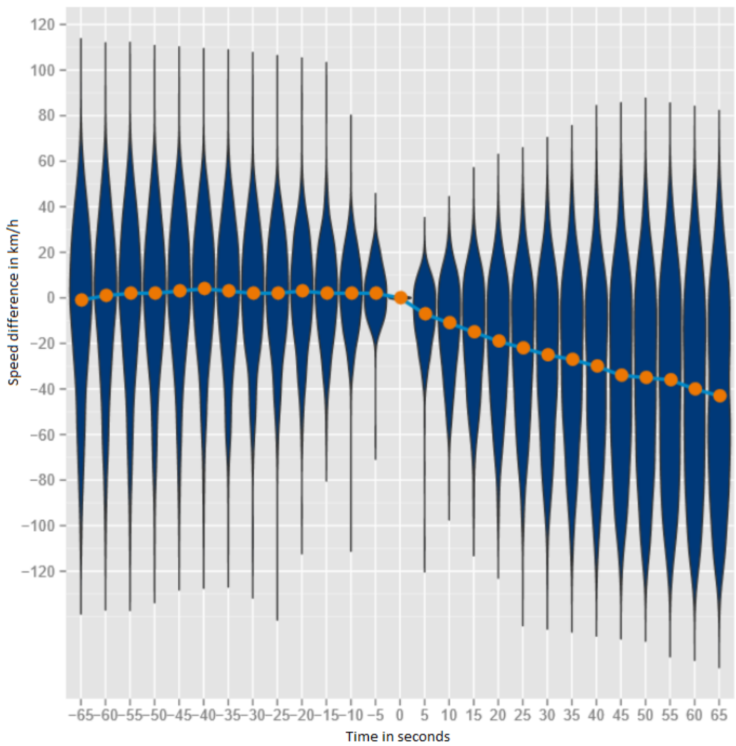


Figure 2.6: Response curve for ZOOF deceleration advice. The orange dot presents the median, the blue area presents the variance. t = 0 is the time of giving speed advice Van der Pas et al. (2017)

Compliance rates were different between the two types of messages. FlowPatrol saw higher compliance rates to the advice given than ZOOF (50% and 40% respectively). However, in case of ZOOF, drivers reduced their speeds by a larger margin on average (although there was a larger variance). Figure 2.6 presents the average speed changes of drivers before and after getting advice from the ZOOF application. From this picture, it can be seen that the average speed is severely reduced after advice is given. If these changes in speed can purely be related to the advice is unknown. In some cases, surrounding traffic could be a reason for slowing down as well.

Also, in many cases, drivers adapted their speed even before the advice was given. Which may be due to the car joining the jam before the advice arrived in the car. This may have affected the results of the research into compliance rates. However, during the test period, users of the services were questioned about their experience multiple times. The results of these surveys show that users had an increasing perception of effectivity of the algorithm on throughput, which can be accounted to significant improvements to the system. There has not been a study on compliance after the latest improvements to the system. However, compliance is expected increase further after these improvements.

2.3.3 Summary and discussion

It is obvious that drivers have to comply to operational advice from traffic management to make it effective, especially for microscopic control of traffic management, such as the creation of voids to dampen shockwaves or the controller that is developed in this thesis. As the effect is not equal to penetration rate of CVs, since advice can be partly complied to, both factors should be taken into account separately. There are multiple factors that play a role in the reason for compliance or non-compliance of a driver. Firstly, the amount of information presented to the driver should be low, as this increases compliance. Secondly, the perception of compliance rates of other drivers should be high. This seems to be related to the first factor, where more information can lead to lower perception of compliance of other drivers. Thirdly, the driver should trust the system and perceive its effectiveness, as they will not comply if they think advice is not beneficial for them or others.

Since the perception of penetration rate of CVs among other drivers seems to be important for compliance, the statements made in Section 2.2 about low penetration rates being effective for improvements in traffic flow, might not be true outside of simulations, as drivers do not feel like their compliance would have any impact when no other drivers seem to reduce speeds. This means that in times of lower penetration rates, more information sources than only in-vehicle messages should be used to provide more (or all) drivers with information on speed limits.

It was also shown that drivers can not estimate their headway accurately without support of technology, which led to non-compliance, while the driver was meaning to comply. Concluding from multiple studies into compliance it can be seen that there is no specific value that can predict the rate of compliance, as it is reliant on multiple factors.

2.4 FINDINGS AND DISCUSSION OF THE LITERATURE REVIEW

In this chapter, the sub-questions defined in Section 1.3.1 are answered through a literature review. The information that was gathered is used to define the control concept developed in this thesis. In the following paragraphs, important findings from the literature for the design of the controller are discussed. This will be done by following the same structure as the sub-questions, the first two sub-questions are discussed together as they are closely related.

2.4.1 Traffic breakdown

From the review of traffic breakdown literature, it was found that disruptions in traffic due to deceleration can lead to shock waves propagating through traffic when traffic load is high enough. When a shock wave occurs, it moves in the downstream direction. However, when the shock wave does not fade out, it may start moving in the upstream direction, which activates the capacity drop. This is a result of asymmetry between the deceleration and acceleration of drivers in response to vehicles slowing down within these shock waves and leads to locally reduced capacity of the road. An important point here, is that the shock waves should be prevented from propagating into the upstream direction, as it is harder to resolve a jam wave when capacity is reduced. The controller developed in this thesis should aim to prevent disturbances or dampen the effects of these disturbances.

It is also found that flow, density and headway distribution are predictors of traffic breakdown. Since it is deemed impossible with current technology to perfectly predict a disturbance in real-time, either of the aforementioned factors could be used as a predictor for breakdown. It was found that controlling the flow by attempting to manipulate density and harmonize headways of vehicles may lead to a promising solution.

The discussed algorithms use multiple types of measurement methods: loop-detectors, video-based monitoring, floating car data or CV sensor data. Depending on the requirements of the algorithm, each of the measurement methods has its strengths and weaknesses.

For the algorithms that rely on queue detection, speed is often used as trigger for control, while for flow restricting algorithms, either flow or density (with a preference for density) is used in combination with a calibrated critical value. The difference between current measurements and the critical value is then used as input for control.

While speed, flow and density are used often, the probability of breakdown is barely used as an input for control in the literature. There is no real consensus on what factors should be used for computation of the probability. Also, real-time estimation requires high-resolution data, which is often not available.

The choice of which real-time measurements should be used is highly dependent on the controller that will be developed. Therefore, this choice is made during the development phase of the controller.

2.4.2 Traffic control algorithms

In the literature, research was conducted to reduce congestion while using CVs in multiple ways. There are many algorithms that try to resolve jam waves, applied to multiple types of bottlenecks, or without the presence of a bottleneck. CVs (Van de Weg et al., 2014) and VMSs (Wang et al., 2016), as well as a combination of them (Han et al., 2017) are all used as actuators in these algorithms to either reduce flow in the first two or create voids to dampen jam waves in the last example. In some, individual cars are controlled to achieve the desired effects (Han et al., 2017; Van de Weg et al., 2014). The methods of actuation and methods to reduce the effects of jam waves from these controllers are shown to be effective. Therefore, the controller will be based on the use of both the flow reduction and the void creation mechanisms to find if the combination can benefit traffic.

Three main approaches for computation of control are found, which are predicting and manipulating traffic states through shock wave theory, proportional feedback based on the fundamental diagram and optimization of actuation through MPC. Since each of these methods show good results, the choice of computation of control will be made later in the report and will depend on the method of actuation used.

2.4.3 Driver compliance to in-car messages

There is not much literature available on the compliance of drivers to in-car control signals. However, some lessons are taken from the literature that was found on compliance. Firstly, an increased amount of information presented to a driver reduces compliance, therefore messages should contain little information and should be easy to follow up on. Secondly, compliance seems to be affected by the perceived rate of compliance of surrounding drivers. Thus, it is important that the information is given to as many vehicles as possible through multiple channels of communication. Thirdly, presenting drivers with a continuous stream of information leads to a high workload and may affect both safety and compliance. This strengthens the fact that the control messages should be easy to comply to. Fourthly, drivers should trust the control system, as it is shown that mistakes in control messages severely reduce compliance for the current signal, but also for future control signals. Lastly, a specific value for compliance is unknown, since it is not widely tested. Therefore, the effects of non-compliance are evaluated in this report in order to be able to account for the effects when more information has been gathered.

Table 2.1: Overview of control algorithms with CVs from the literature, in the same order as mentioned in this section

Paper	Controller type	Computation	Input	Concept	Actuators	#lanes	Drivertypes	Bottleneck type
Wang et al. (2016)	feed-forward	Shockwave theory	Queue detection	Reduce inflow into a jam wave in order to resolve it earlier	VMS, CVs	Any	Homogeneous	Jam wave
Han et al. (2017)	Feed-forward	Shockwave theory	Queue detection + CV location	1. a single CV (per lane) slowing down create a void for shock wave suppression	CVs	1	Homogeneous	Jam wave
	Feed-forward	Shockwave theory	Queue detection + CV location	2. Using single CV in combination with VMS	VMS, CVs	1	Homogeneous	Jam wave
	Feed-forward	Shockwave theory	Queue detection + CV location	3. Creating voids with multiple CVs to mitigate shock waves	CVs	1	Homogeneous	Jam wave
	MPC	Expected delays from control success or failure	Queue detection + CV location	4. Calculating maximum expected delay savings with probabilities	CVs	1	Homogeneous	Jam wave
Hegyi et al. (2013)	Feedback	Shockwave theory	CV sensors	Reduce inflow into a jam wave in order to resolve it earlier	CVs / CAVs	1	Automated	Jam wave
Mahajan (2014)	Feedback	Shockwave theory	On-ramp flow, CV sensors	Reduce inflow into a jam wave in order to resolve it earlier, including on-ramp flow	CVs / CAVs	1	Automated	Jam wave / merge
Perraki et al. (2018)	MPC	Quadratic	CV sensors, loop detectors	Global optimal control for all control measures combined, including VSL, RM and LCC	CVs / CAVs, VMS, RM	Any	Homogeneous	Multiple
Scarinci et al. (2017)	feed-forward	Shockwave theory + Analytical	Vehicle speed and position, on-ramp demand	Create gaps in collaboration with RM to synchronize gaps with merging vehicles	CVs / CAVs	Any	Homogeneous	Merge
Nezafat et al. (2018)	feedback	Analytical	Density measurements from loops	Speed limits for CVs to limit flow into the sag	CVs / CAVs	Any	Homogeneous	Sag
Grumert et al. (2018)	Feedback	Analytical	Density measurements	Proportional VSL for CVs to adjust speed limit, in combination with acc/dec areas	VMS, CVs	Any	Multiple types of vehicles	Regular
Goñi-Ros et al. (2016)	MPC	Nonlinear	Vehicle speed and position, gradient of the road	Induce optimal acceleration control of CVs at the sag in order to minimize total travel time	CVs / CAVs	1	Homogeneous	Sag
Han et al. (2019)	Feed-forward	Shockwave Theory	Breakdown probability	Using Variable Speed Release, which increases speed near bottlenecks to increase flow	CAVs	Any	Automated	Multiple

3 THEORETICAL DEVELOPMENT

In order to develop the controller, some steps of development will be taken. Knoop et al. (2019) describes a methodology for development of a traffic controller which is used as a baseline for the development method. This method consists of multiple complementary steps:

- A problem analysis which clearly describes the problem to be solved;
- A selection of a control approach;
- The development of the theory for the controller in traffic engineering terms, where knowledge gained in the literature review will be used;
- The translation of the developed traffic engineering theory into a control engineering problem.

These steps will be performed in this chapter. Knowledge gained in Chapter 2 in combination with the aforementioned steps will be used for the development. The two complementary steps of theory development are taken in sequence.

Firstly the traffic engineering approach, which is combined with the problem analysis, where the concept is explained in terms that relate to traffic flow theory. Undesired traffic events and the actions required to manipulate these events into desired behaviour are discussed. Shock wave theory will be used to validate and visualize the control concept. Furthermore, required measurements, operational conditions and problems due to delays in communication will be discussed.

Secondly, the control engineering approach, where the traffic engineering theory is translated into a control problem. In the control engineering approach, the control approach is chosen, mathematical relations are formulated and multiple aspects of the controller formulation will be described.

3.1 TRAFFIC ENGINEERING APPROACH

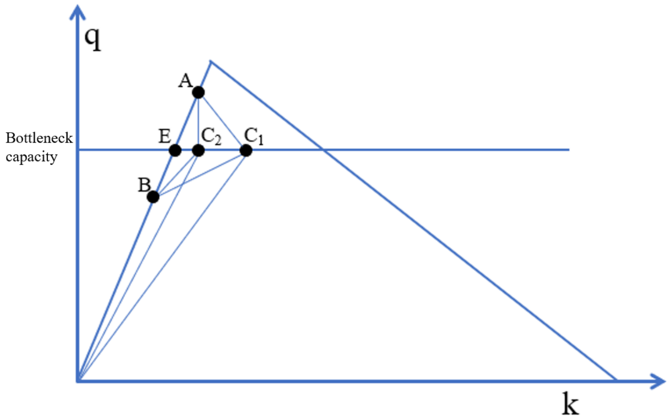
In this problem, the goal of the controller will be to improve bottleneck throughput by preventing the capacity drop. When traffic demand is higher than the bottleneck capacity, disturbances can lead to a breakdown and capacity drop. A disturbance at a bottleneck is often a reduction in speed due to the driver of the car being uncomfortable with their current speed or headway at the bottleneck. This reduction in speed leads to oscillations in traffic, as followers will have to reduce their speed as well. This behaviour increases the risk of causing a breakdown. Due to traffic hysteresis, capacity is reduced at the location of breakdown, as well as at the jam it induces.

By applying control upstream of the bottleneck, short periods of high traffic demand (in free flow) can be spread out over space and time. By keeping the flow at the bottleneck near to, but below its capacity flow, breakdown probability is reduced. Therefore, due to avoiding the capacity drop, average flow will increase.

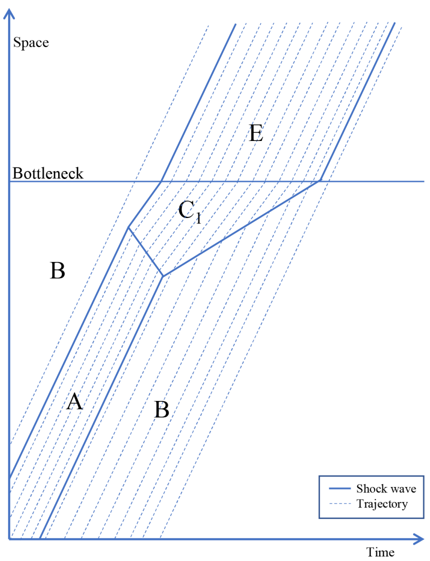
This is achieved by sending speed control signals to CVs. Since positions of CVs are always known and they can be controlled at high accuracy, clusters of vehicles can be guided into a desired state by slowing them down in a controlled way. The concept will be elaborated on further in the following sections.

3.1.1 Macroscopic theory of the concept

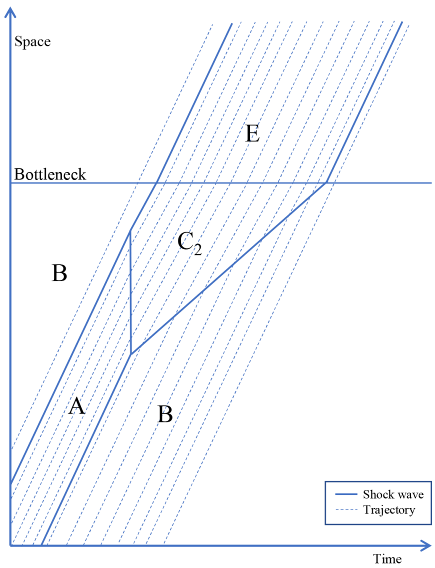
The concept is visualized with a space-time diagram using shock wave theory in Figure 3.1b-c, where the vertical axis represents the location on a freeway with the origin being the most upstream point on the road and the horizontal axis represents time. For the example, a triangular Fundamental Diagram (FD) is assumed with flow on the vertical axis and density on the horizontal axis. The FD can be found in Figure 3.1a and the required states will be discussed below. In the FD, multiple traffic states which relate to the controller are defined. In the x-t diagram, the propagation of the traffic states defined in Figure 3.1a are shown in space and time in such a way that the areas are labelled with the traffic states from the FD. Therefore, when a traffic state is mentioned, refer to the FD and when an area is mentioned, refer to the x-t diagram.



(a) Fundemental Diagram for the control example in the flow-density plane



(b) Starting point of control moving in the upstream direction, including trajectories of vehicles



(c) Starting point of control at an equal point in time, including trajectories of vehicles

Figure 3.1: Two control examples with 100% penetration rate for CVs and the FD used to draw them.

In the FD, multiple traffic states that are important for the example are defined. Firstly, traffic state A which is the traffic state with a too high flow to pass the bottleneck in a stable manner and should therefore always be upstream of the bottleneck. Secondly, state B is a traffic state with lower flow, lower density and speed around the free-flow speed. Thirdly, traffic state C is the target state that is close to the capacity flow of the bottleneck. This state can be chosen specifically for any bottleneck, but is required to have lower flow than state A. For the sake of illustration, two states C (C_1 and C_2) with different control results are shown, the differences between these states will be elaborated on in the following paragraph. Fourthly, traffic state E is the resulting traffic state when control is released and drivers return to their desired speeds.

The propagation of the fronts between the areas in the x-t diagram depends on the chosen target state. For example, in Figure 3.1c, a target state C_2 is chosen with equal density, but lower speed than traffic state A. In the x-t diagram, this results in onset of control for all vehicles in area A around the same time. When a target state with a higher density than state A is chosen, onset of control is spread out and the front moves over space and time in upstream direction, an example of this can be seen in Figure 3.1b. This means that vehicles downstream are controlled first. Furthermore, a traffic state which is on the left of traffic state A can be chosen as well. This results in the onset of control moving in the downstream direction and reduces the density in the area A. However, choosing state C in this way largely increases the complexity of the algorithm and makes the algorithm less flexible for extension to a multi-lane scenario, as a reduction in density may lead to overtaking and lane changing, countering the effects of the algorithm. For that reason, this choice for state C is not implemented into the algorithm.

The concept of this controller does not change much when a different FD is assumed. For example, in case of a inverse-lambda shaped FD, as shown in Figure 3.2, the speeds within the areas A, B and E are slightly different Wu (2002). Higher flow states are then assumed to have lower speeds than lower flow states. This results in areas not moving with the speed of traffic, but with the characteristic wave speed between the two states. Inefficiencies can arise due to a wrong assumption of the FD when the location of measurement is far from the control area, since vehicles that were in the high flow area at the location of measurement could be far away from the actual high flow area, as their speed is faster than the area itself. Therefore, it is important that measurements are made close to the bottleneck, as then there is no estimation required of which vehicles are actually in the high flow area. When this is not possible, the characteristic wave speed should be taken into account when deciding which vehicles require control. Furthermore, since this controller aims to keep traffic within free-flow conditions, a difference in the congested branch of the FD should not make a difference in the dynamics of the controller.

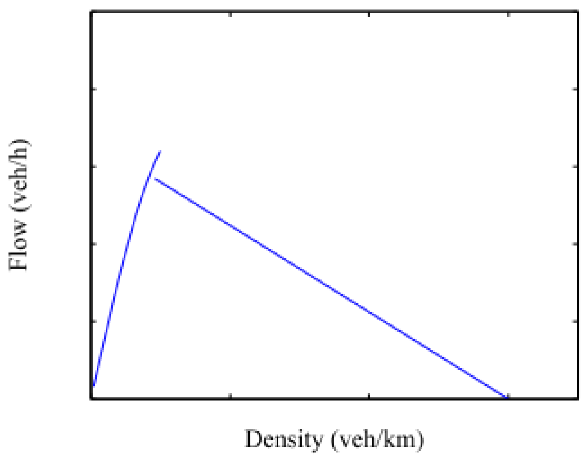


Figure 3.2: The Wu Fundamental Diagram, from Knoop et al. (2019)

3.1.2 Control location in space and time

The location of control in space is an important design choice for this concept. This location is defined as the head of the controlled area, so in the case of the x-t diagrams in Figure 3.1, the upstream end of the bottleneck. Since traffic should not be slowed down unnecessarily, the goal of the controller is to slow down vehicles as close to the bottleneck as possible, as unnecessary speed reductions result in an increased travel time. However, it may be the case that deceleration of a vehicle takes more time than expected, resulting in a longer deceleration distance. This creates a risk for the stability of traffic, as the vehicle can cause disturbances in traffic when passing through the bottleneck. Therefore, a safety margin may be required.

Also, vehicles do not instantly decrease their speeds when they are controlled with a lower speed than their current speed (as is assumed in Figure 3.1). Therefore, control should be applied early enough to allow vehicles to decelerate before passing the bottleneck. However, when drivers do comply, earlier control also results in longer distances driven at lower speeds. Following from this, a balance has to be found between these two factors. All these factors result in requirements for the measurement area.

These requirements for the measurement area are defined based on previously given information and will later be used to define the measurement area for the applied situation. The defined requirements for the measurement area are:

- Measurement has to take place upstream of the bottleneck;
- Vehicles should be able to decelerate to the target speed before reaching the bottleneck after receiving a speed control signal ;
- In order to account for compliance to the speed control signals, some extra distance should be added to make sure that vehicles have completed their deceleration when entering the bottleneck;
- To account for communication delays, the maximum possible delay (two ways) should be added as a safety margin;
- Since traffic is dynamic and shockwaves do not travel at the same speed as traffic itself, the measurement area should be as close to the bottleneck as possible to make problem detection accurate and avoid unnecessary control of vehicles that do not pose any risk anymore;
- The measurement area should be large enough to make sure that the flow of each vehicle is at least measured once within the measurement area and during a controller time step.

3.1.3 Measurements

In order to achieve the aforementioned traffic management goals, measurements of traffic conditions are required. An important part of the measurements are the positions and speeds of CVs, since these vehicles are the main actuators of this control method. These values are expected to be broadcasted by CVs through cooperative awareness messages, which are defined in ETSI (2019). These messages are sent at very low latency, making it possible to apply control within a short period of time.

Another required measurement is the traffic characteristic that the controller uses to decide when to apply control. In Section 2.1, it was found that flow, density and headway distribution were all important factors for predicting traffic breakdown. The measurements should be able to represent when to apply control in both congested and uncongested conditions. When density increases, and thus headways become smaller, while the average speed remains the same, the probability of a breakdown increases. However, when a breakdown occurs, these values still change into the same direction: density increases in congested state and headways decrease in congested state. This is not the case for flow. When a breakdown occurs, flow decreases due to the capacity drop. Furthermore, capacity of roads is generally expressed in terms of flow (veh/h). Therefore, using flow measurements probably results in the most stable controller. How this flow is measured is elaborated on later in the report.

3.1.4 Operational conditions

This concept, with this theory, can be operational in a completely connected environment or an environment with lower penetration rates, which includes connectivity for the vehicles on the road. The roadside is required to have coverage over the whole desired control area. Furthermore, since the concept attempts to prevent breakdown, it can only operate in free-flow conditions, congested traffic can be resolved with other algorithms. For example, this algorithm can be integrated with COSCAL in such a way that after control failure, or when a shock wave occurs due to other disturbances, COSCAL goes into effect instead of the algorithm developed here. Dynamic tuning of the target flow can also be used as a remedy for this, problem. These options are not explored in this thesis.

When traffic demand remains high for extended periods of time, the size of the controlled area increases. When this area becomes too large, spillback to off-ramps may occur. This increases the lost travel time, since drivers that do not have to pass the bottleneck are slowed down as well. When this occurs, congestion could not have been prevented with this algorithm, as the demand is simply too high for the road network to process.

3.1.5 Delays in communication

There are multiple types of communication channels available for sending control signals from the controller to vehicles. This controller does not have to be bound to one communication channel, as this only restricts the potential penetration rate that can be achieved. However, using multiple channels does result in differences in some aspects of the communication. This is primarily the difference in technology used, as well as the communication delays that come with that technology. In this section, the different communication technologies will be discussed and a way of handling the communication delays for the controller will be proposed. The solution for handling the delays will not be implemented or tested during the evaluation.

Communication channels and delays

A variety of communication channels for C-ITS purposes have been tested. However, wireless communication has never become perfectly reliable. In a paper by Bazzi et al. (2019), the communication channels that are currently available or in development are discussed and tested. The following channels are reviewed in the paper:

- IEEE 802.11p
- LTE-V2X
- IEEE 802.11bd
- G5 NR

Since the latter two channels are still in development, only the first two channels were tested in the paper. There are multiple variables that have influence on the delay for each channel:

- Number of vehicles, a higher number of vehicles results in a larger communication delay;
- Modulation and coding schemes, which contain a trade-off between data speed and the communication reliability;
- Required range of the transmitter, where longer range reduces the reliability, but also increases the number of vehicles the transmitter has to handle;
- The resulting packet reception rate, which is the average ratio of vehicles that successfully decode the message sent by the transmitter;
- Distance between the transmitter and the central server of the controller.

The first point is worrying, as communications will be more important when there are a higher number of vehicles on the road. Communication capacity should be sufficient to handle all vehicles on the road. Furthermore, during the tests, the required range was set at 200m, which resulted in update delays between 1 and 10 seconds, of which the variance is low. This means that, while communication itself is within microseconds, each vehicle receives an update each 1 to 10 seconds, depending on the communication channel and settings used and the delays are near constant. While IEEE 802.11bd and G5 NR aim to improve the speed and accuracy of communication in comparison to the other two channels, the controller will be restricted by the slowest communication channel that is used.

Therefore, the controller has to allocate its resources well. Vehicles will have to receive their control signal at least the number of seconds of delay before their onset of deceleration. If the vehicle requires time to process the message, this may even become a minimum of two times the delay. In the worst case scenario, that would result in a 20 second time period between receiving the control signal and the onset of deceleration. On a freeway, this can mean that the vehicle has already travelled more than 600m. This may be harmful to the effectiveness of the algorithm.

How to cope with the delays

The delay per type of communication seems to be near constant. However, since update times seem to be long for some channels and settings for communication, the controller has to take into account that control signals can not be sent to each vehicle individually at the moment a vehicle is required to decelerate.

Therefore, it is proposed to send the control signals in advance and let the vehicles decide on onset of the deceleration. This means that the vehicle tracks its location relative to the information of the control signal it has received and decides on when to decelerate (or present the message to the driver). A result of this, is that the moment of detection has to be early enough to be able to send control signals to vehicles that require deceleration.

Furthermore, the delays are harmful for any feedback loop in the controller. When a problem occurs, faster feedback response is most likely better than a late response. Priority may be given to vehicles that require an update, but the effects of this have to be researched in detail. Due to time constraints, this is not done within this thesis.

3.1.6 Resulting assumptions

- After detection of a high-flow area, it is assumed that the traffic state does not change between the measurement area and the bottleneck for the detected vehicles;
- A static bottleneck capacity is assumed in order to show that the algorithm can effectively reduce flow without inducing congestion. In reality this capacity is dynamic;
- Communication between infrastructure and vehicles is not bottlenecked by any bandwidth limitations and delays are equal to the time step of the simulation;

3.2 CONTROL ENGINEERING APPROACH

In this section, a control engineering approach will elaborate on the controller concept from a control engineering viewpoint, the traffic engineering theory will be translated into control engineering terms. First, important signs and parameters will be presented. Second, the controller will be described in mathematical terms for the detection of high-flow areas and the actuation of the controller.

3.2.1 Controller approach choice

The theory described before uses methods that are similar to the methods used in VSL algorithms such as SPECIALIST and COSCAL. Since the controller will have computations adapted from COSCAL, the choice for the same controller approach seems most useful. Therefore, a feedback controller approach will be used.

3.2.2 Introduction of important parameters

In this paragraph, a short list of important variables is presented. A schematic view of the variables can be found in Figure 3.3. In Appendix A a complete list of variables used in this thesis can be found.

- Vehicles will be referred to with index i, the most downstream vehicle in the network will have have $i = 1$;
- Time of the controller will consist of a time index k^{ctrllr} and a time step length $T^{\text{ctrllr}}(s)$. Therefore, current time of the controller is given by $T^{\text{current}}(s) = k^{\text{ctrllr}} T^{\text{ctrllr}}$;
- The location of a vehicle i in the network at time step k is given by $x_i(k)$ (m), where $x = 0$ is the upstream end of the network;
- The speed of a vehicle i at time step k is given by $v_i(k)$ (m/s)
- The space headway of a vehicle is given by h_i (m)

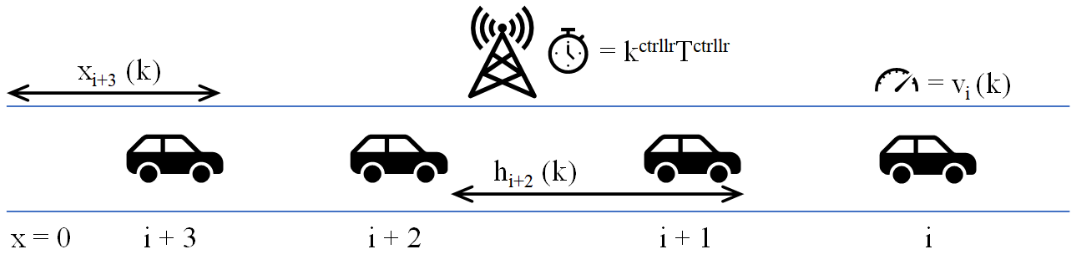


Figure 3.3: Variables in a schematic view

3.2.3 Detection

In order to decide when to start controlling traffic, a trigger has to be developed. As mentioned in Section 3.1.3, using flow as a trigger will result in a stable controller. Since it is assumed that the penetration rate of CVs is 100%, flow can be measured anywhere on the road, for any cluster of cars.

In order to compute the local flow of single vehicles, Edie's definition of flow is used (Edie, 1965). This computation can use the speeds and headways obtained through V2I communication for computation of flow for each vehicle. Since the communication has low latency and high accuracy, flow can be measured for a single vehicle, or over a few vehicles. This results in an an area which is defined by the summed headways of the vehicles over which flow is measured and the time over which flow is measured. An example of this area is given in Figure 3.4.
Since a single vehicle with a high flow is not very risky, flow is aggregated over multiple vehicles. However, this number is determined later, so a generalised computation is given here. In order to calculate the flow over n vehicles every time step, the time step of the simulation T^{ctrllr} (s) is used as the time axis of area X. For the space axis of the area, the sum of the current space headways h_i (m) of the vehicles are used, which is computed with location measurements of the CVs:

$$h_i(k) = x_{i-1}(k) - x_i(k) \tag{3.1}$$

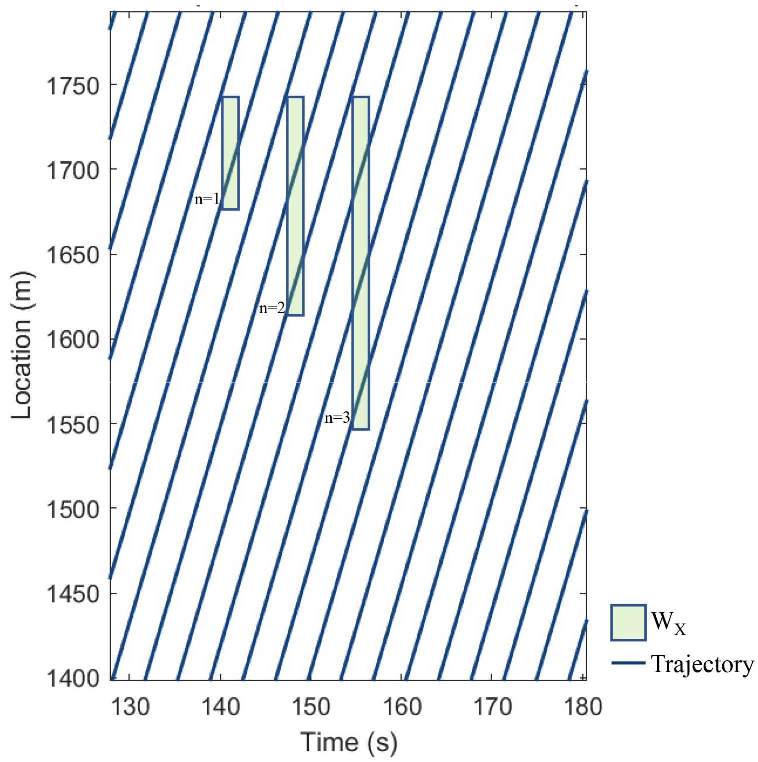


Figure 3.4: Examples of defined areas for measuring flow of vehicles. The box indicates the area W_X for multiple values n.

Then, the surface of area X for vehicle i at time step k is computed with the following formula:

$$W_{X,i}(k) = \sum_{j=i}^{i+n-1} h_j(k) T^{\text{ctrllr}} \tag{3.2}$$

Furthermore, the distance travelled by all vehicles in area $W_{X,i}(k)$, defined as $d_{X,i}(k)$ (m), is computed with the speed $v_i(k)$ of the vehicle times the time step length T^{ctrllr}:

$$d_{X,i}(k) = \sum_{j=i}^{i+(n-1)} v_j(k) T^{\text{ctrllr}} \tag{3.3}$$

When both $W_{X,i}$ and $d_{X,i}$ are known, the flow $Q_i(k)$ (veh/h) over the vehicles can be computed with the following formula:

$$Q_i(k) = \frac{d_{X,i}(k)}{W_{X,i}(k)} 3600 \tag{3.4}$$

When this value exceeds a pre-determined threshold $Q^{\text{threshold}}$, the next steps before actuation are triggered.

3.2.4 Actuation

Since vehicles receive individual control signals, computations have to be made separately for each vehicle. The method of computation from the COSCAL algorithm from Van de Weg et al. (2014) does exactly this. Therefore, the computations from COSCAL are adapted to fit the controller developed in this thesis. When $Q_i(k)$ is over the threshold value $Q^{\text{threshold}}$, control will be computed and actuated with a chosen target speed v^{target} (m/s) and target density ρ^{target} (veh/km). Control is actuated by constructing target lines for the vehicles that are within the high-flow area. These target lines consist of a target speed v^{target} (km/h), a target location and a time factor. These target lines describe a desired trajectory for each vehicle, which is determined by the pre-determined target state and is expected to be stable at the bottleneck. In Figure 3.5, vehicle trajectories of vehicles that have reached their target line and have the target traffic state are presented.

For each vehicle with a defined target line that is not yet decelerated, it is evaluated each time step if decelerating from its current speed and location will result in the vehicle reaching its target line with the desired speed. The computations required to achieve this concept are presented in this section.

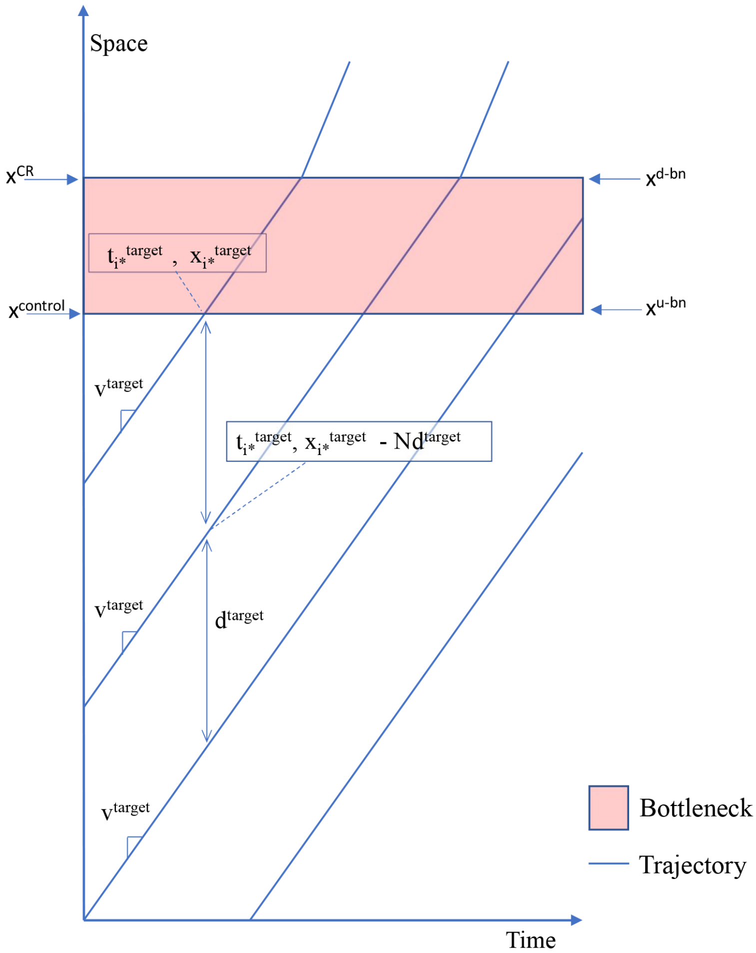


Figure 3.5: Visual representation of target lines with their calibration points and slope

Calculations for actuation are based on the following variables:

- The target state for vehicles, which is defined by the target speed v^{target} (m/s), ρ^{target} (veh/km) and q^{target} (veh/h) with the following relation: $q = kv$;
- $t_i^{target}(k)$ (s) is a point in time of the determined target line for vehicle i;
- $x_i^{target}(k)$ (m) is a point in space of the determined target line for vehicle i.
- $x^{control}$ (m), which indicates a distance from the bottleneck where the first vehicle that will be controlled (i^*) is aimed to have completed its deceleration to v^{target};
- x^{CR} (m) which is the location where vehicles are released from control;

Determining target lines

With the target density, target headways can be determined with:

$$d^{\text{target}} = \frac{1}{\rho^{\text{target}}} \tag{3.5}$$

For the first vehicle in the cluster of vehicles with a flow higher than $Q^{\text{threshold}}$, which we call i^*, a target line has to be determined in order to be able to construct target lines for all following vehicles. This can be done by choosing a location $x_i^{\text{target}}(k)$, which is equal to x^{control} for the first vehicle, and determining a moment in time $t_i^{\text{target}}(k)$ at which the vehicle is expected to reach this target line. Therefore, for the first vehicle:

$$x_{i^*}^{\text{target}}(k) = x^{\text{control}} \tag{3.6}$$

Then, $t_{i^*}^{\text{target}}$ can be determined by computing the expected time that the vehicle still needs to reach x^{control} and adding the desired timeheadway. This makes sure that the headway is large enough at the bottleneck. Then, a calibration point for the target line of the first vehicle has to reach its target line (t_{i^*}) is required. This is determined with the expected arrival time at the bottleneck of vehicle i^* and the target interarrival time between vehicles at the bottleneck. This can be computed with the following formula:

$$t_{i^*} = \frac{x^{\text{control}} - x_{i^*}(k)}{v_{i^*}(k)} + \frac{3600}{q^{\text{target}}} \tag{3.7}$$

$t_{i^*}^{\text{target}}$ can be computed with:

$$t_{i^*}^{\text{target}}(k) = T^{\text{current}} + t_{i^*} \tag{3.8}$$

This results in the target line of vehicle i^*, which is described by the following formula:

$$X_{i^*}^{target}(k) = x_{i^*}^{\text{target}}(k) + v^{\text{target}}(kT^{\text{ctrllr}} - t_{i^*}^{\text{target}}(k)) \tag{3.9}$$

Reaching the target line requires the vehicle to decelerate to the target speed. This requires space and time to brake. When assuming a constant deceleration, deceleration time is given by:

$$t_i^{\text{decelerate}}(k) = \frac{v^{\text{target}} - v_i(k)}{a^{\text{ex}}} \tag{3.10}$$

where a^{ex} (m/s²) is the assumed deceleration of all vehicles. This deceleration time can then be used to calculate the estimated required distance to decelerate:

$$d_i^{\text{decelerate}}(k) = \frac{1}{2}(v_i(k) + v^{\text{target}})t_i^{\text{decelerate}}(k) \tag{3.11}$$

The target line can then be constructed, resulting in the following:

$$X_{i^*}^{\text{target}}(k) = x_{i^*}^{target} - v^{\text{target}}(kT^{\text{ctrllr}} - t_{i^*}^{\text{target}}(k)) \tag{3.12}$$

Then, the vehicle should start to slow down when it fulfills the following condition:

$$x_i(k) \geq X_{i^*}^{\text{target}}(k) - d_i^{\text{decelerate}}(k) \tag{3.13}$$

With the determined target line for the first vehicle, the target lines for each subsequent vehicle can be determined. Firstly, $t_{i^*}^{target}$ remains the same:

$$t_i^{\text{target}}(k) = t_{i^*}^{target}(k) \tag{3.14}$$

Secondly, by subtracting d^{target} from the $x_i^{\text{target}}(k)$ of vehicle i^*, this results in the following target location:

$$x_i^{\text{target}}(k) = x_{i^*}^{target}(k) - Nd^{\text{target}} \tag{3.15}$$

where N is the number of vehicles between the current vehicle and the head of the cluster of vehicles that is slowed down. Then, the deceleration point to reach the target lines for subsequent vehicles can be constructed:

$$X_i^{\text{target}}(k) = x_i^{\text{target}}(k) - v^{\text{target}}(kT^{\text{ctrllr}} - t_i^{\text{target}}(k)) \tag{3.16}$$

The deceleration distance is calculated the same way as in (3.10) to (3.13). This gives the following deceleration condition for subsequent vehicles:

$$x_i(k) \geq X_i^{\text{target}}(k) - d_i^{\text{decelerate}}(k) \tag{3.17}$$

Target line deviations

Since the deceleration used in the calculations is an estimate of the deceleration of drivers, deviations from the target line can occur. While small deviations from the target line average out, large variations can cause problems. If a vehicle decelerates a lot faster than expected, a gap will be created between the vehicle and its predecessor, which reduces capacity where it is needed most. Also, the target lines of successive vehicles might overlap with the current trajectory of the vehicle. If this is not corrected, vehicles will accumulate and the risk of a breakdown increases.

In order to account for this, when checking the location of the vehicle in relation to its target line, two values are evaluated. Firstly, it is checked whether the vehicle is within a threshold distance of its target line. For this, a factor γ is used, which is multiplied by the target headway d^{target}. Secondly, a speed tolerance factor v^{tol} (m/s) is used. When the vehicle is γd^{target} meters upstream of its target line and its speed is nearing the target speed within $v^{\text{target}} + v^{\text{tol}}$, the vehicle is assumed to be off its target line. In order to correct the target lines of vehicles upstream, target lines are recalculated when:

$$X_i^{\text{target}}(k) - x_i(k) \geq \gamma d^{\text{target}} \wedge v_i(k) \leq v^{\text{target}} + v^{\text{tol}} \tag{3.18}$$

When the conditions are met, the target line for the vehicle i is reset to its current location and time, resulting in $x_i^{\text{target}}(k) = x_i(k)$ and $t_i^{\text{target}}(k) = T^{\text{current}}$. This results a re-calibrated target line for each new vehicle that joins the controlled area.

Speed control release

Since reducing the speed of vehicles on the road also has the negative effect of increasing the travel time of all controlled vehicles, the control period for each vehicle should be as short as possible, while making sure control is effective. For this controller, there are two options. Either the controller releases the speed control of vehicles at the upstream end or at the downstream end of the bottleneck.

Releasing speed control of vehicles at the upstream end of the bottleneck results in less delays, since vehicles are speed controlled for a shorter distance. However, in a scenario where penetration rates are lower than 100%, releasing speed control at the beginning of the bottleneck can result in a vehicle accelerating into oscillating traffic, which can create a jam.

In the case of releasing vehicles at the downstream end of the bottleneck, flow through the bottleneck can be controlled more closely. Also, in scenarios with lower penetration rates, uncontrolled oscillating traffic gets space and time to increase their headway and accelerate back to their desired speed.

Both cases can be implemented into the algorithm by changing the parameter x^{CR}, as the controller is designed to be as generic as possible in the sense that it should be applicable to a wide variety of bottlenecks. Therefore, the location of speed release is also a design parameter.

Control release

Releasing speed control for a vehicle does not mean that the control scheme is finished. The moment that the control scheme is finished depends on multiple factors. Firstly, when the difference between the traffic demand and the target state flow becomes higher, the control scheme will last longer. Also, when the demand remains high for a longer period of time, the control scheme will increase in length in space as well. Since the algorithm will be evaluated in isolation of other bottlenecks, this will not be taken into consideration in this thesis.

3.3 CONCLUSION

In this chapter, theory for two approaches to the problem was described. Firstly, the traffic engineering approach, where the theory development was discussed in terms of traffic engineering terms. This resulted in a macroscopic control concept which is expected to prevent breakdown at bottlenecks by manipulating traffic into a desired traffic state. Furthermore, important aspects of the theory are discussed. Requirements for the measurement area, as well as the required measurements are discussed, as well as the operational conditions of the controller and a method to deal with communication delays.

Secondly, the control engineering approach, where the theory from the traffic engineering approach is translated into a control problem. This means that the macroscopic theory is translated into a sequence of formulas to compute control signals for each vehicle. This resulted in a concept of leading individual vehicles to target lines, based on the the existing theory of COSCAL. Furthermore, theory for where control should be released is discussed.

The formulas developed in the control engineering approach are required to compute in the right sequence to get to the desired result that is developed in the traffic engineering approach. This requires the formulation of an algorithm that describes the computational steps of the controller in detail. The algorithm is formulated in the next chapter.

4 DEVELOPMENT OF THE ALGORITHM

In Chapter 3 both the traffic engineering theory and mathematical formulation of the controlled were described. In this chapter, these two pieces of the puzzle will be formed into an algorithm which describes the sequence of computations made by the controller during operation, based on the equations from Section 3.2. Within the algorithm, vehicles are differentiated with driving modes, these are presented first. Then, the steps in the algorithm are described in logical terms, as well as in pseudo-code.

4.1 DRIVING MODES

The control process uses multiple "modes" for the CVs involved. Just as in COSCAL, it is beneficial to use driving modes for multiple reasons (Hegyi, 2013):

- The modes are generic and can be understood by both manual vehicles and automated vehicles. Speed control signals are operationally more feasible for human drivers, since estimating a headway and adhering to a headway advice can not be done reliably without digital support;
- Privacy protection leads to the inability to track vehicles individually. Vehicle ID's may change. Therefore, when vehicles can broadcast their modes, they do not have to be tracked and messaged individually.

For this controller, three modes are used to differentiate between vehicles:

- **Mode N:** The default mode in this algorithm, which consists of normal driving. The vehicle did not receive any control signals from this controller and drives according to its own driving rules;
- **Mode H:** Target lines will be sent to vehicles, after which the vehicles will have to track their position in relation to the target line and decide on when to apply the control signal themselves. Vehicles in mode H have received a target line, but are not yet decelerating;
- **Mode A:** Advised mode, the controller has determined that the vehicle should decelerate to the target speed and the vehicle received the speed control signal from the controller.

There are multiple mode transitions possible:

- **N→H** The vehicle is flagged as high flow and will receive a control signal as soon as it has reached its deceleration point;
- **H→A** The flagged vehicle has passed its deceleration point and receives its speed control signal;
- **A→N** The vehicle has passed the bottleneck, the speed control is withdrawn;
- **H→N** The vehicle has passed the bottleneck without reaching its deceleration point;
- **N→A** The vehicle joins the controlled area at the tail and has reached its deceleration point.

The only mode transition that is not possible is the transition **A→H**, since vehicles that are in the controlled area can only exit the controlled area with a transition to mode N.

4.2 THE ALGORITHM STEPS

The full algorithm consists of a few steps, which are presented in the following section.

In this section, a few more parameters which are important for formulation of the algorithm are introduced. Furthermore, in order to make sure implementation into a simulation goes smoothly, each of the steps is defined as a psuedo-algorithm below.

4.2.1 Introduction of important parameters

- The measurement area is defined by its most upstream location $x^{\text{u-m}}$ (m) and its most downstream location $x^{\text{d-m}}$ (m);
- The mode of a vehicle is defined as M_i, which can be N, H or A;
- In order to track the tail of the speed-controlled area, vehicles are assigned a binary value $Tail_i$, which is 1 of the vehicle is the tail of the controlled area and 0 otherwise;
- A target line for a vehicle consists of both a location $x_i^{\text{target}}(k)$ (m) and a time $t_i^{\text{target}}(k)$ (s), when a step in the algorithm refers to removing the target line, these values are reset to 0;
- $v^{\max}$ (m/s) is the maximum speed a vehicle is expected to drive on the road;
- CVs are detected by the controller with a CV tag, so if $CV_i == 1$, vehicle i is connected.

4.2.2 Detection

Each time step, all CVs within the measurement area are checked on their flow level. However, measurements do not have any effect when there is still a controlled vehicle upstream of the bottleneck. For the aggregation of flow measurements, other vehicles are allowed to be outside of the measurement area. When the flow of a vehicle is too high, the vehicle is flagged for control. This process is described in detail in Algorithm 4.1.

Algorithm 4.1: Detection

Input:
1. All vehicle tags i, positions x_i(k), headways h_i(k), CV status CV_i and speeds v_i(k)
2. Flow threshold value $Q^{\text{threshold}}$
3. Controller time step length T^{ctrllr}
4. Aggregation value n
5. Measurement area which is defined by the locations $x^{\text{u-m}}$ and $x^{\text{d-m}}$

Output: Average local flow ($Q^{\text{threshold}}$) over n vehicles and $M_i(k)$ set to H for each vehicle with $Q_i(k) > Q^{\text{threshold}}$

for *All vehicles i within the defined network* **do**
  Compute local flow using (3.1) to (3.3);
end
for *All vehicles with* $x^{u\text{-}m} \leq x_i(k) \leq x^{d\text{-}m}$ **do**
  Compute aggregated local flow $Q_i(k)$ with (3.4);
  if $Q_i(k) > Q^{threshold} \wedge CV_i == 1 \wedge M_i(k) == N \wedge$ *No control upstream of*$x^{u\text{-}BN}$ **then**
    $M_i(k) = H$;
  end
end

4.2.3 Computing target lines

When the flow of a vehicle is too high according to Algorithm 4.1, the vehicle will receive a target line. The algorithm then gives all vehicles with a measured flow which is higher than q^{target} a target line as well. This is to make sure that vehicles that might have to decelerate earlier than more downstream vehicles receive their target line before they reach their deceleration point. The logic is provided in Algorithm 4.2.

Algorithm 4.2: Computing target lines after detection

Input:
1. Vehicle tags i, positions $x_i(k)$, speeds $v_i(k)$, aggregated flows $Q_i(k)$, CV status CV_i and modes $M_i(k)$
2. Algorithm parameters: $Q^{\text{threshold}}$, x^{control} and a^{ex}
3. Target traffic state consisting of a target flow q^{target}, speed v^{target} and density ρ^{target}

Output: Target lines for all vehicles in the detected high flow area and directly upstream vehicles with $Q_i(k) > q^{\text{target}}$

for *All vehicles i with* $x^{u\text{-}m} \leq x_i(k) \leq x^{CR}$ **do**
    if $M_i(k) == H \wedge$ *vehicle i does not have a target line* **then**
        Compute target line for the vehicle using (3.5) to (3.9);
        $j = i + 1$;
        $N = j - i$;
        while $Q_i(k) \geq Q^{threshold}$ **do**
            if $CV_j == 1$ **then**
                Compute target line using (3.15) to (3.17);
                $M_j(k) = H$;
            end
            $j = j + 1$;
        end
    end
end

4.2.4 Target line deviations

In this part of the algorithm, vehicles are checked for target line deviations by comparing both the location and current speed of the vehicle with the target values. If the location deviates from the target line in the upstream direction, while the vehicle's speed is near or below v^{target}, the target lines of upstream vehicles are reset with the current location and time of the deviating vehicle as new starting point for computation of the target lines. Only following vehicles which already were assigned to a target line receive new target values.

Algorithm 4.3: Check for target line deviations

Input:

1. Vehicle tags i, positions $x_i(k)$, speeds $v_i(k)$ and modes $M_i(k)$
2. Vehicle target lines $X_i^{target}(k)$
3. Algorithm parameters: γ, v^{tol}, d^{target}, v^{target}

Output:

Target line corrections for upstream vehicles when a vehicle deviates from its target line

for *Vehicles i with* $M_i(k) == A \vee M_i(k) == H$ **do**
    if $x_i^{target}(k) - x_i(k) \geq \gamma * d^{target} \wedge v_i(k) \leq v^{target} + v^{tol}$ (3.18) **then**
        $t_i^{\text{target}}(k) = kT^{\text{ctrllr}}$;
        $x_i^{\text{target}}(k) = x_i(\text{k})$;
        $j = i + 1$;
        for *Vehicles with* $x_i(k) \geq x_j(k)$ **do**
            if $M_j(k) == A \vee M_j(k) == H$ **then**
                $N = j - i$;
                $t_j^{\text{target}}(k) = t_i^{\text{target}}(k)$;
                $x_j^{\text{target}}(k) = x_i^{\text{target}}(k) - Nd^{\text{target}}$;
            end
        end
    end
end

4.2.5 Check if vehicles have reached their deceleration point

Each vehicle on the road, which is flagged for a target line, is checked whether it should be slowed down to the target speed to reach its target line. When this is true, vehicles are put into mode A and the tail indicator is passed to the new tail of the controlled area. This follows the logic in Algorithm 4.4.

Algorithm 4.4: Check if high-flow vehicles have to slow down

Input:
1. Vehicle tags i, position x_i(k), modes $M_i(k)$, tail indicator $Tail_i(k)$ and target line values $t_i^{\text{target}}(k)$ and $x_i^{\text{target}}(k)$
2. Algorithm parameters: v^{target}, x^{CR}

Output:
Changes in vehicle modes from N to A when the vehicles are within deceleration range of their target line

for *All vehicles with* $M_i(k) == H$ **do**
    Compute $t_i^{\text{decelerate}}$(k) and $d_i^{\text{decelerate}}$(k) to v^{target} using (3.10) and (3.11);
    if (3.13) *is true* **then**
        $M_i(k) = A$;
        $j = i - 1$;
        while $Tail_j(k) == 0 \wedge x_j(k) < x^{CR}$ **do**
            $j = j - 1$;
        end
        $Tail_j(k) = 0$;
        $Tail_i(k) = 1$;
    end
end

4.2.6 Check for vehicles joining at the tail

Each time step, it is evaluated whether vehicles directly upstream of the controlled area will join the controlled area. In order to find which vehicles may have to slow down, it is evaluated whether the directly upstream CVs would have to slow down if they would have a certain speed $v^{\max}$, which is chosen slightly higher than the maximum speed to make sure outliers are caught as well. Since the target state always contains a density higher than or equal to free-flow, as soon as a vehicle does not have to decelerate with $v^{\max}$, the following vehicle will not have to either. For each of these vehicles, (3.13) is evaluated. When this is true, the vehicles receive a target line, which is computed with the target line of the vehicle that is the tail of the controlled area. When a vehicle is put into mode A with this check, the algorithm searches for the tail of the area and gives the vehicle that just joined the controlled area the tail indicator. The psuedo-algorithm for this part is given in Algorithm 4.5.

Algorithm 4.5: Check tail of the controlled area for incoming vehicles

Input:
1. Vehicle tags i, positions x_i(k), speeds v_i(k), modes M_i, $Tail_i$ and the target lines
2. Algorithm parameters: x^{CR}, v^{target}, ρ^{target}, a^{ex}

Output: Vehicles joining the tail of the controlled area advised when necessary

```
for All vehicles with M_i(k) == A do
    if Tail_i(k) == 1 then
        j = i + 1;
        N = j − i;
        while (3.13) == true when computing t_i^decelerate(k) and d_i^decelerate(k) with v^max
          instead of v_i(k) and t_i^target(k) = t_i^target(k) + T^ctrllr do
            if (3.13) == true then
                Compute target line using (3.15) to (3.17);
                M_j(k) = A;
                u = j − 1;
                while Tail_u(k) == 0 ∧ x_u(k) < x^CR do
                    u = u − 1;
                end
                Tail_u(k) = 0;
                Tail_j(k) = 1;
            else
                Compute target line using (3.17) to (3.15);
                M_j(k) = H;
            end
            j = j + 1;
            N = j − i;
        end
    end
end
```

4.2.7 Release of control

All vehicles in mode A that have passed the point of control release are released and put back into mode N. Their association with the tail is removed. Therefore, when the tail of the controlled area passes the bottleneck, the control scheme is finished and no new target lines are assigned until a new high flow area is detected. The psuedo-code is given in Algorithm 4.6.

Algorithm 4.6: Release of control

Input:

1. Vehicle tags i, positions $x_i(k)$, $Tail_i$ and modes $M_i(k)$
2. Control release location x^{CR}

Result:

Vehicles are released from control when passing x^{CR}

```
if x_i(k) > x^CR ∧ (M_i(k) == H ∨ M_i(k) == A) then
    Set M_i(k) = N;
    Set Tail_i(k) = 0;
    Remove target line;
end
```

4.2.8 Send control signals

In the last step, the target lines are sent to the vehicles that require it. The vehicles either follow it, or present it to their driver at the right moment, depending on the level of automation. Also, vehicles that have passed x^{CR} are released from control and can accelerate to their desired speeds. This results in the following changes for each transition:

- **N→H** No speed control, the vehicle has not reached its deceleration point but has received a target line;
- **H→A** The vehicle has passed its deceleration point and has determined it should decelerate to v^{target};
- **A→N** The speed control signal is withdrawn, the vehicle can accelerate to its desired speed;
- **H→N** The vehicle has passed the bottleneck without reaching its deceleration point, no deceleration has taken place due to control signals.

4.3 CONCLUSION

In this chapter, the algorithm used for the computation of the control concept which is developed in Chapter 3 is presented. To distinguish vehicles on the road on control signals received, driving modes were introduced and discussed. The three driving modes are:

- **Mode N:** The default mode in this algorithm, which consists of normal driving without control signals;
- **Mode H:** Vehicles in mode H have a target line assigned, but are not given a deceleration control signal yet;
- **Mode A:** Advised mode, the controller has determined that the vehicle should decelerate to the target speed and the vehicle received the speed control signal from the controller.

For all driving modes, each viable mode transition is discussed and the corresponding control signals are presented. Six sequential steps of computation are developed to determine when to trigger the algorithm and send the target lines and driving modes to the vehicles, how to check for vehicles joining the controlled area, when to adjust the target lines due to deviations from them and when to release vehicles of control. For each step, its goals, inputs and pseudo-code are given. The pseudo-code can be implemented into a microscopic traffic simulation model. The development of the simulation model, which will be used for validation of the control concept, as well as evaluation of the controller, is discussed in the following chapter.

5 DEVELOPMENT OF THE SIMULATION

In this chapter, important aspects of the microscopic traffic simulation study performed to evaluate the effectiveness of the controller in this thesis are discussed. Firstly, the software used for the simulation is presented. Secondly, the choice for the driver model is elaborated and the chosen driver model is described. Thirdly, the network layout is presented, including the method to simulate a single-lane bottleneck. Fourthly, each of the tuning parameters of the algorithm is discussed and it is determined how the parameters will be tuned. Fifthly, the demand pattern is presented. Lastly, the method for modelling driver compliance is explained.

For some explanations in this and the following chapters, simulation output in the form of vehicle trajectories is presented. In Figure 5.1, an example of this output is presented. In the figure, four plots can be found which show speed, density, flow and driving modes of vehicles respectively. on the right side of each figure, a color map is shown, which indicates the meaning of the colors within each figure. The speed, density and flow plots show how the controller manipulates the three traffic variables in order to keep traffic stable. The mode plot shows the modes of cars, which were explained in Section 4.1.

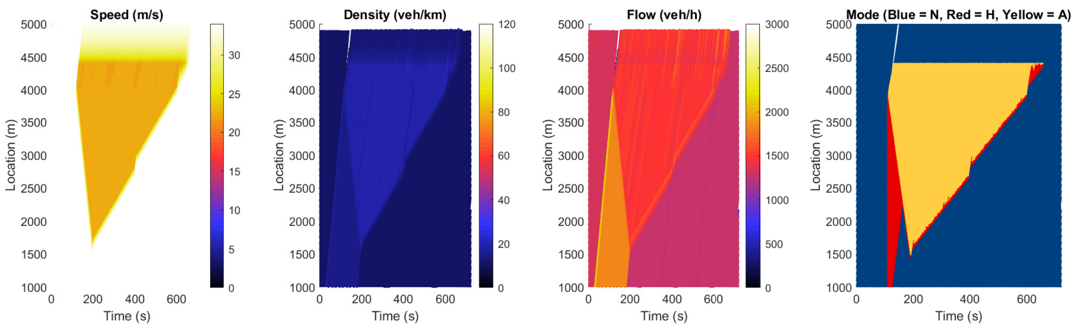


Figure 5.1: Example of the output figures of the simulation

For the simulation, a small traffic micro-simulation model in MATLAB is used. This gives access to influencing all the factors in the simulation, direct control into every detail of the model and easy access to all output data. Furthermore, computation times will be low, which makes it easy to test easily during development. Also, many scenarios can be run during the evaluation. Further arguments and how to receive the model can be found in Appendix C.

5.1 DRIVER MODEL

The driver model is an important part of a traffic simulation model, as it describes the behaviour of the vehicles in the simulation. In this section, the requirements for the driver model are described. Also, the chosen driver model is explained in detail.

5.1.1 Requirements for the driver model

In order to test the developed controller well, the driver model has to reproduce the behaviour on the road that the controller tries to counteract. Therefore, the following requirements for the driver model are identified:

- The model should allow for a bottleneck to be created in a single lane network;
- The model should be able to reproduce traffic instability and jams;
- The model should have a capacity drop in congested conditions;
- Since one lane is simulated, only longitudinal behaviour has to be described.

A driver model that complies to these requirements is the Intelligent Driver Model (IDM), which reproduces instability and congestion conditions well due to the non-linear behaviour it produces (Perraki et al., 2018; Schakel et al., 2010).

5.1.2 Description of the Intelligent Driver Model

IDM was originally developed in Treiber et al. (2000). For IDM+, the following parameters are used:

- v is the standard desired speed of vehicles;
- $v_i^{\text{des}}(k)$ (m/s) is the desired speed of the vehicle for the current time step;
- a (m/s²) is the maximum value for acceleration;
- δ is the rate at which acceleration decreases as the speed approaches the desired speed
- s_0 (m) is the preferred distance between vehicles during standstill
- $s_i(k)$ (m) is the spacing between vehicle i and $i-1$
- b (m/s²) is the maximum comfortable deceleration
- $T_i(k)$ (s) is the desired time headway of drivers
- l (m) is the vehicle length

In this driver model, the determination of acceleration is split up into two parts. Firstly, the determination of the acceleration due to the difference between the current speed of the vehicle and its desired speed. Secondly, an interaction term, which determines the acceleration by looking at the speed and distance differences with the leading vehicle. This results in the following formula:

$$v_i(k+1) = v_i(k) + a\left[1 - \left(\frac{v_i(k)}{v_i^{\text{des}}(k)}\right)^{\delta} - \left(\frac{s^*(v_i(k), \Delta v_i(k), k)}{s_i(k)}\right)^2\right] \tag{5.1}$$

where v is the current speed of the vehicle and $\Delta v_i(k)$ is the speed difference with the leading vehicle

$$\Delta v_i(k) = v_i(k) - v_{i-1}(k) \tag{5.2}$$

The value for $s_i^*(k)$ is determined with:

$$s^*(v_i(k), \Delta v_i(k), k) = s_0 + v_i(k)T_i(k) + \frac{v_i(k)\Delta v_i(k)}{2\sqrt{ab}} \tag{5.3}$$

Since the capacity of the original IDM does not produce realistic capacity values for reasonable parameter values, Schakel et al. (2010) adapted the model to increase the resulting capacity to a reasonable value. The free-flow and interaction terms are separated, resulting in the following adaptation of (5.1):

$$v_i(k+1) = v_i(k) + a \cdot \min\left[1 - \left(\frac{v_i(k)}{v_i^{\text{des}}(k)}\right)^{\delta}, 1 - \left(\frac{s^*(v_i(k), \Delta v_i(k), k)}{s_i(k)}\right)^2\right] \tag{5.4}$$

The separation results in a triangular FD instead of a topped-off triangular FD (Schakel et al., 2010). Furthermore, acceleration changes more aggressively in response to changes in either $v_i(k)/v_i^{\text{des}}(k)$ or $s_i^*(k)/s_i(k)$.

Furthermore, a small adjustment was made to the model for situations where the desired speed of a vehicle is lower than its current speed. The original IDM is not built for these situations, as it results in very heavy decelerations. For example, when a vehicle with a speed of 120km/h receives a control signal to decelerate to 80km/h, deceleration would be:

$$1 - \left(\frac{120}{80}\right)^4 = -4.06\text{m/s}^2$$

which is an unrealistic response on a freeway. Therefore, when $v_i(k) > v_i^{\text{des}}(k)$, vehicles are decelerated with a constant deceleration of b. Furthermore, the desired speed of a vehicle changes when it receives a control signal.

$$v_i^{\text{des}}(k) = \min\left[v, v_i^{\text{ctrl}}(k)\right] \tag{5.5}$$

where $v_i^{\text{ctrl}}(k)$ is determined with:

$$v_i^{\text{ctrl}}(k) \begin{cases} v^{\text{target}}, & \text{if } M_i(k) == A \\ \infty, & \text{otherwise} \end{cases}$$

Also, as mentioned before, the desired time headway value $T_i(k)$ is changed when vehicles enter the bottleneck. The current value is determined with:

$$T_i(k) = \begin{cases} T_i^{\text{base}} + T^{\text{BN}}, & \text{if } x^{\text{u-BN}} < x_i(k) < x^{\text{d-BN}} \\ T_i^{\text{base}}, & \text{otherwise} \end{cases}$$

where T^{BN} (s) is the additional desired time headway value in the bottleneck. Determination of T_i^{base} is explained in the next paragraph.

5.1.3 Parameter values used for the driver model

In Table 5.1, the parameters used during the simulations are presented. Since only the dynamics of the controller are evaluated and not the expected effects on a real case, there is no need for calibrated values. Therefore, the original IDM+ values for MOTUS are used (Kan et al., 2019). However, since some driver variability is desired, a variable that can be varied within this model has to be identified.

Since in the developed model, only a single lane is simulated, varying the desired speed v_0 among individual vehicles is not logical. Where on a multi-lane road, this would lead to lane-changes and overtaking, in this model it would lead to empty gaps between vehicles with a large difference in desired speed. Since this results in voids that would dampen any oscillations without control signals, varying the desired speed is not possible.

The desired space headway of a vehicle is primarily determined by its current speed, s_0 and $T_i(k)$. Since the headway plays a big role in IDM, varying a parameter that relates to the desired headway results some heterogeneity in the simulated traffic. From these parameters, varying T is the obvious choice, as s_0 describes the minimal stopping distance, which does not affect the driving task a lot.

For these reasons, the desired time headway of drivers is varied during the simulations. This is done by giving vehicles an individual value T_i^{base} with a normal distribution with $\mu = 1.2$ and $\sigma = 0.15$, which does not go below zero. In Table 5.1, the μ of the normal distribution for T is given.

Another parameter that affects the controller is b. Since the controller uses an expected deceleration parameter (a^{ex}), varying it between vehicles will affect the results of the simulations. In order to analyse effects of this well, this will be taken into account during the evaluation separately.

Table 5.1: IDM+ parameter values used during the simulations

Parameter	v	a	δ	s_0	b	T_i^{base}	l
Value	34.36	1.25	4	3	2.09	1.2	4

5.2 NETWORK

For the simulation, a one-lane freeway with a length of 5km is simulated. The length of the network is chosen to make sure that the controlled area does not reach the end of the network in any of the simulations used during evaluation. A time period of 12 minutes is simulated to make sure that the controlled area is resolved in most simulations.

Since no real disturbances occur naturally when simulating a one-lane freeway with IDM+, the bottleneck in the network is defined as in Treiber et al. (2006). At a certain point in the network which is called $x^{\text{u-BN}}$ (m) and is at 4000m, the T value of IDM+ is increased. At the end of the bottleneck $x^{\text{d-BN}}$, which is at 4400m, the value T is set back to the usual value. Without control, this bottleneck in combination with the demand pattern should result in a jam. When simulating the network with the later defined demand pattern, a jam reliably occurred for $T^{\text{BN}} = 0.9$, lower values resulted in fewer jams. As the controller aims to prevent breakdown, it has to be guaranteed that a jam is created without control. Therefore, the T value was increased from an average of 1.2s to an average of 2.1s when entering the bottleneck, resulting in $T^{\text{BN}} = 0.9$.

5.3 TUNING OF THE CONTROLLER PARAMETERS

The algorithm contains many parameters which influence the behaviour of the controller. In this section, the parameters are re-visited and the tuning process for each parameter is described. When possible, the resulting value is given.

5.3.1 Target state

The target traffic state is a design variable in this concept. Vehicles can be guided to the desired speed and density, such as one of the two states C_1 and C_2, presented in Figure 3.1a. The target values should be chosen in such a way that the flow remains high, but stable enough to not cause any disturbances that may result in a jam. These choices depend on the road on which the control scheme is applied, as well as the type of bottleneck that is present on the road. Furthermore, there is an upper bound for v^{target} of 80km/h, since when applying the algorithm in a real case, trucks with a speed limit of 80km/h will be present on the road as well. Another requirement for the target state, which was determined in Section 3.1.1, is that the target state should have a higher density than the density in the detected high-flow area.

Estimation of the bottleneck capacity

In order to make an estimation of capacity of the bottlenecks for the given speeds, Knoop et al. (2019) proposes a method that makes use of the relation between the acceleration formula of the driver model (5.4) and the FD. In order to find the equilibrium spacing, the acceleration of the driver model should be zero. This was implemented into an Excel Solver model consisting of (5.4) and all IDM+ parameters used in the simulation. The equilibrium conditions (v^{eq}, ρ^{eq} and q^{eq}) can be found for the desired time headway inside the bottleneck by letting the solver adjust the headway with the objective of making the acceleration zero.

When applying the solver for $T = 2.1$ and $v = 34.36$, $\rho^{\text{eq}} = 13.7$ veh/km is found. This results in $q^{\text{eq}} = 1645$. This means that when flow is above this value, traffic will be unstable as vehicles will start decelerating and thus traffic will oscillate. Therefore, this target flow will be used as a starting value for tuning the controller. Since the controller will have $v^{\text{target}} = 80$km/h during the evaluation, a starting value for the density can be calculated with the relation $q = \rho v$, which results in a starting value of the tuning parameter $\rho^{\text{target}} = 20.5$ However, since vehicles receive a random value for T, the final value for the target density is expected to be lower. Furthermore, this is not the maximum bottleneck capacity, as this is defined as the maximum throughput. However, it is the maximum stable capacity for the given speeds in this simulation.

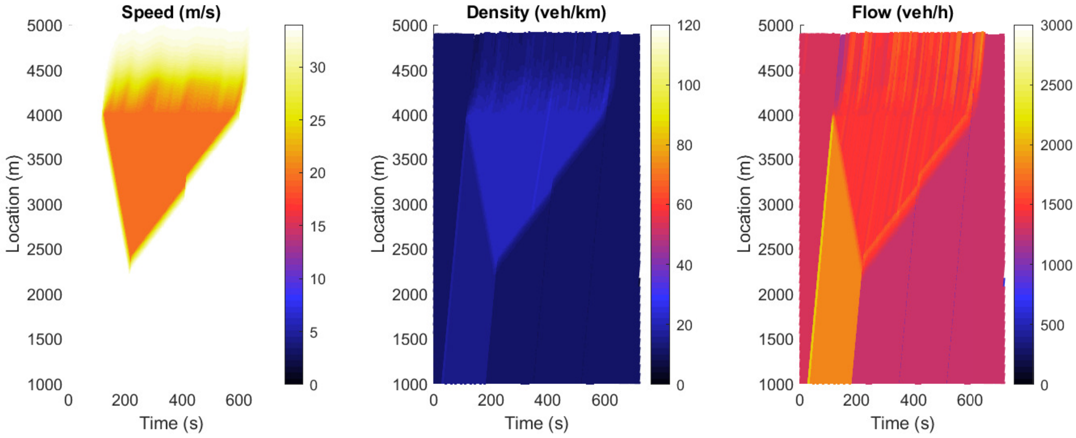
During the tuning of the controller in a real-world setting, there is no driver model. Then, the bottleneck capacity will have to be found by looking at real data from loop-detectors or Video-Based Monitoring (VBM). The highest flow measured before a jam occurs can be used as a starting point for tuning the controller. However, since it is expected that the target flow should be a little lower than the measured flow (as instability in the controlled area may occur if the flow is at or near capacity), it is advised to start with a value that is slightly lower.

Since this bottleneck restricts flow by a heavy margin, there could be many cases that flow has to be reduced from a density that is not the equilibrium density. Since speeds can be kept higher at lower densities, it would be beneficial to let the target state be decided by the current state of the flow. Automatic tuning could provide a solution for this. However, due to time constraints this is not developed within this thesis.

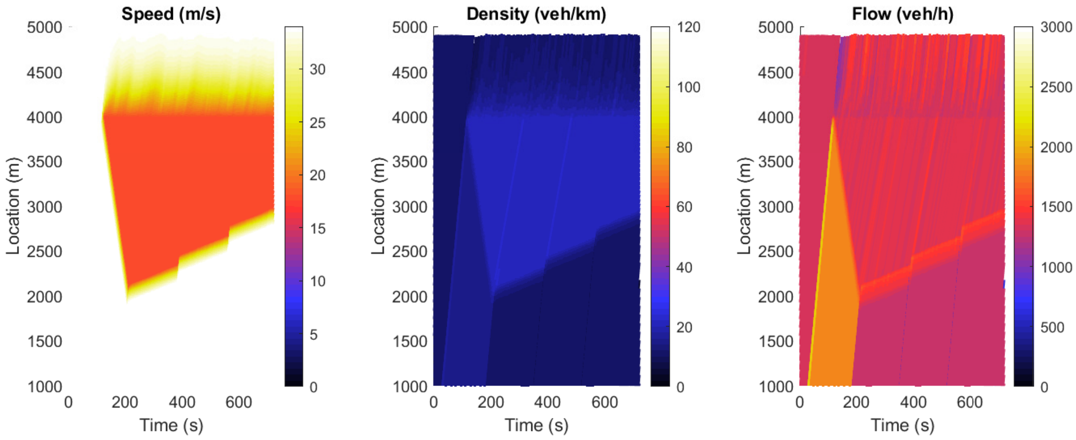
Effects of changing the target state

When changing either of the two target state variables, the resulting target flow changes. The change in flow follows the $q = \rho v$ relation. When comparing the speed, flow and density between Figure 5.2a and Figure 5.2b, it can clearly be seen that the reduced flow results in a larger control area. Reducing either of the parameters will result in this effect. However, when both parameters are changed in such a way that the resulting q^{target} remains the same, flow through the bottleneck remains the same. However, the shape of the control scheme does change. This can be seen when comparing Figure 5.2a with Figure 5.2c. Due to the lower ρ^{target}, vehicles have to be decelerated earlier, resulting in control area that is more stretched out on the space-axis.

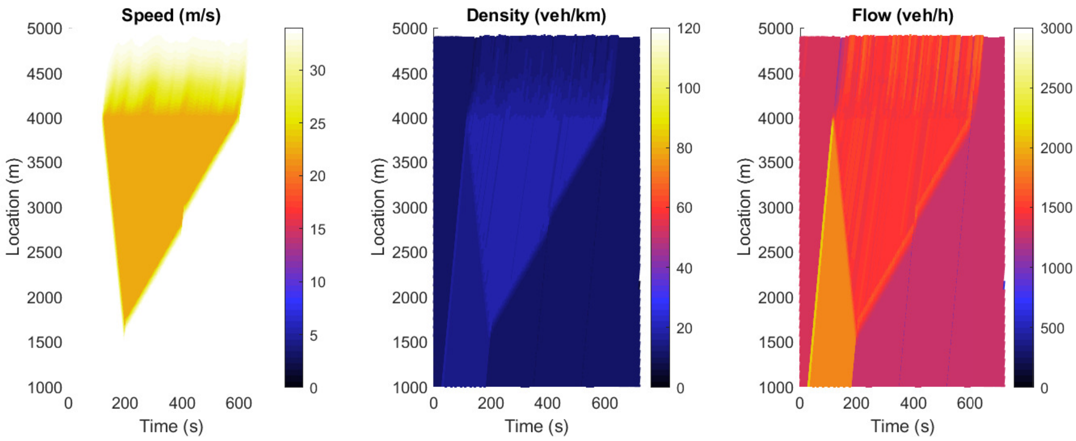
When density is chosen too low, the high flow area is not decelerated properly, since vehicles upstream should be decelerated earlier due to the lower density (and thus higher spacing between vehicles). This is shown in Figure 5.3. As can be seen in the figure, the high-flow area is not exactly decelerated into the target state, which is due to vehicles being downstream of their target lines and receiving control signals at the same time. This results in a higher density than the target density, since vehicles do not spread out in the initial high-flow area. Vehicles upstream of the high-flow area do decelerate in time, so these vehicles do create the target traffic state. Even though in this example traffic remains stable, it is undesired that traffic is not controlled into the desired state.



(a) Trajectories for $v^{\text{target}} = 70\text{km/h}$, $\rho^{\text{target}} = 20.5\text{veh/km}$ and $q^{\text{target}} = 1435$



(b) Trajectories for $v^{\text{target}} = 65$ km/h, $\rho^{\text{target}} = 20.5$ veh/km and $q^{\text{target}} = 1333$



(c) Trajectories for $v^{\text{target}} = 80$ km/h, $\rho^{\text{target}} = 17.9$ veh/km and $q^{\text{target}} = 1432$

Figure 5.2: Examples of trajectory plots with different tuning, showing different behaviour. When comparing a and b, it can be seen that reduced flow severely increases the size of the controlled area. When comparing a and c, near-equal flow can still result in a different shape of the controlled area.

There is a solution for this problem. Relocating the measurement area more upstream will result in earlier detection of the high-flow area. In this case, the controller will be able to identify the high-flow area in time and reduce the speed of upstream vehicles first. However, steering to lower densities is still undesired. When the measurement area is further away from the bottleneck, it is unknown if the traffic state is still the same when traffic arrives at the bottleneck. Therefore, a lower density would require the model to predict which

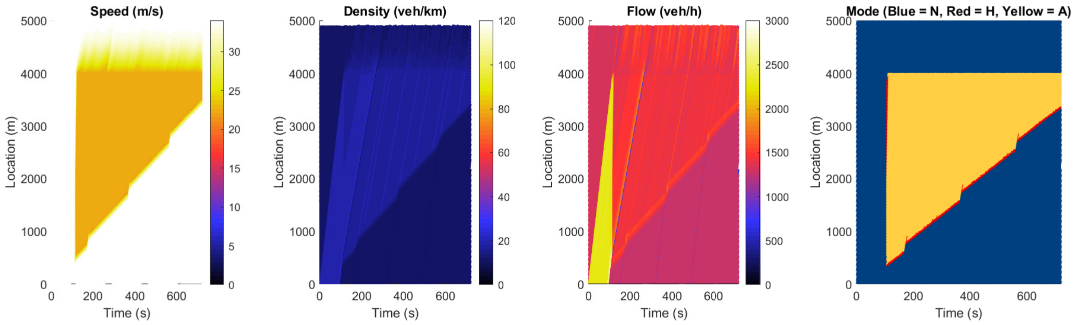


Figure 5.3: Trajectories for a scenario where ρ^{target} is chosen too low

vehicle has to be controlled upstream first. When this prediction is wrong, this may result in unnecessary control of vehicles, or not enough control of vehicles. Besides, overtaking may occur, which would change the order of target lines. If this is not accounted for, confusing control signals will be sent.

5.3.2 Threshold for activation of the control scheme

The threshold $Q^{\text{threshold}}$ is determined just like the starting value for the target state, since the threshold should be equal to maximum bottleneck flow. This value was found to be 1643 when determining the starting value for the target state, so $Q^{\text{threshold}} = 1645$.

5.3.3 Measurement area

In Section 3.1.2, requirements for the measurement area are given. These requirements can be changed into constraints, which will be used to determine the measurement area. When all these requirements are combined, the following condition for the minimum distance between the measurement area and x^{control} is determined:

$$x^{\text{control}} - x^{\text{d-m}} \geq d_i^{\text{decelerate}} * d^{\text{compliance-margin}} + 2 * D^{\max} * v^{\max} \tag{5.6}$$

with $d^{\text{compliance-margin}}$ (m) being the safety margin for compliance to the advice and $D^{\max}$ (s) being the maximum communication delay between the controller and a CV. Furthermore, the minimum length of the measurement area is given by:

$$x^{\text{d-m}} - x^{\text{u-m}} \geq 2D^{\max}v^{\max} \tag{5.7}$$

Please note that these equations are designed for the synthetic case used in this thesis. Other factors not considered within this thesis may change these equations.

5.3.4 Control target location

The target location for control, x^{control}, decides the target line of the first vehicle i^*. The most obvious choice for this parameter is to aim at the upstream end of the bottleneck ($x^{\text{u-BN}}$), as that is the location where traffic should have the target state.

When choosing a point downstream of $x^{\text{u-BN}}$, vehicles from the detected high-flow area, that are still decelerating (or did not even start deceleration) will enter the bottleneck. This is undesired, as the flow will be too high and the risk of instability is increased.

For a point upstream of $x^{\text{u-BN}}$, vehicles are slowed down for a longer distance and thus delays due to control are increased. However, in the cases of slower deceleration than expected, vehicles have some extra space to decelerate and thus increasing the chance that vehicles are decelerated to the target speed when entering the bottleneck.

In general, this parameter will be chosen $x^{\text{control}} = x^{\text{u-BN}}$. This may be changed during evaluation when the scenario may benefit from it.

5.3.5 Aggregation over n cars

This parameter decides the number of vehicles that are taken into account for each flow measurement of the controller. Depending on the demand pattern and the scale at which the controller should operates, this parameter can be chosen as desired. When it is desired that the controller responds when a single vehicle measures a high flow, n should be chosen 1. However, since the speed difference between v^{target} and v_0 is quite large, reacting to a single vehicle with high flow probably is not favorable for delays. As this parameter will not make a difference for the previously defined demand pattern, this parameter will be chosen to be 4 during the simulations, but is not expected to make a difference.

For real world application, this parameter will also depend on data availability. When only data from CVs is present for the controller, n can only be two, as a CV can measure the headway on its downstream and upstream side and compute flow over itself and its follower.

5.3.6 Deceleration

In order to compute the expected deceleration time and distance ((3.10), (3.11) and (3.13)), a value has to be assumed for deceleration. This deceleration is based on the parameters of the driver model. Acceleration and deceleration do have a stochastic factor within the model. However, this is a uniform distribution, adding a factor between -0.5 and 0.5 to the acceleration each time step.

When overestimating the deceleration value, vehicles need more time to slow down than the algorithm predicts ((3.10) becomes lower). This results in vehicles being advised at a later point in time, which leads to vehicles ending downstream of their target lines. This may not be a problem for the 100% connected scenario. However, when taking into account compliance, many vehicles are likely to end up downstream of their target line already. In this case, it is desirable to have as many vehicles as possible on their target line. Therefore, a^{ex} should not be overestimated.

In the case that a^{ex} is underestimated, the opposite effect happens. Vehicles receive their advice earlier. This leads them to a more upstream position than their target line. While this gives some room for vehicles ending up downstream of their target lines, this results in a higher number of target line resets. This causes a drop in the realised density of the algorithm, which is inefficient.

From simulation tests, it was found that slightly overestimating the deceleration resulted in more homogeneous traffic within the controlled area. This was mainly due to more reliable onset of deceleration, which is important for the effectivity of the algorithm. Therefore, during the evaluation of the algorithm, a^{ex} will be overestimated by 10%, resulting in $a^{\text{ex}} = 2.299$.

5.3.7 Tolerance for deviation from target lines of vehicles

There are two parameters relating to target line tolerance: v^{tol} and γ. Firstly, v^{tol} determines at what speed the vehicle is expected to be close to its target line. A higher v^{tol} results in a higher chance to trigger a target line reset. This is undesirable, as it reduces the flow on the road when lines are unnecessarily reset. Therefore, v^{tol} should be small. The initial value for v^{tol} will be based on the value chosen for COSCAL in Van de Weg et al. (2014), which is 1km/h.

The second parameter, γ, describes how many meters a vehicle can be upstream of its target line when it has reached v^{target}. As explained earlier, γ is multiplied with d^{target} in order to get a tolerance factor for location. When $\gamma = 1$, this means that a vehicle can be on the target line of the following vehicle before a reset happens. Since this may result in short headways and possibly oscillations at the bottleneck, γ should be lower than 1. The value will be determined in the same way, as v^{tol}, based on the value chosen for COSCAL in Van de Weg et al. (2014), which gives $\gamma = 0.5$.

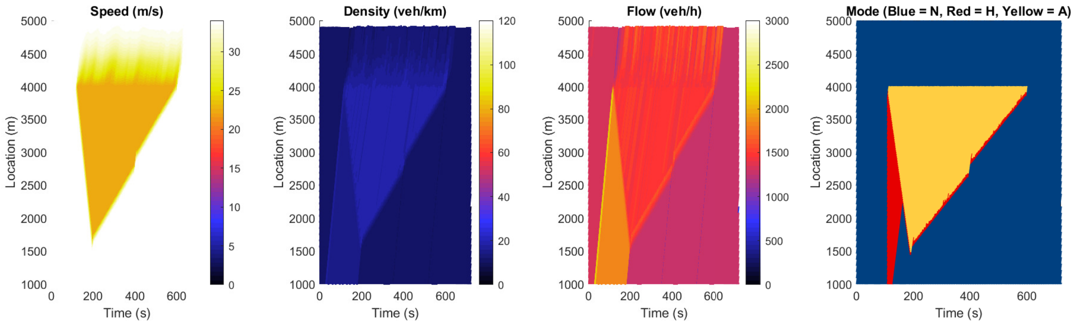


Figure 5.4: Trajectories for $v^{\text{target}} = 80$ km/h, $\rho^{\text{target}} = 17.9$ veh/km and $x^{\text{CR}} = x^{\text{u-BN}}$.

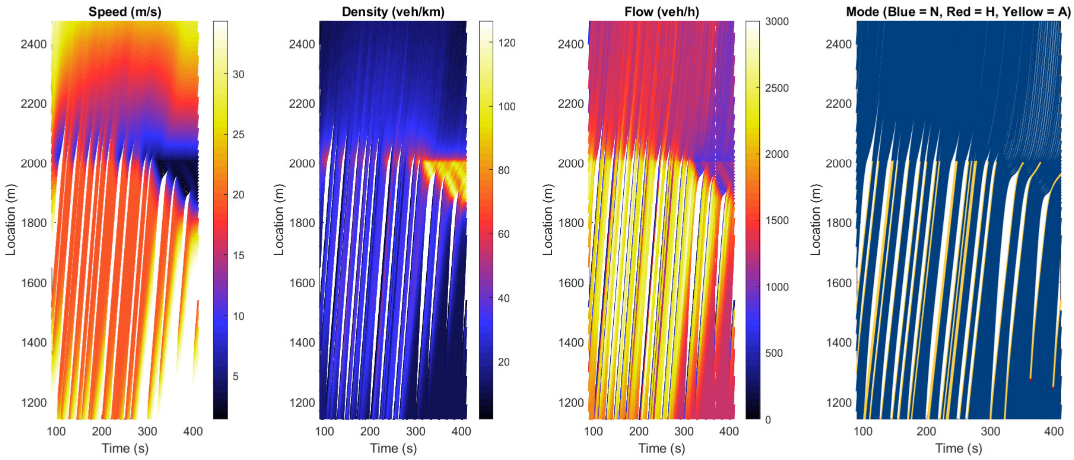


Figure 5.5: Trajectories for $v^{\text{target}} = 70$ km/h, $\rho^{\text{target}} = 20.5$ veh/km and $x^{\text{CR}} = x^{\text{u-BN}}$.

5.3.8 Speed control release

The point where control is released is defined as x^{CR} and is logical to be at two possible locations. Either control is released at the beginning of the bottleneck, or after passing the end of the bottleneck. For further elaboration, read back in Section 3.2.4.

In the example given at the beginning of this chapter (Figure 5.1), the release parameter was set to $x^{\text{CR}} = x^{\text{d-BN}}$. However, when changing this parameter, behaviour of the algorithm is changed slightly. As long as the penetration rate remains high, the discharge rate from the controlled area can not become higher than the bottleneck capacity (state E from Figure 3.1). Therefore, when releasing traffic at the upstream end of the bottleneck, traffic remains stable as shown in Figure 5.4.

However, when penetration rates are reduced, the algorithm relies on instabilities being dampened by the voids created by upstream CVs. When vehicles are released at the upstream end of the bottleneck, this may result in these voids not being dampened, as upstream vehicles will accelerate into unstable traffic, resulting in oscillations. An example of this is given in Figure 5.5. The point where x^{CR} should be changed from the upstream end to the downstream end of the bottleneck therefore depends on the penetration rate, but also the spread of CVs along the traffic stream. Large clusters of non-equipped vehicles could cause oscillations to develop past the precursor stage, as explained in Section 2.1.1. As long as the oscillations do not get to the growth stage of development, they can be dampened by upstream CVs with the right tuning.

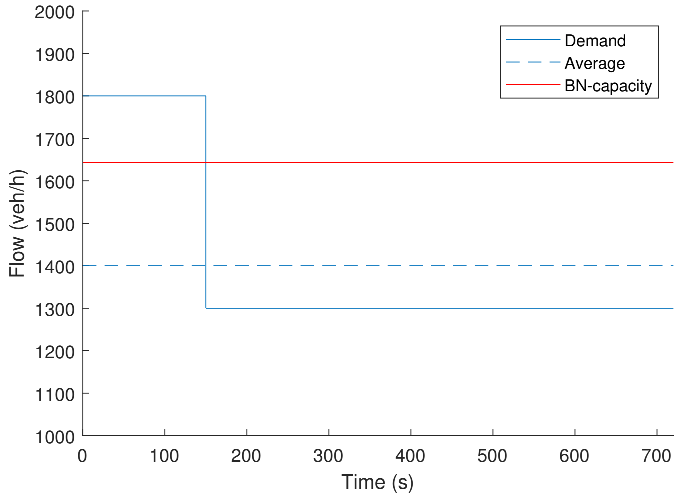


Figure 5.6: Demand pattern during the simulations

5.4 DEMAND PATTERN

In Section 5.3.1, the theoretical bottleneck capacity was defined as 1643. Since the goal of the controller is to handle demand situations which would fit through the bottleneck on average, but cause a breakdown due to heterogeneity, a demand pattern has to be found with an average lower than the bottleneck capacity. Furthermore, in order to reduce the simulation times, it is desired that the controlled area remains small. Therefore, the flow after the high-flow area is preferred to be relatively low. The initial state of the network is filled with homogeneous vehicles at a flow of 1300 veh/h, followed by the demand pattern displayed in Figure 5.6.

5.5 EVALUATION CRITERIA

For the evaluation of the algorithm, multiple criteria will be used. In the following paragraphs the indicators for network performance and traffic safety will be presented, as well as the method for face validation of stability in the network.

5.5.1 Network performance

Since in the network demand is equal for each simulation run, TTS is a good indicator of the impact of the controller. Lower values of TTS mean that vehicles leave the network earlier and that bottleneck outflow is therefore higher. For this indicator, base cases for uncontrolled and controlled scenarios will be defined for comparison of the further evaluation of the algorithm. The TTS is calculated in minutes by taking the sum over all vehicles i in the set I of the total number of time steps each vehicle was in the system, which is defined as K_i.

$$TTS = \frac{T^{\text{ctrllr}}}{60} \sum_{i=1}^{I} K_i \tag{5.8}$$

where T^{ctrllr} is the time-step length of the controller, which is 0.2s, which is divided by 60, since the TTS is measured in minutes.

5.5.2 Safety analysis

Since the driver model used is crash free, a surrogate safety measure has to be used. Multiple of these safety measures are discussed and evaluated by Mullakkal-Babu et al. (2017). From this evaluation, it was concluded that the inverse Time-To-Collision (iTTC) (1/s) is a good indicator for evaluation of longitudinal safety, which is widely used in Adaptive Cruise Control systems and to asses human driver behaviour (Moon et al., 2009). As the simulations are performed with a single lane, this is a good choice.

The iTTC is computed with the following formula:

$$iTTC = \max\left[0, \frac{v_i(k) - v_{i-1}(k)}{s_i(k)}\right] \tag{5.9}$$

where s_i is the spacing between vehicle i and $i-1$. For this indicator, higher values mean lower safety. The iTTC is expected to be high in simulation runs where a jam occurs, since speed differences are higher and headways are shorter in these situations. Finally, the maximum value for the iTTC for each run is stored. From all simulation runs, the average of the maximum, as well as the minimum and the maximum of the maximum value of iTTC is given.

5.5.3 Stability

From the trajectory plots it could be derived if the flow in a simulation run was stable or not. This can be derived by looking at the speed, density and flow plots. When speeds become lower than v^{target} and density increases as well, there are instabilities in the network. When this starts to propagate in the upstream direction, a jam was created. Furthermore, oscillations can be detected with the same method. However, these do not have to propagate in the upstream direction. This will be important especially during the evaluation of the controller for lower rates of penetration and compliance.

5.6 MODELLING COMPLIANCE

Since it is desired to look into the effects of compliance to the advice given by the controller, as it may seriously reduce the effectivity of the algorithm, compliance has to be incorporated into the evaluation of the algorithm.

In Section 2.3 it was found that the main way of not complying to the control signals, was not reducing speed to v^{target}, but to a higher value. In order to model compliance, a compliance factor ω_i^{speed} is defined, which is a factor between zero and one, where zero means no compliance at all and one results in full compliance.

5.6.1 Changes to the driver model for compliance

For speed compliance, the desired speed difference of the algorithm can be multiplied with ω_i^{speed} when $M_i(k) == A$:

$$v_i^{\text{ctrl}}(k) = (1 - \omega_i^{\text{speed}}) * (v - v^{\text{target}}) + v^{\text{target}} \tag{5.10}$$

When adding speed compliance to the model, it should also be determined how the compliance factor is varied between drivers. For the case that each driver receives the same compliance factor, it becomes a tuning problem, since setting a lower target speed would then result in the controller working as intended. This is not desired during the evaluation, as it is more interesting to see what would happen if the compliance of vehicles differs. Therefore, a random distribution should be used. During the evaluations for compliance, a uniform distribution is used with the interval $[\omega^{\text{speed}}, 1]$, where ω^{speed} is the value that defines the minimum level of compliance.

From this interval, each vehicle receives a individual value for its compliance with the following formula which is called each time a vehicle is created:

$$\omega_i^{\text{speed}} = \omega^{\text{speed}} + (1 - \omega^{\text{speed}})\phi \tag{5.11}$$

where ϕ is a random number drawn from a uniform distribution between 0 and 1.

5.7 CONCLUSION

In this chapter, the simulation model that is used for the evaluation of the controller is developed. Firstly, the driver model is chosen, which is IDM+. Some adaptations to the model, related to changes in desired speed of vehicles, were required to make it usable for the evaluation of this controller. Secondly, the network is presented, including the method of simulating a bottleneck, which was based on increasing the desired time headway of vehicles when entering the bottleneck. Thirdly, all parameters of the controller that can be varied are discussed and tuned when possible. Fourthly, the demand pattern that will be used for the majority of the evaluation is presented. Fifthly, the evaluation criteria are discussed. Lastly, the method of modelling compliance is presented. These steps are preparations for the evaluation of the controller, which is performed in the following chapter.

6 RESULTS

In this chapter, the algorithm developed in Chapter 4 will be evaluated with the simulation results. This will be done by simulating a wide variety of scenarios of the synthetic case in the simulation model developed in Chapter 5. The goals of the evaluation are to show the following:

- The algorithm improves throughput on the freeway;
- The algorithm prevents jams;
- The algorithm can cope with multiple demand situations;
- The effects of reduced penetration rate on the effectiveness of the algorithm
- The effects of non-compliance to control signals;
- The effects of a deceleration mismatch;
- The safety benefits of the algorithm.

As mentioned in Section 5.5, there are multiple indicators that will be used for the evaluation. Throughput will be measured with Total Time Spent (TTS) (min), safety with the inverse Time-To-Collision (iTTC) and stability can be validated by looking at the trajectory plots that come from the simulation model. Moreover, the changes in dynamics within the controlled area due to changes in i.e. penetration rate will be analysed with these trajectory plots as well. For each performance measure, the average, minimum and maximum values over the 10 runs will be presented and compared with both the uncontrolled and controlled base cases.

Since there are random varieties in the IDM parameter T of drivers in the simulation model, each scenario will be run 10 times with the same set of seeds. Firstly, to be able to compare results, a base case without control will be simulated. Secondly, the algorithm will be tuned for maximum performance in the synthetic case as described in Chapter 5, which will also be used for comparison with the following scenarios.

Then, the ability of the algorithm to cope with multiple variations in demand will briefly be validated. A small set of different demands will be simulated. For analysis of the trajectory plots, face validation of the onset of control will be used.

Subsequently, the effects of reduction in penetration rate will be presented with simulation results for multiple rates of penetration. Each time the rate is reduced, the ρ^{target} will be tuned to optimize the TTS. This is done by running 10 simulations for multiple values of ρ^{target}, which will be reduced with 0.1veh/km per step, until the optimal average TTS is found. During testing, it was found that the average TTS over 10 runs follows a convex curve when tuning ρ^{target}. Therefore, when the average TTS increases on both sides of a certain value for ρ^{target}, it is accepted as the optimal value for the current scenario.

Next, the same process will be performed for a reduction in compliance of drivers, which is implemented as explained in Section 5.6. The same tuning process as for the evaluation of reduction in penetration rate is used.

The last factor that will be evaluated is the effect of a deceleration mismatch between the expectation of the algorithm (a^{ex}) and the actual deceleration of the vehicles. The mismatch will be applied to multiple scenarios with varying penetration and compliance rates.

6.1 BASE CASES

In this section, the base cases which are used for comparison during the evaluation are defined and the tuning parameters used in the base case are given. Furthermore, the results for the quantitative performance measures are presented for both the uncontrolled and controlled base cases.

6.1.1 Uncontrolled base case

The uncontrolled scenario is used as a base case to compare the results of the control scenarios in terms of improvements to the aforementioned evaluation criteria. This scenario consists of a high-flow area passing through the bottleneck as defined in Section 5.2, which causes instability in traffic. As can be seen in Figure 6.1, the instability causes a jam and activates the capacity drop with a discharge flow of around 1100 veh/h when looking at the figure. The jam does not resolve and traffic is delayed. The quantitative results can be found in Table 6.2.

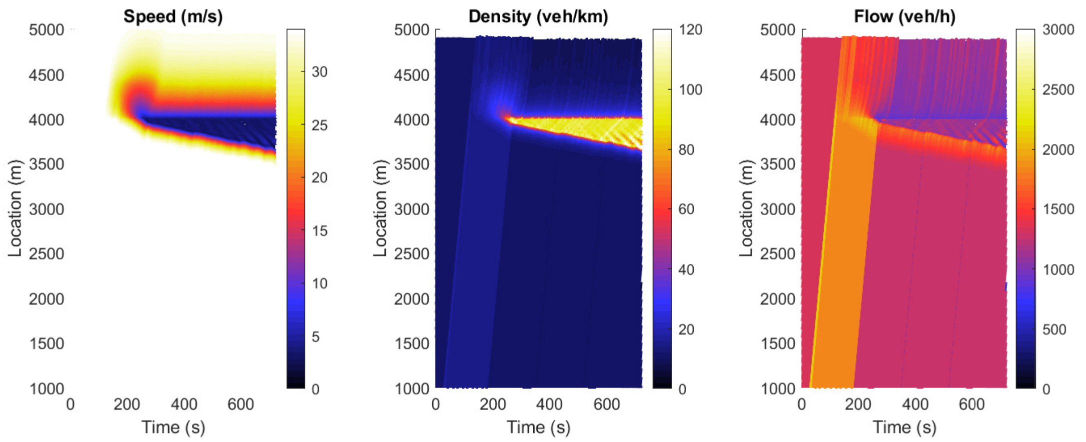


Figure 6.1: Trajectories for the uncontrolled case. Due to the high-flow area, a jam occurs at the bottleneck entrance and the capacity drop is activated. The jam does not resolve, but grows.

6.1.2 Controlled base case

In this scenario the flow is optimized with the maximum target speed of 80km/h. Since this is maximum speed for trucks, higher speeds can not be used when the algorithm would be implemented onto real roads and are therefore not taken into account. The values used for each of the other tuning parameters can be found in Table 6.1. Unless stated otherwise, these parameters are used during the evaluation of the algorithm.

Table 6.1: Values for tuning parameters

Parameter	n	γ	a^{ex}	$x^{\text{u-m}}$	$x^{\text{d-m}}$	x^{control}	x^{CR}	v^{tol}	v^{max}	$Q^{\text{threshold}}$
Value	4	0.5	2.299	3798	3809	4000	4000	1	130	1643

In the simulations, it was found that $\rho^{\text{target}} = 18.6\text{veh/km}$ is the maximum density for $v^{\text{target}} = 80\text{km/h}$ that remains reliably stable through the bottleneck over the simulation runs. Higher target densities could not reliably pass the bottleneck. This shows that for a constant flow, the bottleneck capacity is 1488veh/h for a speed of 80km/h. For higher speeds, the possible target flow is higher, so it may be useful to find a way to increase speeds in some lanes. Due to time constraints this is not further investigated within this thesis.

The trajectories of one of the runs for these settings can be found in Figure 6.2, showing that the jam which was created in the uncontrolled case is prevented. In the trajectory plot of flow, the traffic states from Figure 3.1b were drawn. A homogeneous traffic state is created in the controlled area, which is able to pass through the bottleneck without any instabilities, just as expected from the theory.

Furthermore, the quantitative results can be found in Table 6.2. As can be seen, the algorithm has strong positive effects for both TTS and iTTC. Since the jam is prevented, there is no capacity drop and vehicles pass through the network faster. An important note, is that in the uncontrolled case the jam is not resolved and will cause more delays when a longer time period is simulated. Moreover, large safety benefits are found for the base case. This is due to the reduction in speed differences between vehicles (since there is no jam), as well as the controlled onset of deceleration of vehicles.

From this point, the resulting TTS and iTTC will be compared with both base cases. Comparison with the uncontrolled base case will be presented under Δ_u, while comparison with the controlled base case will be presented under Δ_c. Both are computed as the difference in percentage between the result of the current case against the result of either the uncontrolled base case for Δ_u and for the controlled base case for Δ_c. Only for the varying demand patterns, face validation is the only method used for evaluation.

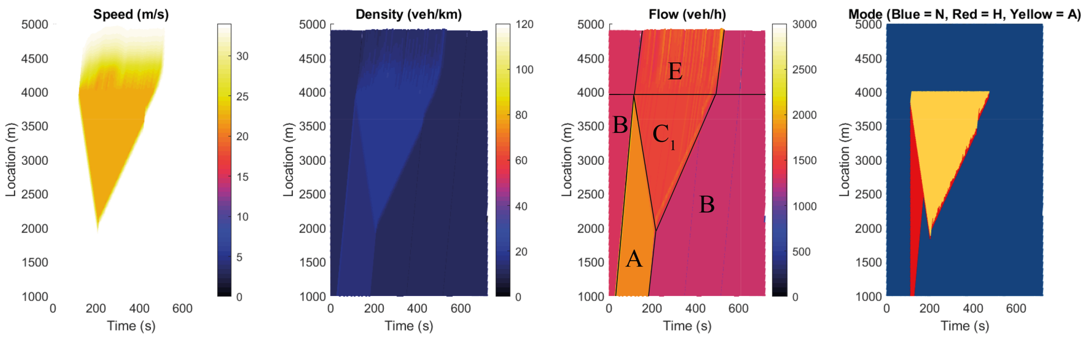


Figure 6.2: Trajectories for $v^{\text{target}} = 80$km/h and $\rho^{\text{target}} = 18.6$ veh/km. The traffic state upstream of the bottleneck is successfully manipulated into a state that can stably pass the bottleneck due to the control signals.

Table 6.2: Results for the uncontrolled and controlled base case. From this, it can be concluded that the controller can prevent a jam and its negative effects on throughput and safety in the base case.

Scenario	Target density		TTS	SD	Δ_u	Δ_c	iTTC	SD	Δ_u	Δ_c
Uncontrolled	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
		Min.	826.89	-	-	+14%	0.345	-	-	+190%
		Max.	850.01	-	-	+16%	0.399	-	-	+224%
Controlled	18.6	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
		Min.	728.02	-	-12%	-	0.119	-	-66%	-
		Max.	730.67	-	-14%	-	0.123	-	-69%	-

6.2 CHANGES IN DEMAND PATTERNS

In this section, multiple cases of demand will be presented to show that the algorithm is able to cope with multiple demand scenarios. Since a reduced penetration rate may influence the behaviour of the controller, the situations are simulated with two penetration rates: 100% and 50%. First, two high-flow areas will be sent into the network. Secondly, many small clusters of high-flow vehicles will be created to see how to the controller copes.

6.2.1 Multiple areas A

The trajectories for this demand pattern can be found in Figure 6.3. When multiple areas of high flow are inserted into the network, the controller handles it well in the 100% penetration rate scenario. As can be seen from the figure, a homogeneous traffic state is created in the controlled area. When simulating this demand pattern with a lower penetration rate, the same conclusions could be made, so these trajectories can be found in Appendix B.

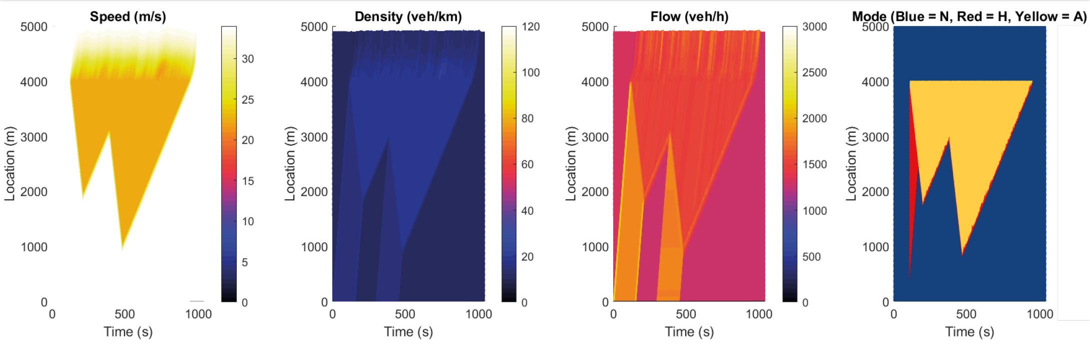


Figure 6.3: Trajectories for two high-flow areas with 100% penetration rate. The controller is able to successfully attach the second high-flow area to the first.

6.2.2 Small clusters of high flow

In this demand scenario, short clusters of high-flow vehicles follow each other in the network. The trajectories for the demand scenario can be found in Figure 6.4. What can be seen in the figure, is that the vehicles join the controlled area successfully as long as the areas of high flow follow each other close enough. When the distance between the controlled area and the following high-flow area is one vehicle too large, not all vehicles receive a target line, since these vehicles would get a target line that is too far downstream. In this case, this results in two vehicles with a short headway passing the bottleneck, but not causing any oscillations.

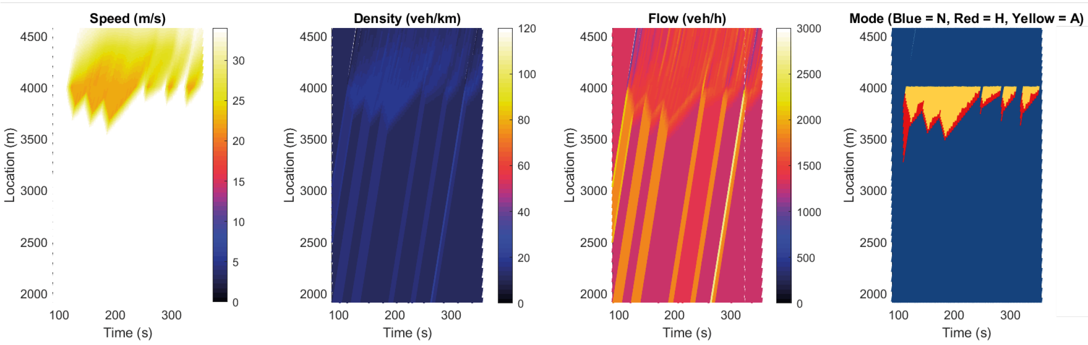


Figure 6.4: Trajectories for multiple areas of high flow in short succession with 100% penetration rate. Traffic remains stable and the controller is able to handle the multiple bursts of short demand and is reactivated when required.

When simulating the same demand pattern with a penetration rate of 50%, a problem occurs. As can be seen from the trajectory plots in Figure 6.5, when the upstream end of the controlled area passes the bottleneck while there are no Connected Vehicles (CVs) following closely, the control scheme finishes. This results in small oscillations in the cluster of vehicles that was following the tail of the controlled area.

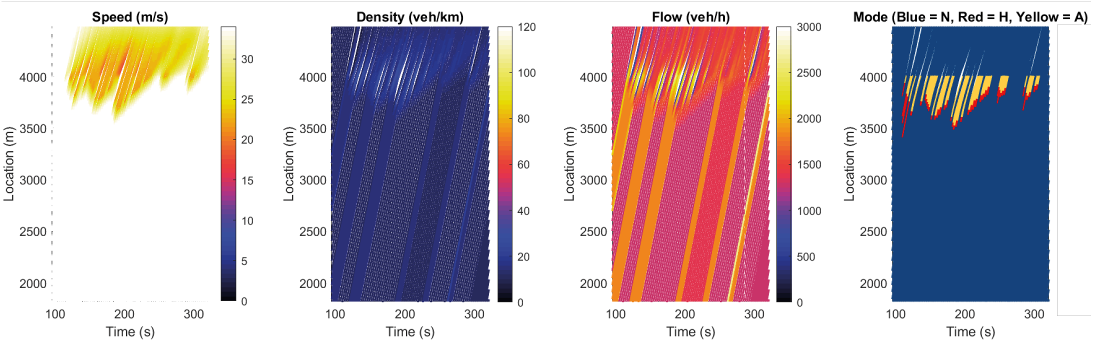


Figure 6.5: Trajectories for multiple areas of high flow in short succession with 50% penetration rate. Due to the reduced span of control, traffic becomes slightly heterogeneous in the controlled area, but remains stable.

6.3 REDUCTION IN PENETRATION RATE

In this section, the effects of a reduction in penetration rate are analysed. For the analysis, the penetration rate of CVs is reduced in steps. Also, all CVs in the network are randomly distributed. The target density of the controller is tuned for each step with the procedure explained earlier in this chapter.

As mentioned in Section 5.3, the location of speed release (x^{CR}) should be taken into account for reduced penetration rates. Therefore, the chosen location is also presented in the table. In the following paragraphs, each level of penetration that is tested, will be discussed.

6.3.1 Penetration rate of 50%

For the first step in reduced penetration rate of CVs, there is not much that changes compared to the controlled base case. A 0.1 reduction in ρ^{target} resulted in slightly better performance in terms of TTS, so the results for that tuning are presented in the first row of results in Table 6.3. As can be seen from the table, there is only a slight performance loss compared to the controlled base case. Since only half of the vehicles on the road are CVs, small clusters of vehicles are formed behind a CV that has received a control signal. These small clusters oscillate slightly when entering the bottleneck. This resulted in traffic instability for some cases with a higher ρ^{target}, thus some tuning was required.

When looking at the safety effects, it is obvious that reduced penetration results in smaller safety benefits. Since the controlled area is not homogeneous, speed differences between vehicles increase and stronger deceleration occurs when clusters of non-equipped vehicles enter the bottleneck. However, since no jam occurs, there is still a large improvement for traffic safety when comparing to the uncontrolled base case.

6.3.2 Penetration rate of 25%

For a penetration rate of 25%, it becomes harder to keep traffic stable. Since there are less CVs, the distribution of CVs along the road is less equal. This results in a higher probability of a large cluster of uncontrolled vehicles forming behind a CV. These clusters increase the risk of instabilities in traffic, since entering the bottleneck causes oscillations in the non-equipped clusters of vehicles. The target density was therefore required to be severely reduced to $\rho^{\text{target}} = 17.8\text{veh/km}$. Also, since even larger clusters of non-equipped vehicles enter the bottleneck in this scenario, oscillations are larger. Therefore, x^{CR} is set to $x^{\text{d-BN}}$ for this scenario.

From the figures (show fig with late breakdown), it can be seen that the algorithm performs well as long as the distance between consecutive CVs is not too large. When the distance becomes too large, the oscillations that occur require more space to fade out. This occurred twice out of the ten simulation runs.

The increase in oscillations can also be seen in the results for the iTTC. While still improving safety on the road when compared to the uncontrolled base case, the increase in oscillations decreases the safety benefits when compared to the controlled case.

6.3.3 Penetration rate of 10%

In the scenario with 10% penetration rate, jams become hard to prevent. Many large clusters of non-equipped vehicles form behind CVs and cause large oscillations when entering the bottleneck. This results in earlier onset of a jam than without control and thus more delays. In Figure 6.6, an example is given of a situation where CVs are distributed along the road well enough for a short while to almost prevent a jam. However a breakdown still occurs. So with the current state of this algorithm, a penetration rate of 10% is not enough to reliably prevent jams. Therefore, other algorithms which have the goal of resolving a jam wave may perform better. Besides the bad performance in TTS, safety is severely affected as well. In the worst situation, the iTTC scored higher than in the uncontrolled scenario. This leads to the same conclusion, that a 10% penetration rate is not effective.

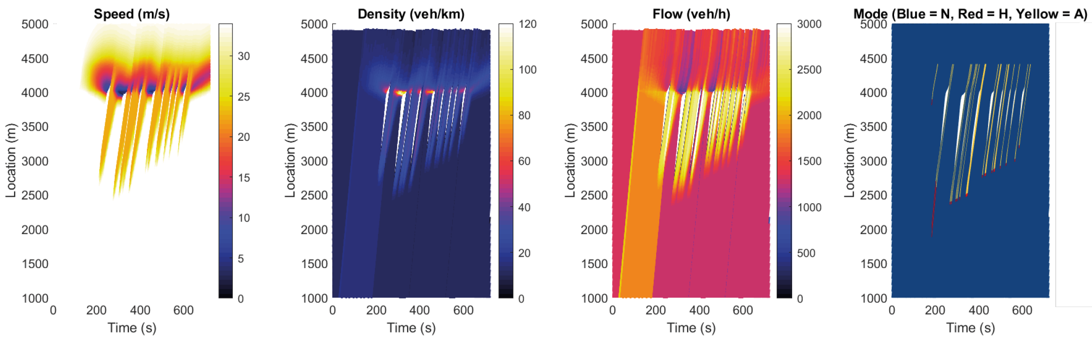


Figure 6.6: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 16.5$ veh/km and **10% penetration rate**. The random distribution of CVs leads to large clusters of unequipped vehicles which oscillate when entering the bottleneck.

6.3.4 Low penetration rates with equal spread of CVs

In the previous simulations, CVs were placed at random points in traffic. However, it is also of interest to see what is possible when CVs are equally spread among traffic. In Figure 6.7, an example of an equal spread of CVs at 5% penetration rate can be found. As can be seen, each time a CV with a cluster of non-equipped vehicles behind it passes the bottleneck, there is just enough space to let the oscillations fade out. Traffic remains stable, but since the flow has to be constrained to $q^{\text{target}} = 1280\text{veh/h}$, it does not improve the performance of the network.

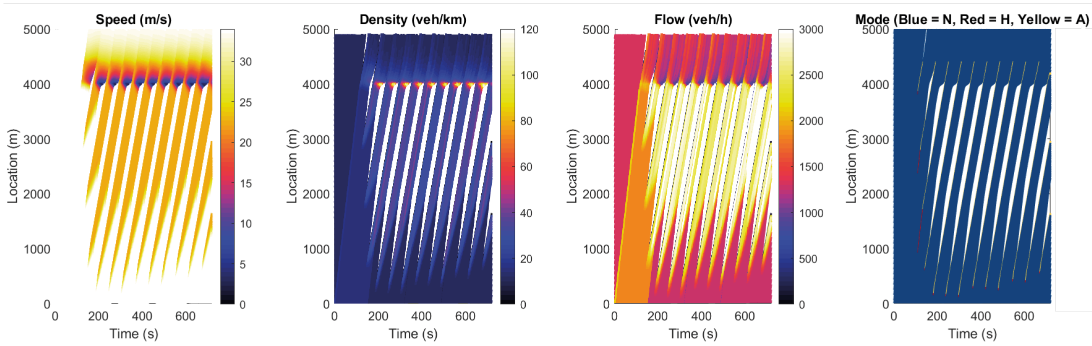


Figure 6.7: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 16$ veh/km and **a CV every 20 vehicles**. Traffic remains stable for a short while, but breaks down eventually.

Table 6.3: Results for multiple levels of penetration in comparison with the base cases. It can be seen that decreased penetration rate of CVs severely reduces the effectiveness of the controller for both throughput and safety. However, at 50% penetration rate there is only a slight difference in TTS. The best location for speed-control release switches to the downstream end of the bottleneck for lower penetration rates

Scenario	**Target density**	**Control release**		**TTS**	SD	Δ_u	Δ_c	**iTTC**	SD	Δ_u	Δ_c
Base case: Uncontrolled	-	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
			Min.	826.89	-	-	+14%	0.345	-	-	+190%
			Max.	850.01	-	-	+16%	0.399	-	-	+224%
Base case: Controlled	18.6	Upstream end	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
			Min.	728.02	-	-12%	-	0.119	-	-66%	-
			Max.	730.67	-	-14%	-	0.123	-	-69%	-
50% penetration	18.5	Upstream end	Avg.	735.85	6.11	-12%	+1%	0.136	0.013	-63%	+13%
			Min.	728.97	-	-12%	0%	0.125	-	-64%	+5%
			Max.	748.56	-	-12%	+2%	0.161	-	-60%	+30%
25% penetration	17.8	Downstream end	Avg.	783.54	45.66	-6%	+8%	0.217	0.079	-41%	+80%
			Min.	736.31	-	-11%	+1%	0.129	-	-63%	+8%
			Max.	903.72	-	+6%	+24%	0.353	-	-12%	+187%
10% penetration	16.5	Downstream end	Avg.	860.09	46.98	+3%	+18%	0.336	0.080	-8%	+180%
			Min.	758.14	-	-8%	+4%	0.191	-	-45%	+61%
			Max.	917.67	-	+8%	+26%	0.443	-	+11%	+261%

6.4 REDUCTION IN DRIVER COMPLIANCE

While both penetration rate and non-compliance result in vehicles not following up on control signals, their exact effects are still different. Non-compliance is not binary, vehicles can follow up the speed-control signals partly. Therefore, in this section the effects of reduced compliance are analysed.

In this test, compliance is modelled as explained in Section 5.6. Since there is no real information on what percentage of people comply to in-car speed control signals, since compliance is dependent on many factors, as found in Section 2.3. Furthermore, multiple levels of non-compliance are simulated. Each vehicle receives an individual value for ω_i^{speed} from a uniform distribution. Furthermore, the penetration rate in this section is constant at 100% and $x^{\text{CR}} = x^{\text{u-BN}}$.

The simulation results can be found in Table 6.4. As can be seen, larger reductions in compliance to control signals reduce the performance for both network performance and safety more. Also, not every jam could be prevented in scenarios with 40% or lower potential minimum compliance, since these scenarios resulted in more heterogeneous traffic, which can become unstable when entering the bottleneck.

Especially the dynamics within the controlled area are interesting for these scenarios. As can be seen from Figure 6.8, the homogeneity of the controlled area is affected by the reduction in compliance. However, the effects are found to be different than the reduced penetration rate. Reduced compliance causes less heterogeneity than reduced penetration rate, since vehicles still decelerate, but to a lesser extend. High-density clusters of vehicles are also formed, but they are smaller. This is of course reliant on the distribution of compliance that is present on the road.

Moreover, during testing it was found that, unlike in the reduced penetration scenarios, moving x^{CR} to $x^{\text{d-BN}}$ was not beneficial for the performance of the algorithm. Further analysis showed that this is the case due to the unequal spread of vehicles. Where the voids between clusters of vehicles were able to dampen the oscillations in the reduced penetration scenarios, the voids were slightly smaller in the non-compliance scenarios due to CVs that fill the gaps due to non-compliance. Therefore, it was better to release speed-control earlier, making it possible to increase the distance with the follower CV.

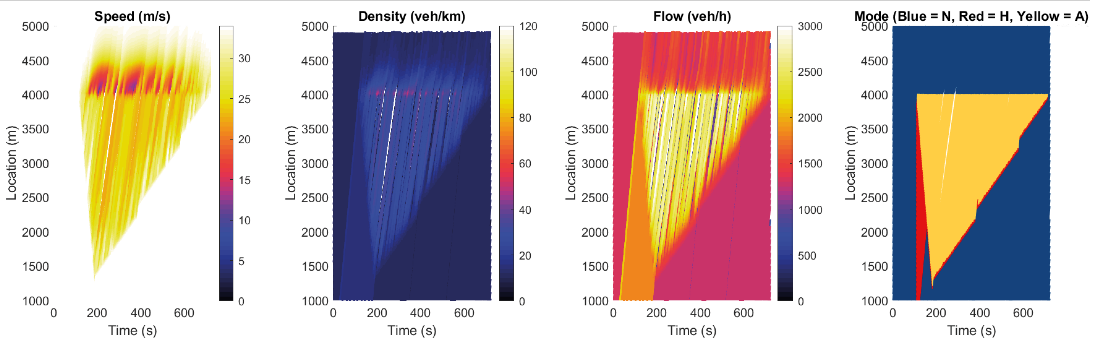


Figure 6.8: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 17.5$ veh/km and max. non-compliance of 0%. Due to non-compliance of vehicles, traffic is heterogeneous, resulting in some oscillations downstream of the bottleneck entrance but traffic remains stable.

Table 6.4: Results for multiple levels of non-compliance in comparison with the base cases. A reduction in compliance results in higher TTS, as traffic becomes unstable due to the small high-density clusters that form in the traffic stream. A reduction in target density helps with stability but does not result in stable traffic each run. Due to an increase in heterogeneous traffic, safety benefits are also reduced.

Scenario	**Target density**		**TTS**	SD	Δ_u	Δ_c	**iTTC**	SD	Δ_u	Δ_c
Base case: Uncontrolled	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
		Min.	826.89	-	-	+14%	0.345	-	-	+190%
		Max.	850.01	-	-	+16%	0.399	-	-	+224%
Base case: Controlled	18.6	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
		Min.	728.02	-	-12%	-	0.119	-	-66%	-
		Max.	730.67	-	-14%	-	0.123	-	-69%	-
Compliance range of [0.8 , 1]	18.6	Avg.	734.72	9.48	-12%	1%	0.130	0.014	-65%	+8%
		Min.	724.13	-	-12%	-1%	0.117	-	-66%	-2%
		Max.	750.82	-	-12%	+3%	0.162	-	-59%	+32%
Compliance range of [0.6 , 1]	18.0	Avg.	744.56	9.37	-11%	+2%	0.153	0.027	-58%	+28%
		Min.	733.55	-	-11%	+1%	0.116	-	-66%	-3%
		Max.	775.72	-	-9%	+6%	0.279	-	-30%	+127%
Compliance range of [0.4 , 1]	18.0	Avg.	761.14	20.09	-9%	+4%	0.196	0.074	-47%	+63%
		Min.	742.24	-	-10%	+2%	0.0.126	-	-63%	+6%
		Max.	797.14	-	-6%	+9%	0.339	-	-15%	+176%
Compliance range of [0.2 , 1]	17.7	Avg.	776.95	24.29	-7%	+7%	0.224	0.069	-39%	+87%
		Min.	756.56	-	-9%	+4%	0.147	-	-57%	+24%
		Max.	833.7	-	-2%	+14%	0.328	-	-18%	+167%
Compliance range of [0 , 1]	17.5	Avg.	790.96	24.54	-6%	+9%	0.244	0.070	-34%	+103%
		Min.	769.56	-	-7%	+6%	0.171	-	-50%	+44%
		Max.	845.38	-	-1%	+16%	0.365	-	-9%	+197%

6.5 DECELERATION MISMATCH

Since differences in deceleration are also a factor for reducing the homogeneity in traffic resulting from control, it is analysed how this affects the controller. To simulate the effects of this, vehicles receive a value for the parameter b of the driver model according to a normal distribution with $\mu = 2.09$ and $\sigma = 0.5$ in the 100% penetration scenario. The results can be found in Table 6.5.

Table 6.5: Results for the base case with deceleration differences in comparison with the other base cases. It can be seen that in terms of TTS, there are only slight differences between the controlled base case and the deceleration mismatch base case. However, safety is reduced due to an increase in heterogeneity in the traffic stream.

Scenario	Target density		TTS	SD	Δ_u	Δ_c	iTTC	SD	Δ_u	Δ_c
Uncontrolled	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
		Min.	826.89	-	-	+14%	0.345	-	-	+190%
		Max.	850.01	-	-	+16%	0.399	-	-	+224%
Controlled	18.6	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
		Min.	728.02	-	-12%	-	0.119	-	-66%	-
		Max.	730.67	-	-14%	-	0.123	-	-69%	-
Mismatch	18.6	Avg.	726.18	3.12	-13%	0%	0.134	0.003	-63%	+12%
		Min.	724.77	-	-12%	0%	0.130	-	-62%	+10%
		Max.	730.03	-	-14%	0%	0.140	-	-65%	+14%

As could be expected, the controlled area becomes slightly heterogeneous. However, the heterogeneity does not cause any instabilities in simulations for the 100% penetration rate scenario.

One thing that would be expected with larger differences in the deceleration, is that target line resets would occur. However, with the original tuning of $\gamma = 0.5$, resets rarely occur. Even when reducing γ to a value of 0.2, target line resets are rare, meaning that the deceleration differences in the model do not cause vehicles to decelerate so fast that they reach v^{target} far enough from their target line to trigger a reset. Even with differences between a^{ex} and the actual deceleration, the target line resets only occur when traffic breaks down. This is currently not in the design domain of this controller and is therefore not discussed.

To still be able to show the effects of strong deceleration, a scenario was run where some vehicles were given $b = 15$. These vehicles do trigger a target line reset. The trajectory results are displayed in Figure 6.9. As can be seen, the target line reset results in all vehicles joining the controlled area to be decelerated earlier than they would have been without the reset. As can be seen from the density and flow figures in Figure 6.9, small heterogeneities occur due to the reset. However, these small areas actually have lower flow and density, meaning that they do not pose any risk for stability.

6.5.1 With reduced penetration rate

When simulating the deceleration differences in combination with reduced penetration rate, only minor changes occur, as can be found in Table 6.6. Both the TTS and iTTC perform slightly worse than in the scenario with no deceleration differences. This is a positive result, since this means that small heterogeneities due to different decelerations do not necessarily cause extra delays or instability as long as the flow is controlled and on average remains homogeneous.

Other than the small heterogeneities, which were also present in the 100% penetration rate scenario, no new dynamics within the controlled area were found in this scenario.

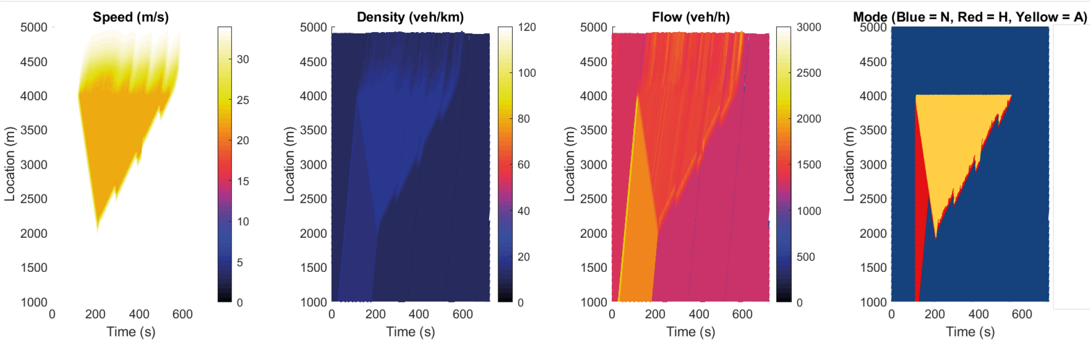


Figure 6.9: Three vehicles with $b = 15$ result in successful target line resets while traffic remains stable.

6.5.2 With reduced compliance

In this paragraph, the results of a combination of non-compliance and deceleration differences are presented. The quantitative results can be found in Table 6.7.

For the combination of deceleration differences and reduced compliance, only small reductions in performance can be found when compared to the scenario without deceleration differences. Notable is that for the 60% and 0% case, a higher density resulted in slightly better performance than without deceleration differences. The iTTC is affected the most by the deceleration differences when compared to the non-compliance scenarios without it. This is obvious, since the increased heterogeneity in the controlled area results in shorter headways.

Since the dynamic of the slowest vehicles forming clusters that pass the bottleneck does not change much, the differences in TTS are small. The only difference is that the clusters of vehicles might shift a little bit in the upstream or downstream direction.

Table 6.6: Results for multiple levels of penetration in combination with a deceleration mismatch in comparison with the base cases. Important to note is that the differences with the results from Table 6.3 are small and therefore a deceleration mismatch does not result large performance losses for the controller.

Scenario	**Target density**	Control release		**TTS**	SD	Δ_u	Δ_c	**iTTC**	SD	Δ_u	Δ_c
Base case: Uncontrolled	-	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
			Min.	826.89	-	-	+14%	0.345	-	-	+190%
			Max.	850.01	-	-	+16%	0.399	-	-	+224%
Base case: Controlled	18.6	Upstream end	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
			Min.	728.02	-	-12%	-	0.119	-	-66%	-
			Max.	730.67	-	-14%	-	0.123	-	-69%	-
Base case: Mismatch	18.6	Upstream end	Avg.	726.18	3.12	-13%	0%	0.134	0.003	-63%	+12%
			Min.	724.77	-	-12%	0%	0.130	-	-62%	+10%
			Max.	730.03	-	-14%	0%	0.140	-	-65%	+14%
50% penetration + mismatch	18.5	Upstream end	Avg.	735.33	5.94	-12%	+1%	0.148	0.011	-60%	+23%
			Min.	728.83	-	-12%	0%	0.134	-	-61%	+13%
			Max.	746.14	-	-12%	+2%	0.168	-	-58%	+37%
25% penetration + mismatch	17.7	Downstream end	Avg.	785.73	43.88	-6%	+8%	0.242	0.093	-34%	+102%
			Min.	740.89	-	-10%	+2%	0.143	-	-59%	20%
			Max.	903.96	-	+6%	+24%	0.407	-	+2%	+232%
10% penetration + mismatch	16.5	Downstream end	Avg.	844.85	27.11	+1%	+16%	0.310	0.068	-15%	+159%
			Min.	804.65	-	-3%	+11%	0.205	-	-40%	+73%
			Max.	893.57	-	+5%	+22%	0.388	-	-3%	+216%

Table 6.7: Results for multiple levels of non-compliance in combination with a deceleration mismatch in comparison with the base cases. Important to note is that the differences with the results from Table 6.4 are small and therefore a deceleration mismatch does not result large performance losses for the controller.

Scenario	**Target density**		**TTS**	SD	Δ_u	Δ_c	**iTTC**	SD	Δ_u	Δ_c
Base case: Uncontrolled	-	Avg.	837.94	5.99	-	+15%	0.367	0.015	-	+206%
		Min.	826.89	-	-	+14%	0.345	-	-	+190%
		Max.	850.01	-	-	+16%	0.399	-	-	+224%
Base case: Controlled	18.6	Avg.	728.72	0.71	-13%	-	0.120	0.001	-67%	-
		Min.	728.02	-	-12%	-	0.119	-	-66%	-
		Max.	730.67	-	-14%	-	0.123	-	-69%	-
Base case: Mismatch	18.6	Avg.	726.18	3.12	-13%	0%	0.134	0.003	-63%	+12%
		Min.	724.77	-	-12%	0%	0.130	-	-62%	+10%
		Max.	730.03	-	-14%	0%	0.140	-	-65%	+14%
Compliance range of [0.8 , 1] + mismatch	18.5	Avg.	731.88	6.26	-13%	0%	0.129	0.005	-65%	+8%
		Min.	725.89	-	-12%	0%	0.122	-	-65%	+3%
		Max.	746.22	-	-12%	+2%	0.141	-	-65%	+14%
Compliance range of [0.6 , 1] + mismatch	18.1	Avg.	745.14	11.26	-11%	+2%	0.161	0.044	-56%	+34%
		Min.	737.09	-	-11%	+1%	0.122	-	-65%	+3%
		Max.	775.18	-	-9%	+6%	0.270	-	-32%	+120%
Compliance range of [0.4 , 1] + mismatch	17.9	Avg.	759.86	19.16	-9%	+4%	0.205	0.080	-45%	+68%
		Min.	745.38	-	-10%	+2%	0.130	-	-62%	+9%
		Max.	801.83	-	-6%	+10%	0.355	-	-11%	+189%
Compliance range of [0.2 , 1] + mismatch	17.7	Avg.	775.78	24.82	-7%	+6%	0.226	0.074	-38%	+88%
		Min.	754.33	-	-9%	+4%	0.143	-	-59%	+20%
		Max.	834.67	-	-2%	+14%	0.337	-	-16%	+174%
Compliance range of [0 , 1] + mismatch	17.6	Avg.	787.43	26.14	-6%	+8%	0.268	0.094	-27%	+124%
		Min.	760.64	-	-8%	+4%	0.162	-	-53%	+36%
		Max.	842.57	-	-1%	+15%	0.406	-	+2%	+230%

6.6 CONCLUSIONS

In this chapter, the algorithm developed in this thesis is evaluated with the results of the simulations. The goals described at the beginning of this chapter will be discussed briefly in this conclusion.

At first, the algorithm was evaluated in terms of the ability to cope with multiple demand situations. Due to time constraints, this was only briefly done. Both large and small variations in demand were successfully controlled by the algorithm for multiple penetration rates. However, in the low penetration rate scenario, when the tail of the controlled area passes the bottleneck and speed-control is released, small oscillations occur which are not dampened by application of control. This is a point of improvement for the algorithm.

The algorithm was also able to improve throughput of the synthetic case in most scenarios by preventing the jam that occurred in the uncontrolled scenario. However, it was found that for the lower penetration rates, the algorithm could not reliably prevent the jams for this case. The cause of the instabilities in the network for the control scenarios was insufficiently large voids to dampen the oscillations that occurred when clusters of vehicles entered the bottleneck. As following vehicles enter the bottleneck while the leading vehicle is still part of an oscillation, instabilities arise and traffic breaks down.

The dynamic that results in a break down is slightly different for the different types of scenarios analysed during the evaluation. Firstly, the reduced penetration rate results in clusters of vehicles forming a high-density area behind each CV in the controlled area. When the void between a cluster and a following CV is not large enough to dampen the oscillations of the non-equipped vehicles, traffic instability arises.

Secondly, reduced speed compliance results in a similar dynamic. However, clusters of vehicles with a high density are not necessarily formed behind each CV, but depend on the distribution of vehicles that comply the most to the control signals. This effect results in a heterogeneous traffic state and can cause oscillations at the bottleneck. Furthermore, it was found that moving x^{CR} to $x^{\text{d-BN}}$ was not beneficial for this scenario. How this works for a combination of reduced compliance and penetration rate still has to be analysed.

Thirdly, when implementing differences in the deceleration of vehicles, the dynamics previously described remained the same. The only difference that was found, was that the controlled area was slightly more heterogeneous. This was also the case when combining these deceleration differences with both reducing the penetration rate, as well as reducing compliance.

Furthermore, the impact on traffic safety was found to be strong. Due to preventing a jam and decelerating vehicles in a controlled manner with the control signals, the iTTC is reduced severely, as speed differences are reduced. In the reduced penetration and compliance rate scenarios, the safety benefits were smaller, as the speed differences between vehicles increased again. Especially in the lower penetration rate scenarios, the clusters of non-equipped vehicles following a CV oscillated, increasing the speed differences and reducing safety. However, even in these cases, safety was improved.

7 CONCLUSIONS AND RECOMMENDATIONS

There has been a lot of research on traffic management strategies. However, there were no generic controllers available that proactively manipulated traffic on the mainline against the formation of a jam near recurring bottlenecks. Clusters of vehicles with a high flow can cause unnecessary traffic breakdowns. With the uprise of C-ITS, these vehicles become more and more controllable and thus, strategies to manipulate traffic using this technology should be developed. Therefore, the objective of this thesis was to design a traffic controller with the following properties:

- Aims to either prevent breakdown or reduce the probability of a breakdown near bottlenecks;
- Uses C-ITS for in-car messaging with low latency and high accuracy;
- Is mainly applied on freeways;
- Has a relatively short control scheme;
- Can be applied to a wide variety of bottlenecks with some tuning or adjustments.

The following sections will repeat the research questions that were formulated with the aforementioned design goals, present the important findings, discuss the control concept and its evaluation and give recommendations for both future work and policy.

7.1 FINDINGS FROM THE LITERATURE

To develop a traffic controller with these properties, a literature study had to be performed to build a sufficiently large knowledge base in as many important aspects as possible. Firstly, the properties resulted in the following main research question:

In what way can throughput of bottlenecks on freeways be improved, while using C-ITS technology to give in-car operational advice to drivers of CVs, without causing traffic instability in the network?

Then, multiple questions were determined of which the answers could be of importance for the development of the controller:

1. What are the mechanisms behind traffic breakdown on freeways?
2. What are important factors for predicting traffic breakdown on freeways and which of these factors can be influenced through control of CVs?
3. What are currently available mainline traffic control algorithms that use CVs and what mechanisms do they use?
4. What factors are found to influence compliance of drivers to control signals given through C-ITS communication channels and which of these can be taken into account when designing a controller?

Findings of these questions will be presented in the following paragraphs in their respective order.

Traffic jams can occur on any road as long as there are both a high traffic load and a disturbance in traffic. When these two ingredients are accompanied by any type of bottleneck, the chance on a disturbance is higher and thus the chance of a breakdown is higher as well. After a disturbance has occurred, oscillations propagate through traffic. The important finding from the literature is that this has multiple stages, of which the precursor stage is important for this controller. In this stage, oscillations are small and propagate in the downstream direction. When an oscillation grows large enough to start propagating in the upstream direction, the capacity drop is activated. Therefore, it is of high importance that oscillations do not grow past the precursor stage.

Since the controller aims to prevent the occurrence of jams, a predictor of jams had to be chosen. In the literature, it was found that flow, density and time headway distribution were important predictors of breakdown. Flow relates to the first ingredient of breakdown mentioned earlier: high traffic load. Density and time headway distribution are related, but not equal. Increased density increases the probability of a breakdown, while an equal time headway distribution is favorable in lower density situations, high-density traffic situations stay more stable with larger variance in time headways, but damping oscillations then becomes a game of chance. The important takeaway here was that harmonization of traffic near a bottleneck and controlling both flow and density can help with reducing the occurrence of jams near these bottlenecks.

From the review of the already available controllers, two conclusions were taken into the development of the controller. Firstly, none of the currently available controllers is generic for proactively manipulating mainline traffic to prevent jams at bottlenecks. However, the second conclusion that could be made was the idea of creating voids to dampen oscillations, which was used as a responsive measure to jam waves. This was taken into the theory for control with lower penetration rates. Furthermore, the computations from COSCAL, where the concept of target lines originates from, are adapted to the controller developed in this thesis.

There is not much literature available on the compliance of drivers to in-car control signals. The notion that an increased amount of information presented to a driver reduces compliance was incorporated by restricting control signals to speed control. Furthermore, a specific value for compliance is unknown, since it is not widely tested. Therefore, the effects of non-compliance are evaluated in this report to be able to account for the effects when more information has been gathered.

7.2 THE CONTROL CONCEPT AND EVALUATION

After the literature review, theory for the controller is developed and implemented in a control strategy which was later evaluated in a micro-simulation study. The controller aims to manipulate traffic upstream of a bottleneck by slowing vehicles down at a moment where they are expected to reach a vehicle-specific target line. Through this method, a harmonized traffic state is created at a desired speed and density. This target state can be chosen as a designer parameter. During the evaluation, the target speed was chosen to be 80km/h, as this is the maximum speed for trucks in the Netherlands. Many other parameters were introduced to make the controller applicable for a wide variety of bottlenecks and situations with some tuning. For each parameter, the logic and trade-offs for tuning these parameters are extensively discussed.

In the 100% CV penetration rate scenarios, the controller concept reduces the speed of vehicles and harmonizes traffic upstream of the bottleneck. It was found that in a perfect world scenario, the controller could prevent jams, as well as improve both throughput and safety greatly. When testing for non-compliance, the controller could still keep traffic stable for the majority of the simulations, while improving both throughput and safety. This did require some tuning of the target density. The same conclusion was made for this scenario when differences in deceleration of vehicles were implemented. Only a slight performance reduction was found. Although, safety benefits were reduced to a small degree.

For lower penetration rate scenarios, voids are used proactively to dampen oscillations that may occur when unequipped vehicles pass the bottleneck. Due to the CVs slowing down, clusters of non-equipped vehicles create a high-density area behind these CVs. When this cluster enters the bottleneck, oscillations occur and are then dampened by the voids. When reducing the penetration rate, these clusters of non-equipped vehicles become larger, requiring more space for oscillations. Therefore, the target density had to be reduced. While the controller was still effective for a penetration rate of 25%, the simulations for 10% were unsuccessful except for one run. This shows that the penetration rate severely limits the effectiveness of the controller, but that it may still be effective under the right circumstances. In a test with equally spread CVs and a penetration rate of 5%, the controller showed to be effective in preventing the jam. However, the target density had to be reduced to a value that resulted in a flow close to the capacity drop. Therefore, jam wave resolving algorithms will most likely result in better network performance in these cases.

This thesis has demonstrated that the developed control strategy fits within the description defined in the main design objective and research question. It is effective for preventing jams near bottlenecks on freeways by making use of C-ITS, as is shown in the evaluation. The length of the control scheme depends on the demand and parameters such as penetration rate of CVs, but is short and reduces delays effectively while keeping traffic stable. The number of tuning parameters make the controller flexible in its application, but it has not been tested if it could be applied to a wide variety of bottlenecks.

7.3 RECOMMENDATIONS FOR FUTURE WORK

In this section, options for further research are proposed and discussed. This is split into three parts. Firstly, some tests that could be performed on the current algorithm will be discussed. Secondly, algorithmic improvements that could be made are presented. Thirdly, the steps required to take the control concept closer to real-world application are discussed.

7.3.1 Further testing

While both penetration rate and non-compliance are tested separately, due to time constraints, a combination of these two important factors is not evaluated. Even though the consequences seem obvious (more clusters of vehicles that are either non-equipped or non-compliant), unforeseen interactive effects could occur. This may be important for adjusting the theory of the controller.

Furthermore, in the literature review, it was found that combinations of aggressive and timid drivers could affect the growth and decay of oscillations. Thus, types of drivers could affect the control strategy in currently unknown ways. This is a downside of using random distributions during the evaluation. Since the T and b values were not correlated, nothing can be said about the actual effects of different driver types.

Even though multiple tests could still be performed, it is advised to take the lessons from the tests in this thesis and make improvements to the algorithm first, since the current version is not ready for real-world application yet.

7.3.2 Improvements to the algorithm

There are many aspects of the controller that could still be extended or improved. Currently, the algorithm has one feedback loop which activates when vehicles end up too much upstream of their target line. However, stability is far more at risk when many vehicles reach far downstream of their target line and create a high-density cluster. Further research could add a feedback loop that attempts to respond to these high-density clusters by creating a larger void or other ideas that may come to mind.

A direct solution is to add a growth factor to the distance between target lines, depending on the number of non-equipped vehicles between two consecutive CVs. This results in more space for oscillations and can be calibrated depending on the amplitude of the oscillations. However, it is only expected to work for lower penetration rates and to a lesser extend for coping with non-compliance to control signals, While the two options mentioned before are still proactive against oscillations, a module could be added that responds to possible oscillations that occur in the controlled area. When the algorithm can rapidly manipulate traffic upstream of these oscillations, a breakdown within the controlled area could be prevented. This may also result in the possibility of more aggressive tuning of the algorithm with a higher flow in the controlled area since a probable breakdown due to control could be prevented as well. Furthermore, when a breakdown does occur within the controlled area, a jam wave resolving algorithm should be triggered. This means that all control from the controller developed in this thesis should be released and overwritten by the complementary algorithm.

Another reason that could be given for releasing all control is the growth of the controlled area. When the controlled area increases in length in the upstream direction, it may approach another bottleneck in the network. When this bottleneck is an on-ramp the inflow of vehicles is not taken into account and may cause oscillations in traffic. This also counts for other bottlenecks, such as lane-drops. However, when the algorithm is extended to handle these types of bottlenecks as well (i.e. by making room for merging vehicles or reducing density to ease the merging), control does not have to be released but is extended.

In the literature review on compliance, the notion was made that compliance is affected by the perception of compliance of other drivers. When approaching lower penetration rates, this may severely reduce the compliance of the CVs that are present on the road. The algorithm could be extended to also make use of the existing infrastructure that supports current Variable Speed Limit (VSL) strategies. This may also help against the growth of oscillations, as densities may be reduced due to earlier deceleration of non-equipped or non-compliant vehicles.

Since vehicles control their onset of deceleration, estimates could be made about the length of deceleration by the vehicle itself. This increases the chance that a vehicle does follow its target line. Since supportive indicators to follow the target lines can not be used due to workload issues, increasing the accuracy with this system may benefit the controller.

Also, in the current version of the algorithm, detection only works with measurements made at CVs. This may cause late detection of a high-flow area, which may result in oscillations before the algorithm responds. When loop-detectors are also used for flow-measurements, the algorithm can respond in time. A difference will be that the first vehicle is not the detected vehicle, but a CV upstream. Since oscillations may still occur, the first decelerated CV may already have to create a void to dampen the oscillations.

Moreover, the detection currently responds to single flow measurements. However, a single measurement of high-flow may not necessarily cause problems, i.e. when it is followed by a low-flow area. Also, when flow remains just under the threshold flow for a longer period, instabilities may occur as well. Since the controller will result in unnecessary delays when either of these situations occurs, a method should be developed that makes a better estimate of the probability of a breakdown at the bottleneck.

7.3.3 Extensions for real-world application

The controller that was developed in this thesis is not ready for real-world application yet. The most crucial reason for this is the single-lane design. Freeways generally consist of multiple lanes. Therefore, a solution has to be found for a multi-lane scenario. For a 100% penetration rate scenario, the control concept can be extended with little effort. However, when reducing the penetration rate, voids start being used for damping oscillations. In a multi-lane scenario, vehicles could merge into a void. This completely negates the effect when the void was optimized for the traffic situation, as oscillations will not be dampened completely, increasing the risk of traffic instability. This could be solved by synchronizing target lines of CVs, which removes the possibility of overtaking. However, this is highly dependant on factors such as deceleration and compliance as long as vehicles are manually driven.

Another factor that is currently not taken into account, is the variability in vehicle lengths. In the current evaluation, all vehicles are 4 meters long. For example, when trucks are present in traffic, the algorithm has to account for the extra space the truck requires. This can be done by using the vehicle length itself when the information is available through i.e. Central Awareness Messages (CAM) or Video-Based Monitoring (VBM) or estimations with passenger car units, when information is less accurate.

Furthermore, the communication framework is not provided within this thesis. This framework will be the basis of the controller and should, therefore, be developed and tested together with the controller. Variability in communication delay, determination of position and non-synchronous clocks of different systems are examples of important factors to take into account during real-world application of this controller.

When these extensions are made, extensive testing will be required to make sure the algorithm performs as intended and does not make unnecessary mistakes in communication. This is especially important since compliance is highly reliant on trust in the system, as found in the literature. Therefore, it is important that the moment the system is open for the public, it should work perfectly.

7.3.4 Policy recommendations for real-world application

Besides the aforementioned extensions that have been made to the algorithm before it is applicable for a real-world scenario, both the roadside and vehicles will have to be connected. While for vehicles, a penetration rate that is not 100% is acceptable, the roadside will need full coverage of communication technology near the bottlenecks that the algorithm developed in this thesis will be applied to. It was found that the roadside units currently developed have a range near 300m, which means that for full coverage a roadside unit has to be placed at least every 600m. Furthermore, when VBM is used as a data-source besides the CVs, which is highly advised in situations with lower penetration rates, cameras have to be placed near the desired measurement areas.

Communication technology is not standardized into new commercial vehicles yet. Since it takes a long time to refresh the fleet of commercial vehicles, it will take time before the penetration rate of standardized vehicles will rise above an acceptable level for the controller. Therefore, it is advised to make sure the application of this controller is also possible using other communication channels than with roadside units that communicate through WiFi. This can be achieved by adding the controller to agendas of projects such as SOCRATES2.0 or other currently ongoing large Cooperative Intelligent Transportation Systems (C-ITS) projects, where giving in-car information through i.e. mobile phones is researched. This must happen in collaboration with all important parties, such as the automotive industry, communication technology providers and road authorities, as an increased number of communication channels will likely increase the complexity of applying the controller by a large margin due to matching multiple levels of delay and accuracy.

When technology on both the roadside, as well as in vehicles is available for testing, it is advised to organize pilots at known bottlenecks to test the controller in reality. This way, unaddressed shortages can be identified and the controller or the communication technology could be improved. It is advised to perform tests on a test road before applying the controller to real traffic situations, as mistakes in control signals may result in unnecessary jams, but also in distrust in the system, which may lead to non-compliance later.

BIBLIOGRAPHY

Bazzi, A., Cecchini, G., Menarini, M., Masini, B. M., and Zanella, A. (2019). Survey and perspectives of vehicular Wi-Fi versus sidelink cellular-V2X in the 5G era. *Future Internet*, 11(6):1–20.

Bekiaris-Liberis, N., Roncoli, C., and Papageorgiou, M. (2016). Highway Traffic State Estimation With Mixed Connected and Conventional Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(12):3484–3497.

Bekiaris-Liberis, N., Roncoli, C., and Papageorgiou, M. (2017). Highway traffic state estimation per lane in the presence of connected vehicles. *Transportation Research Part B: Methodological*, 106:1–28.

Brilon, W., Geistefeldt, J., and Regler, M. (2005). Reliability of Freeway Traffic Flow: A stochastic Concept of Capacity. In *Proceedings of the 16th International Symposium on Transportation and Traffic Theory*, pages 125–144, Maryland.

Cassidy, M. J. and Bertini, R. L. (1999). Some traffic features at freeway bottlenecks. *Transportation Research Part B: Methodological*, 33(1):25–42.

Chen, D., Ahn, S., Laval, J. A., and Zheng, Z. (2014). On the periodicity of traffic oscillations and capacity drop: The role of driver characteristics. *Transportation Research Part B: Methodological*, 59:117–136.

Chow, A. H., Lu, X.-Y., and Qiu, T. Z. (2009). An Empirical Analysis of Freeway Traffic Breakdown Probability Based on Bivariate Weibull Distribution. *IFAC Proceedings Volumes*, 42(15):472–477.

Deng, H. and Zhang, H. M. (2015). On Traffic Relaxation, Anticipation, and Hysteresis. *Transportation Research Record: Journal of the Transportation Research Board*, 2491(1):90–97.

Edie, L. (1965). Discussion of traffic stream measurements and definition. In *2nd International Symposium On the Theory of Traffic Flow.*

ETSI (2019). EN 302 637-3 - V1.3.1 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service. Technical report, European Telecommunications Standard Institute.

Forbes, T. W. (1963). Human factor considerations in traffic flow theory. *Highway Research Record*, 15:60–66.

Goñi-Ros, B., Knoop, V. L., Takahashi, T., Sakata, I., van Arem, B., and Hoogendoorn, S. P. (2016). Optimization of traffic flow at freeway sags by controlling the acceleration of vehicles equipped with in-car systems. *Transportation Research Part C: Emerging Technologies*, 71:1–18.

Goñi-Ros, B., Knoop, V. L., Van Arem, B., and Hoogendoorn, S. P. (2014). Mainstream traffic flow control at Sags. *Transportation Research Record*, 2470(1):57–64.

Grumert, E. F. and Tapani, A. (2018). Bottleneck mitigation through a variable speed limit system using connected vehicles. *Transportmetrica A: Transport Science*, pages 213–233.

Hall, F. L. and Agyemang-Duah, K. (1991). Freeway capacity drop and the definition of capacity. *Transportation Research Record*, 1320:91–98.

Han, Y. and Ahn, S. (2018). Stochastic modeling of breakdown at freeway merge bottleneck and traffic control method using connected automated vehicle. *Transportation Research Part B: Methodological*, 107:146–166.

Han, Y. and Ahn, S. (2019). Variable Speed Release (VSR): Speed Control to Increase Bottleneck Capacity. *IEEE Transactions on Intelligent Transportation Systems*, 21(1):298–307.

Han, Y., Chen, D., and Ahn, S. (2017). Variable speed limit control at fixed freeway bottlenecks using connected vehicles. *Transportation Research Part B: Methodological*, 98:113–134.

Hegyi, A. (2013). A roadside-cooperative speed control algorithm to resolve moving jams (COSCAL v2). Technical report, TU Delft, Delft.

Hegyi, A., Hoogendoorn, S., Schreuder, M., Stoelhorst, H., and Viti, F. (2008). SPECIALIST: A dynamic speed limit control algorithm based on shock wave theory. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 827–832, Beijing. IEEE.

Hegyi, A. and Hoogendoorn, S. P. (2010). Dynamic speed limit control to resolve shock waves on freeways-Field test results of the SPECIALIST algorithm. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 519–524.

Hegyi, A., Netten, B., Wang, M., Schakel, W., Schreiter, T., Yuan, Y., van Arem, B., and Alkim, T. (2013). A cooperative system based variable speed limit control algorithm against jam waves - an extension of the SPECIALIST algorithm. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 973–978. IEEE.

Kan, X., Xiao, L., Liu, H., Wang, M., Schakel, W. J., Lu, X.-Y., van Arem, B., Shladover, S. E., and Ferlis, R. A. (2019). Cross-Comparison and Calibration of Two Microscopic Traffic Simulation Models for Complex Freeway Corridors with Dedicated Lanes. *Journal of Advanced Transportation*, 2019.

Khondaker, B. and Kattan, L. (2015). Variable speed limt: an overview. *Transportation Letters*, 7(5):264–278.

Knoop, V. L., Hegyi, A., Salomons, M., van Lint, H., Yuan, Y., and Landman, R. (2019). Traffic Flow Modelling & Control: lecture notes. Technical report, Technical University of Delft, Delft.

Laval, J. A. (2011). Hysteresis in traffic flow revisited: an improved measurement method. *Transportation Research Part B: Methodological*, 45(2):385–391.

Mahajan, N. (2014). *Integrated Approach to Variable Speed Limits, Ramp Metering extension of COSCAL v2 algorithm with ramp metering and ramp queue constraints*. MSc. Thesis, TU Delft.

Mahajan, N., Hegyi, A., Weg, G. S. v. d., and Hoogendoorn, S. P. (2015). Integrated Variable Speed Limit and Ramp Metering Control Against Jam Waves – A COSCAL v2 Based Approach. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 1156–1162. IEEE.

Middelham, F. and Taale, H. (2006). Ramp metering in the Netherlands: An overview. *IFAC Proceedings Volumes*, 39(12):267–272.

Moon, S., Moon, I., and Yi, K. (2009). Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice*, 17(4):442–455.

Mullakkal-Babu, F. A., Wang, M., Farah, H., van Arem, B., and Happee, R. (2017). Comparative assessment of safety indicators for vehicle trajectories on highways. *Transportation Research Record*, 2659(1):127–136.

Netten, B., Hegyi, A., Wang, M., Schakel, W., Yuan, Y., Schreiter, T., Van Arem, B., and Alkim, T. (2013). Improving moving jam detection performance with V2I communication. In *Proceedings of the 20th ITS world congress on intelligent transport systems*, Tokyo.

Newell, G. F. (1962). Theories of instability in dense highway traffic. *The Operations Research Society of Japan*, 1:9–54.

Nezafat, R. V., Beheshtitabar, E., Cetin, M., Williams, E., and List, G. F. (2018). Modeling and Evaluating Traffic Flow at Sag Curves When Imposing Variable Speed Limits on Connected Vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, 2672(20):193–202.

Papadopoulou, S., Roncoli, C., Bekiaris-Liberis, N., Papamichail, I., and Papageorgiou, M. (2018). Microscopic simulation-based validation of a per-lane traffic state estimation scheme for highways with connected vehicles. *Transportation Research Part C: Emerging Technologies*, 86:441–452.

Perraki, G., Roncoli, C., Papamichail, I., and Papageorgiou, M. (2018). Evaluation of a model predictive control framework for motorway traffic involving conventional and automated vehicles. *Transportation Research Part C: Emerging Technologies*, 92:456–471.

Risto, M. (2014). *Cooperative In-Vehicle Advice*. PhD thesis, University of Twente, Twente.

Roncoli, C., Papageorgiou, M., and Papamichail, I. (2015a). Traffic flow optimisation in presence of vehicle automation and communication systems – Part I: A first-order multi-lane model for motorway traffic. *Transportation Research Part C: Emerging Technologies*, 57:241–259.

Roncoli, C., Papageorgiou, M., and Papamichail, I. (2015b). Traffic flow optimisation in presence of vehicle automation and communication systems – Part II: Optimal control for multi-lane motorways. *Transportation Research Part C: Emerging Technologies*, 57:260–275.

Roncoli, C., Papamichail, I., and Papageorgiou, M. (2016). Hierarchical model predictive control for multi-lane motorways in presence of Vehicle Automation and Communication Systems. *Transportation Research Part C: Emerging Technologies*, 62:117–132.

Saffarian, M., de Winter, J. C. F., and Happee, R. (2013). Enhancing Driver Car-Following Performance with a Distance and Acceleration Display. *IEEE Transactions on Human-Machine Systems*, 43(1):8–16.

Scarinci, R., Hegyi, A., and Heydecker, B. (2017). Definition of a merging assistant strategy using intelligent vehicles. *Transportation Research Part C: Emerging Technologies*, 82:161–179.

Schakel, W. J. and Van Arem, B. (2014). Improving Traffic Flow Efficiency by In-Car Advice on Lane, Speed, and Headway. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1597–1606.

Schakel, W. J., Van Arem, B., and Netten, B. D. (2010). Effects of cooperative adaptive cruise control on traffic flow stability. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 759–764. IEEE.

Sharma, A., Zheng, Z., Bhaskar, A., and Haque, M. M. (2019). Modelling car-following behaviour of connected vehicles with a focus on driver compliance. *Transportation Research Part B: Methodological*, 126:256–279.

Srivastava, A. and Geroliminis, N. (2013). Empirical observations of capacity drop in freeway merges with ramp control and integration in a first-order model. *Transportation Research Part C: Emerging Technologies*, 30:161–177.

Treiber, M., Hennecke, A., and Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *PHYSICAL REVIEW E*, 62(2):1805 – 1824.

Treiber, M. and Kesting, A. (2013). *Traffic Flow Dynamics - Data, Models and Simulation*. Springer-Verlag Berlin Heidelberg.

Treiber, M., Kesting, A., and Helbing, D. (2006). Delays, inaccuracies and anticipation in microscopic traffic models. *Physica A: Statistical Mechanics and its Applications*, 360(1):71–88.

Treiterer, J. and Myers, J. (1974). The hysteresis phenomenon in traffic flow. *Transportation and traffic theory*, 6:13–38.

Van de Weg, G., Hegyi, A., Hellendoorn, H., and Shladover, S. E. (2014). Cooperative Systems Based Control for Integrating Ramp Metering and Variable Speed Limits. In *Proceedings of the 93rd Annual Meeting of the Transportation Research Board*, Washington D.C.

Van der Pas, J., Stolz, C., van der Knaap, R., Schoemakers, M., and van Engelenburg, B. (2017). Evaluatie spookfiles A58. Technical report, DTV Consultants, Breda.

Wang, M., Daamen, W., Hoogendoorn, S. P., and van Arem, B. (2016). Connected variable speed limits control and car-following control with vehicle-infrastructure communication to resolve stop-and-go waves. *Journal of Intelligent Transportation Systems*, 20(6):559–572.

Wu, N. (2002). A new approach for modeling of Fundamental Diagrams. *Transportation Research Part A: Policy and Practice*, 36(10):867–884.

Xu, T.-D., Hao, Y., Peng, Z.-R., and Sun, L.-J. (2013). Modeling probabilistic traffic breakdown on congested freeway flow. *Canadian Journal of Civil Engineering*, 40(10):999–1008.

Yuan, Y., Van Lint, J. W. C., Hoogendoorn, S. P., Wilson, R. E., and Van Wageningen-Kessels, F. (2012). Real-Time Lagrangian Traffic State Estimator for Freeways. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):59.

Zheng, Z., Ahn, S., Chen, D., and Laval, J. (2011). Freeway traffic oscillations: Microscopic analysis of formations and propagations using Wavelet Transform. *Transportation Research Part B: Methodological*, 45(9):1378–1388.

A LIST OF CONTROLLER VARIABLES

Variable	Units	Short Description
γ	-	Acceptable distance factor for target line deviations
ρ^{target}	veh/km	Target density of the algorithm
ω_i^{speed}	-	Compliance rate in terms of speed adaptation
a^{ex}	m/s²	Expected average deceleration of vehicles
CV_i	-	Indicator if vehicle i is a Connected Vehicle
$d^{\text{CheckDist}}$	m	Distance the algorithm checks giving a target line
$d_{X,i}(k)$	m	Distance travelled by all vehicles with area $W_{X,i}(k)$
$d_i^{\text{decelerate}}(k)$	m	Distance that a vehicle travels during deceleration
d^{target}	m	Target space headway of the algorithm
h_i(k)	m	Current space headway of vehicle i
i	-	Index of vehicles in the network, starting from downstream
i^*	-	Index of the first controlled vehicles after detection
k^{ctrllr}	-	Time index of the controller
$M_i(k)$	-	Current mode of vehicle i
N	-	Number of vehicles between a CV and the CV for which a target line is computed
n	-	Aggregation factor for flow measurement
q^{target}	veh/h	Target flow of the algorithm
Q_i(k)	veh/h	Aggregated flow for vehicle i at time step k
$Q^{\text{threshold}}$	veh/h	Threshold flow for activation of the control scheme
$Tail_i$	-	Indicator if vehicle i is a the tail of the controlled area
T^{ctrllr}	s	Time step length of the controller
$T^{\text{current}}(k)$	s	Current time in the simulation
$t_i^{\text{target}}(k)$	s	Point in time on the target line
$t_i^{\text{decelerate}}(k)$	s	Time a vehicle is expected to decelerate to v^{target}
$v_i(k)$	m/s	Current speed of vehicle i
v^{max}	m/s	Maximum speed of vehicles on the road
v^{target}	m/s	Target speed of the algorithm
v^{tol}	km/h	Speed tolerance factor for target line deviation
$W_{X,i}(k)$	m*s	Surface area in the space-time plane for flow measurement
$x_i(k)$	m	The location of vehicle i in the network
$x_i^{\text{target}}(k)$	m	Point in space on the target line
$x^{\text{u-BN}}$	m	Location of the upstream end of the bottleneck in the network
$x^{\text{u-m}}$	m	Location of the upstream end of the measurement area in the network
x^{control}	m	Location at which vehicles should be decelerated when controlled
x^{CR}	m	Location of the location of speed-control release in the network
$x^{\text{d-BN}}$	m	Location of the downstream end of the bottleneck in the network
$x^{\text{d-m}}$	m	Location of the downstream end of the measurement area in the network
$X_i^{\text{target}}(k)$	m	Target line of a vehicle

B TRAJECTORY PLOTS

B.1 DEMAND PATTERNS

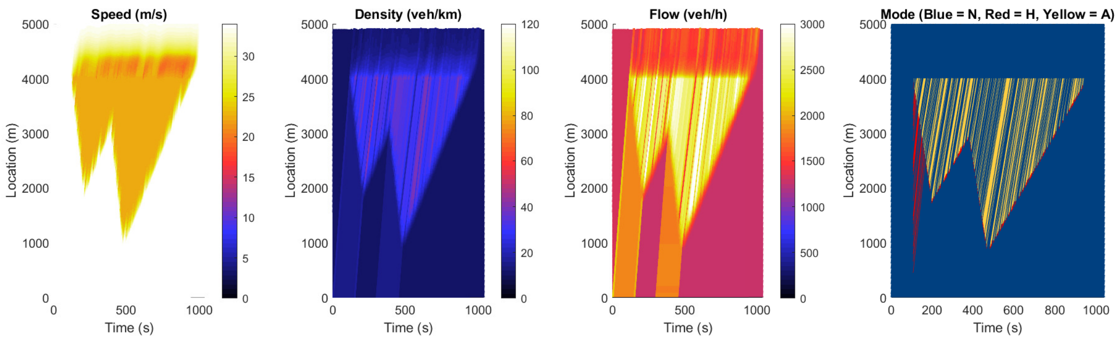


Figure B.1: Trajectories for two high-flow areas with 50% penetration rate

B.2 REDUCTION IN PENETRATION LEVEL

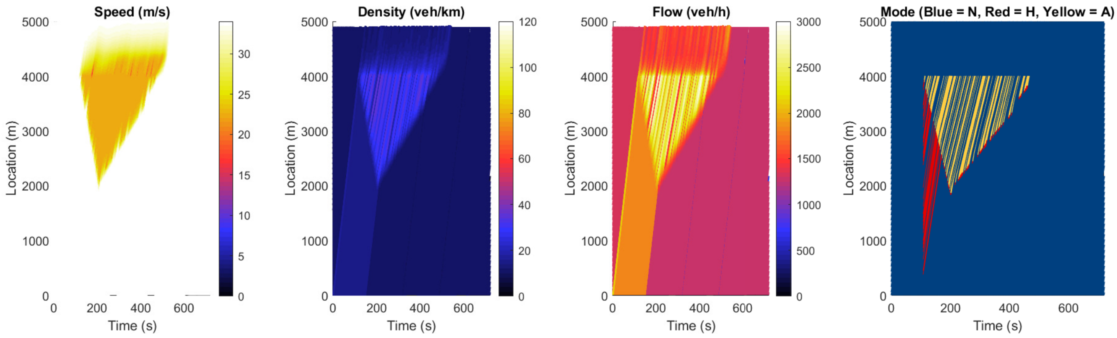


Figure B.2: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 18.5$ veh/km and **50% penetration rate**

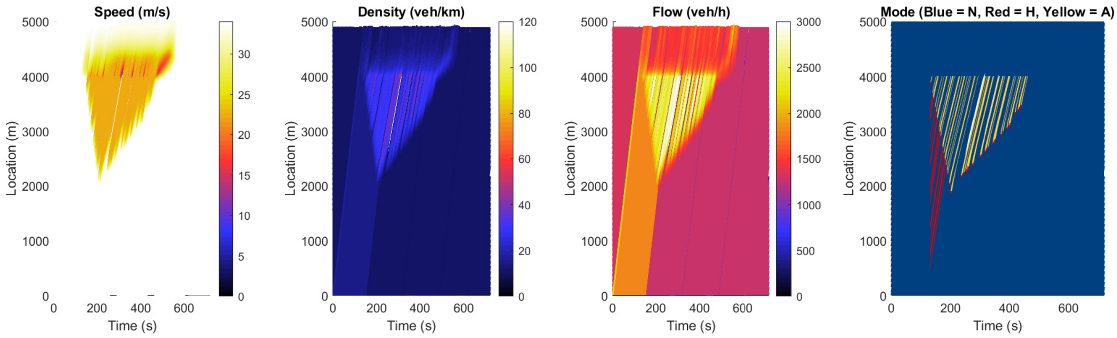


Figure B.3: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 17.8$ veh/km and **25% penetration rate**

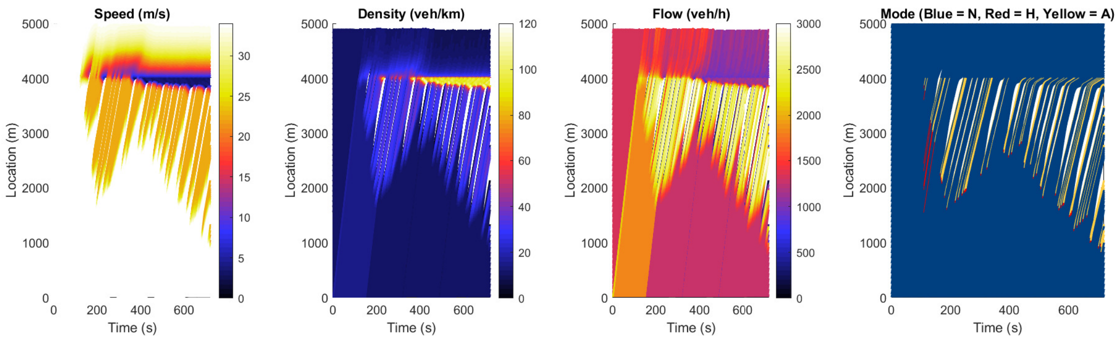


Figure B.4: Trajectories for an **unsuccessful** run with $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 17.8$ veh/km and **25% penetration rate**

B.3 REDUCTION IN COMPLIANCE

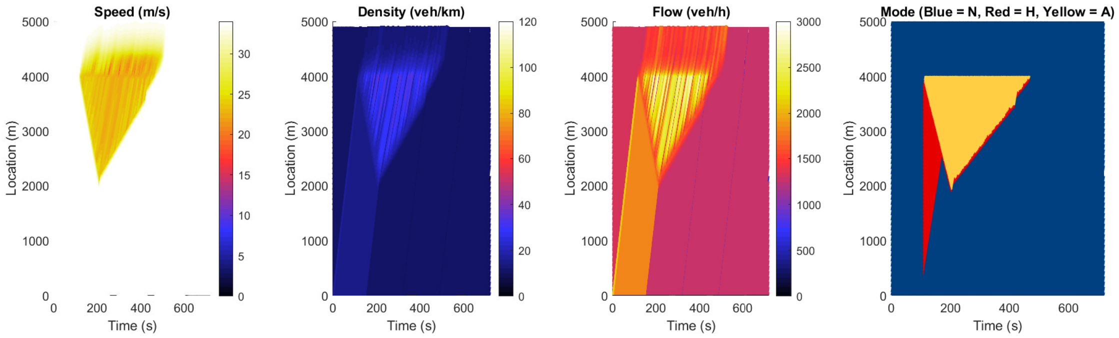


Figure B.5: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 18.6$ veh/km and **max. non-compliance of 80%**

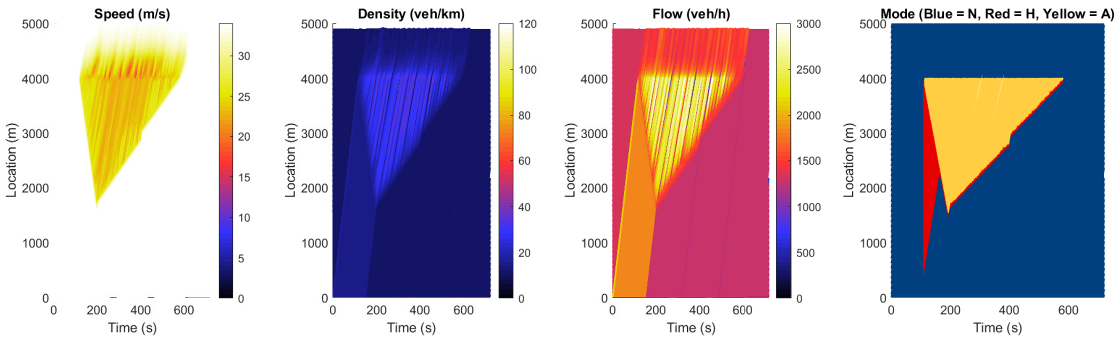


Figure B.6: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 18.0$ veh/km and **max. non-compliance of 60%**

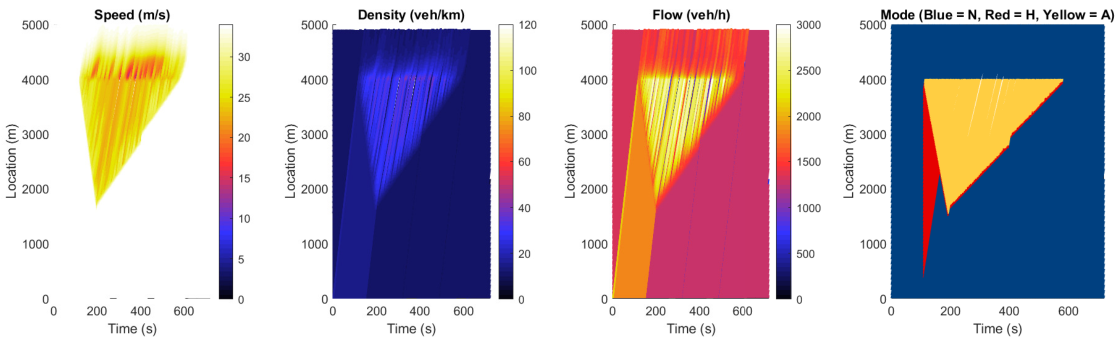


Figure B.7: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 18.0$ veh/km and **max. non-compliance of 40%**

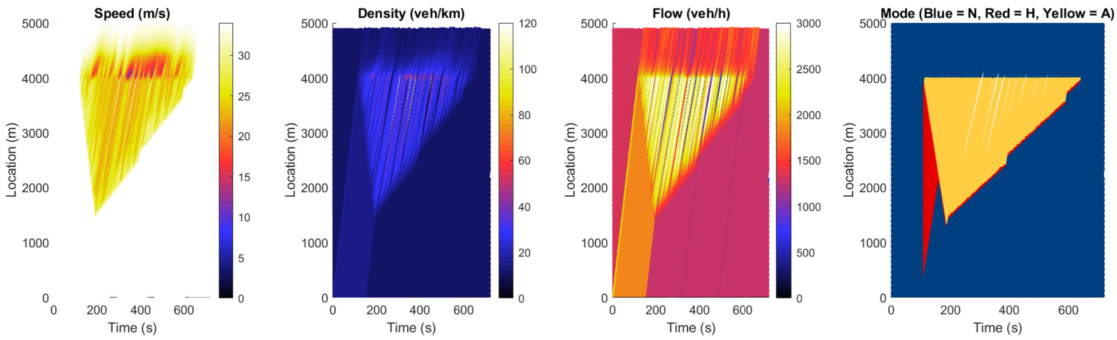


Figure B.8: Trajectories for $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 17.7$ veh/km and **max. non-compliance of 20%**

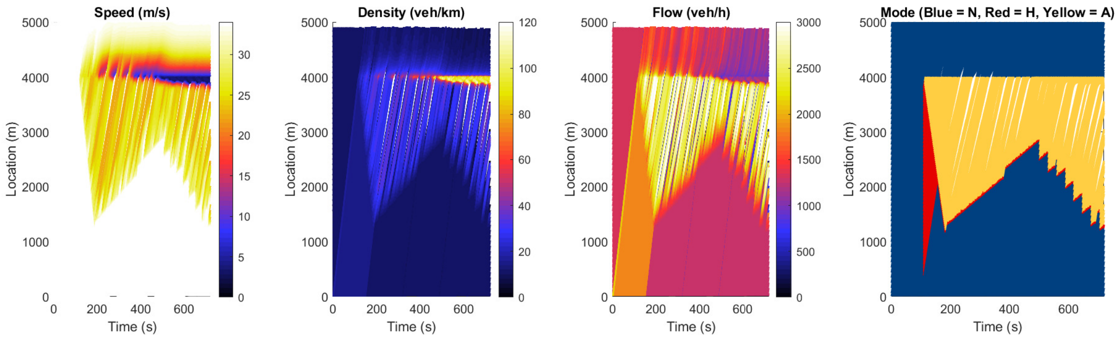


Figure B.9: Trajectories for an **unsuccessful** run with $v^{\text{target}} = 80\text{km/h}$ and $\rho^{\text{target}} = 17.5$ veh/km and **max. non-compliance of 0%**

C MATLAB

For the evaluation of the controller, multiple software packages able to simulate traffic were considered. The considered packages were VISSIM, MOTUS and MATLAB. Since development of a controller requires significant amounts of testing, short run times are important for testing efficiently. Experience of other students and researchers has shown that both VISSIM and MOTUS have long run times of around 15 minutes. Especially VISSIM in combination with the COM interface is slow. As MOTUS requires strong java programming skills, which the writer does not have, this was not an option within the given time-frame. Furthermore, a small traffic simulation model within MATLAB, which is used for showing the dynamics of COSCAL (made by Andreas Hegyi), is available for use. The run times of this model are close to 30 seconds, which is extremely short in comparison to VISSIM and MOTUS. Therefore, it is chosen to adjust the small traffic micro-simulation model in MATLAB. This gives access to influencing all the factors in the simulation, direct control into every detail of the model and easy access to all output data. Furthermore, computation times will be low, which makes it easy to test easily during development. Also, many scenarios can be run during the evaluation due to the low computation time, which makes the evaluation more valuable.

For interested readers, the code can be received upon request by emailing to Rick@Overvoorde.net. The code was written in MATLAB 2019b and is expected to be compatible with other versions, but this can not be guaranteed.