Collision avoidance in cooperative trajectory planning via time scaling

Hendriks, K.W.J.J.

Award date:
2020

Department of Mechanical Engineering
Dynamics & Control research group

Collision avoidance in cooperative trajectory planning via time scaling

Master thesis

K.W.J.J. Hendriks (1033665)

DC 2020.033

Coach: ir. R.B.A. van Hoek

Supervisor: prof.dr. H. Nijmeijer

Eindhoven, Friday 1st May, 2020

Abstract

As a result of the increasing traffic density nowadays, new technologies are being investigated to prevent traffic jams and increase the highway capacity by decreasing the inter-vehicle distance. Ideally a vehicle uses both cooperative driving and a trajectory planner. Such a vehicle can make use of close vehicle following, while still having the possibility to drive completely autonomous. This research only considers one single trajectory in the longitudinal direction.

The trajectory planner used in this project ensures cooperative driving of automated vehicles with the use of B-splines. The update frequency of the trajectory planner is relatively low due to computational constraints of the planner algorithm. A disturbance in between two consecutive plan cycles might result in a dangerous situation. A radar is used to measure the inter-vehicle distance between two vehicles and samples with a higher frequency than the frequency at which the trajectory planner updates. If this distance is smaller than planned, in between two plan cycles, the time scaling mechanism becomes active. The values of the planned trajectory are read out at a lower pace than real time to obtain a lower velocity and acceleration, without leaving the prescribed route. The planned trajectory is adapted instantaneous instead of generating a new one.

A time scaling mechanism is derived where a new desired velocity and acceleration are defined. The time scaling mechanism is simulated in a MATLAB Simulink environment before executing experiments. The two vehicles in the experiment are both a Renault Twizy. The vehicle parameters used for the final vehicle model are obtained by earlier performed experiments. Different use cases are considered to test the functionality of the time scaling mechanism during the experiments.

The time scaling mechanism works properly to cover several drawbacks of the trajectory planner. A following vehicle maintains on its own trajectory, while adapting the velocity if it encounters an object detected by the radar. Dangerous situations, resulting from V2V impairment or failure of the lead vehicle, can be overcome by using time scaling.

Contents

Abstract **iii**

Contents **v**

1 Introduction **1**
1.1 Background . . . 1
1.1.1 Cooperative Adaptive Cruise Control . . . 2
1.1.2 String stability . . . 2
1.1.3 Automated connected vehicles . . . 2
1.2 Problem definition . . . 3
1.3 Research objective . . . 4
1.4 Outline . . . 4

2 Cooperative trajectory planning **7**
2.1 Vehicle following objective . . . 7
2.2 Vehicle model . . . 8
2.3 Cooperative trajectory planning . . . 8
2.3.1 B-splines . . . 9
2.3.2 Gap closing . . . 13
2.4 Summary . . . 15

3 Time scaling **17**
3.1 Deriving the scaled time . . . 17
3.2 Acceleration based on $\ddot{\tau}$ . . . 20
3.3 Analytical and numerical comparison . . . 24
3.3.1 Internal vehicle noise . . . 25
3.3.2 Tracking error . . . 27
3.4 Upper bound for $\ddot{\tau}$ . . . 28
3.5 Summary . . . 30

4 Experiments **31**
4.1 Experimental setup . . . 31
4.2 Vehicle modelling . . . 32
4.2.1 Longitudinal vehicle dynamics . . . 32
4.2.2 Acceleration control . . . 33
4.3 Simulation results . . . 35
4.4 Experimental results . . . 39

4.4.1 Constant standstill distance $\underline{r}_i$ 39
4.5 Summary 46

5 Conclusions and recommendations **47**
5.1 Conclusions 47
5.2 Recommendations 48

Bibliography **49**

A Simulated noise signals **53**

B Schematic representation of the physical vehicle model **63**

Chapter 1

Introduction

1.1 Background

Roads and highways are becoming more and more densely populated by vehicles these days due to the demand for mobility for business and personal transportation. A major problem that arises due to the high traffic density are the so called *ghost* traffic jams [17]. These ghost traffic jams are formed by a human driver overreacting to a momentary disturbance (e.g. a slight deceleration by the preceding vehicle), and a delay in response time of the following drivers. Due to the reaction time of the driver, the following vehicle needs to brake harder than the preceding vehicle. This results in an amplification of the disturbance in the upstream direction Follower vehicles eventually can come in a situation where they become stationary when a platoon is long enough, Or even worse, a rear-end collision.

Vehicles are becoming more equipped with modern embedded control systems with safety improvements and driver comfort, such as Anti-lock Braking System (ABS), Electronic Stability Program (ESP) and cruise control. Luxury cars are being equipped with even more advanced collision avoidance systems. These systems decrease the risk of having collisions by actively intervening on the braking systems and increase the traffic throughput by actuating the powertrain. One of these systems is called Adaptive Cruise Control abbreviated as ACC. Vehicles equipped with Adaptive Cruise Control will automatically maintain a safe distance towards the preceding vehicle by adjusting the velocity, even if the velocity of the preceding vehicle is lower than the cruise control speed set by the driver of the ACC equipped vehicle. The system makes the vehicle slow down when it approaches a vehicle with a lower speed. And the vehicle increases speed to the cruise control set speed when the preceding vehicle accelerates or disappears (e.g., by changing lanes). A vehicle equipped with Adaptive Cruise Control will have a much better response time than an average human driver with as a result that less acceleration is needed, which reduce the possibility of collisions. The human driver will no longer have to concentrate on keeping a safe distance to its preceding vehicle. ACC systems accomplish the task of maintaining the desired spacing by measuring the inter-vehicle distance and relative velocity by means of a radar or a scanning laser (lidar) systems [18].

However, Adaptive Cruise Control is primarily intended as comfort system. Relatively large inter-vehicle distances are adopted [22], with a standardized minimum of 1 second time headway, which is defined as the distance between two vehicles divided by the vehicle velocity [17]. As a result, ACC is not capable of attenuating upstream disturbances to prevent formation of ghost traffic jams when using a smaller then 1 second time headway.

1.1.1 Cooperative Adaptive Cruise Control

Vehicle-to-Vehicle (V2V) communication can further advance the development of Adaptive Cruise Control systems. Extending the standard ACC functionality with wireless inter-vehicle communication link enables driving at smaller distances between vehicles without forming so-called shockwaves [18], formed by braking and accelerating, and ghost traffic jams. This extension is called Cooperative Adaptive Cruise Control, abbreviated as CACC [15],[16],[17],[24]. Cooperative Adaptive Cruise Control is known to allow a time headway significantly less than 1 second, compared to the standard ACC systems, with as a result an increase in traffic throughput [23]. In addition is the aerodynamic drag reduced and therefore the fuel consumption [17],[18], which is especially beneficial to heavy duty vehicles.

1.1.2 String stability

The main objective of CACC systems is to prevent disturbance attenuation in upstream direction. Disturbance amplifications can for instance be induced by velocity variations of the leading vehicle. The disturbance propagation along an interconnected system, such as a vehicle platoon, is known as string stability [6],[18]. Three main approaches of string stability are considered in the literature, being a formal stability-like approach [21], a stability approach for strings of infinite length [3] and a performance-oriented approach [17],[18].

The formal stability-like approach is based on Lyapunov stability, focussing on perturbations of the initial condition. Here string stability is interpreted as asymptotic stability of interconnected systems. The downside of this approach is that it only considers the response of a perturbation of the initial condition of a single vehicle in a platoon [18].

The stability approach for strings of infinite length formulates the model of a string in the state-space and transformed using the bilateral Z-transform. The Z-transform is executed over the vehicle index instead of over time which results in a discrete-frequency domain formulated model. By evaluating the eigenvalues of state matrix, stability can be assessed. This method, however, is only applicable for linear strings of infinite length, which is never the case for a platoon in practice.

A performance-oriented approach for string stability is adopted since this method directly offers tools for controller design for linear strings. In the performance-oriented approach, string stability is characterized by the amplification in upstream direction of the platoon in terms of distance error, velocity or acceleration. The latter approach for string stability is considered in this research.

1.1.3 Automated connected vehicles

A string of vehicles equipped with CACC controllers can be considered string-stable in the longitudinal direction of the string. However, string stability in the lateral direction is an unsolved problem. Unless the system is marginally string stable, corner cutting will occur. This leads to question if vehicle following in the lateral direction is a suitable approach altogether. Autonomous vehicles do not suffer from corner cutting, nor from string instability in the lateral direction [4],[12],[20]. Autonomous vehicles aim to navigate on highways and urban areas without a human driver. These type of vehicles rely on environmental perceptions to follow a trajectory specified by a trajectory planner [10],[19]. A trajectory refers to a spatial path in time. An approach that is often followed, is sampling of the state-space for terminal conditions that are related to a reference path [2],[11],[13],[25]. Within this method a set of

trajectories is generated. The autonomous vehicle is able to deviate from its reference path by selecting one of these trajectories to prevent collisions or circumvent obstacles. Compared with CACC systems, trajectory planners provide desired trajectories for the vehicle, where the CACC systems contain no information about future trajectories.

Ideally the automated vehicles can drive autonomous and cooperative using a single trajectory planner, without a prespecified formation of the platoon. The planner of such a vehicle would retain the comprehensive functionality of autonomous driving, such as passing other vehicles, but with the advantages of cooperative systems.

1.2 Problem definition

In addition to autonomous navigation, it is desired to ensure that automated vehicles move more efficiently and at as short as possible inter-vehicle distances. This is important for increasing the traffic throughput and preventing ghost traffic jams. Two type of automated vehicles can be distinguished. The two types are cooperative vehicles and autonomous vehicles. Cooperative vehicles use Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communication to share information and handle certain situations (to a certain extent) automated. These type of vehicles control a relative position with respect to the preceding vehicle. In contrast, autonomous vehicles try to regulate a relative position with respect to the road. These type of vehicles are able to autonomously navigate in traffic using trajectory planners to create feasible and collision-free trajectories. A trajectory planner, that uses for example B-splines [8], generates a set of possible trajectories. Afterwards these trajectories are checked for possible violations of constraints and on the possibility for collisions. The trajectory planner considers multiple potential trajectories simultaneously. The generated trajectories contain various actions such as changing lanes, maintaining velocity and coming to a stop. Cooperative trajectories can simply be added to this set of generated trajectories. A cost function is used to select the most optimal trajectory [25]. Hence the vehicle can automatically switch between for example cooperation in the form of platooning or simply overtaking an other vehicle which is driving at a lower speed than desired. Simple CACC systems, which follows the preceding vehicle at all times, lack this functionality.

The problem definition is defined as follows. The trajectories planned by the planner should result in string stable behaviour, but are updated at a relatively low frequency due to the computational load of the planning algorithm on the onboard CPU. The trajectory planner plans a trajectory for a constant number of seconds ahead with a relatively low update frequency. A disturbance in between plan cycles might result in dangerous situations. The velocity of the vehicle may deviate from the prescribed velocity trajectory, whereas the following vehicle still maintains the velocity prescribed by the planned trajectory, which might result in rear-end collision.

This can be realized by generating a new trajectory with a lower speed, which is likely not possible due to computational constraints, or by adapting the existing desired trajectory in real time. This thesis focusses on the latter approach because of the fact that a vehicle can adapt its trajectory instantaneous in between plan cycles, without leaving the desired route. The velocity and acceleration can be adapted with a higher frequency than the update frequency of the trajectory planner, which results in safer vehicle following behaviour. When an automated vehicle encounters another automated vehicle driving in the same lane with a lower speed, or an (stationary) object, it has to adapt its velocity to match the one of

the preceding vehicle. A time scaling method will be used to obtain this functionality. This method ensures a lower velocity if needed in between two consecutive plan cycles. A desired trajectory, which will in practice be stored in a table containing sampled-time values, will be read out a lower pace then real time. This method can adapt the desired trajectory in the longitudinal an lateral direction without leaving the desired route. This thesis only considers the longitudinal direction of the planned trajectory, but can be used for further research purposes when the lateral component is incorporated in a trajectory planner [5],[11].

1.3 Research objective

The research objective of this thesis is to derive a time scaling method to obtain collision avoidance. The time scaling policy enables the automated vehicle to adapt the existing trajectory, in between plan cycles, without leaving the desired route.

A time scaling method is designed where a suitable progression scheme for a scaled time τ as function of real time t is derived. The sampled-time values of the trajectory can be read out at a lower pace than real time with the scaled time $\tau(t)$. The velocity will be determined with this scaled time. Various approaches of computing the corresponding acceleration are compared.

Before the time scaling method can be simulated in a MATLAB Simulink environment, one needs to implement a trajectory planner which generates desired trajectories as a function of time, where $t \in [t_c, t_c+T]$ with t_c the current time and T the planning horizon. The planned trajectories are updated with a frequency of $\frac{1}{t_p}$ Hz.

A low level vehicle controller is needed to minimize the vehicle tracking error. This controller consists of a feedback part on the velocity and acceleration, which minimizes the tracking error. A feedforward controller is considered which provides a desired acceleration for the vehicle. The low level controller that is tracking the planned trajectory is running at a frequency of $\frac{1}{t_s}$ Hz with $t_s < t_p$. The planned trajectory will be adjusted at the same rate as the low level controller samples.

Eventually experiments are performed to show the functionality of the time scaling mechanism. Before one can implement this on real vehicles, an as accurate as possible vehicle model is needed. Also a controller for the longitudinal acceleration vehicle control is needed to minimize the error between the acceleration delivered and asked by the vehicle. Different use cased will be tested to ensure the safety properties of the time scaling method.

1.4 Outline

This report is organised as follows. Chapter 2 starts with the vehicle following objective in a platoon. Here is the spacing policy defined. Next a basic longitudinal first order vehicle model is explained, which is used in the beginning of the project. This is followed by a part about the trajectory planner used for cooperative driving. This is a trajectory planner based on B-splines. The chapter is ended with a part where comfortable gap closing by the trajectory planner is explained.

Chapter 3 first discusses the time scaling method and how the scaled time is derived and used to obtain a lower velocity than planned. This is followed by an extension on the time scaling mechanism where the acceleration is derived analytical by means of the second time derivative of the scaled time. Next a part where a schematic representation describes how

the time scaling mechanism is implemented in a vehicle, in cooperation with the trajectory planner, spacing policy and closed-loop vehicle dynamics.

This chapter is followed by a chapter which describes simulations of the experiments to be performed. This starts with an explanation of how a low level controller is designed to minimize the error between the desired vehicle states and the actual vehicle states. Next a longitudinal vehicle model is designed which describes the actual experimental vehicle. Different scenarios that activates the time scaling mechanism are explained, followed by the simulation results of the scenarios. This is followed by the executed experiments with the results.

Chapter 5 finalises the report with conclusions and recommendations for future research purposes.

Chapter 2

Cooperative trajectory planning

A vehicle that drives in a platoon often aims for a certain following objective when using cooperative driving. This objective is often defined by a spacing policy. A longitudinal vehicle model is considered. This model contains the three states position, velocity and acceleration. The desired acceleration is described as a first order model. Automated vehicles using a trajectory planner are driving cooperatively by communicating the planned trajectory. Note that the purpose of this chapter is to provide background information of how the trajectory planner is constructed and used in the research.

2.1 Vehicle following objective

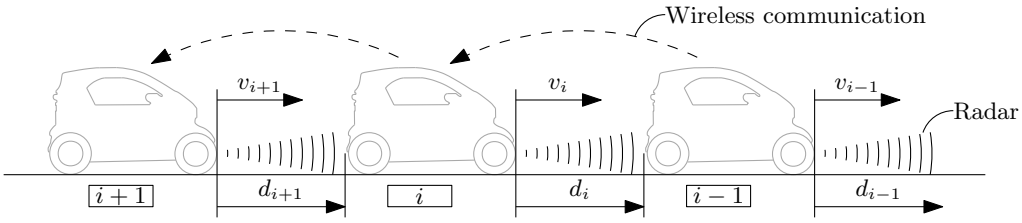


Figure 2.1: Vehicle platoon.

Consider a platoon of m vehicles, schematically depicted in Figure 2.1 with d_i being the distance between vehicle i and its preceding vehicle, vehicle $i-1$. Vehicle i is driving with a velocity v_i. The objective of each vehicle is to follow its preceding vehicle at a desired distance $d_{r,i}$. The desired distance is defined as

$$d_{r,i}(t) = r_i + hv_i(t), \quad i \in S_m \tag{2.1}$$

where r_i is the distance between two preceding vehicles when standing still and h is the time headway, or in other words, the difference between the time when the front of vehicle $i-1$ arrives at a point and the time the front of vehicle i arrives at the point r_i backwards. $S_m = \{i \in \mathbb{N} \mid 1 \leq i \leq m\}$ is the set of all vehicles in a platoon of length $m \in \mathbb{N}$. The spacing policy (2.1) is known to achieve string stability [17],[20] and safety [14],[18]. The actual inter-vehicle distance (from front bumper of vehicle i to the rear bumper of vehicle $i-1$) of vehicle i with respect to the preceding vehicle is defined as:

$$d_i(t) = s_{i-1}(t) - s_i(t) - L_i \tag{2.2}$$

with $s_i(t)$ and $s_{i-1}(t)$ corresponds to the position of the front of vehicle i and $i-1$ respectively.

The spacing error, with respect to the spacing policy, of a vehicle can now be defined using the difference between the desired distance (2.1) and the real distance (2.2):

$$\begin{aligned} e_i(t) &= d_i(t) - d_{r,i}(t) \\ &= (s_{i-1}(t) - s_i(t) - L_i) - (r_i + hv_i(t)) \end{aligned} \tag{2.3}$$

The vehicle-following objective is based on minimizing the spacing error as $\lim_{t\to\infty} e_i(t) = 0 \; \forall \; i \in S_m$.

2.2 Vehicle model

The following state space model represents the longitudinal dynamics of each vehicle:

$$\begin{pmatrix} \dot{s}_i(t) \\ \dot{v}_i(t) \\ \dot{a}_i(t) \end{pmatrix} = \begin{pmatrix} v_i(t) \\ a_i(t) \\ -\frac{1}{\eta_i} a_i(t) + \frac{1}{\eta_i} u_i(t) \end{pmatrix} \tag{2.4}$$

with $s_i(t)$ the position of vehicle i at time t, whereas $v_i(t)$ and $a_i(t)$ are the velocity and acceleration respectively. The external input $u_i(t)$ is the desired acceleration, which is determined by the acceleration of the desired trajectory. How this is determined will be discussed later in the report. The time constant representing the driveline dynamics is η_i. The value for the driveline dynamics constant is chosen as 0.1 s for the remainder of this report. The input $u_i(t)$ is allowed to make a step in signal as a function of time, which means that the signal may be discontinuous. The acceleration, however, is always a continuous signal according to the vehicle model. The current states are determined by the vehicle model. The initial states for the position, velocity and acceleration are needed for the trajectory planner. Since these states are continuous in time, the same must hold for the planned states from the trajectory planner.

2.3 Cooperative trajectory planning

A trajectory planner framework is presented in [25] in a Frenet frame, $\vec{e}^2$, where the planner uses a reference path $X(s)$, expressed in an interial cartesian coordinate frame, where $x(s)$ and $y(s)$ are spatial coordinates in the reference frame $\vec{e}^1$. These are parametrized with the distance along the reference path s. The trajectory is planned along this path using a lateral offset d perpendicular to the reference path, illustrated in Figure 2.2.

Parameterizing s and d in time using class $\mathcal{C}^2$ functions[1], a trajectory in time is obtained using $s(t)$ and $d(t)$. A vehicle can not move in the lateral direction without moving in the longitudinal direction, with as a result that the lateral offset is parametrizes as $d(s(t))$. A planner will always generate multiple trajectories. An example of this is illustrated in Figure 2.3. This thesis only considers one single trajectory in the longitudinal direction. The planner of a vehicle can plan both autonomous and cooperative trajectories. Such an automated vehicle is capable of short-distance vehicle-following in a string-stable manner. At the same time such a vehicle can break the platoon formation when, for example, the

[1] First and second derivatives are continuous.

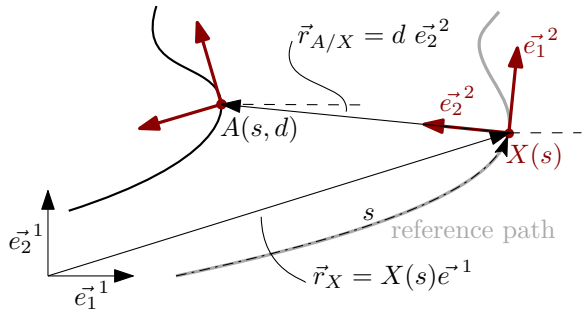


Figure 2.2: Reference path $X(s)$ and the lateral offset $d(s)$ in Frenet frame $\vec{e}_2^2$.

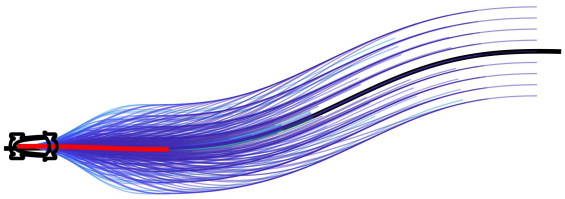


Figure 2.3: A set of generated trajectories with the planned trajectory.

preceding vehicle is driving too slow or the platoon has a different destination than the host vehicle. The vehicle is then still capable of navigating autonomously or be the lead vehicle in a platoon. The vehicle communicates the planned trajectory via a wireless communication to its following vehicle.

The longitudinal direction of the vehicle is based on the planned trajectory of the preceding vehicle. The trajectory of vehicle i are signals that are planned over a certain time horizon. The planned trajectory $\underline{\xi}_{r,i}(t)$ is given by

$$\underline{\xi}_{r,i}(t) = \begin{pmatrix} s_{r,i}(t) \\ v_{r,i}(t) \\ a_{r,i}(t) \end{pmatrix}, \quad t \in [t_{c,i},\ t_{c,i} + T_i] \tag{2.5}$$

with $s_{r,i}(t)$ the desired position states, $v_{r,i}(t)$ the desired velocity states and $a_{r,i}(t)$ the desired acceleration states of vehicle i. The current time of vehicle i is depicted by $t_{c,i}$ and the plan horizon given by T_i. The trajectory planner used in this project uses B-splines to generate desired trajectories.

2.3.1 B-splines

A method that uses B-splines to generate longitudinal trajectories for position, velocity and acceleration is used to generate cooperative string-stable trajectories for cooperative driving of automated vehicles [7]. Like polynomial trajectories [25], a trajectory can be generated with a few parameters which are communicated between two vehicles. But the advantage of using B-splines compared to polynomial trajectories is that B-splines can be modified locally.

The trajectory planner used in this project makes use of B-splines, which is a piecewise polynomial of degree p to construct trajectories [8]. The trajectories which are considered,

express the position $s_{r,i}$ of vehicle i [8],

$$s_i(t) = \mathbf{N}_{n,p}(t)\boldsymbol{\mathcal{P}}_i, \quad t \in [t_{c,i}, t_{c,i} + T_i] \tag{2.6}$$

$$\mathbf{N}_{n,p}(t) = [N_{0,p}(t) \quad N_{1,p}(t) \quad ... \quad N_{n,p}(t)] \tag{2.7}$$

where $\mathbf{N}_{n,p}(t)$ is a row vector of basis functions, $\boldsymbol{\mathcal{P}}_i = [P_{0,i}, P_{1,i}, ..., P_{n,i}]^T$ are $n+1$ control points, $t_{c,i}$ is the current time and T_i the planning horizon for vehicle i. The basis functions $N_{j,p}(t)$ are defined as

$$\begin{aligned} N_{j,0}(t) &= \begin{cases} 1 & \text{if} \quad u_j \leq t < u_{j+1} \\ 0 & \text{otherwise} \end{cases}, \\ N_{j,p}(t) &= \frac{t - u_j}{u_{j+p} - u_j} N_{j,p-1}(t) + \frac{u_{j+p+1} - t}{u_{j+p+1} - u_{j+1}} N_{j+1,p-1}(t) \end{aligned} \tag{2.8}$$

and are based on the knotvector $\mathbf{U}_i$. The uniform knotvector that spans from the current time $t_{c,i}$ to the end of the planning horizon $t_{c,i} + T_i$. The knotvector $\mathbf{U}_i$ is defined as

$$\begin{aligned} \mathbf{U}_i &= [\underbrace{t_{c,i}, ..., t_{c,i}}_{p+1}, \underbrace{u_{p+1}, ..., u_n}_{n-p}, \underbrace{T_i + t_{c,i}, ..., T_i + t_{c,i}}_{p+1}], \\ u_j &= \frac{j-p}{n-p+1} T_i + t_{c,i}, \quad j = \{p+1, ..., n\} \end{aligned} \tag{2.9}$$

An important feature of a B-spline is that the derivative is also a B-spline. The derivative of B-spline of degree p is one with degree $p-1$. The control points of the kth derivative are determined by:

$$P_{j,i}^{(k)} = \begin{cases} P_{j,i} & \text{for } k = 0, \\ \frac{p-k+1}{u_{j+p+k+1} - u_{j+1+k}} \left(P_{j+1,i}^{(k-1)} - P_{j,i}^{(k-1)} \right) & \text{otherwise.} \end{cases} \tag{2.10}$$

The trajectory $s_{r,i}(t)$ and its first and second time derivatives, $v_{r,i}(t)$ and $a_{r,i}(t)$ respectively, can be easily expressed as a function of the original control points because of the fact that the derivative of a B-spline is also a B-Spline.

The first and second time derivatives of $s_{r,i}(t)$ are defined as

$$v_{r,i}(t) = \mathbf{N}_{n-1,p-1}(t)\boldsymbol{\Gamma}^1(\mathbf{U}_i)\boldsymbol{\mathcal{P}}_i \tag{2.11}$$

$$a_{r,i}(t) = \mathbf{N}_{n-2,p-2}(t)\boldsymbol{\Gamma}^2(\mathbf{U}_i)\boldsymbol{\Gamma}^1(\mathbf{U}_i)\boldsymbol{\mathcal{P}}_i \tag{2.12}$$

where $\boldsymbol{\Gamma}^{(k)}(\mathbf{U}_i)$ is a matrix to transform the control points $\boldsymbol{\mathcal{P}}_i$ to the control points of the k-th derivative of the B-Spline as $\boldsymbol{\mathcal{P}}_i^{(k)} = \boldsymbol{\Gamma}^{(k)}(\mathbf{U}_i)\boldsymbol{\mathcal{P}}_i^{(k-1)}$ [8],[7], given by

$$\boldsymbol{\Gamma}^{(k)}(\mathbf{U}_i) = \begin{bmatrix} \frac{p-k+1}{u_{p+k+1} - u_{1+k}} I & & & \\ & \frac{p-k+1}{u_{p+k+2} - u_{2+k}} I & & \\ & & \ddots & \\ & & & \frac{p-k+1}{u_{n+p+1} - u_{n+1}} I \end{bmatrix} \begin{bmatrix} -I & I & & \\ & -I & I & \\ & & \ddots & \ddots \\ & & & -I \quad I \end{bmatrix} \tag{2.13}$$

Figure 2.4 gives a representation of the trajectory and its derivatives with the corresponding control polygon which is used to shape the trajectory. A one dimensional trajectory

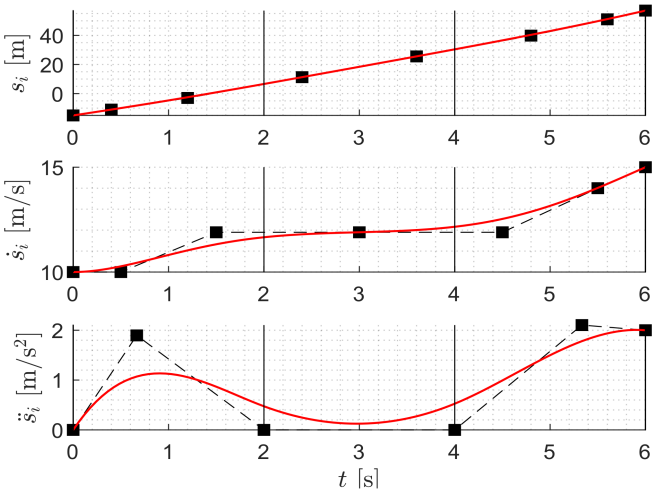


Figure 2.4: B-Spline trajectory with $n = 7$, $p = 5$ for $s_{r,i}(t)$. Control points $(\mu_j, P_{j,i})$ that construct the control polygon and knots indicated by vertical black lines [7].

is desired thus the control polygon can be generated using the control points $P_{j,i}$ and the Greville abscissae [8] and is given by

$$\mu_j = \frac{1}{p}\sum_{k=1}^{p} u_{j+k}, \quad j \in \{0, ..., n\} \tag{2.14}$$

The vehicle following scenario makes use of the spacing policy as is defined in (2.3). Substituting (2.6) and (2.11) in (2.3) results in

$$e_i(t) = s_{i-1}(t) - L_i - r_i - \left[\mathbf{N}_{n,p}(t) + h\mathbf{N}_{n-1,p-1}(t)\mathbf{\Gamma}^1\right]\mathcal{P}_i \tag{2.15}$$

The same approach as in [7] is followed to obtain the desired trajectory for $s_i(t)$ based on the planned trajectory $s_{i-1}(t)$ of the preceding vehicle. To find the control points for the spline trajectory, a linear system of equations is solved, that minimizes the summed square of the error (2.15) evaluated at the time coordinates μ_j, with $j \in \{3, \ldots, n\}$.

$$\begin{bmatrix} P_{3,i} \\ P_{4,i} \\ \vdots \\ P_{n,i} \end{bmatrix} = \begin{bmatrix} \mathbf{N}_{n,p}(\mu_3) + h\mathbf{N}_{n-1,p-1}(\mu_3)\mathbf{\Gamma}^1(\mathbf{U}_i) \\ \mathbf{N}_{n,p}(\mu_4) + h\mathbf{N}_{n-1,p-1}(\mu_4)\mathbf{\Gamma}^1(\mathbf{U}_i) \\ \vdots \\ \mathbf{N}_{n,p}(\mu_n) + h\mathbf{N}_{n-1,p-1}(\mu_n)\mathbf{\Gamma}^1(\mathbf{U}_i) \end{bmatrix}^+ \begin{bmatrix} s_{i-1}(\mu_3) - L_i - r_i \\ s_{i-1}(\mu_4) - L_i - r_i \\ \vdots \\ s_{i-1}(\mu_n) - L_i - r_i \end{bmatrix} \tag{2.16}$$

where $.^+$ denotes the Moore-Penrose inverse.

An advantage of trajectories constructed by B-Splines is that it can be fully described by the control points. Hence the required communication bandwidth is less then when communicating for example a whole polynomial [6].

The results of the B-spline planner are illustrated in Figure 2.5 and 2.6. Figure 2.5 represents the planned trajectories of a platoon of 5 vehicles that are accelerating to a constant velocity of 5 m/s from a standstill position whereas Figure 2.6 represents a deceleration. The update frequency of the planner is $f_p = 5$ Hz in both figures. The parameters used in the

spacing policy (2.3) are set as follows: the time headway $h = 1$ s, standstill distance $r = 5$ m and the vehicle length $L = 0$ m (the vehicle length has no influence on the vehicle dynamics and it can therefore be set to zero). It is assumed that the platoon is homogeneous. As one can see in both figures, the maximum acceleration and deceleration of vehicle i is always less then vehicle $i - 1$, which indicates string stable behaviour in terms of performance.

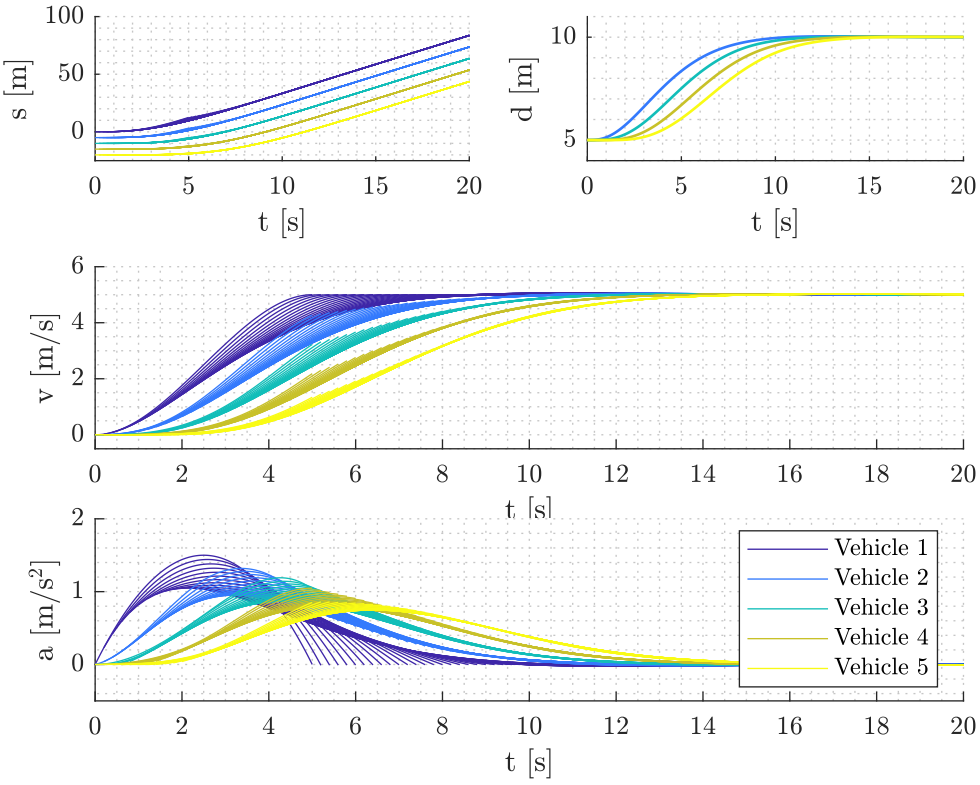


Figure 2.5: Planned trajectories of 5 vehicles when accelerating from 0 to 5 m/s.

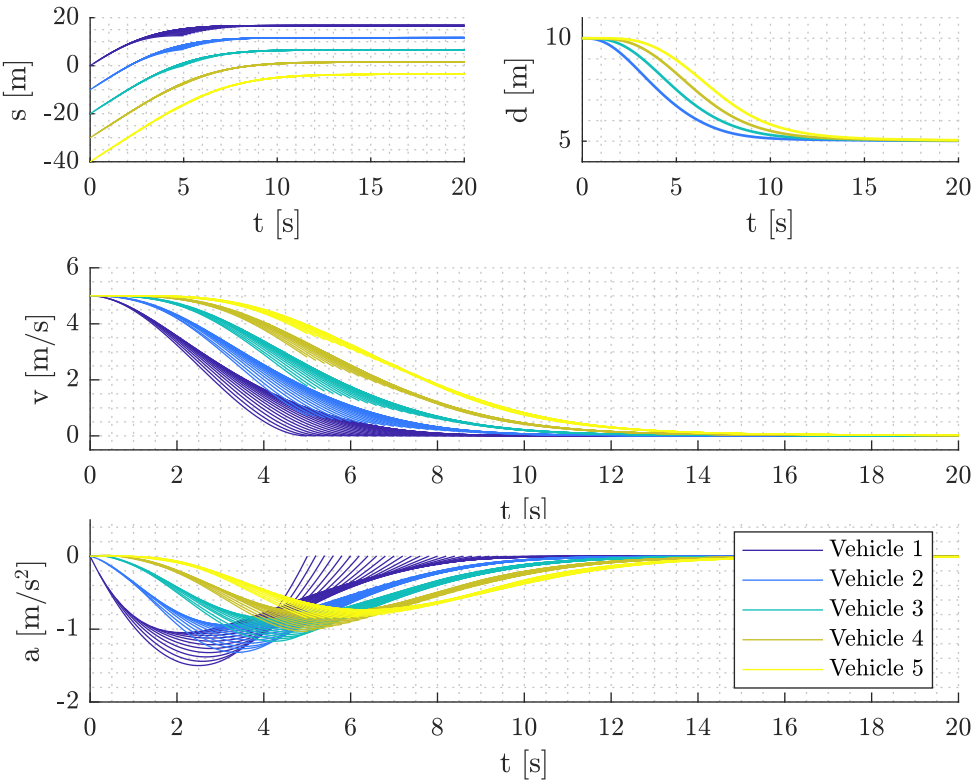


Figure 2.6: Planned trajectories of 5 vehicles when decelerating from 5 to 0 m/s.

2.3.2 Gap closing

The same spacing policy as is often used for CACC systems is used to enable short distance inter vehicle following [17] as is described in (2.1). This spacing policy tries to minimize the distance error, which is defined in (2.3), as fast as possible, which can result in uncomfortable accelerating. The result of this is illustrated in Figure 2.7. The initial velocity of both vehicles is 15 $\mathrm{m\,s^{-1}}$ and the following vehicle is positioned 25 m back with respect to the spacing policy. A result of this is that the following vehicle closes the gap in between both vehicles as fast as possible. To make the gap closing more comfortable, the spacing policy (2.1) is slightly modified. The trajectory planner now ensures comfortable gap closing with a modified standstill distance [8]. The standstill distance at time t is defined as

$$r_i(t) = s_{i-1}(t) - s_i(t) - hv_i(t) - L_i \tag{2.17}$$

The rate at which the gap size is decreases is determined and given by

$$\underline{\psi} = \left(\frac{1}{1+\underline{\phi}_i} - 1\right) \underbrace{\frac{1}{T_i}\left(s_{i-1}(t_{c,i}) + s_{i-1}(t_{c,i}+T_i)\right)}_{V_{\mathrm{avg},i-1}} \tag{2.18}$$

with $\underline{\phi}_i$ a used defined change rate with $\underline{\phi}_i \in [-1, 0]$. The transition of fixed rate change to

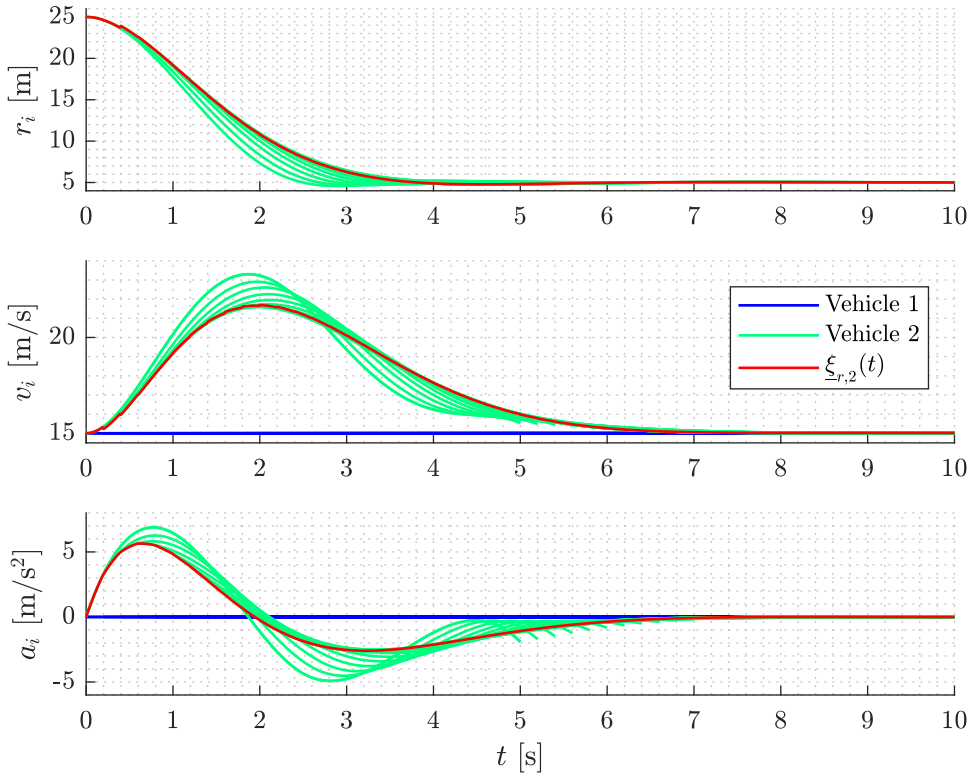


Figure 2.7: Uncomfortable gap closing

first order dynamics can be determined via

$$r_{\text{tr}} = \min[\underline{r}_i + \underline{\psi} T_i,\ r(t_{c,i})] \tag{2.19}$$

$$t_{\text{tr}} = \frac{r(t_{c,i}) - r_{tr}}{\underline{\psi}} + t_{c,i} \tag{2.20}$$

with $\underline{r}_i$ the minimum standstill distance. Now $\underline{\psi}$ can be used to determine the trajectories for the desired standstill distance $r_{d,i}(t)$ [8]:

$$r_{d,i}(t) = \begin{cases} r_i(t_{c,i}) + \underline{\psi}(t - t_{c,i}), & \text{if } t < t_{c,i} \\ (r_{\text{tr}} - \underline{r}_i) \exp\left(\dfrac{t_{\text{tr}} - t}{T_i}\right) + \underline{r}_i, & \text{otherwise} \end{cases} \tag{2.21}$$

Figure 2.8 illustrates how the standstill distance vary as a function of time. Here two vehicles are considered. The lead vehicle is driving at a constant velocity of $v = 15$ m/s. The velocity of the following vehicle is also 15 m/s at $t = 0$ s. The following vehicle is positioned 25 m back with respect to the spacing policy given by (2.3). With as a result that the following vehicle tries to close the gap without extremely or uncomfortably accelerating and decelerating.

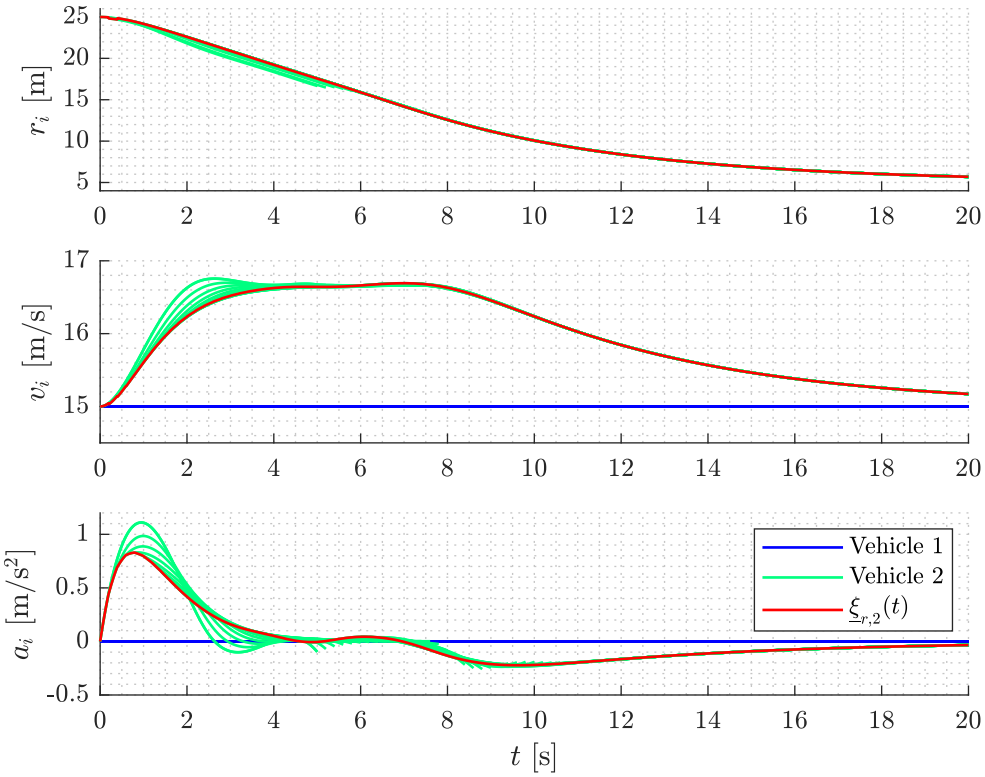


Figure 2.8: Comfortable gap Closing

2.4 Summary

A string of automated vehicles aim in this thesis for a following objective, defined by a spacing policy. All vehicles are equipped with a radar that measures the position and velocity relatively to the preceding vehicle. A vehicle is modelled as a state space model with a time constant that represent the driveline dynamics. The initial states of this model are needed for the trajectory planner. The trajectory planner plans trajectories with the use of B-splines and updates at a given frequency and determines the planned trajectory based on the communicated of a preceding vehicle, and the desired spacing policy. This allows for string stable cooperative driving to be integrated in the framework of the autonomous vehicle, such that a single framework can be used for both operating modes.

Chapter 3

Time scaling

The method from Chapter 2.3.1 results in string stable behaviour by means of trajectory planning using B-splines. It also provides comfortable gap closing as is described in Section 2.3.2. A lead vehicle will never exactly execute the communicated trajectory due to various disturbances. These disturbances can be compensated by the next plan cycle of the host vehicle. However, the update frequency of the trajectory planner is relatively low due computational constraints. Therefore an other method is desired to compensate for these disturbances.

An automated vehicle is considered which is equipped with a controller aiming to follow a desired trajectory $\underline{\xi}_{r,i}(t)$, which is defined in (2.5), as a function of time t. The desired trajectory is, in practice, stored in a table containing sampled-time values. These sampled-time values can be read out at a lower pace than real time to obtain a lower velocity without leaving the desired path, in other words, the combination of the x and y coordinates stay the same. The objective is to determine a scaled time as a function of time as input to the table. The output of this 'lookup-table' is scaled with the scaled time to determine the new adapted desired states for the vehicle.

This chapter starts with the explanation of how the scaled time is derived and a new scaled velocity is determined with the corresponding acceleration. This is followed by a part where an alternative is proposed of how the scaled acceleration is determined. Next a situation is simulated where the time scaling mechanism activates. This is followed by comparing the two ways of determining the scaled acceleration. The chapter is concluded with a boundary condition that prevents the vehicle from accelerating more than prescribed by the desired trajectory.

3.1 Deriving the scaled time

A desired velocity $v_{c,i}(t) > 0$, not necessarily equal to $|v_{r,i}(t)|$, with which the trajectory $s_{r,i}(t)$ must be driven, is used to determine the 'scaled time'. Note that $v_{c,i}(t)$ could for example come from a used defined cruise control speed or a CACC controller that aims to realize the spacing policy. In this work the latter option is chosen to obtain the velocity $v_{c,i}(t)$, and is defined as

$$v_{c,i}(t) = \frac{d_i(t) - r_{c,i}}{h} \tag{3.1}$$

The scaled time is an increasing function $\tau(t)$ where t is real time. This function is defined by means of its derivative, given by

$$\dot{\tau}(t) := \begin{cases} \dfrac{v_{c,i}(t)}{|v_{r,i}(\tau(t))|}, & \text{subject to } |v_{r,i}(\tau(t))| > 0, \frac{v_{c,i}(t)}{|v_{r,i}(\tau(t))|} \leq 1 \\ 1, & \text{otherwise.} \end{cases} \tag{3.2}$$

Setting $\tau(0) = 0$,

$$\int_0^t \dot{\tau}(z)dz = \tau(t) \tag{3.3}$$

is then the scaled time, with $0 \leq \tau(t) \leq t$ since $0 \leq \dot{\tau}(t) \leq 1$. Note that $\tau(t)$ is only used for a host vehicle. Hence, the scaled time does not go faster then real time, with as a result that the new scaled velocity can never be higher than the velocity obtained by the trajectory planner. A time-scaling function $\tau(t)$ is defined for a given desired trajectory as is given in (2.5). Then

$$\underline{\xi}_{\tau,i}(t) = \begin{pmatrix} s_{\tau,i}(t) \\ v_{\tau,i}(t) \\ a_{\tau,i}(t) \end{pmatrix} = \begin{pmatrix} s_{r,i}(\tau(t)) \\ \dot{s}_{r,i}(\tau(t)) \\ \ddot{s}_{r,i}(\tau(t)) \end{pmatrix} \tag{3.4}$$

represents the new desired trajectory. Consider the derivative of $s_{\tau,i}(\tau(t))$ with respect to t to obtain an adapted velocity:

$$\begin{aligned} v_{\tau,i}(t) = \dot{s}_{r,i}(\tau(t)) &= \frac{ds_{r,i}(\tau(t))}{d\tau}\frac{d\tau(t)}{dt} \\ &= v_{r,i}(\tau(t))\dot{\tau}(t) \end{aligned} \tag{3.5}$$

The time derivative of $v_{\tau,i}(t)$ is determined to obtain the adapted acceleration and is given by:

$$\begin{aligned} a_{\tau,i}(t) = \ddot{s}_{r,i}(\tau(t)) &= \frac{d^2 s_{r,i}(\tau(t))}{d\tau^2}\left(\frac{d\tau(t)}{dt}\right)^2 + \frac{ds_{r,i}(\tau(t))}{d\tau}\frac{d^2\tau}{dt^2} \\ &= a_{r,i}(\tau(t))\dot{\tau}(t)^2 + v_{r,i}(\tau(t))\ddot{\tau}(t) \end{aligned} \tag{3.6}$$

As a result the second time derivative of $\tau(t)$ is required to determine the adapted acceleration $a_{\tau,i}(t) = \ddot{s}_{r,i}(\tau(t))$. Due to the definition of $\dot{\tau}(t)$ from (3.2), the time scaling mechanism is only causes retarding actions for safety reasons. Since $\ddot{\tau}$ is not yet available it can be determined by means of numerical differentiation. However, it is more efficient to directly determine $a_{\tau,i}(t)$ by numerical differentiation of the adapted desired velocity $v_{\tau,i}(t)$. Although numerical differentiation is not particularly elegant, it will be feasible assuming that the velocity desired trajectory $v_{r,i}(t)$ is continuous differentiable. Figure 3.1 gives a representation of how the time-scaling mechanism works illustrated in a block-scheme. Here the velocity $v_{c,i}(t)$, given by (3.1), aims to realise the spacing policy.

A schematic representation of how the time-scaling mechanism is implemented is a vehicle is illustrated in Figure 3.2. The parameters to generate the planned trajectory of the lead vehicle are communicated through a wireless connection. The trajectory planner plans a trajectory for vehicle i using these parameters and is stored in a look-up table containing time based values. The vehicle effectively upperbounds the desired velocity $v_{r,i}(t)$ by $v_{c,i}(t)$ as can be seen by:

$$v_{\tau,i}(t) = \begin{cases} v_{r,i}(\tau(t))\frac{v_{c,i}(t)}{|v_{r,i}(\tau(t))|}, & \text{if } \frac{v_{c,i}(t)}{|v_{r,i}(\tau(t))|} \leq 1 \\ v_{r,i}(\tau(t)) & \text{else} \end{cases} \tag{3.7}$$

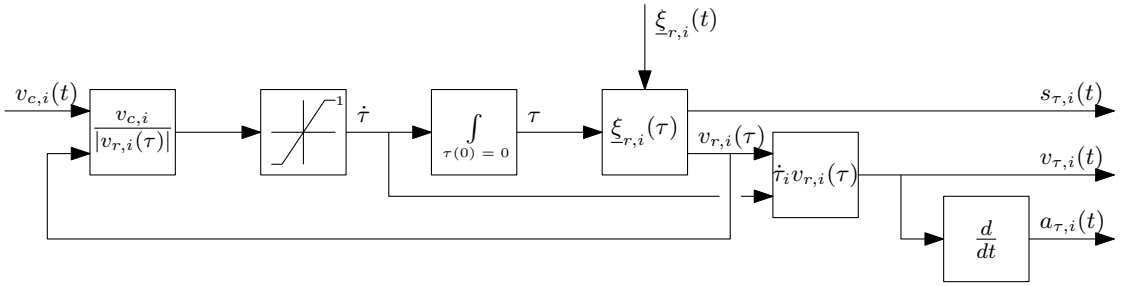


Figure 3.1: Block-scheme of the time-scaling mechanism.

according to the spacing policy using the current states of the preceding vehicle, $r_{c,i}$ and its own current states. The velocity coming from the spacing policy is used as an input for the time-scaling mechanism, as is the velocity according to the trajectory planner. With both the velocities $v_{c,i}(t)$ and $v_{r,i}(t)$ (assuming that $v_{r,i}(t)$ is non-negative) is the scaled time $\tau(t)$ derived, which acts as input to the lookup-table to define a new desired states which the vehicle aims to follow.

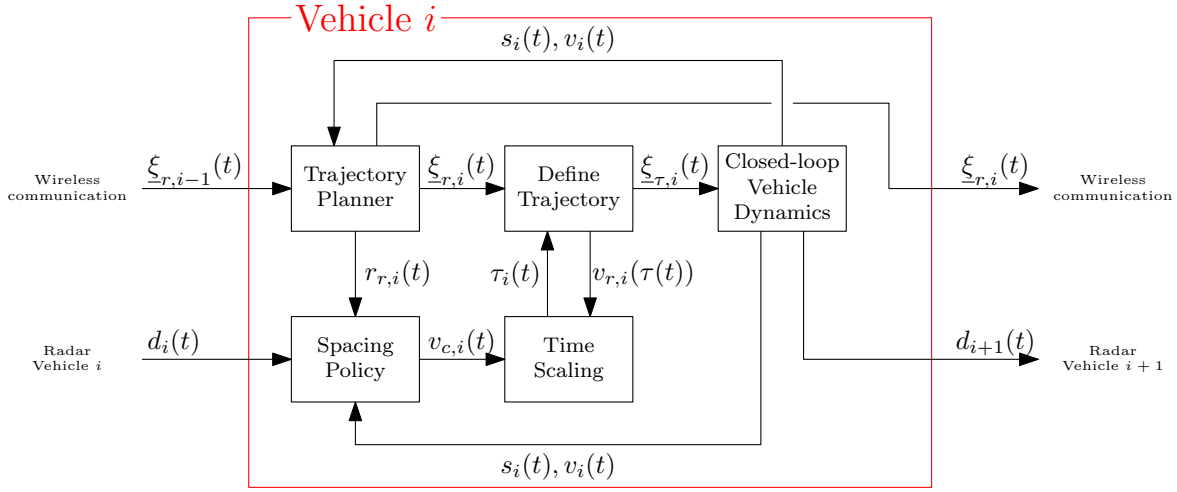


Figure 3.2: Schematic representation of the time-scaling implementation in vehicle i.

An illustration of how $\dot{\tau}$ acts with respect to time is given in Figure 3.3. A situation where two consecutive vehicles are driving with a constant velocity of 5 $\mathrm{m\,s^{-1}}$ is simulated. The lead vehicle drives autonomous using the B-spline planner, whereas the second vehicle aims to follow the first one according to the spacing policy (2.3). The trajectory planner plans a trajectory over a time horizon $T_i = 5$ s with a update frequency of $f_p = 1$ Hz. The lead vehicle decelerates unexpectedly in between two plan cycles, while the planning still aims to maintain a constant velocity of 5 $\mathrm{m\,s^{-1}}$. The inter-vehicle distance decreases and thus the velocity coming from the spacing policy. As one can see in the figure, $\dot{\tau}(t)$ decreases and resets to 1 when a new planning is initiated. This will not affect the control input because at this moment is the desired stand-still distance to the actual stand-still distance, in other words $r_i(t) = r_{r,i}(t)$.

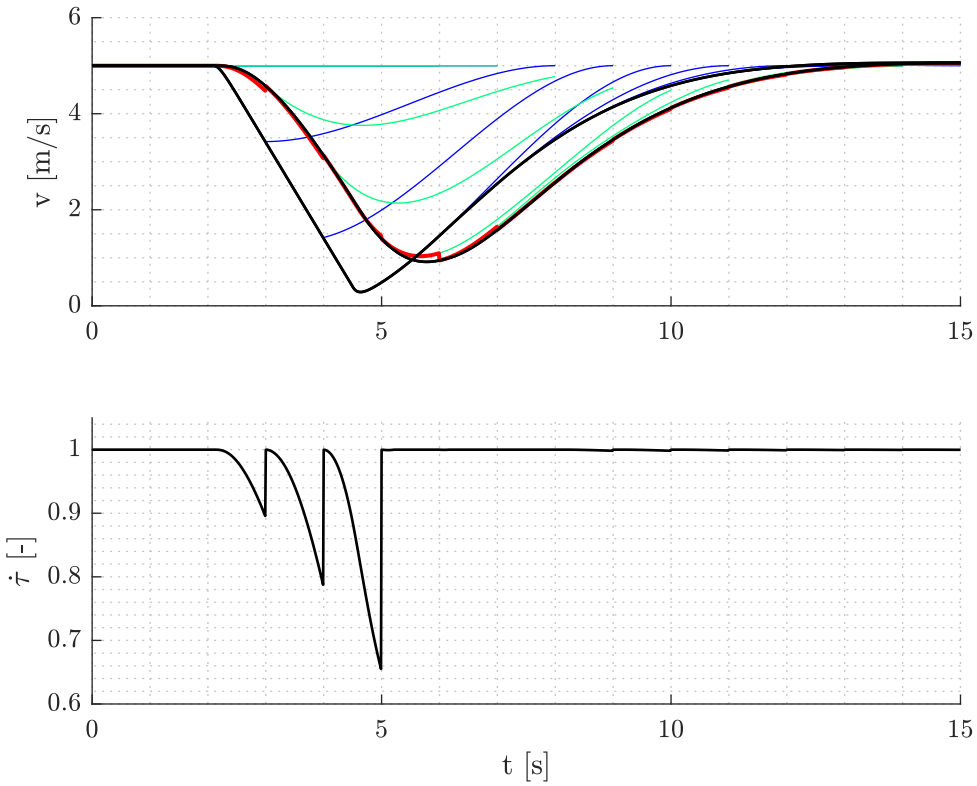


Figure 3.3: Time-scaling mechanism active during braking in between trajectory plan cycles with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired states of vehicle 2 $\underline{\xi}_{\tau,2}(t)$ (—) and the real vehicle states $\underline{\xi}_i(t)$ (—).

3.2 Acceleration based on $\ddot{\tau}$

In previous section, the acceleration is determined by numerically differentiation of the adapted velocity $v_{\tau,i}(t)$. Despite the assumption that the trajectory $\underline{\xi}_{r,i}(t)$ is continuously differentiable, numerical differentiation is not desired. Instead of numerical differentiation of $v_{\tau,i}(t)$, the acceleration $a_{\tau,i}(t)$ can also be determined by means of equation (3.6) and $\ddot{\tau}(t)$. Therefore the derivative of (3.2) with respect to time is composed:

$$\ddot{\tau}(t) := \begin{cases} \frac{|v_{r,i}(\tau(t))|\dot{v}_{c,i}(t) - v_{c,i}(t)a_{r,i}(\tau(t))\dot{\tau}(t)\frac{v_{r,i}(\tau(t))}{|v_{r,i}(\tau(t))|}}{|v_{r,i}(\tau(t))|^2}, & \text{subject to } |v_{r,i}(\tau(t))| > 0, \frac{v_{c,i}(t)}{|v_{r,i}(\tau(t))|} \leq 1, \\ 0, & \text{otherwise.} \end{cases} \tag{3.8}$$

As one can see, $\dot{v}_{c,i}(t)$ is required to determine $\ddot{\tau}(t)$ and is given by

$$\begin{aligned} \dot{v}_{c,i}(t) &= \frac{\dot{d}_i(t) - \dot{r}_{r,i}(t)}{h} \\ &= \frac{v_{i-1}(t) - v_i(t) - \dot{r}_{r,i}(\tau(t))}{h} \end{aligned} \tag{3.9}$$

Note that de desired standstill distance, which is defined as

$$r_{r,i}(t) = s_{r,i-1}(t) - s_{r,i}(t) - hv_{r,i}(t) - L_i \tag{3.10}$$

and is a function of time and follows from the trajectory planner. A result of this is that the standstill distance also becomes a function of the scaled time τ. The time derivative of $r_{r,i}(\tau(t))$ is defined as:

$$\begin{aligned} \dot{r}_{r,i}(\tau(t)) &= \frac{dr_i(\tau(t))}{d\tau}\frac{d\tau}{dt} \\ &= [v_{r,i-1}(\tau(t)) - v_{r,i}(\tau(t)) - ha_{r,i}(\tau(t))]\dot{\tau}(t) \end{aligned} \tag{3.11}$$

Now the equation for $\ddot{\tau}$ is defined, it is bounded in the same way as is done for $\dot{\tau}$ in (3.2). The purpose of the time scaling mechanism is to slow down a vehicle if needed. So to prevent the time scaling mechanism from accelerating more than prescribed by the trajectory planner, $\ddot{\tau}$ is set to zero if $\dot{\tau}$ is equal to 1. A schematic representation of the time scaling mechanism as is described before is illustrated in Figure 3.4. The new desired acceleration $a_{\tau,i}(t)$ as

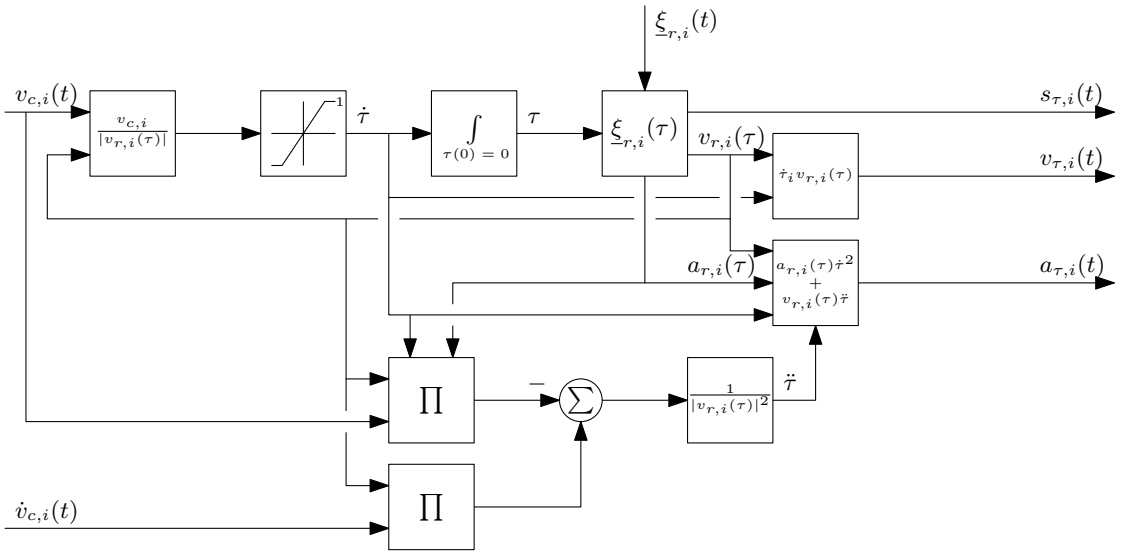


Figure 3.4: Schematic representation of the time-scaling implementation in vehicle i.

is defined in (3.6) can now be determined by means of the time scaling mechanism. An extra advantage of this is that there is extra information available by means of $\dot{d}_i(t)$, instead of looking backwards to determine the acceleration by means of numerical differentiation. To get a better insight of how this changes the implementation, a schematic representation is illustrated in Figure 3.5. To check whether this approach to determine a scaled desired acceleration corresponds with the definition for $\dot{\tau}$, a comparison is made between the the scaled velocity $v_{\tau,i}(t)$ and the integral of $a_{\tau,i}(t)$ over time. This comparison is represented in Figure 3.6. The same simulation is executed as is done for Figure 3.3. As one can see in the figure, the integral of scaled acceleration the corresponds to the scaled velocity.

To see how the time scaling mechanism affects the states of a vehicle, a simulation is done for two following vehicles. The lead vehicle brakes unexpectedly at approximately 3 s. To simulate a faulty vehicle, the controller output of the lead vehicle is replace by a fixed

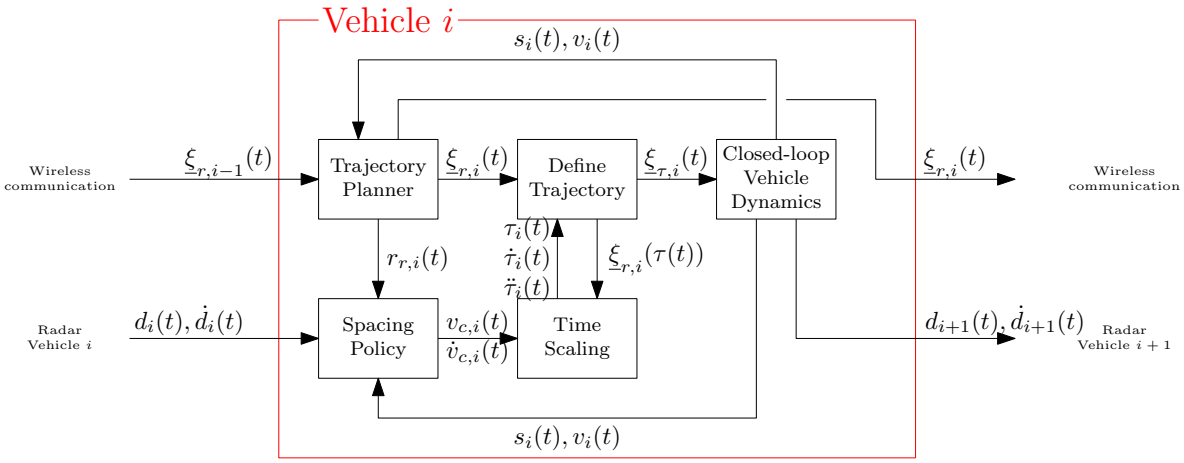


Figure 3.5: Schematic representation of the time-scaling implementation in vehicle i.

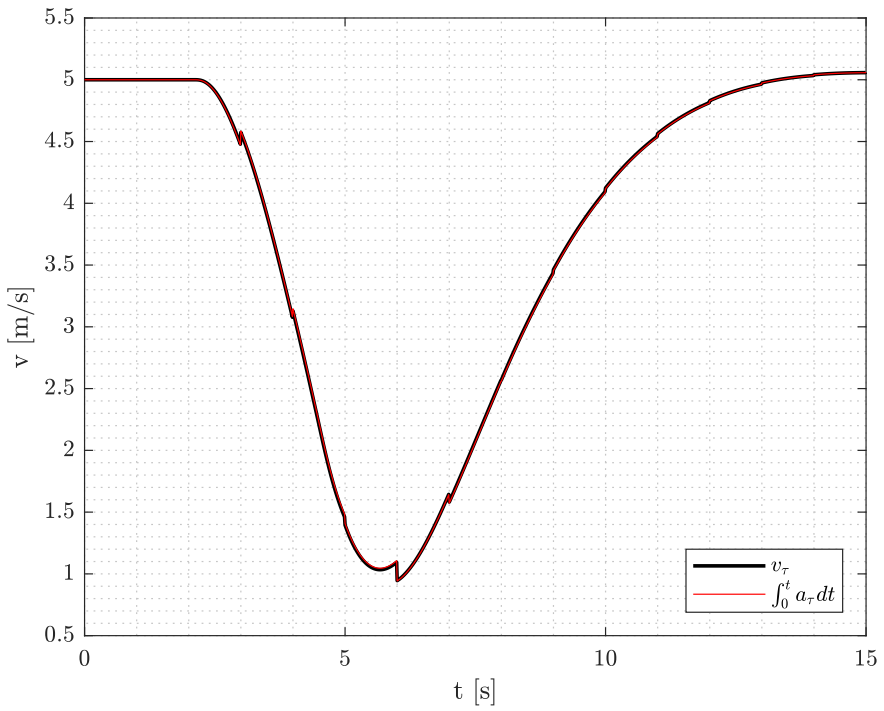


Figure 3.6: The scaled velocity $v_{\tau,i}(t)$ compared with the integral of the scaled acceleration $a_{\tau,i}$ over time. Two consecutive vehicles driving with a constant speed of 5 $\mathrm{m\,s^{-1}}$ where the lead vehicle decelerates unexpectedly in between two plan cycles.

deceleration of 2 $\mathrm{m\,s^{-2}}$, while the trajectory planner still plans to a final velocity of 5 $\mathrm{m\,s^{-1}}$. After a couple of seconds the lead vehicle continues normal operation, by following the output of the planned trajectory. The result of this is illustrated in Figure 3.7 and Figure 3.8. The three states of both vehicles, position, velocity and acceleration, are plotted in Figure 3.7. The black lines represent the actual vehicle states of the vehicles. The blue coloured lines are

the outputs of the B-spline trajectory planner. The update frequency is set to $f_p = 1$ Hz for the sake of visibility of the effect by the timescaling mechanism. The scaled values $\underline{\xi}_{\tau,i}$ are illustrated by the red lines. Figure 3.8 illustrates the relation between the the velocity coming from the spacing policy $v_{c,i}(t)$, as is defined in (3.1) and the read-out velocity $v_{r,i}(\tau(t))$. The figure also illustrates how $\dot{\tau}$ and $\ddot{\tau}$ change over time.

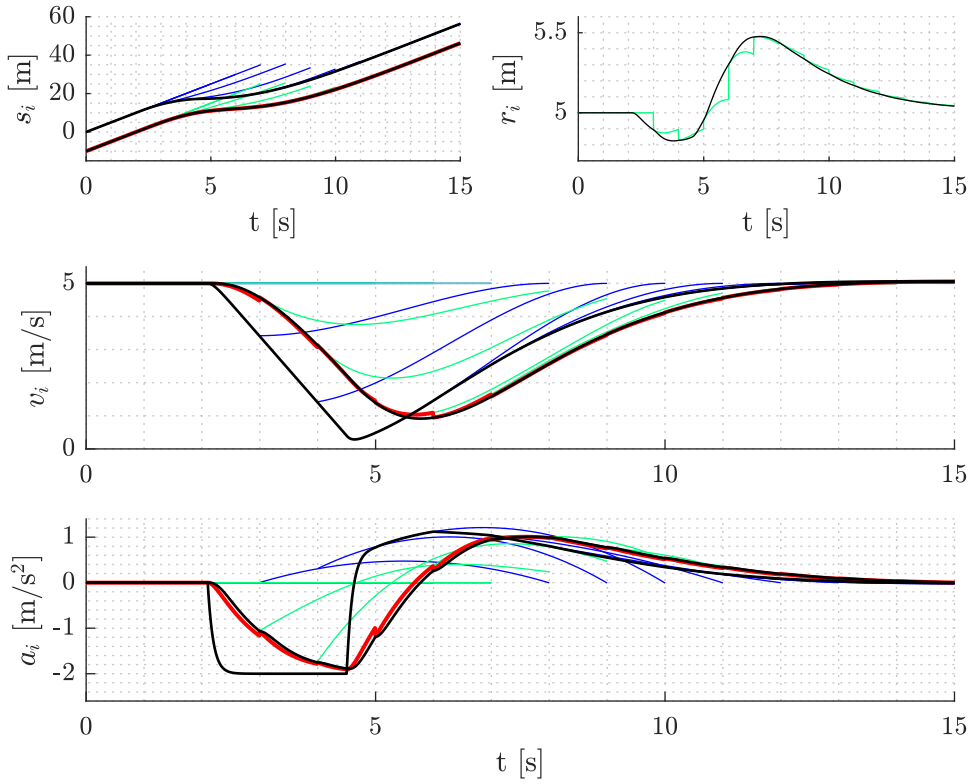


Figure 3.7: States and inter-vehicle distance of two following vehicles when braking not prescribed by the planned trajectory with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the read-out values of vehicle 2 $\underline{\xi}_{r,2}(t)$ (—) and the real vehicle states $\underline{\xi}_i$ (—).

When the lead vehicle brakes unexpected, the following vehicle encounters the preceding vehicle. The velocity, regulated by the spacing policy, $v_{c,i}(t)$ becomes smaller, while trajectory planner still plans to the final velocity of 5 m s^{-1}, due to the communicated trajectory of the lead vehicle. This results in a $\dot{\tau}$ smaller than 1. The trajectory will be read out at a lower pace than real time. This is clearly visible in Figure 3.7 where the actual vehicle states of the velocity and acceleration deviates from the desired trajectory. At the start of a new plan cycle, $\dot{\tau}$ resets back to 1 because of the fact that a new plan cycle initiates from the current states of the vehicle, where the standstill distance, as is defined in (3.10), resets to the measured value of the current states.

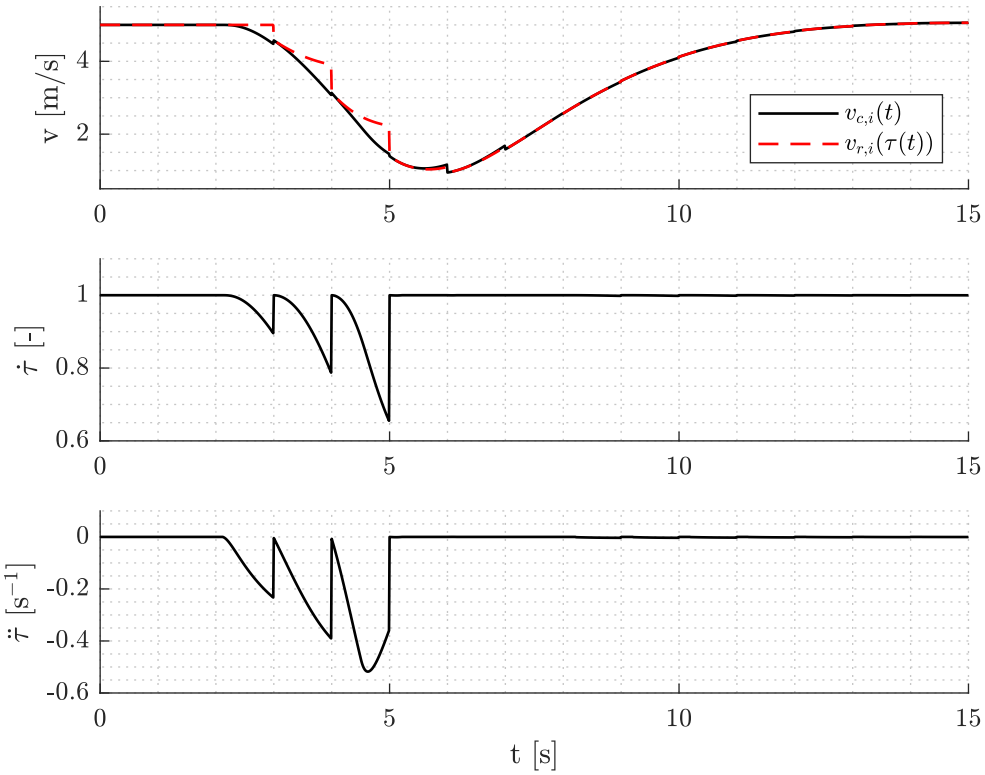


Figure 3.8: Comparing the velocity from the spacing policy $v_{c,i}$ with the read-out value $v_{r,i}(\tau)$ and the behaviour of $\dot{\tau}$ and $\ddot{\tau}$ as function of time.

3.3 Analytical and numerical comparison

Ideally a vehicle follows the same trajectory as the planned (scaled) trajectory with the corresponding velocity and acceleration at time t. Because of the vehicle dynamics, (input) delays, and response time of the engine, vehicle state tracking errors may occur. The state tracking error is defined as the difference between the planned trajectory and the actual vehicle states, as is given below

$$\underline{e}_{tr}(t) = \begin{pmatrix} s_{\tau,i}(t) \\ v_{\tau,i}(t) \\ a_{\tau,i}(t) \end{pmatrix} - \begin{pmatrix} s_i(t) \\ v_i(t) \\ a_i(t) \end{pmatrix} \tag{3.12}$$

A feedback loop with a proportional gain on the velocity and acceleration is designed that minimizes the tracking error. The position is not included in the feedback loop. The position of vehicle i will always be a relative position with respect to the preceding vehicle $i-1$.

A feedforward controller is used, based on the external input in (2.4), to improve the tracking performance of the vehicle, which is defined as

$$u_{ff,i}(t) = \eta \dot{a}_{\tau,i}(t) + a_{\tau,i}(t) \tag{3.13}$$

where $a_{\tau,i}(t)$ and $\dot{a}_{\tau,i}(t)$ are the scaled acceleration and its time derivative respectively. A

schematic representation of the low level vehicle controller is given in Figure 3.9. The output of this controller is given as $u_i(t)$ which is the input for the vehicle.

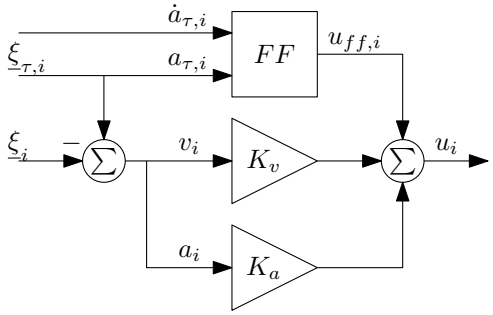


Figure 3.9: Schematic representation of the low level tracking controller

As is described in previous sections, different ways of obtaining the scaled desired acceleration $a_{\tau,i}(t)$ are described. This results in the following options for the feedforward controller:

1. $u_{ff,i}(t) = \eta \dot{a}_{\tau,i}(t) + a_{\tau,i}(t)$, acceleration and jerk numerical derived
2. $u_{ff,i}(t) = a_{\tau,i}(t)$, acceleration numerical derived
3. $u_{ff,i}(t) = \eta \dot{a}_{\tau,i}(t) + a_{\tau,i}(t)$, acceleration analytical and jerk numerical derived
4. $u_{ff,i}(t) = a_{\tau,i}(t)$, acceleration analytical derived

At option 1 and 2 are both the acceleration and jerk numerical derived as is described in Section 3.1, whereas at option 3 and 4 the acceleration is analytical derived by means of $\ddot{\tau}(t)$. To compare these four, noise is added to the 'measured' velocity and acceleration by the vehicle in the simulation. Two situations are simulated to compare the different feedforward options. The first situation is a normal acceleration, from 5 to 10 m/s, by the lead vehicle, so the time scaling mechanism is inactive. The second situation illustrates situation where the lead vehicle suddenly brakes in between plan cycles, while driving at a constant velocity of 5 m/s. This is not prescribed by the planning, so the time scaling mechanism activates. To compare these four situations properly, and see what the feedforward term does, the feedback gains are set to $K_v = 0$ and $K_a = 0$.

3.3.1 Internal vehicle noise

The noise used for the simulations is extracted from real experimental data where the same vehicle is used as is used for the experiments. Vehicle data from earlier performed experiments is extracted and filtered with a zero phase lowpass filter to obtain a more clear signal. The velocity and acceleration signals are filtered and illustrated in Figure 3.10. The remainder of these two signals can be considered as the noise signal, which is illustrated in 3.11 for both the velocity and acceleration. The variance of these signals is determined so that a reasonable noise signal can be simulated, which are $\sigma_v^2 = 0.0013\ \mathrm{m^2\,s^{-2}}$ and $\sigma_a^2 = 0.1791\ \mathrm{m^2\,s^{-4}}$ for the velocity and acceleration respectively.

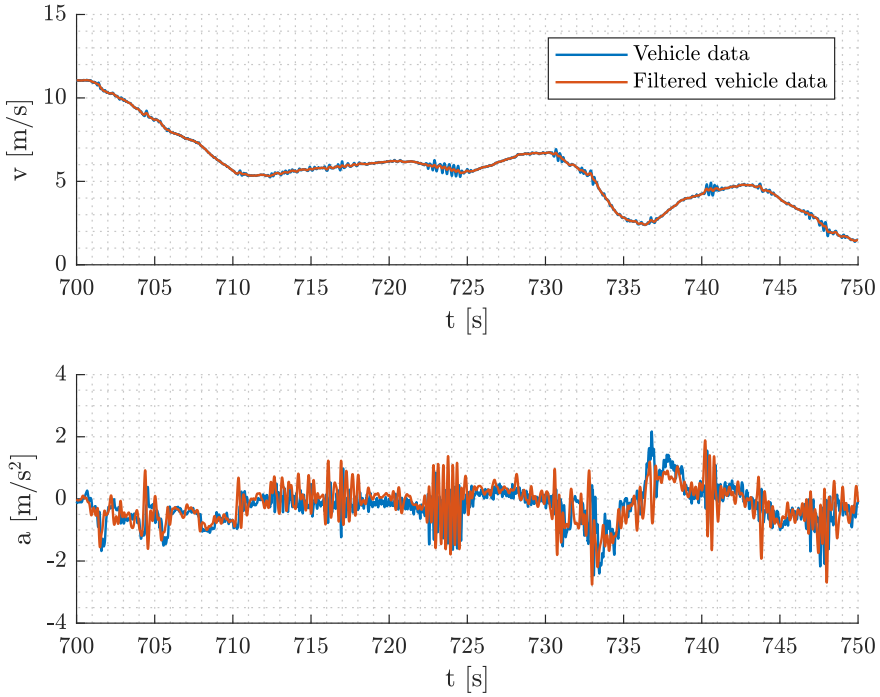


Figure 3.10: Velocity and acceleration data from earlier performed experiments

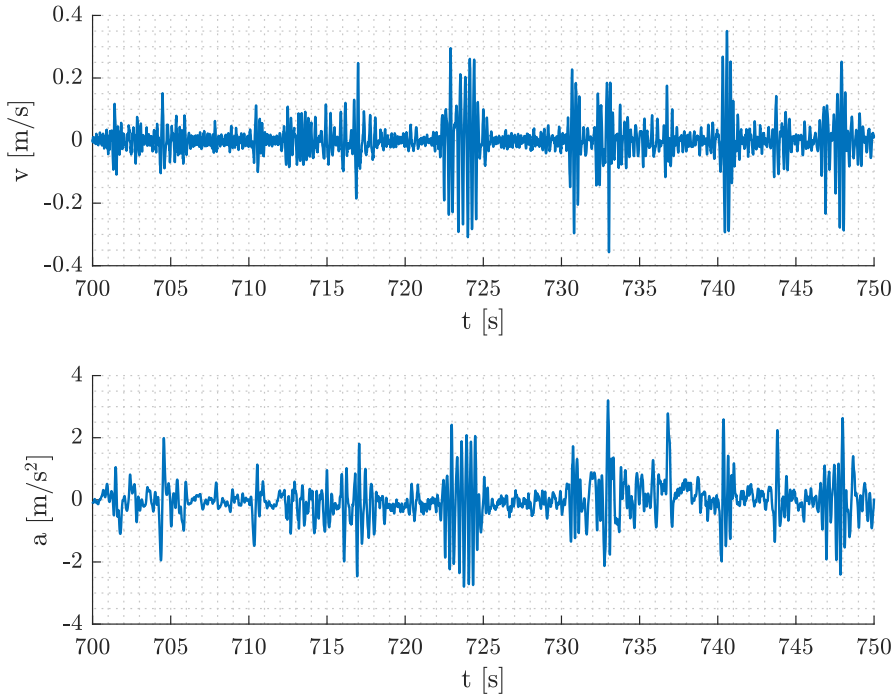


Figure 3.11: Extracted noise signals from the experimental data of Figure 3.10

Measurements done by the radar, which measures the relative position and velocity, also come with noise on the measurements. The radar noise is simulated by a random Band-Limited white noise signal, with both a sample rate of $f_{\text{RadarNoise}} = 10$ Hz.

3.3.2 Tracking error

The four options for the feedforward part of the low level controller are compared based on error measurements to decide which one is the 'best' option. The tracking error is defined in (3.12). The noise signal is added in the simulation model and the tracking error signal is calculated for every time sample of the simulations which is sampled with a frequency of $f_s = 100$ Hz. To make a fair comparison between the four feedforward options, the RMS and a maximum error value are calculated using the whole time domain of the simulation. The RMS value for the tracking error is calculated as follows:

$$\underline{e}_{tr,RMS} = \sqrt{\frac{1}{k}\sum_{j=0}^{k} \underline{e}_{tr}^2(t_j)} \tag{3.14}$$

The graphs corresponding with the different controller options are illustrated in Appendix A. The values for the trajectory error during an velocity change from 5 to 10 m/s is given in Table 3.1. The values for the trajectory error during time scaling is given in Table 3.2.

As one can see in both tables, the errors of the third signal are the smallest compared to the others. Nevertheless, this option still contains the variable $\dot{a}_{\tau,i}$ which is determined through numerical differentiation. The error signals of the fourth option are larger than option 3, but the differences are marginal. As one can see in the figures in Appendix A, option 3 contains more jerk than option 4. So for the sake of comfortable driving, it is concluded that option 4 is the best option to use as feedforward term. This option will be used for the remainder of this report.

Table 3.1: RMS and max error of the second vehicle when accelerating from 5 to 10 m/s.

	Option 1	**Option 2**	**Option 3**	**Option 4**
$\underline{e}_{tr,RMS}$	$\begin{pmatrix}0.0020\\0.0403\\1.3814\end{pmatrix}$	$\begin{pmatrix}0.0028\\0.0417\\1.0163\end{pmatrix}$	$\begin{pmatrix}0.0044\\0.0501\\0.4416\end{pmatrix}$	$\begin{pmatrix}0.0045\\0.0506\\0.4400\end{pmatrix}$
$\underline{e}_{tr,max}$	$\begin{pmatrix}0.0081\\0.1649\\12.4939\end{pmatrix}$	$\begin{pmatrix}0.0119\\0.1657\\9.4610\end{pmatrix}$	$\begin{pmatrix}0.0214\\0.1789\\1.7350\end{pmatrix}$	$\begin{pmatrix}0.0221\\0.1797\\1.7133\end{pmatrix}$

Table 3.2: RMS and max error of the second vehicle with active time scaling

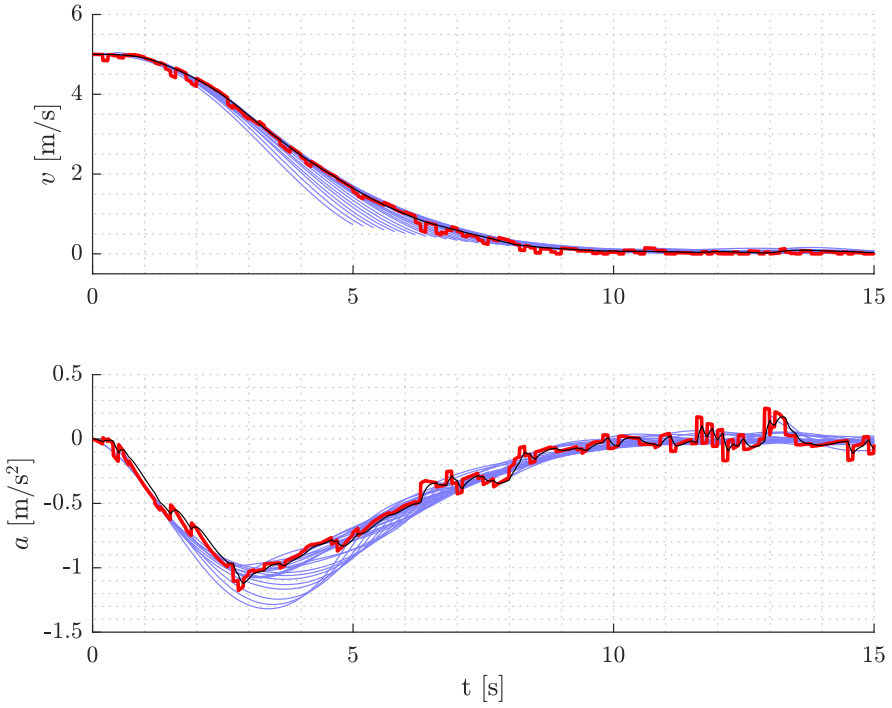
	Option 1	**Option 2**	**Option 3**	**Option 4**
$\underline{e}_{tr,RMS}$	$\begin{pmatrix}0.0019\\0.0401\\1.4283\end{pmatrix}$	$\begin{pmatrix}0.0031\\0.0479\\1.4692\end{pmatrix}$	$\begin{pmatrix}0.0044\\0.0502\\0.4437\end{pmatrix}$	$\begin{pmatrix}0.0044\\0.0506\\0.4461\end{pmatrix}$
$\underline{e}_{tr,max}$	$\begin{pmatrix}0.0069\\0.1649\\12.5433\end{pmatrix}$	$\begin{pmatrix}0.0136\\0.1644\\14.1527\end{pmatrix}$	$\begin{pmatrix}0.0213\\0.1789\\1.7149\end{pmatrix}$	$\begin{pmatrix}0.0218\\0.1792\\1.6168\end{pmatrix}$

3.4 Upper bound for $\ddot{\tau}$

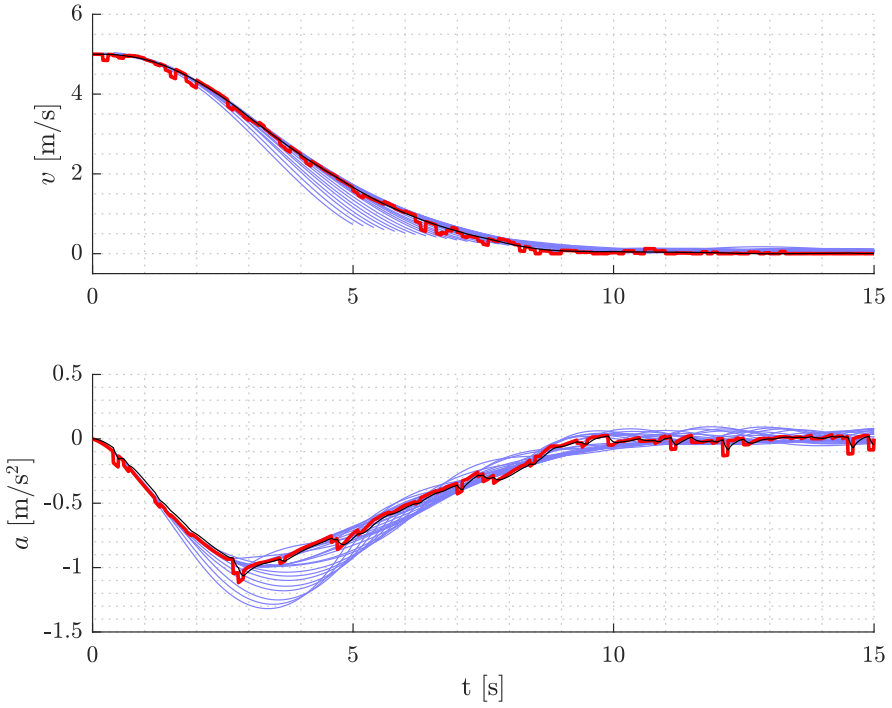
When the time scaling mechanism is inactive, the trajectory is read-out at real time with $\dot{\tau}(t) = 1$ and $\ddot{\tau}(t) = 0$. Noise on the measured vehicle states and on the radar signals may result in situations where $\dot{\tau}(t) \neq 1$ and $\ddot{\tau} \neq 0$. Approaching low velocities may then result in unwanted high values for $\ddot{\tau}(t)$ due to the fraction in (3.8). With as a result very high unwanted amplifications in the desired accelerations. Since the time scaling method is only allowed to perform retarding actions, the desired acceleration of the vehicle has to be less or equal to the acceleration of the planned trajectory at time t, in other words: $a_{\tau,i}(t) \leq a_{r,i}(\tau(t))$. An upper bound for $\ddot{\tau}(t)$ can be defined as:

$$\begin{aligned} a_{r,i}(\tau(t))\dot{\tau}(t)^2 + v_{r,i}(\tau(t))\ddot{\tau}(t) &\leq a_{r,i}(\tau(t)) \\ \ddot{\tau}(t) &\leq \frac{a_{r,i}(\tau(t)) - a_{r,i}(\tau(t))\dot{\tau}(t)^2}{v_{r,i}(\tau(t))} \\ \ddot{\tau}(t) &\leq \frac{a_{r,i}(\tau(t))(1 - \dot{\tau}(t)^2)}{v_{r,i}(\tau(t))} \end{aligned} \tag{3.15}$$

The result of the functionality of this upper bound for $\ddot{\tau}$ is illustrated in Figure 3.12a and Figure 3.12b with the planned trajectories of the host vehicle $\underline{\xi}_{r,2}(t)$ (—), the desired states of the vehicle $\underline{\xi}_{\tau,2}(t)$ (—) and the vehicle states $\underline{\xi}_2(t)$ (—). A situation is simulated where a lead vehicle accelerates from a standstill position. The following vehicle follows using a trajectory planner and the spacing policy. The parameters for the simulated noise and input delays are obtained by real vehicle data from earlier performed experiments. If the velocity coming from the trajectory planner $v_{r,i}(\tau(t))$ approaches zero, the acceleration of the vehicle becomes larger.



(a) Without the condition $a_{\tau,i}(t) \leq a_{r,i}(\tau(t))$



(b) With the condition $a_{\tau,i}(t) \leq a_{r,i}(\tau(t))$

Figure 3.12: Planned trajectories of the following vehicle $\underline{\xi}_{r,2}(t)$ (—), the desired states of the vehicle $\underline{\xi}_{\tau,2}(t)$ (—) and the real vehicle states $\underline{\xi}_2(t)$ (—).

3.5 Summary

A method is developed to prevent vehicles from colliding when a preceding vehicle drives with a slower velocity than prescribed by the trajectory planner. This time scaling method has shown that it can adapt the velocity of a vehicle when the inter-vehicle distance is smaller than prescribed by the trajectory planner. A comparison is made between different implementations of the time scaling mechanism. Eventually the choice where the acceleration is determined analytical is preferable in comparison with a numerical differentiation of the adapted velocity. An upper-bound is defined to prevent the scaled acceleration resulting in a higher acceleration than is prescribed by the trajectory planner. The resulting method acts as desired.

Chapter 4

Experiments

The time scaling method is tested on real vehicles. In preparation of the experiments on the full-scale vehicles, a simulation study using a high fidelity model was uses. This simulation study is used to verify that the time scaling approach still operates as desired subject to noise and delays. Until now a simplified vehicle model was used. At first a more extensive vehicle model is created in the MATLAB/Simulink environment. Some different use cases are considered. This is followed by performing experiments on two real vehicles, where also different use cases are considered.

The chapter starts with describing the experimental setup on which the experiments will be performed. This is followed by describing the more extended vehicle model which corresponds with the experimental setup. Next the simulation results of the more extended vehicle model are shown and compared with the simplified model from (2.4). The chapter is concluded with the performed experiments and the results.

4.1 Experimental setup

The experimental setup contains two Renault Twizy's. One Renault Twizy is depicted in Figure 4.1 These two vehicles are automated as a part of the i-CAVE project [9]. The vehicles are equipped with Simulink Real-time computers. These real-time computers have acces to the vehicles native CAN-bus. The CAN-bus provides the real-time computer with the vehicle's velocity by means of a motor RPM. The acceleration and gyroscopic measurements are done by a Bosch mm5.10 inertial measurement unit, abbreviated as IMU. The vehicles location is provided by a u-Bloks EVK-M8T GPS system, as well as a common clock. A Bosch MRRevo14F radar sensor measures the inter-vehicle distance and relative velocity. Two Wi-Fi ITS-G5 routers from Rendits Routers ensures the V2V communication between the two Renault Twizy's. The routers enable the Simulink platform to transmit ITS-G5 messages, such as the CAM and DENM message. The routers are also capable of transmitting a custom message that can be used to communicate arbitrary bytes between the vehicles. In this experiment, this additional message is used to communicate the coefficients of the planned trajectory, denoted by $\mathcal{P}_i$, as well as the current planning time $t_{c,i}$ and the planing time horizon T_i. The trajectory planner plans with a frequency of $f_p = 5$ Hz, the radar that measures the inter-vehicle distance and relative velocity samples with a frequency of approximately $f_r = 10$ Hz, whereas the rest of the setup samples with a frequency of $f_s = 100$ Hz.

Figure 4.1: Renault Twizy

4.2 Vehicle modelling

Before executing experiments on real vehicles, simulations are done. These are used to verify that the time scaling method still functions properly with noise and delays. To make the simulation as accurate as possible, a more realistic vehicle model is implemented and described in following subsections.

4.2.1 Longitudinal vehicle dynamics

Until now the driveline dynamics were represented by the time constant η. Before performing the simulations which can be compared with experiments afterwards, a more accurate vehicle model needs to be considered. This vehicle model includes for example some resistance forces and acceleration limitations.

The physical model of the vehicle can be described using Newton's second law:

$$\begin{aligned} F_{res} &= M \cdot a \\ F_x - F_{fr} &= M \cdot a \end{aligned} \tag{4.1}$$

with F_{res} the resulting forces on the system, m the effective mass of the vehicle and a the acceleration of the vehicle in longitudinal direction. The resulting forces can be defined as the difference between the external force put on the system F_x, coming from the electrical engine, and a summation of the resistive forces F_{fr}. The resistive forces can be divided in three different forces, the rolling resistance, drag resistance and viscous friction, which are

respectively defined as:

$$F_r = M \cdot g \cdot C_r \tag{4.2}$$

$$F_d = \frac{1}{2} \cdot \rho \cdot v^2 \cdot C_d A \tag{4.3}$$

$$F_v = v \cdot C_v \tag{4.4}$$

with v being the vehicle speed. The other parameters are given in Table 4.1. It is assumed

Table 4.1: Parameters used in the vehicle simulation model

Parameter	**Value**	**Unit**
Effective vehicle mass M	755	$[kg]$
Gravitational acceleration constant g	9.81	$[m/s^2]$
Air density ρ	1.225	$[kg/m^3]$
Rolling resistance coefficient C_r	0.0162	$[-]$
Aerodynamic drag $C_d A$	0.877	$[m^2]$
Viscous friction C_v	5.55	$[Ns/m]$

that the road inclination is zero, otherwise the tangential gravity component is also considered in the summation of the friction forces. Now the acceleration of the vehicle can be determined for a known F_x on the vehicle. The force on the system is provided by the engine. This force corresponds with a certain throttle position TPS. In simulation environment, the force on the vehicle F_x is determined by a torque map [1]. The torque map used in the simulations is given by Figure 4.2. This torque map has input limitations which are dependant of the velocity of the vehicle, which was not the case as the simplified vehicle mode. A schematic representation of how the physical vehicle is modelled, is given by Appendix B, with the parameters from Table 4.1 included and where R_w represents the effective rolling radius and i_d the driveline transmission ratio and ω_e the engine speed.

4.2.2 Acceleration control

The required force that is needed on the system, to obtain the adapted desired acceleration $a_{\tau,i}$, is also defined by.

$$F_{x,r} = M \cdot a_{\tau,i} + F_{fr} \tag{4.5}$$

The desired acceleration is known, as is the vehicle mass. The required force on the system is controlled by a feedback PID controller and a feedforward controller, which compensates for the resistive force losses. The actual vehicle has an internal vehicle input delay of $\phi = 0.14$ s. The electrical engine has an impulse response which is simulated as an first order dynamical model. The total driveline dynamics are of the form

$$G(s) = \frac{1}{\theta s + 1} e^{-\phi s} \tag{4.6}$$

with $\theta = 0.0687$ s. The data for the inverted torque map and the parameters for the vehicle input delay and response time are obtained by earlier performed experiments [9]. The controller consists of a weak integrator

$$C_{wi}(s) = \frac{s + \omega_{wi}}{s} \tag{4.7}$$

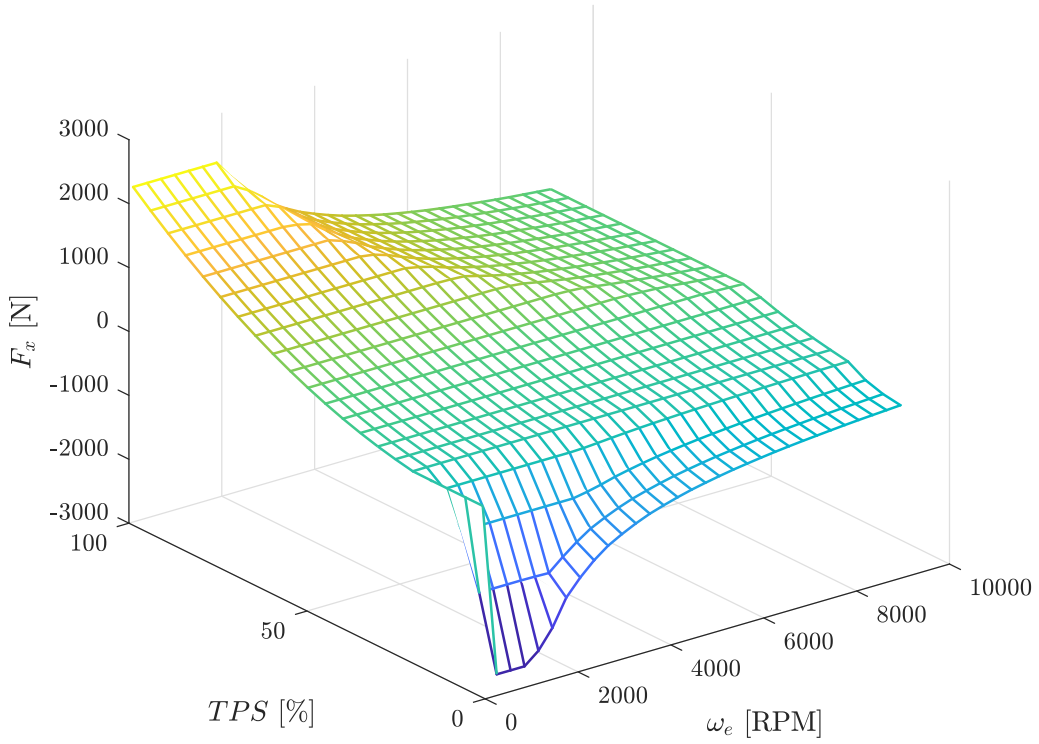


Figure 4.2: Torque map to determine the F_x by the current engine speed and requested TPS [1]

with $\omega_w i = 100$ rad s^{-1} and a low pass filter

$$C_{lp}(s) = \frac{1}{\frac{1}{\omega_c 6 s} + 1} \tag{4.8}$$

with the crossover frequency $\omega_c = 2\pi 0.55$ rad s^{-1}. A Bode diagram of the sensitivity function of the closed loop driveline dynamics is given in Figure 4.3. The feedback controller $C_{wi}(s)C_{lp}(s)$ is with the use of MATLAB converted to a parallel-form PID controller. The PID controller is of the form (note that the PID controller is given in discrete time):

$$C_{PID}(z) = K_p + K_i \frac{T_s}{z-1} + K_d \frac{1}{T_f + \frac{T_s}{z-1}} \tag{4.9}$$

with $Kp = -0.137$, $K_i = 3.58$, $K_d = 0.0073$, $T_f = 0.0543$ and $T_s = 0.01$.

The required torque T_r for the engine to deliver, is obtained by multiplying the required force by the vehicle wheel radius. The input to the engine is a required throttle position, TPS_r, and is found by inverting the torque map. The inputs of the map are the engine speed, expressed in RPM, and the required throttle position. A schematic representation of the total vehicle control model is given in Figure 4.4.

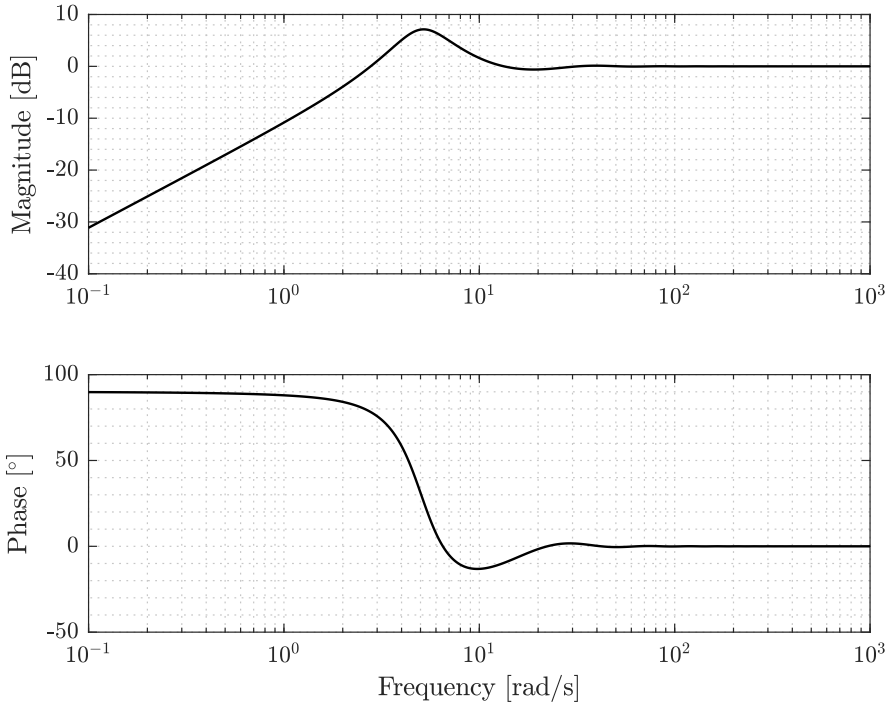


Figure 4.3: Bode diagram of the sensitivity function of the closed loop driveline dynamics.

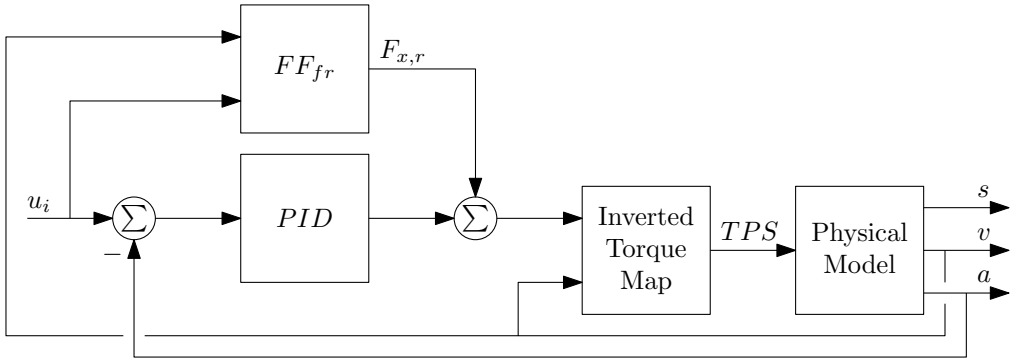


Figure 4.4: Schematic representation of the vehicle control model.

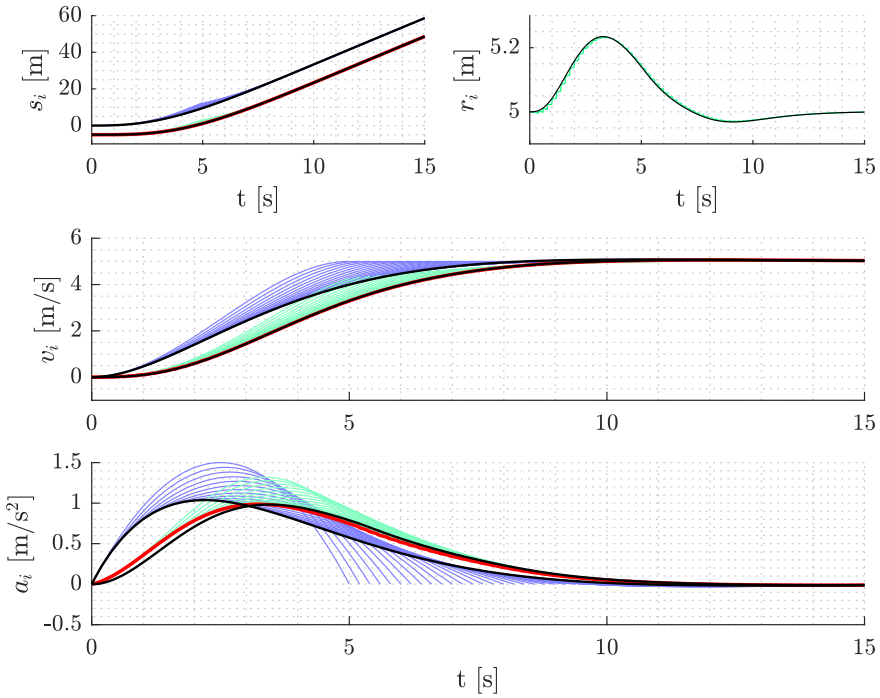
4.3 Simulation results

The more extended vehicle model, as is described in 4.2, is adopted in the simulation environment. More accurate simulations included with delays and noise are performed in the MATLAB/Simulink environment. The simulations are done with a sample frequency of $f_s = 100$ Hz, while the trajectory planner updates at a frequency of $f_p = 5$ Hz and the radar 'measures' the relative position and velocity with a frequency of $f_r = 10$ Hz. The sample frequencies of the sensors are made identical to those on the experimental setup. The vehicle string parameters are during the simulations defined as $h = 1$ s, $\underline{r}_i = 5$ m and $L_i = 0$ m (note that the choice for $L_i = 0$ m has no consequence for the platoon dynamics). The constant represent-

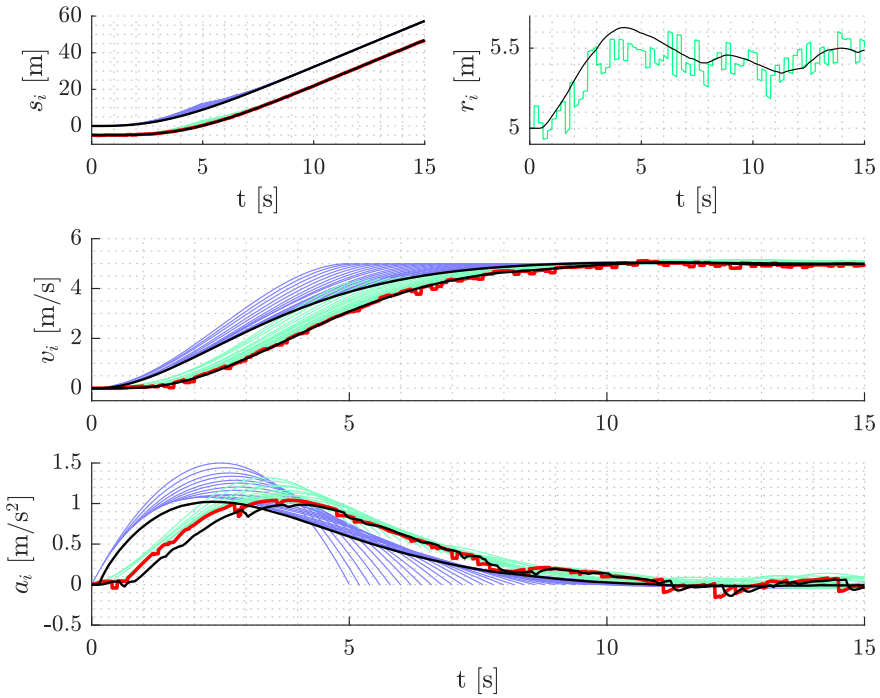
ing the driveline dynamics is set to $\eta = 0.2$ s and used in the simplified vehicle model. The standstill distance $r_{c,i}(t)$ used to determine $v_{c,i}(t)$ as defined in (3.1) is substituted by $r_{r,i}(t)$ which is defined in (3.10).

Two different use cases are considered during the simulations. The first one is a situation where two vehicles are following the planned trajectory, starting from a standstill situation and accelerating to a final desired velocity. This simulation is done for the simple vehicle model (2.4) from Chapter 2 and for the more extensive vehicle model as is described in previous section to make a comparison between the models. The result of the simple and more extensive vehicle model are illustrated in Figure 4.5a and 4.5b respectively, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}$ (—) and the vehicle states $\underline{\xi}_i$ (—). Note that noise and delays are also incorporated in the more extensive vehicle model. As one can see in the figures the full vehicle model acts similar to the simplified vehicle model, with the main difference that the time scaling becomes active in Figure 4.5b as a consequence of simulated measurement noise. This is visible in the little downward peaks in the acceleration plot. The delays and noise signals are noticeable in Figure 4.5b. The vehicle states $\underline{\xi}_2(t)$ are noisy and deviate from the desired trajectory of the following vehicle $\underline{\xi}_{r,2}(t)$, but does not affect the vehicle following objective. Note that in the figure the vehicle states are plotted before injecting the noise on the acceleration and velocity for the sake of visibility.

The second use case is a situation where the two vehicles are driving at a constant velocity, and the lead vehicle decelerates by a simulated disturbance, while the trajectory planner of the lead vehicle still aims for a certain final desired velocity. In this case, the time scaling mechanism is required to adapt the desired trajectory in order to maintain a safe distance. The result of this simulation is illustrated in Figure 4.6. As one can see in the figure, both vehicles drive at a constant velocity of $v_i(t) = 5\ \mathrm{m\,s^{-1}}$. After a couple of seconds, a disturbance is added to the lead vehicle by means of a deceleration of $2\ \mathrm{m\,s^{-2}}$. The trajectory planner still plans trajectories to the defined velocity of $5\ \mathrm{m\,s^{-1}}$. The second vehicle is able to decelerate by means of the time scaling method, $\underline{\xi}_{\tau,2}(t)$ (—) deviates from the planned trajectories $\underline{\xi}_{r,2}(t)$ (—). Even when the communicated parameters result in planned trajectories to the final defined velocity. After the applied disturbance on the lead vehicle, the trajectory planner plans to the terminal velocity and both vehicles follow the planned trajectories until they reach the terminal velocity of $5\ \mathrm{m\,s^{-1}}$. Both the noise and delays on the vehicles are visible in the figure without affecting the time scaling method. The planned trajectories for the acceleration of both vehicles are initialised from $a_{\tau,i}(t)$ instead of $a_i(t)$. This is because the trajectory planner is very sensitive for noise signals in the acceleration of the vehicle.



(a) Using vehicle model from (2.4)



(b) Using the more extensive vehicle model

Figure 4.5: Simulation of two following vehicles while accelerating. The planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}$ (—) and the real vehicle states $\underline{\xi}_i$ (—).

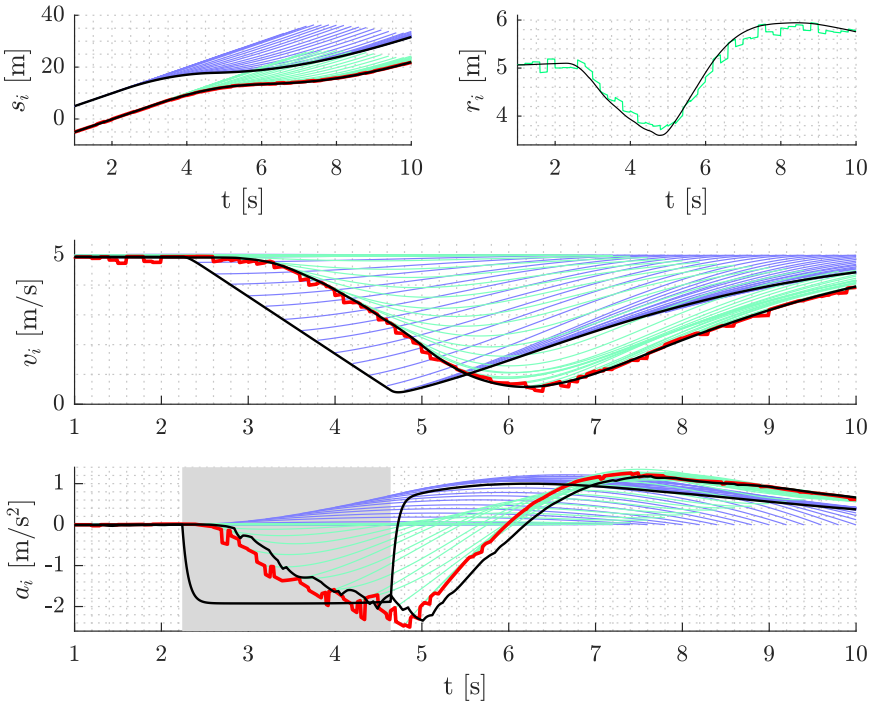


Figure 4.6: Simulation of two following vehicles while decelerating by disturbance, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}$ (—), the vehicle states $\underline{\xi}_i$ (—) and the area where the time scaling is active (▫).

4.4 Experimental results

The experiments are done on two vehicles following each other by means of cooperative trajectory planning. The headway time $h = 0.6$ s, the vehicle length $L_i = 0$ m and the minimum standstill distance $\underline{r}_i = 8$ m during the experiments. To create a benchmark, the time scaling method is not included in this experiment, in other words, the values for $\dot{\tau}(t)$ and $\ddot{\tau}(t)$ are set to 1 and 0 respectively during the whole experiment. Note that the feedback gains K_v and K_a (Figure 3.9) are both set to zero such that $u_i(t) = a_{\tau,i}(t)$. The effect of these feedback terms are very small compared to the low-level acceleration control. Leaving out both feedback terms K_v and K_a makes the control structure simpler. The result of this experiment is illustrated in Figure 4.7. Both vehicles are initially stationary and are spaced according to the desired spacing policy. The trajectory plans towards a velocity of $v_{r,i}(t) = 0$ $\mathrm{m\,s^{-1}}$ with as a result that both vehicles remain stationary. After 5 seconds the terminal velocity of the trajectory planner is set to 10 $\mathrm{m\,s^{-1}}$. When the velocity is achieved, both vehicles drive at a constant speed for a couple of seconds. Finally the trajectory planner is set to a velocity of 0 $\mathrm{m\,s^{-1}}$, and both vehicles come to a completely standstill situation. When both vehicle decelerate to a stationary situation situation, they both roll out because the engine no longer supplies braking torque. Note that the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}$ are the planned trajectories from the moment they are *'known'* for the following vehicle.

After this experiment the same procedure is executed for both vehicles, but now with the time scaling method active. For safety reasons the velocity for the trajectory planner to aim for is set to 5 instead of 10 $\mathrm{m\,s^{-1}}$. The result of this experiment is given in Figure 4.8. The figure illustrates unwanted behaviour in the second vehicle. The following vehicle has an departure delay which is a consequence of a significant incorrect measurement by the radar. As one can see in the figure, the inter-vehicle distance measured by the radar smaller than measured at the standstill position. With as a result that the following vehicle does not leave its initial position. Also a desired acceleration of -10 $\mathrm{m\,s^{-2}}$ is visible in the figure. A result of this is that the vehicle even longer remains stationary as a consequence of the integrator in the feedback controller. After the departure of the following vehicle, extreme oscillations are visible in both the velocity and acceleration of the following vehicle, while the radar measurement gives no significant errors in the relative position.

The actual inter-vehicle distance is defined as (2.17). A new plan cycle initialises with a inter-vehicle distance of $r_{r,i}(t_{c,i})$ with as a consequence that $v_{c,i}(t_{c,i}) = v_{r,i}(t_{c,i})$ at the start of every plan cycle. Differences in the velocity ($v_{c,i}(t) < v_{r,i}(t)$), measured by the radar, may result in decelerating by time scaling, while that is not necessary from a safety point of view. If the actual inter-vehicle distance $r_i(t)$ is smaller than the planned inter-vehicle distance $r_{r,i}(t)$ at time t, the time scaling mechanism activates as a result of this. The acceleration input for the low-level controller will be adapted with as a result unwanted oscillations in the velocity and acceleration.

4.4.1 Constant standstill distance $\underline{r}_i$

Because of the oscillations in the velocity and acceleration, $r_{c,i}$ will be substituted for the minimal inter-vehicle distance $\underline{r}_i$ from Section 2.3.2. A new experiment is performed where the lead vehicle plans from 0 to 10 $\mathrm{m\,s^{-1}}$, depicted in Figure 4.9, followed by a deceleration to a standstill situation. As one can see in the figure, the oscillations are disappeared. So the option to use a constant value for the standstill distance to obtain $v_{c,i}(t)$ works well during

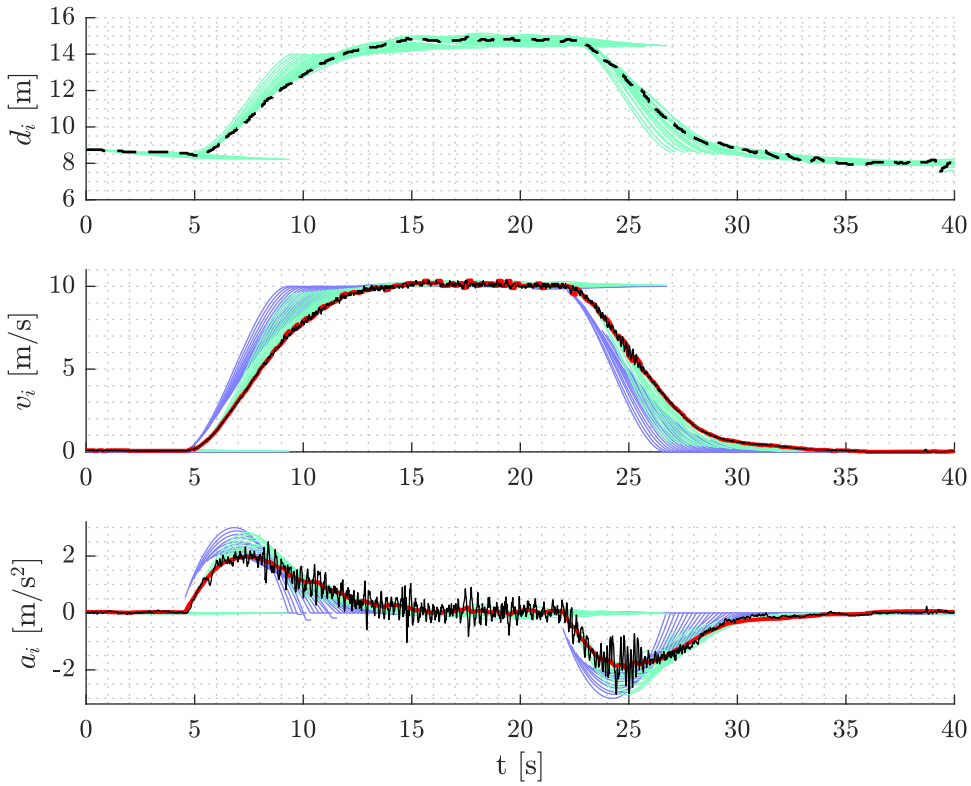


Figure 4.7: Experiment of two following vehicles, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}(t)$ (—), the vehicle states $\underline{\xi}_i(t)$ (—) and relative position measured by the radar $d_i(t)$ (- -)

accelerating, constant driving and decelerating.

Next an experiment is done where a deceleration by disturbance activates the time scaling mechanism. The results of this experiment are illustrated in Figure 4.10 and 4.11. The lead vehicle executes its planned trajectory at constant velocity of 5 $\mathrm{m\,s^{-1}}$ and the host vehicle follows according to the spacing policy. This is followed by an unintended deceleration of the first vehicle, applied by manually pressing the footbrake. The trajectory planner of the lead vehicle still plans towards 5 $\mathrm{m\,s^{-1}}$. In Figure 4.10 is clearly visible that the following vehicle also decelerates according to the spacing policy by means of time scaling. Figure 4.11 gives a good representation of the velocity profile of both vehicles. Both the measured velocities of the vehicles are originating from separate vehicles contrary to previous figures, where the plots are coming from the host vehicle. This is done by synchronising the clocks of both vehicles using the MATLAB function `alignsignals` which provides the delay between the two clock signals which originate from a common GPS clock. The delay is used to shift the signal of the host vehicle so that both signals are synchronised. The result of this method is illustrated in Figure 4.12. As one can see in Figure 4.11 the host vehicle decelerates more than the lead vehicle, which implies that the string of vehicles is not string stable any more. This is not an issue, as time scaling is a safety feature, intended for preventing the vehicles from colliding.

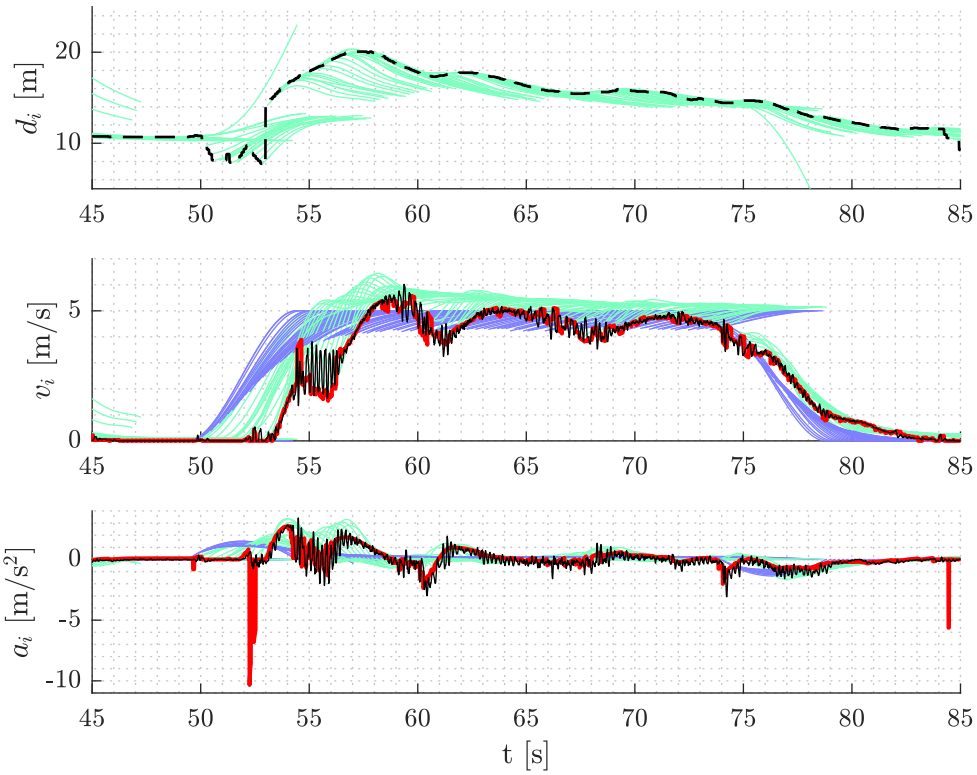


Figure 4.8: Experiment of two following vehicles, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}(t)$ (—), the real vehicle states $\underline{\xi}_i(t)$ (—) and relative position measured by the radar $d_i(t)$ (- -)

The tracking error, as is described in (3.12), of the four experiments is compared. The position of the following vehicle is a relative position with respect to the preceding vehicle. Therefore the position state is neglected in tracking error. The results are given in Table 4.2. Every column in Table 4.2 represents a experiment, indicated by the corresponding figure number. As one can see in the table, the benchmark has the lowest values for the RMS

Table 4.2: RMS and the absolute maximum error values of the executed experiments

	Figure 4.7	**Figure 4.8**	**Figure 4.9**	**Figure 4.10**
$\underline{e}_{tr,RMS}$	$\begin{pmatrix}0.1082\\0.2173\end{pmatrix}$	$\begin{pmatrix}0.2954\\0.6902\end{pmatrix}$	$\begin{pmatrix}0.1465\\0.1609\end{pmatrix}$	$\begin{pmatrix}0.1936\\0.2789\end{pmatrix}$
$\underline{e}_{tr,max}$	$\begin{pmatrix}0.6795\\1.2406\end{pmatrix}$	$\begin{pmatrix}2.3133\\10.9153\end{pmatrix}$	$\begin{pmatrix}0.6325\\1.0023\end{pmatrix}$	$\begin{pmatrix}1.1346\\1.6910\end{pmatrix}$

and maximum error. The experiment with a variable standstill distance and the time scaling mechanism active, corresponding with Figure 4.8, has a RMS value of the acceleration which is 218% higher than the benchmark of Figure 4.7. The RMS value of the acceleration of Figure 4.9 is 26% lower compared to the benchmark. The time scaling experiment, corresponding

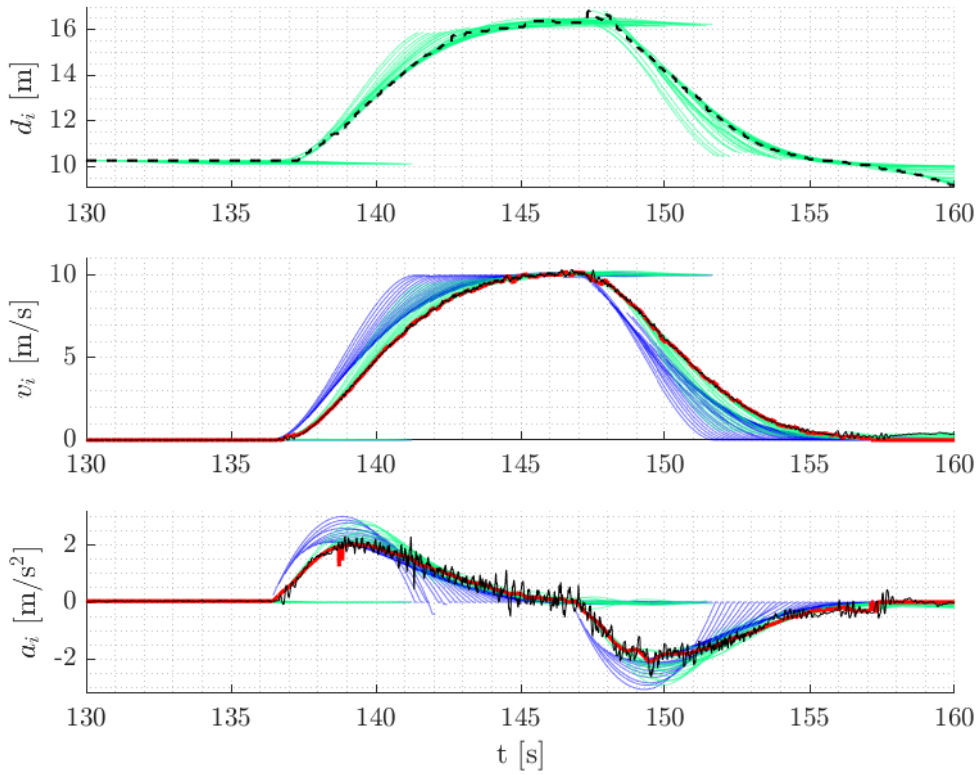


Figure 4.9: Experiment of two following vehicles, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}(t)$ (—), the real vehicle states $\underline{\xi}_i(t)$ (—) and relative position measured by the radar $d_i(t)$ (- -)

with Figure 4.10, has a RMS value which is 28% higher compared to the acceleration of the benchmark. Concluded is that the using a constant standstill distance will not affect the vehicles tracking performance much, even when the time scaling mechanism activates.

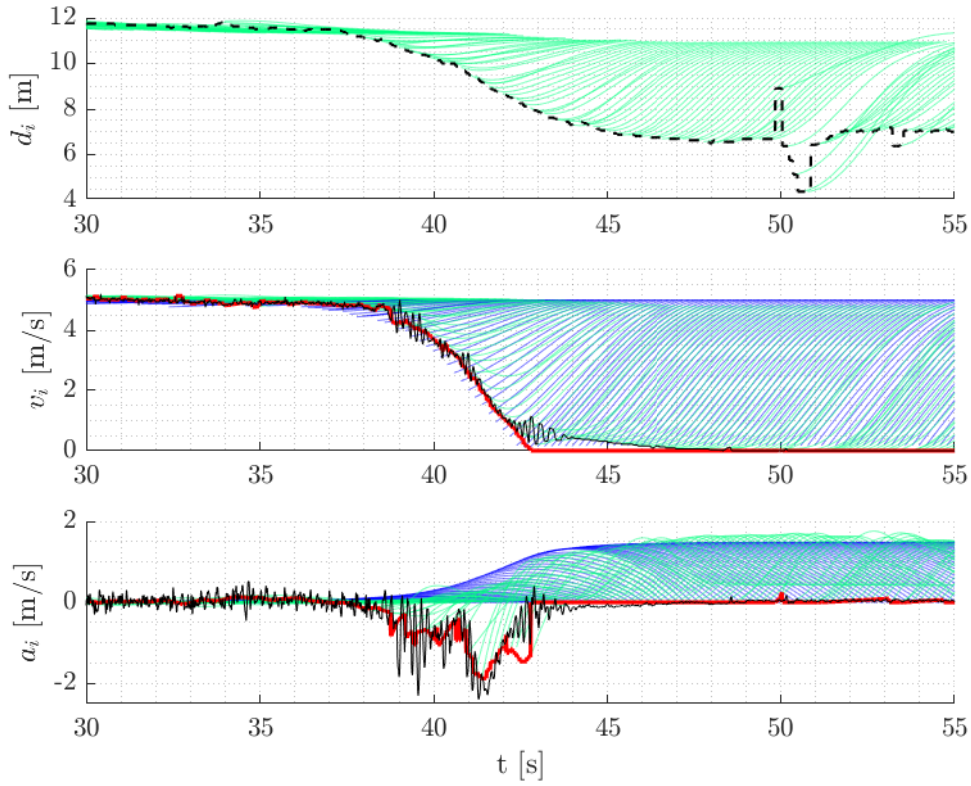


Figure 4.10: Experiment of two following vehicles, with the planned trajectories of vehicle 1 $\underline{\xi}_{r,1}(t)$ (—) and 2 $\underline{\xi}_{r,2}(t)$ (—), the desired trajectory of vehicle 2 $\underline{\xi}_{\tau,2}(t)$ (—), the real vehicle states $\underline{\xi}_i(t)$ (—) and relative position measured by the radar $d_i(t)$ (- -)

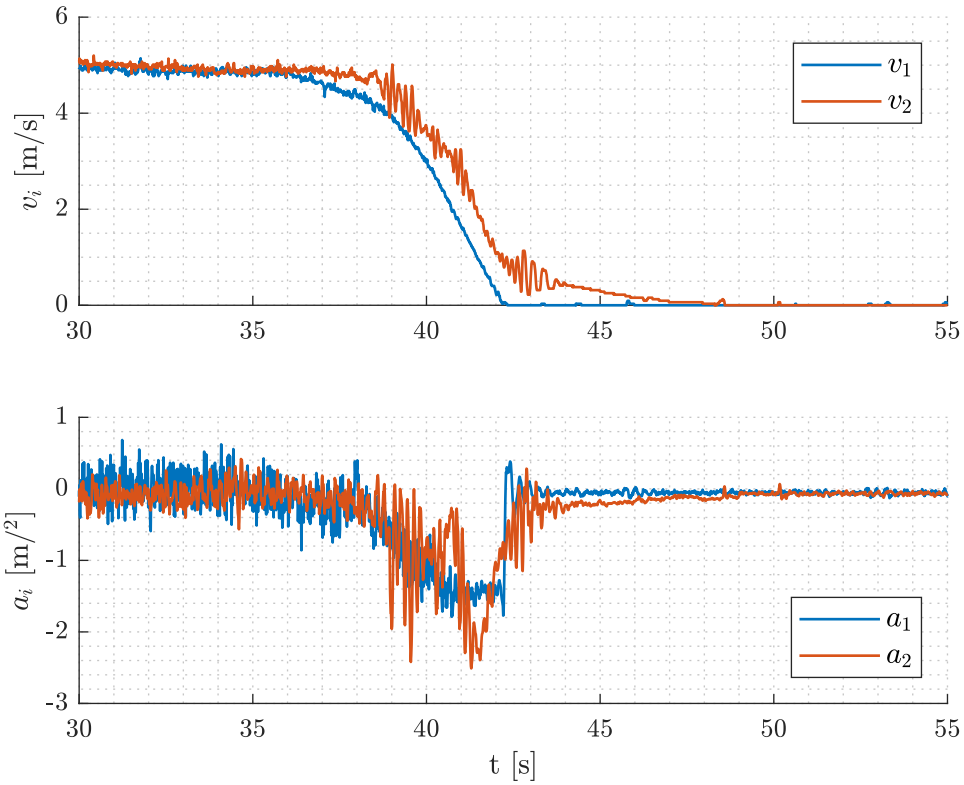


Figure 4.11: Velocity and acceleration of both vehicles during the time scaling experiment

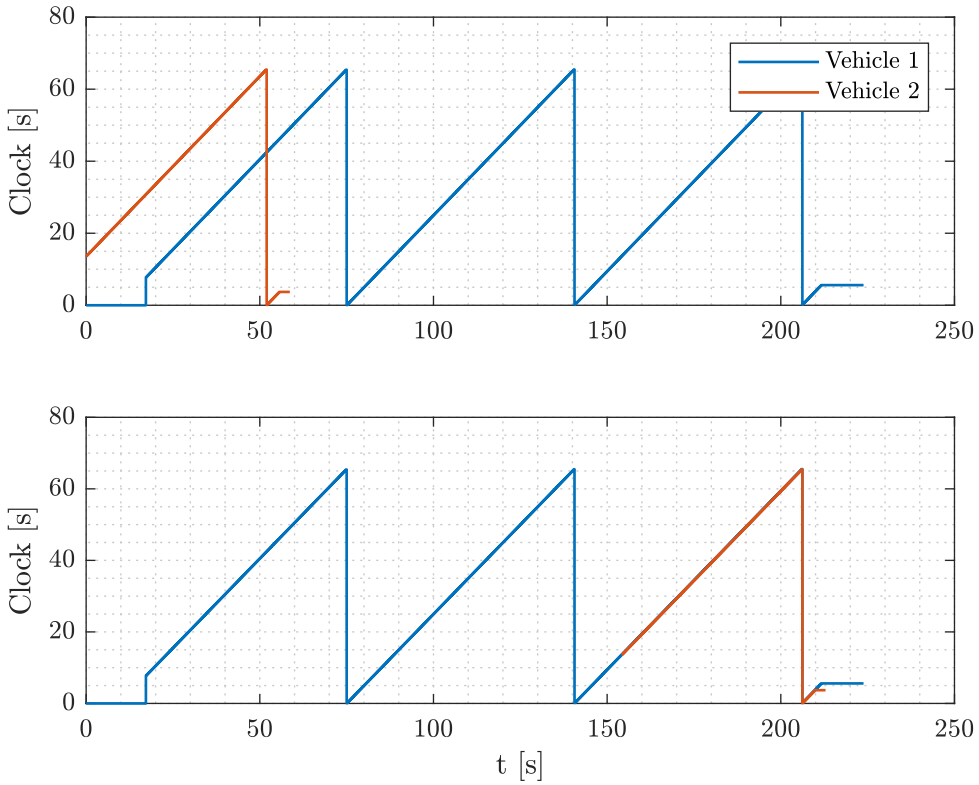


Figure 4.12: Synchronising the individuals clocks of both vehicles.

4.5 Summary

Simulations in a MATLAB/Simulink environment are done to prepare for the experiments with real vehicles. These are two Renault Twizy's which are automated as a part of the i-Cave project [9]. A more realistic vehicle model is implemented to make the simulations as accurate as possible. The more accurate vehicle model is compared to the the simplified vehicle model from Section 2.2. Also the time scaling is simulated on the more accurate vehicle model, included with input delays and noise. A disturbance is applied to the input of the lead vehicle while the trajectory planner still plans towards a final velocity.

Next these method is implemented on the real vehicles. At first a benchmark is done where the time scaling is inactive, in other words $\dot{\tau} = 1$ and $\ddot{\tau} = 0$ during the whole experiment. Next the same experiment is executed again but without the fixed values for $\dot{\tau}$ and $\ddot{\tau}$. This experiment has shown that a variable inter-vehicle gap size results in oscillations in the acceleration and velocity of the host vehicle. Therefore $r_{c,i}$ is substituted by a constant inter-vehicle distance $\underline{r}_i$. This has no consequence for the vehicle following objective because it is only used in the safety feature of time scaling.

Eventually the results of the executed experiments differ from the results of the simulations. The radar of the vehicle does not always work properly, with as a result high unwanted desired accelerations or decelerations. Also the time scaling method causes string unstable braking. Note that this is however not an issue, as time scaling is a safety feature, intended for preventing the vehicles from colliding. This is also the case with ACC equipped vehicles, in other words, communication fails causes also string unstable braking. This may be solved further in the string of vehicles by communicating $\tau(t)$, $\dot{\tau}(t)$ and $\ddot{\tau}(t)$ with following vehicles.

Chapter 5

Conclusions and recommendations

In this final chapter, the conclusions of the thesis are given as well as recommendations for future research.

5.1 Conclusions

Vehicles driving in a platoon using cooperative driving in this thesis aim for a certain following objective, which is defined by a spacing policy. To realise this spacing policy, each vehicle is equipped with a radar which measures the relative position and relative velocity of the vehicle. The vehicle model is expressed in simple state space model with a constant representing the driveline dynamics. The initial states of this model are needed for the trajectory planner. The trajectory planner plans trajectories with the use of B-splines. This results in cooperative string-stable trajectories for cooperative driving of automated vehicles. A trajectory is generated with a few parameters which are communicated between two vehicles. The required communication bandwidth is less then when communicating whole sampled trajectories. This will act as the foundation of the time scaling mechanism.

A vehicle will never exactly execute its desired trajectory due to disturbances. Disturbances can be compensated by the next plan cycle. Due to the relatively low update frequency of the trajectory planner or unreliable communicated information between vehicles, disturbances may occur in between plan cycles. The goal is to develop a method to overcome these disturbances, without leaving the desired path. The combination of the x and y coordinates stay the same while adapting the desired velocity of a vehicle. A time scaling method is designed that compensates for these disturbances. Time scaling will ensure that the host vehicle decelerates, even when its current trajectory states otherwise. The scaled time, to read out the table containing sampled-time values, does not progress faster then real time. As a result, the scaled velocity will always be smaller then the velocity obtained by the trajectory planner. The scaled acceleration can be determined by numerical differentiation or in an analytical way. By means of simulation, these two methods are compared. In the simulations, realistic measurement noise is added in the simulation for fair comparison. Concluded from this is that determining the scaled acceleration analytical gives lower tracking error values then when the acceleration is determined by numerical differentiation. Implementing noise on the vehicle states, resulting in unwanted desired accelerations when approaching standstill conditions. An upper bound is defined for the second derivative of the scaled time which ensures the acceleration to remain lower than the desired acceleration coming from the tra-

jectory planner. This corresponds to the retarding nature of the time scaling mechanism, which is uses as a safety measure.The time scaling mechanism is successfully implemented in the MATLAB/Simulink environment.

A more accurate vehicle model is made before executing experiments on real vehicles. The time scaling method is simulated with this extended vehicle model. The host vehicle deviates from the planned velocity, while the trajectory planner still plans to a predefined final velocity. After performing the first set of experiments, it was concluded that a variable stand-still distance $r_{r,i}(t)$ (to determine the velocity $v_{c,i}(t)$) in between the lead vehicle and the host vehicle results in unwanted oscillations in the velocity and acceleration of the host vehicle, therefore a fixed stand-still distance is used in the experiments. After the experiments, where a deceleration by disturbance on the lead vehicle activates the time scaling method, it is concluded that the braking by time scaling is not string stable, which is also the case at ACC equipped vehicles where also only radar information is used. Note that this is not an issue, as time scaling is a safety feature, intended for preventing the vehicles from colliding. In conclusion, the time scaling mechanism proved to be a promising solution for deployment on automated connected vehicles using trajectory planning. Dangerous situations, resulting from V2V impairment or failure of the lead vehicle, can be overcome by using time scaling.

5.2 Recommendations

The following recommendations are given for future research purposes:

- The lateral component can be included in the trajectory planner. Since the time scaling method adapts the velocity and acceleration without changing the X and Y coordinate combinations of the prescribed trajectory, such that the same spacing path will be followed. The lateral component will not affect the time scaling mechanism.
- The drop-out of wireless communication can be included in the experiments. Whenever a host vehicle does not get new trajectories due to communication failure, the host vehicle will execute its latest update of the planned trajectory. This can be an important time scaling application where it is needed.
- When a string with more than two vehicle is considered, the scaled time $\tau(t)$, its time derivative $\dot{\tau}(t)$ and its second time derivative $\ddot{\tau}(t)$ can be communicated by wireless communication. This may result in string stable behaviour while decelerating by disturbance. The following vehicles will have as much information available as is with CACC equipped vehicles. The first vehicle that initiates the time scaling will never act string stable while decelerating by the time scaling mechanism, but all following vehicles probably will.
- The radar selects sometimes the wrong object, which can result in a departure delay of the host vehicle. Therefore the way of selecting the most important object can be optimised or combining the radar with vision data. The radar even does not detect any object at all which results in unreliable information for the time scaling mechanism. This can be solved by optimising the whole target tracking of the radar.

Bibliography

[1] S. Baaij. Development and validation of a multibody model of a Renault Twizy. Technical Report August, 2019.

[2] Mohammad Bahram, Anton Wolf, Michael Aeberhard, and Dirk Wollherr. A prediction-based reactive driving strategy for highly automated driving function on freeways. *IEEE Intelligent Vehicles Symposium, Proceedings*, (Iv):400–406, 2014.

[3] Mohamed L. El-Sayed, P. S. Krishnaprasad, and P. S. Krishnaprasad. Homogeneous Interconnected Systems: An Example. *IEEE Transactions on Automatic Control*, 26(4):894–901, 1981.

[4] Thor I Fossen and Kristin Y Pettersen. From Cooperative to Autonomous Vehicles. In *Sensing and Control for Autonomous Vehicles*, volume 474, pages 435–452. 2017.

[5] Timothy J Graettinger and Bruce H Krogh. Evaluation and Time-Scaling of Trajectories for Wheeled Mobile Robots. *IEEE American Control Conference*, pages 511–516, 1988.

[6] R B A Van Hoek, J Ploeg, and H Nijmeijer. Motion Planning for Automated Connected Vehicles. *Proceedings of the 14th International Symposium on Advanced Vehicle Control (AVEC' 18)*, pages 3–8, 2018.

[7] Robbin Van Hoek, Jeroen Ploeg, and Henk Nijmeijer. Cooperative Driving of Automated Vehicles using B-splines for Trajectory Planning. X(X):1–11, 2019.

[8] Robbin Van Hoek, Jeroen Ploeg, and Henk Nijmeijer. Gap Closing for Cooperative Driving in Automated Vehicles using B-splines for Trajectory Planning. *Accepeted for publication for Intelligent Vehicles Symposium 2020*, 2020.

[9] Frans Hoogeboom. *Safety of Automated Vehicles: Design, Implementation, and Analysis.* Phd. dissertation, Eindhoven University of Technology, 2020.

[10] Junxiang Li, Bin Dai, Xiaohui Li, Chao Li, and Yi Di. A real-Time and predictive trajectory-generation motion planner for autonomous ground vehicles. *Proceedings - 9th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2017*, 2(1):108–113, 2017.

[11] Xiaohui Li, Zhenping Sun, Dongpu Cao, Daxue Liu, and Hangen He. Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles. *Mechanical Systems and Signal Processing*, 87:118–137, 2017.

[12] Xiaohui Li, Zhenping Sun, Zhen He, Qi Zhu, and Daxue Liu. A practical trajectory planning framework for autonomous ground vehicles driving in urban environments. *IEEE Intelligent Vehicles Symposium, Proceedings*, 2015-Augus(June):1160–1166, 2015.

[13] Xiaohui Li, Zhenping Sun, Arda Kurt, and Qi Zhu. A sampling-based local trajectory planner for autonomous driving along a reference path. *IEEE Intelligent Vehicles Symposium, Proceedings*, (Iv):376–381, 2014.

[14] Peter Martin. Autonomous intelligent cruise control incorporating automatic braking. *SAE Technical Papers*, 4(4), 1993.

[15] G Naus, R Vugts, J Ploeg, R van de Molengraft, and M Steinbuch. Cooperative adaptive cruise control, design and experiments. (1):6145–6150, 2010.

[16] Gerrit J.L. Naus, René P.A. Vugts, Jeroen Ploeg, Marinus J.G. Van De Molengraft, and Maarten Steinbuch. String-stable CACC design and experimental validation: A frequency-domain approach. *IEEE Transactions on Vehicular Technology*, 59(9):4268–4279, 2010.

[17] Jeroen Ploeg, Bart T.M. Scheepers, Ellen Van Nunen, Nathan Van De Wouw, and Henk Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 260–265, 2011.

[18] Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2014.

[19] Xiangjun Qian, Arnaud De La Fortelle, and Fabien Moutarde. A hierarchical Model Predictive Control framework for on-road formation control of autonomous vehicles. *IEEE Intelligent Vehicles Symposium, Proceedings*, 2016-Augus(Iv):376–381, 2016.

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

[21] D. Swaroop and J. K. Hedrick. Constant spacing strategies for platooning in automated highway systems. *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, 121(3):462–470, 1999.

[22] Azim Vahidi, Ardalan and Eskandarian. Research Advances in Intelligent Collision Avoidance and Adaptive Cruise Control. *IEEE Intelligent Transportation Systems Magazine*, 4(3):143–153, 2003.

[23] B Van Arem, J G Van Driel, and R Visser. The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow Characteristics MIXIC Microscopic simulation of Intelligent Cruise Control View project Taking the Fast Lane View project The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow C. *Ieee Transactions on Intelligent Transportation Systems*, 7(4):429, 2006.

[24] Ellen Van Nunen, Jan Verhaegh, Emilia Silvas, Elham Semsar-Kazerooni, and Nathan Van De Wouw. Robust model predictive cooperative adaptive cruise control subject to

V2V impairments. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2018-March:1–8, 2018.

[25] Moritz Werling, Sören Kammel, Julius Ziegler, and Lutz Gröll. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds. *International Journal of Robotics Research*, 31(3):346–359, 2012.

Appendix A

Simulated noise signals

The simulations are done with a a sample rate of $f_s = 100$ Hz whereas the trajectory planner updates at $f_p = 5$ Hz. The radar noise is sampled with a frequency of $f_{\text{RadarNoise}} = 10$ Hz.

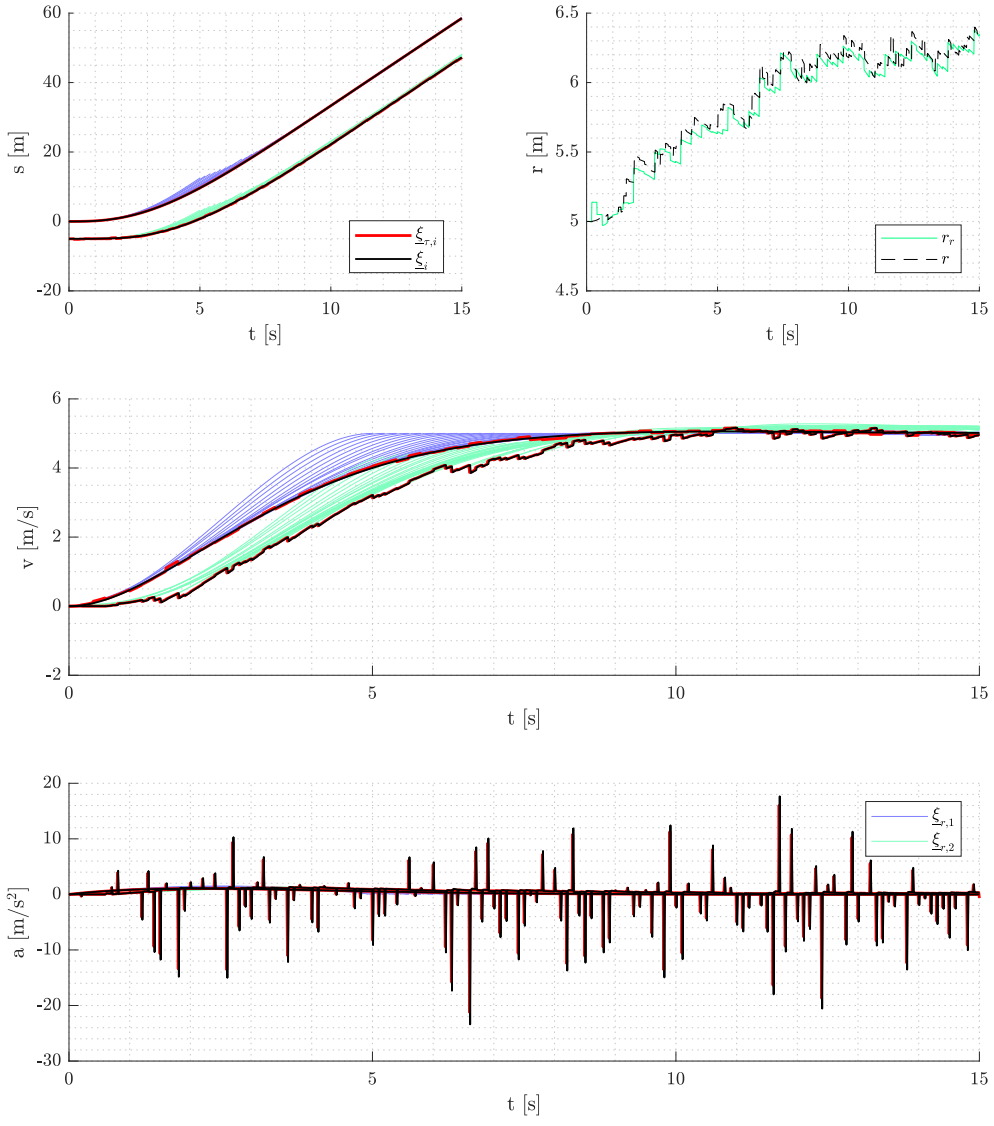


Figure A.1: A velocity change with added noise obtained from real vehicle data. Option 1

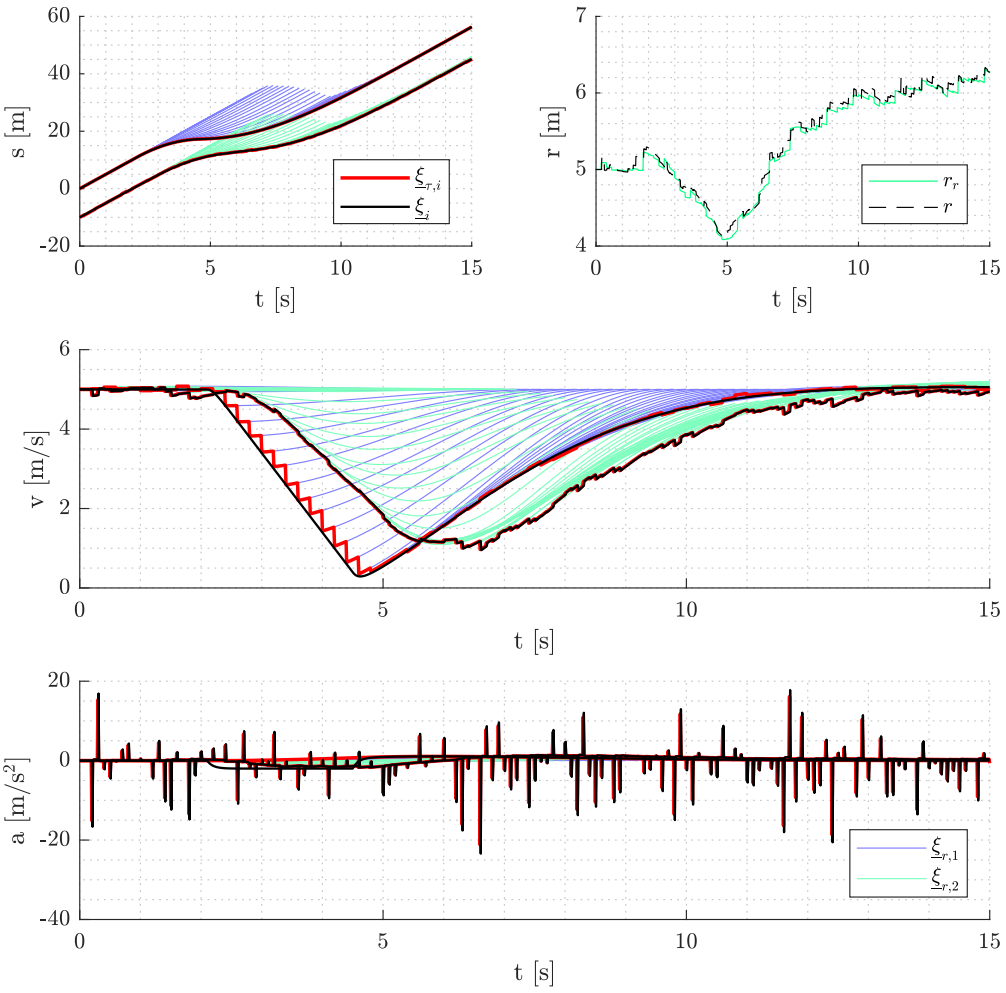


Figure A.2: Suddenly braking of the first vehicle with added noise obtained from real vehicle data. Option 1

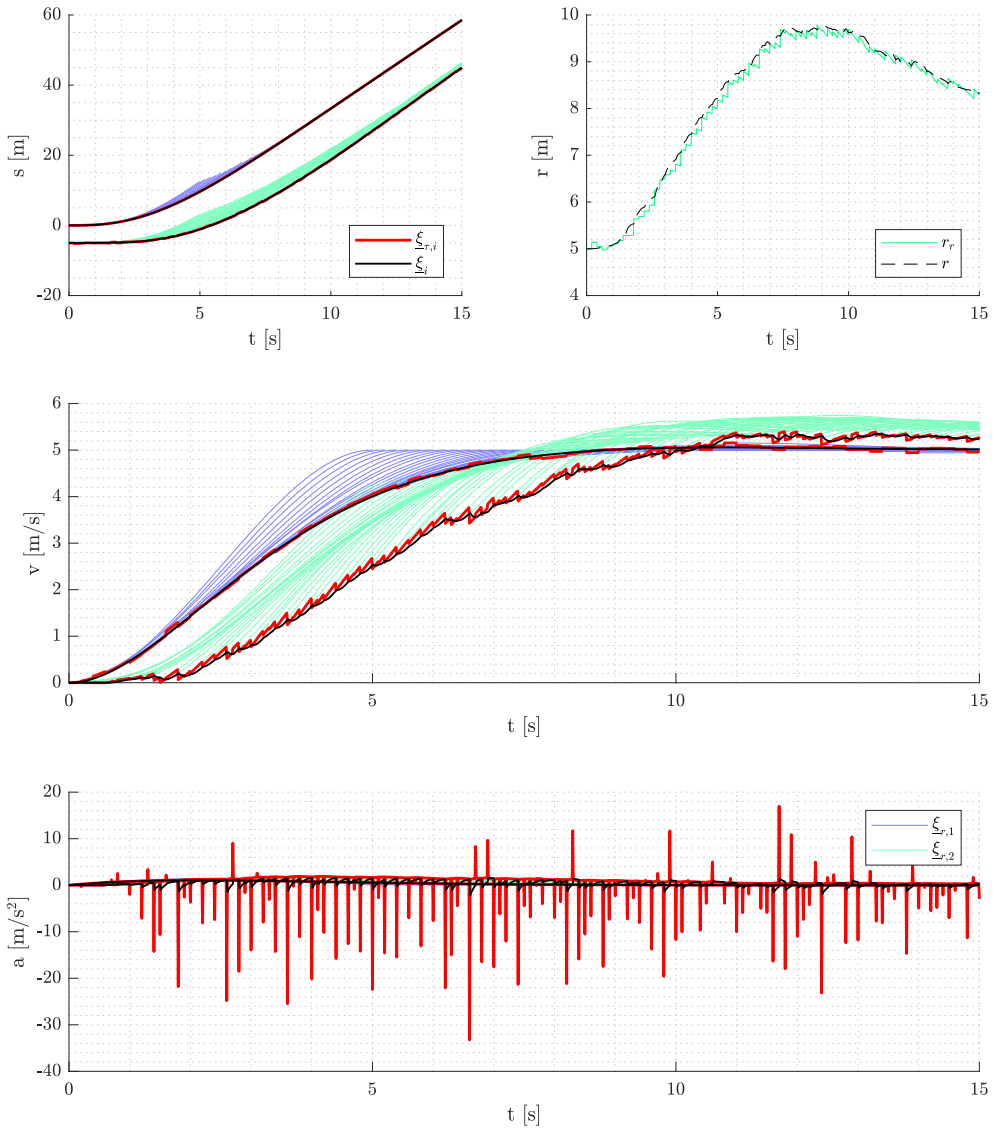


Figure A.3: A velocity change with added noise obtained from real vehicle data. Option 2

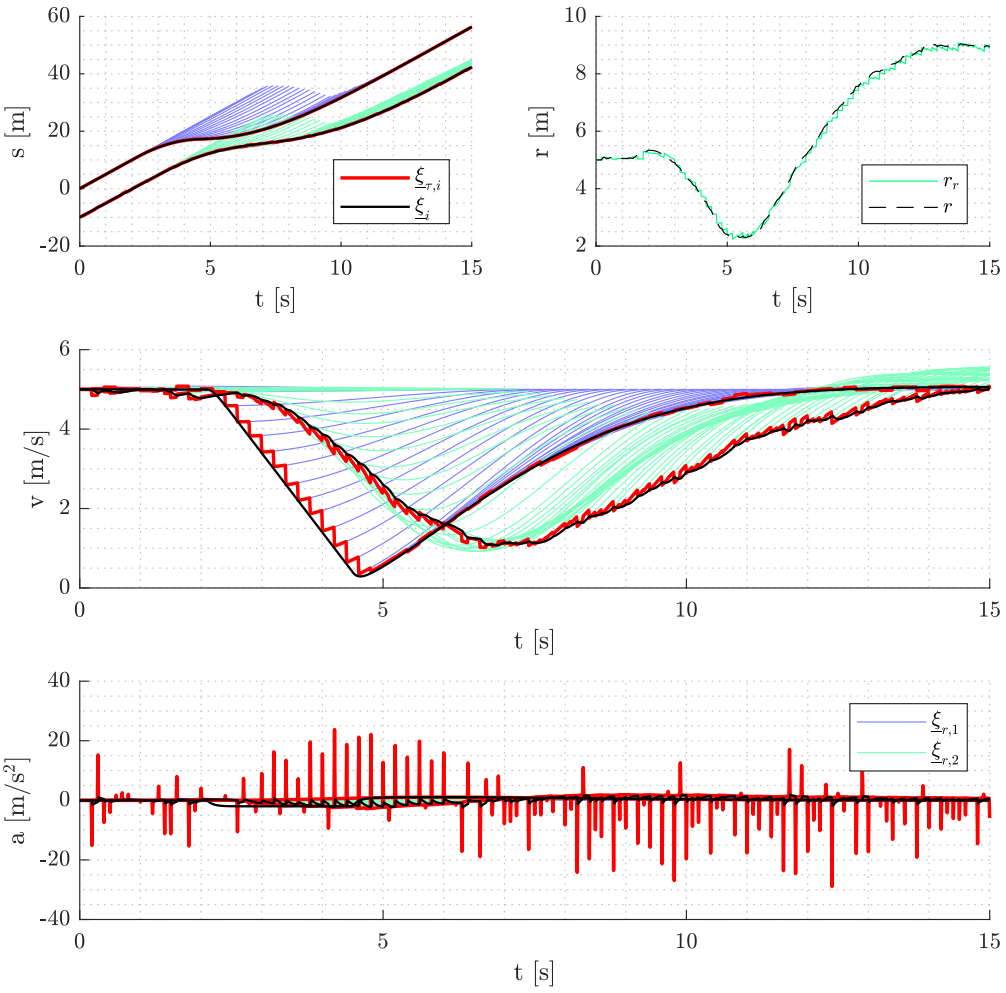


Figure A.4: Suddenly braking of the first vehicle with added noise obtained from real vehicle data. Option 2

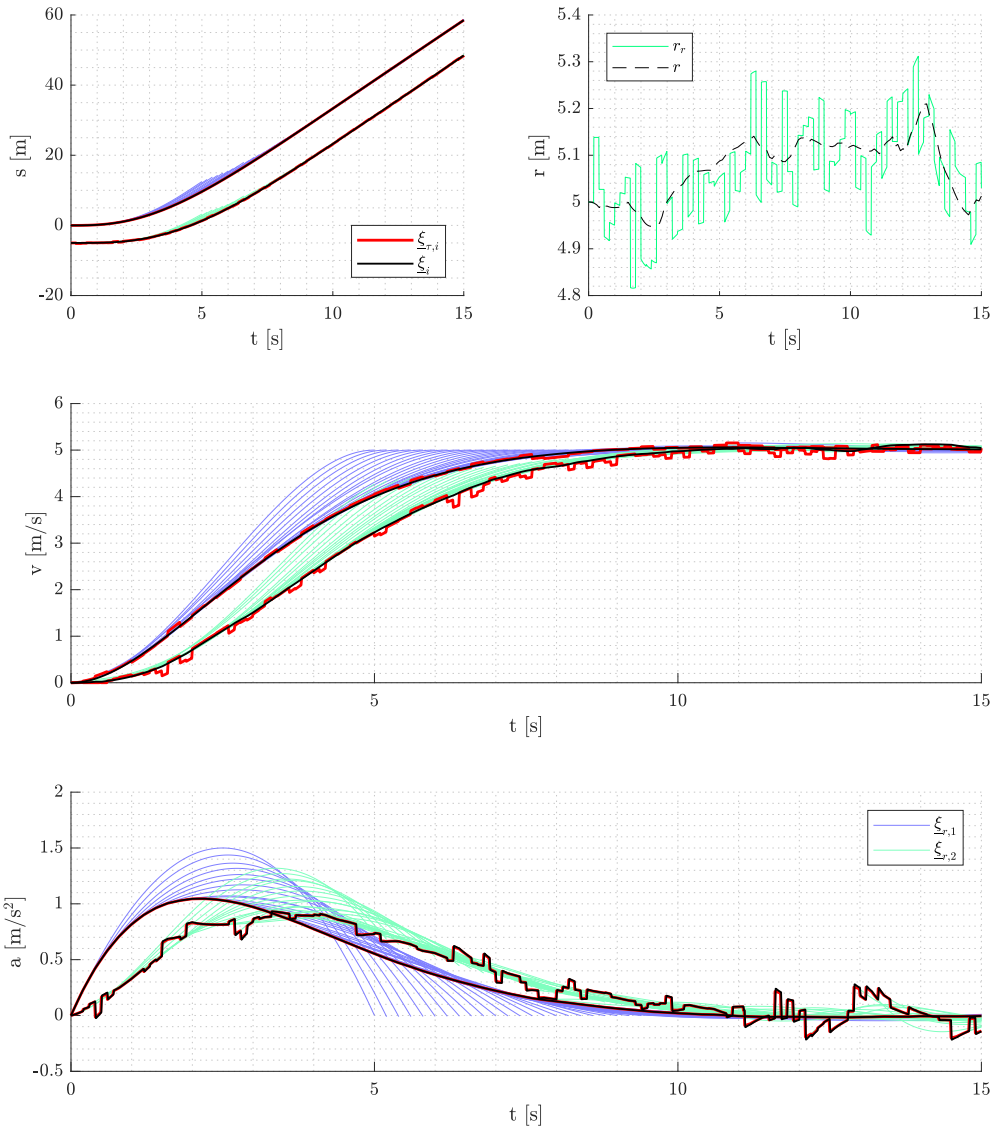


Figure A.5: A velocity change with added noise obtained from real vehicle data. Option 3

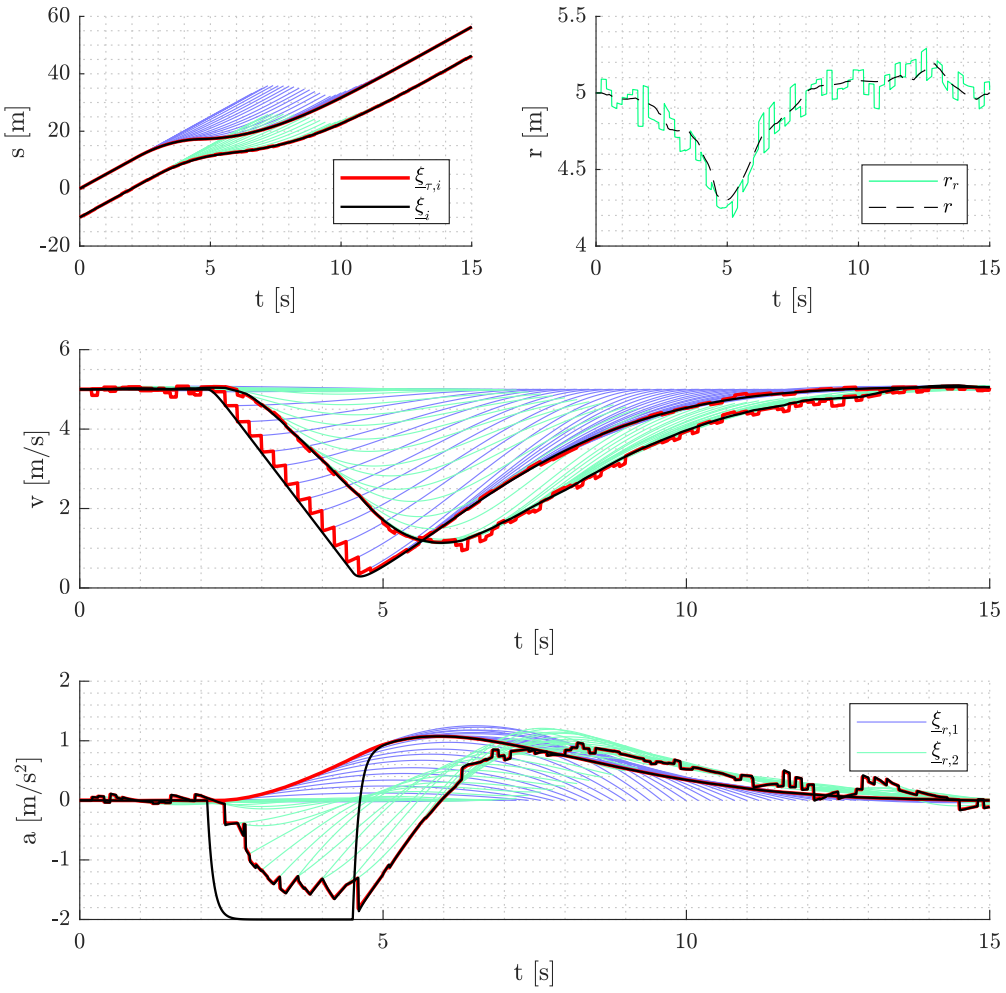


Figure A.6: Suddenly braking of the first vehicle with added noise obtained from real vehicle data. Option 3

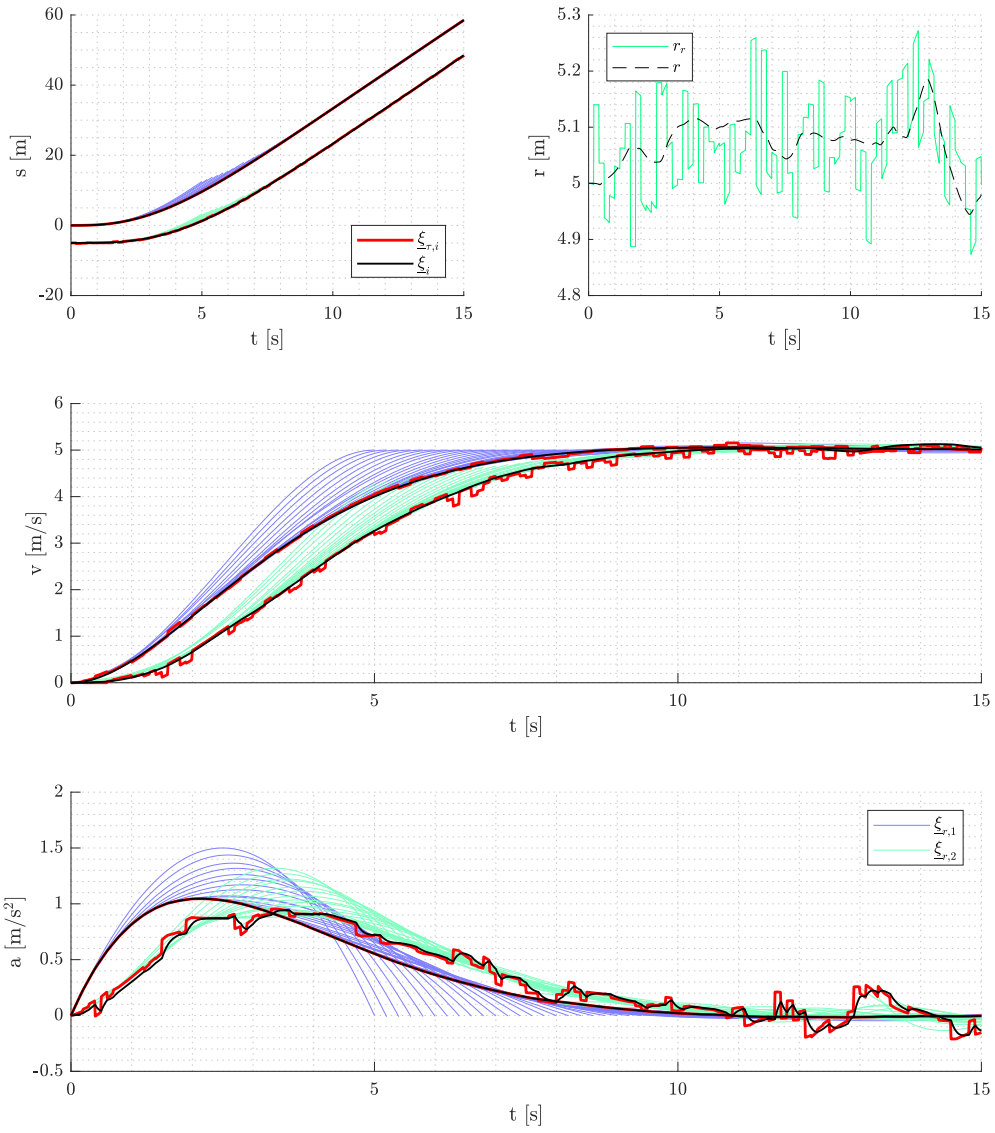


Figure A.7: A velocity change with added noise obtained from real vehicle data. Option 4

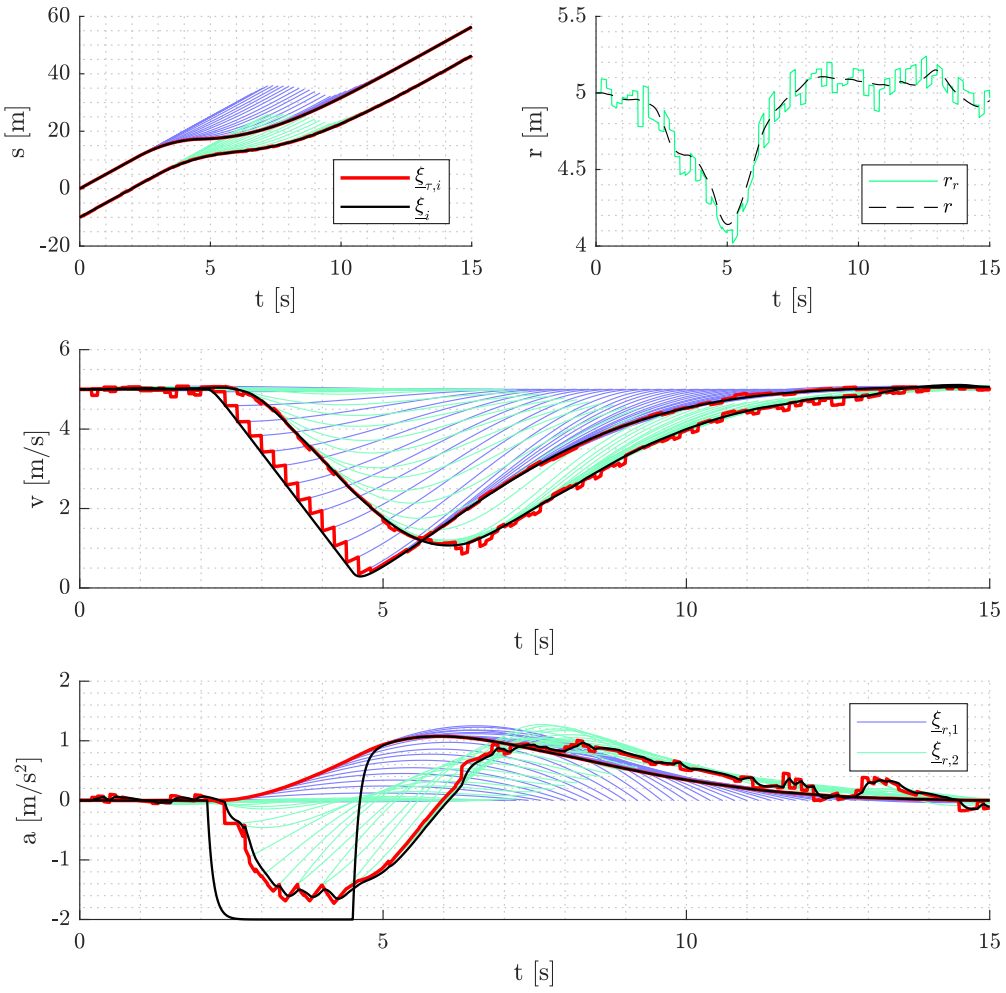


Figure A.8: Suddenly braking of the first vehicle with added noise obtained from real vehicle data. Option 4

Appendix B

Schematic representation of the physical vehicle model

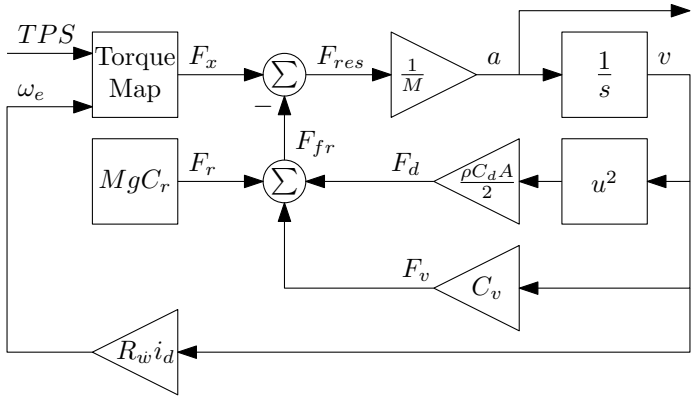


Figure B.1: Schematic representation of the physical vehicle model.

Technische Universiteit
Eindhoven
University of Technology

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

01-05-2020

Name

Karsten Hendrikus

ID-number

1033665

Signature

K.Hendrikus

Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU