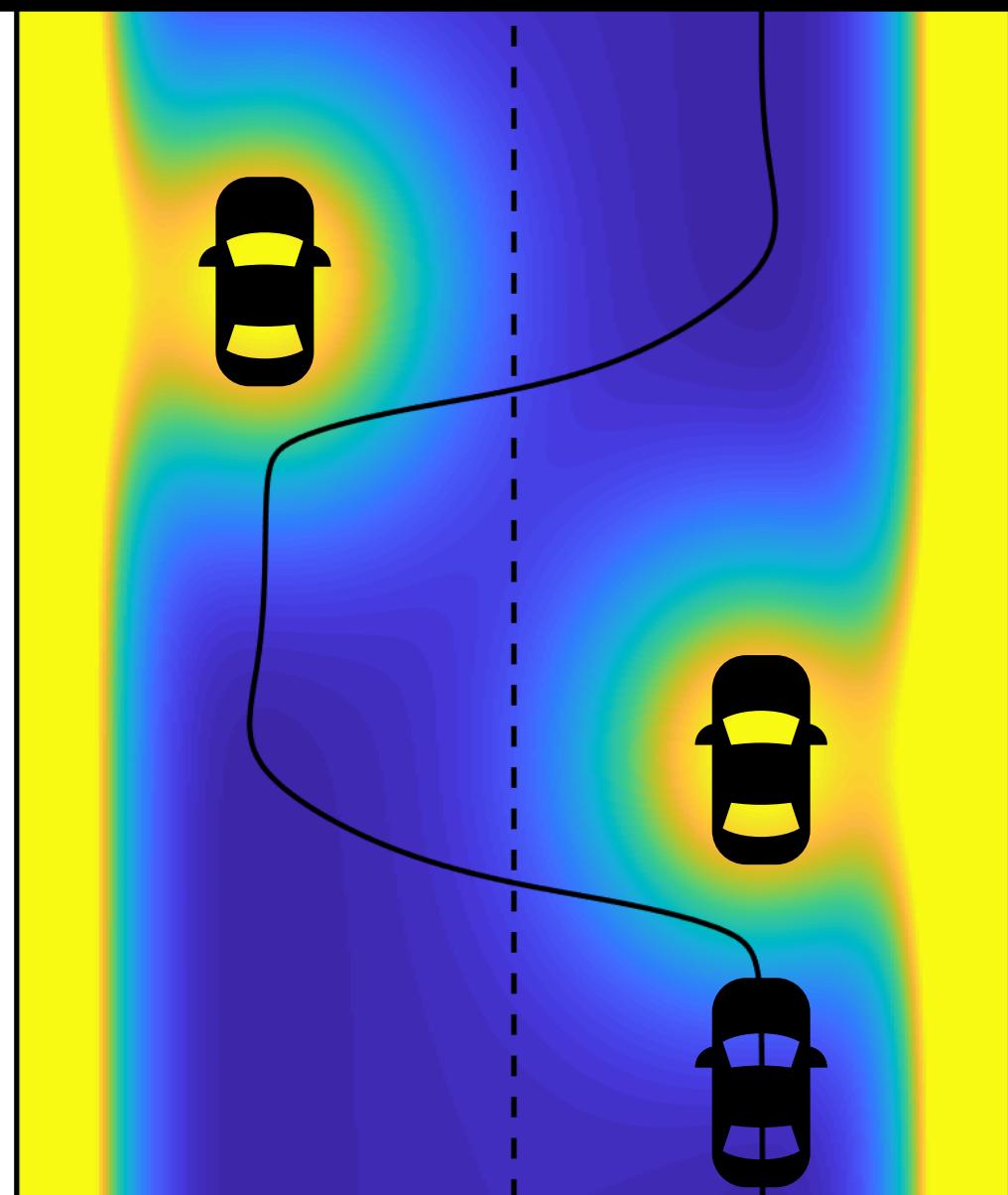


Model-based Path Planning and Control for Autonomous Vehicles using Artificial Potential Fields

E.Y. Snapper



Model-based Path Planning and Control for Autonomous Vehicles using Artificial Potential Fields

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Systems and Control at Delft
University of Technology

E.Y. Snapper

January 10, 2018

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of
Technology



The work in this thesis was supported by the Nederlandse organisatie voor Toegepast Natuurwetenschappelijk Onderzoek (TNO). Their cooperation is hereby gratefully acknowledged.



Copyright © Delft Center for Systems and Control (DCSC)
All rights reserved.



DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
DELFT CENTER FOR SYSTEMS AND CONTROL (DCSC)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

MODEL-BASED PATH PLANNING AND CONTROL FOR AUTONOMOUS VEHICLES
USING ARTIFICIAL POTENTIAL FIELDS

by

E.Y. SNAPPER

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE SYSTEMS AND CONTROL

Dated: January 10, 2018

Supervisor(s): _____
prof.dr.ir. J. Hellendoorn

dr.ir. M. Alirezaei

dr.ir. E. Semsar-Kazerooni

Reader(s): _____
First Reader

Second Reader

Abstract

This report presents the results of the graduation thesis from the TU Delft, performed at the Integrated Vehicle Safety (IVS) department of TNO. The goal of this graduation project is to develop a unified path-planning and -tracking method for autonomous vehicles in highway-driving scenarios, by making use of Artificial Potential Fields (APFs).

Most autonomous vehicles base their navigation control on first planning a path, which is then tracked by using a combination of feedback and feedforward control. The strength of using APFs is the possibility to integrate the path-planning and -tracking process. This concept has already been extensively used in the field of robotics. The attractive and repulsive forces coming from the APF guide the robot towards the final goal while avoiding obstacles. This offers an intuitive way to represent the level of hazard experienced in the direct environment. However, considerably less research has been devoted to the application of APFs in the field of autonomous vehicles. Furthermore, the available research mostly treats the vehicle as a particle, thereby leaving out the more complicated vehicle dynamics.

Therefore, this research is aimed at including the vehicle dynamics into the path-planning process such as to generate feasible and desirable paths. A Model Predictive Control (MPC) framework is proposed to fulfill this task. The adopted vehicle model is given by the linear bicycle model, which represents the vehicle dynamics sufficiently well for highway applications. The two main maneuvers of lane keeping and lane changing are executed with the aid of two different potential fields that were designed for these specific purposes: the road and obstacle APF, respectively. A second order Taylor approximation is used to incorporate the APFs into the quadratic MPC cost function. The Simulink model from TNO to simulate the controlled vehicle is modified by extending it to curved roads and by including the developed APF MPC controller. The resulting algorithm is capable of following curved highway lanes and overtaking slower vehicles at different velocities in the simulation environment. The results are compared with the previously developed path planner and lateral controller from TNO. In order to also deliver longitudinal control action, the model can be connected to the Adaptive Cruise Control (ACC) application of TNO.

It is concluded that the suggested method potentially offers a powerful solution to the navigation control of autonomous vehicles. Real-life experiments have to be done in the future to validate the performance of the controller.

Table of Contents

Preface	xi
1 Introduction	1
1-1 Problem statement	3
1-2 Literature study	5
1-3 Thesis outline	6
2 Autonomous driving	7
2-1 Autonomous vehicle system	7
2-2 Driver behavior	10
2-3 Highway environment	10
2-4 Sensors	11
2-5 Vehicle maneuvers	14
2-6 Collision avoidance	15
3 Controller design	17
3-1 Model predictive control	17
3-2 Vehicle model	19
3-3 Potential field functions	23
3-3-1 Road potential field	27
3-3-2 Obstacle potential field	33
3-3-3 Artificial potential field	37
3-4 Combining APF with MPC	39
4 Simulations	49
4-1 TNO controller	49
4-2 Quadratic programming	52
4-3 Weight tuning	54
4-4 Results	57
4-5 Experimental considerations	63

5 Conclusions	65
5-1 Recommendations	66
A Bicycle model derivation	69
B Quadratic Taylor approximation	73
C MPC matrix derivation	79
D Simulink model	85
E Tuned obstacle PF parameters	91
F Lane-change characteristics results	99
Bibliography	111
Glossary	117
List of Acronyms	117
List of Symbols	118
Index	125

List of Figures

1-1	The different domains that come together in autonomous vehicles	2
1-2	Highway platooning for CACC applications	3
2-1	The complex architecture of an autonomous vehicle system	9
2-2	Sensor measurements of the environment: 1) lateral lane-center distance, 2) lane heading, 3) lane curvature, 4) longitudinal obstacle distance, 5) lateral obstacle distance, 6) longitudinal velocity difference	12
2-3	Flow chart of vehicle maneuvers for highway driving	15
2-4	Two options for collision avoidance: 1) comfortable lane change, 2) normal braking	16
3-1	Simplified block diagram of autonomous vehicle system with APF MPC controller	18
3-2	Transformation from car to bicycle model	20
3-3	Different coordinate frames used in simulation	20
3-4	Free-body diagram of the bicycle model	21
3-5	Spring and damper analogy for the APF method	23
3-6	Morse PF U_M , with $k_1 = 0.3$, $k_2 = 0.042$, and $k_3 = 15.5$	26
3-7	Gradient of Morse PF U_M , with $k_1 = 0.3$, $k_2 = 0.042$, and $k_3 = 15.5$	26
3-8	Right and left lane potential fields $U_{r\ell,sr}$ and $U_{\ell\ell,sr}$ for a straight road, with $A_r = 0.5$ and $b_r = 1$	28
3-9	Straight road potential field $U_{r,sr}$, with $A_r = 0.5$ and $b_r = 1$	29
3-10	Curved road PF computation	30
3-11	Curved road potential field $U_{r,cr}$, with $A_r = 0.5$, $b_r = 1$, and $c_2 = 5e-6 \text{ m}^1$. . .	32
3-12	Obstacle projection to lane center	34
3-13	Obstacle potential field U_o , with $A_o = 1$, $(X_o, Y_o) = (375, 0)$ m, $x_\sigma = 375$ m, and $\sigma_y = 0.865$ m	36
3-14	Total artificial potential field of curved road with two obstacles, with $\lambda_r = 1$, $A_r = 0.5$, $b_r = 1$, $c_2 = 5e-6 \text{ m}^{-1}$, $\lambda_o = 25$, $A_o = 1$, $(X_{o,1}, Y_{o,1}) = (375, 0.7031)$ m, $(X_{o,2}, Y_{o,2}) = (775, 6.5031)$ m, $x_\sigma = 375$ m, and $\sigma_y = 0.865$ m	38

4-1	Graphical representation of tracking error definitions used by TNO	50
4-2	TTC-instances for tuning the host vehicle lane change, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$	55
4-3	Angled view of total APF with obstacle PF blocking the left lane, with $V_h = 120 \text{ km/h}$, $\Delta V_{o,h} = -20 \text{ km/h}$, $c_2 = -0.00025 \text{ m}^{-1}$, $x_\sigma = 130 \text{ m}$, and $\sigma_y = 0.865 \text{ m}$	56
4-4	Characteristics to evaluate the lane-change performance: 0) lane-change initiation, 1) rise time: t_r , 2) lane-change duration: t_d , 3) settling time: t_{se} , 4) longitudinal IVD: d_{lo} , 5) lateral IVD: d_{la} , 6) overshoot: d_o , 7) lateral acceleration a_y , 8) lateral jerk j_y	58
4-5	Time characteristics of lane-changing host vehicle, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$	59
4-6	Less challenging lane-change maneuvers towards the inside of the curve	60
4-7	More challenging lane-change maneuvers towards the outside of the curve	60
4-8	Host and obstacle vehicle path results, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$	61
4-9	Lateral acceleration and jerk for $c_2 = 5e-6 \text{ m}^{-1}$, $V_h = 120 \text{ kmh}^{-1}$, $V_o = 100 \text{ kmh}^{-1}$	61
4-10	Host vehicle states for $c_2 = 5e-6 \text{ m}^{-1}$, $V_h = 120 \text{ kmh}^{-1}$, $V_o = 100 \text{ kmh}^{-1}$	62
A-1	Bicycle model configuration expressed in global road coordinate frame	69
A-2	Free-body diagram of the bicycle model	69
B-1	Lateral QAs of straight road PF $U_{r,sr}$	75
B-2	Longitudinal QA of curved road PF $U_{r,cr}$	76
B-3	Lateral QA of obstacle PF U_o	77
B-4	Longitudinal QA of obstacle PF U_o	78
D-1	Simulink model	85
D-2	World model	86
D-3	Host vehicle model	87
D-4	APF MPC control system	88
D-5	TNO control system	89
D-6	APF MPC control system details	90
E-1	Obstacle PF weight λ_0 versus V_h for different $\Delta V_{o,h}$ for all c_2	91
E-2	Obstacle PF σ_x -measure x_σ versus V_h for different $\Delta V_{o,h}$ with $c_2 \geq 0$	93
E-3	Obstacle PF σ_x -measure x_σ versus V_h for different $\Delta V_{o,h}$ with $c_2 \leq 0$	94
F-1	Characteristics to evaluate the lane-change performance: 0) lane-change initiation, 1) rise time: t_r , 2) duration: t_d , 3) settling time: t_{se} , 4) longitudinal IVD: d_{lo} , 5) lateral IVD: d_{la} , 6) overshoot: d_o	99
F-2	Lane-change characteristics box plots for $c_2 = 0\text{m}^{-1}$	100

F-3	Lane-change characteristics box plots for $c_2 = 1.0\text{e}{-6} \text{ m}^{-1}$	101
F-4	Lane-change characteristics box plots for $c_2 = 5.0\text{e}{-6} \text{ m}^{-1}$	102
F-5	Lane-change characteristics box plots for $c_2 = 1.0\text{e}{-5} \text{ m}^{-1}$	103
F-6	Lane-change characteristics box plots for $c_2 = 5.0\text{e}{-5} \text{ m}^{-1}$	103
F-7	Lane-change characteristics box plots for $c_2 = 1.0\text{e}{-4} \text{ m}^{-1}$	104
F-8	Lane-change characteristics box plots for $c_2 = 2.5\text{e}{-4} \text{ m}^{-1}$	105
F-9	Lane-change characteristics box plots for $c_2 = -1.0\text{e}{-6} \text{ m}^{-1}$	106
F-10	Lane-change characteristics box plots for $c_2 = -5.0\text{e}{-6} \text{ m}^{-1}$	106
F-11	Lane-change characteristics box plots for $c_2 = -1.0\text{e}{-5} \text{ m}^{-1}$	107
F-12	Lane-change characteristics box plots for $c_2 = -5.0\text{e}{-5} \text{ m}^{-1}$	108
F-13	Lane-change characteristics box plots for $c_2 = -1.0\text{e}{-4} \text{ m}^{-1}$	109
F-14	Lane-change characteristics box plots for $c_2 = -2.5\text{e}{-4} \text{ m}^{-1}$	109

List of Tables

1-1	The levels of autonomy outlined by SAE	2
2-1	Minimum horizontal lane curve radii in [m] for different design speeds and cants .	11
2-2	Symbols from the Mobileye sensor measurements	13
2-3	Symbols from the lateral lane center position equation	14
2-4	Symbols from the obstacle information calculations	14
3-1	Symbols from the straight road PF function	30
3-2	Symbols from the curved road PF function	33
3-3	Symbols from the obstacle PF function	35
3-4	Symbols from the obstacle PF rotation coefficients	37
3-5	Parameters of the host vehicle	42
3-6	Symbols from the cost function	44
4-1	Mean lane-change characteristics of APF MPC and TNO controller	59
E-1	λ_o in [-] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for all c_2	92
E-2	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 0 \text{ m}^{-1}$	92
E-3	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0e-6$ and $5.0e-6 \text{ m}^{-1}$.	95
E-4	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0e-5$ and $5.0e-5 \text{ m}^{-1}$.	95
E-5	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0e-4$ and $2.5e-4 \text{ m}^{-1}$.	96
E-6	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0e-6$ and $-5.0e-6 \text{ m}^{-1}$.	96
E-7	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0e-5$ and $-5.0e-5 \text{ m}^{-1}$.	97
E-8	x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0e-4$ and $-2.5e-4 \text{ m}^{-1}$.	97
F-1	Mean lane-change characteristics for $c_2 = 0 \text{ m}^{-1}$	100

F-2	Mean lane-change characteristics for $c_2 = 1.0e-6$ and $5.0e-6 \text{ m}^{-1}$	101
F-3	Mean lane-change characteristics for $c_2 = 1.0e-5$ and $5.0e-5 \text{ m}^{-1}$	102
F-4	Mean lane-change characteristics for $c_2 = 1.0e-4$ and $2.5e-4 \text{ m}^{-1}$	104
F-5	Mean lane-change characteristics for $c_2 = -1.0e-6$ and $-5.0e-6 \text{ m}^{-1}$	105
F-6	Mean lane-change characteristics for $c_2 = -1.0e-5$ and $-5.0e-5 \text{ m}^{-1}$	107
F-7	Mean lane-change characteristics for $c_2 = -1.0e-4$ and $-2.5e-4 \text{ m}^{-1}$	108

Preface

This is the Master of Science graduation thesis from the Delft Center for Systems and Control (DCSC) department within the faculty of Mechanical, Maritime and Materials Engineering (3mE) at Delft University of Technology. The topic of this research, namely using Artificial Potential Fields (APFs) for path planning and control of autonomous vehicles, is of interest to the Integrated Vehicle Safety (IVS) department of TNO in Helmond.

This research was performed within the Cooperative Control Systems (CCS) group of TNO, under supervision of dr. ir. Mohsen Alirezaei and dr. ir. Elham Semsar-Kazerooni, to whom I want to express my enormous gratitude. There were some difficult times where you showed great understanding and gave me the space I needed, which I really appreciated. Thank you for your time and guidance during our weekly progress meetings. Your thoughts and advice have helped a lot to move forward.

I look back at my time at TNO with good memories, even though traveling back and forth between Delft and Helmond could sometimes be quite exhausting. However, it gave me the opportunity to read many books in the estimated number of 260 train journeys. Luckily, I was able to stay in hotels and explore the area of Eindhoven. Fun fact: I have tried eight different hotels, slept 39 times in either Eindhoven or Helmond, and went to see the movies fourteen times on these occasions.

I would also like to thank my TUD supervisor, prof. dr. ir. Hans Hellendoorn, for his monthly coaching sessions (as I liked to call it) where we had some good talks. Although we did not discuss my work in much detail, I always left your office with new motivation and ideas. Also thank you for your feedback and review of my final thesis.

Of course I want to acknowledge my parents, who have supported me in every possible way, for which I remain forever thankful. I could not have done this without you. Furthermore, I want to mention my fantastic friends, who were there when I needed them, for distraction and doing fun things together. Lastly, special thanks go out to my girlfriend. Graduation could be a very though ride for both of us, with many doubts and uncertainties. I am glad to have shared this experience with you. You were my rock in all of this. Furthermore, I will not forget to mention my girlfriend's help with the graphic design of the illustrations in this report. This future control engineer knows nothing about InDesign and EPS-figures.

Delft, University of Technology
January 10, 2018

E.Y. Snapper

Chapter 1

Introduction

The number of vehicles has been steadily increasing in the last few decades. In the Netherlands, this number increased from over 6.3 million cars in 2000 to almost 8 million in 2005 [1]. This increase poses severe challenges for transportation systems. Traffic delays and congestions, for example, increase travel time and therefore impact society and economy negatively. This has raised the interest in developing intelligent traffic solutions, like automated platoons, where vehicles are driving in platoon form, with small inter-vehicle time gaps of well below one second [2]. Fully autonomous driving makes this possible, since it can greatly improve on the human reaction time. This will significantly increase the road capacity of existing roads and reduce the fuel consumption, due to lower aerodynamic drag. Moreover, autonomous driving also promises to improve transportation in terms of traffic throughput, passenger comfort, and road safety.

The interest of both the academic and commercial industry in the field of autonomous vehicles has grown increasingly in the past couple of years because of the many promising benefits for society [3]. Another reason for it being so popular is that many research domains come together in autonomous vehicles, like robotics, computer science, and engineering. The illustration from Figure 1-1 shows the knowledge from different backgrounds that is combined in the autonomous vehicle design. Many open issues remain to be researched, like environment perception, modeling, localization, map building, path planning, decision making, and motion control [4]. Software is the key driving factor in all of this, but social acceptance also plays a critical role.

The increased interest in autonomous driving has led to the rapid development of new technologies and big improvements on already existing technologies. The first successful implementation of the autonomous vehicle dates back to the 80's, by the pioneering Carnegie Mellon University [5]. Nowadays, many car manufacturers, like Tesla, Audi, Mercedes, Renault, and Fiat, have already developed their own intelligent vehicle solutions [6]. However, it is hard to determine the current state-of-the-art, since car manufacturers generally do not publicly disclose any details on their scientific advances. Nevertheless, the expectation is that autonomous vehicles will be the most viable means of transportation by 2040 [7].

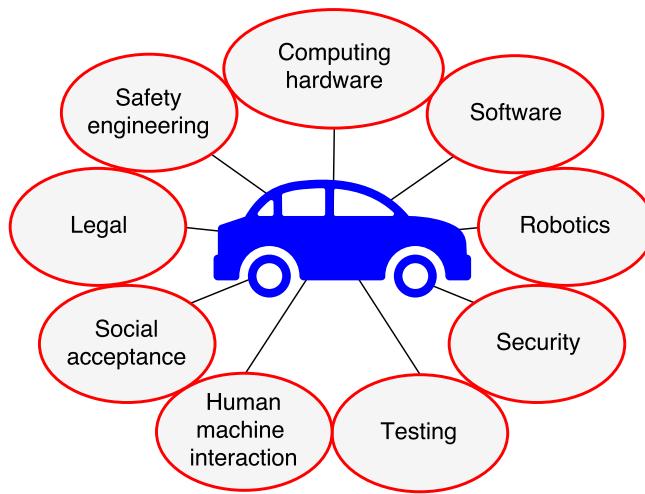


Figure 1-1: The different domains that come together in autonomous vehicles

Passive safety solutions such as crumple zones, seat belts, and airbags have been standard vehicle features to reduce the number of fatalities [8]. Modern vehicles are equipped with sensing, computational and actuating capabilities, which allows for the design of active safety features, called Advanced Driver Assistance Systems (ADAS), such as Adaptive Cruise Control (ACC) and lane-keeping systems [9]. ADAS roadmaps show that the number of assistance functionalities will increase steadily in the near future [10]. Some of these systems, like Electronic Stability Control (ESC) are already mandatory in the EU since 2011 [11]. To indicate the level of vehicle automation, the Society of Automotive Engineers (SAE) outlines six levels for self-driving cars, from no automation to full automation. This system has become the standard and is summarized in Table 1-1 [12].

Table 1-1: The levels of autonomy outlined by SAE

	Classification	Description
Level 0	No automation	You drive it
Level 1	Driver assistance	Hands on the wheel
Level 2	Partial automation	Hands off the wheel, eyes on the road
Level 3	Conditional automation	Hands off the wheel, eyes off the road - sometimes
Level 4	High automation	Hands off, eyes off, mind off - sometimes
Level 5	Full automation	Steering wheel is optional

The research for this graduation thesis was conducted at the Integrated Vehicle Safety (IVS) department of TNO. The research focus of this department is to further develop the technology behind active safety systems and get closer towards fully autonomous vehicles. These solutions improve the safety of vehicles and prevent human fatalities and severe injuries in a robust and reliable manner [13]. The daily work was performed in collaboration with the Cooperative Control Systems (CCS) group. They specialize in building novel real-time safety critical cooperative and automated systems, covering fundamental theory, model-based development, simulation, and experimental evaluation in proof-of-concept demonstrators. The unique expertise, tools and facilities of TNO have led to the development of several high and

low level control software, such as the following ADAS systems:

- Vehicle State Estimation (**VSE!**),
- Anti-Lock Braking System (ABS),
- Autonomous Emergency Braking (AEB),
- Cooperative Adaptive Cruise Control (CACC),
- Fully Automated Highway Platooning.

CACC is a good example of the capabilities of an automatic vehicle. It aims at achieving small inter-vehicle distances, which can increase the road capacity, reduce traffic jams, and improve on fuel consumption. The method is based on inter-vehicle data exchange through wireless communication and data obtained by radar or LiDAR. Figure 1-2 depicts this, with v_{i+1} and d_{i+1} the communicated velocity and vehicle-distance of vehicle $i + 1$, respectively.

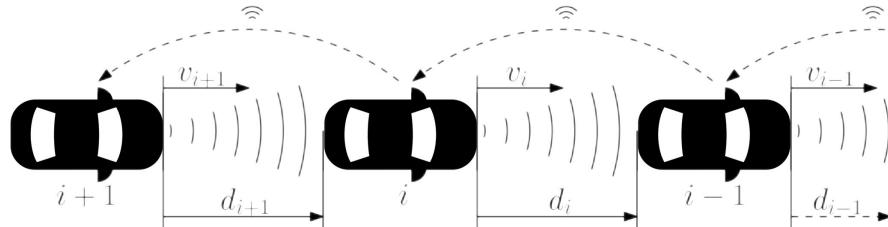


Figure 1-2: Highway platooning for CACC applications

The IVS department is located at the Automotive Campus in Helmond, that promotes itself as the national and international hotspot, meeting point and place of business in the field of automotive and smart mobility [14]. Companies and institutes ranging from the fields of education, research and industry, like Fontys, Lightyear, Rijkswaterstaat, VDL, and many more are based at this location. The Automotive Campus provides shared high-tech automotive testing facilities and flexible working places, where smart, safe, and sustainable mobility solutions for people, roads and vehicles are realized.

1-1 Problem statement

The task of path planning plays an critical role in developing an autonomous vehicle, which makes it an important field of research. The navigation control in most autonomous vehicles is based on first planning a path to a goal. The task of tracking this planned path is then achieved by feedback and feedforward controllers that apply correctional lateral and longitudinal control, consisting of steering input and braking or throttle action, respectively.

The strength of using the Artificial Potential Field (APF) method is the possibility to integrate this path-planning and -tracking process. It is based on the generation of repulsive Potential Fields (PFs) associated with obstacles, whereas the PF fixed to the goal position has an attractive effect on the vehicle. The sum of these fields yields the final APF, from which the artificial force is computed that directs the vehicle through the environment. One way

to produce a planned path is to simulate the vehicle motion under influence of the field, after which the simulated path can then be used as the planned path. The APF-method is often referred to as being a local method [15]. The applied artificial force namely includes the contributions from all PFs, but the forces are stronger close to an obstacle and have less effect at a larger distance. This allows for a local representation of the environment, which reduces computational difficulties significantly [16]. Another advantage is that it offers a simple and intuitive way to represent the level of hazard experienced by the vehicle in the environment. Furthermore, it results in paths that can be driven with some acceptable tolerance, as opposed to following a strictly defined trajectory.

The main contribution of this research is the development of a unified path-planning and -control solution for autonomous vehicles driving on highways using the APF-method. The vehicle dynamics should be incorporated into the planning of feasible and safe paths with the ability to re-plan in changing traffic situations. The research question is phrased as follows:

How to plan and track desirable and feasible paths for highway-driving autonomous vehicles using the APF-method?

The research aims at planning paths for controlled vehicles driving in steady traffic consisting of homogeneous vehicles on a two-lane highway. The planned paths should satisfy the criteria for being desirable and feasible, which are formally defined next. The paths are considered to be desirable if they are planned such that the vehicle:

- Stays on the road while driving according to common traffic rules and conventions;
- Follows the center of the lane, with a preference for the rightmost lane;
- Performs lane-change maneuvers in order to overtake slower preceding vehicles; and
- Keeps appropriate safety margins to static and dynamic obstacles.

In addition to being desirable, the planned paths are only allowed if they are considered to be feasible. This is the case if the planned paths can be tracked in a successful way that guarantees safety by avoiding collisions and satisfies the vehicle limitations. The comfort criteria that are used to evaluate the quality of the planned path are given by the path smoothness and having comfortable accelerations.

These desirable and feasible paths are to be generated by the APF-method. It should therefore have the capability to continuously re-plan the path based on the changing traffic situations. In simulation, it is assumed that all required sensor measurements are available. This corresponds to having the position and heading of the controlled vehicle on the road as well as the respective relative positions and velocities of the surrounding traffic. Furthermore, the layout of the highway environment is assumed to be known. The method is designed for real-time application such that it can be implemented on a real autonomous vehicle, which means that it should be computationally efficient.

The aim is to reproduce human-like driving behavior, since the controlled vehicle will be operating among vehicles that still have a driver behind the wheel. This means, for example, that common driving rules need to be incorporated, like left-side overtaking, having a preference to stay in the lane center, and returning to the right most lane. Only typical highway conditions

are considered, which means appropriate road curvatures and velocities, and normal driving conditions. Emergency maneuvers are beyond the scope of this thesis.

The paths resulting from the APF-method have to be tracked in order to evaluate the path-planning performance. The lateral controller developed by TNO is initially used for this purpose. In case the TNO controller proves to be unsuitable for integration with the APF-method, a new controller will be developed. The vehicle is also equipped with an ACC (longitudinal) controller. Tracking paths at constant longitudinal velocity is preferred, so this has to be taken into account in the design.

For practical implementation of the lateral control algorithm, a Toyota Prius is available TNO to serve as the experimental vehicle. It has a park assistant system that can be used to control the angle of the steering wheels. This test platform can be used to test and validate the lateral control performance. However, due to implementation issues that are described in Section 4-5, no experimental testing was done, unfortunately.

The general problem statement is summarized with the following research objectives:

- Modeling the static and dynamic road environment using PFs;
- Formally defining solution requirements in terms of safety, comfort, and performance;
- Adopting a controller capable of tracking the proposed trajectory from the APF; and
- Evaluation of the integrated path-planning and -tracking method in simulation.

1-2 Literature study

This section presents a summary of the findings from the literature study that was performed in preparation for the main research. The most common path-planning methods were discussed, together with the main challenges encountered in this field. Next, an overview of the most general path-tracking controllers was given and their advantages and disadvantages were compared. After that, the APF-method was explained and the challenges to obtain good path-planning and -tracking performance were identified. Several examples were given to show the different applications of APFs in path planning and control. Some general comments on the APF design requirements for automated driving were given eventually. The most important results from the literature study are presented below.

Path planning is a very exciting field of research for which new solutions are provided almost daily. The main challenges of path planning for autonomous vehicles are given by the limited sensor data of the environment, the dynamic and kinematic vehicle constraints, and the computational demand for real-time implementation. Classical path-planning approaches, like cell decomposition methods and rapidly-exploring random trees, have the major drawback of leaving out the vehicle dynamics [17]. This leads to a suboptimal tracking performance at high speeds and for varying path curvatures. The dynamics of both the controlled and the obstacle vehicle play namely an important role in successful path planning. The absolute and relative velocities, for example, determine the safe distance that should be maintained. Current research focuses on generating trajectories that are similar to human driving behavior,

in order to yield user-acceptable paths [18]. Human-like driving is characterized by smooth and continuous steering action, while minimizing the lateral jerk and acceleration.

Trajectory tracking is concerned with computing the actuator input that makes the vehicle track the predefined trajectory. This can either be a correction to the steering input to adjust the lateral position, or an application of the brake or throttle to change the longitudinal speed. The used vehicle model and the quality of the planned path influence the performance of the tracking controller. The bicycle model is the mostly used dynamic model to describe the lateral dynamics of the vehicle. The linearized equations give a good representation of the vehicle behavior in steady-state highway-driving scenarios. The controller can be tuned by evaluating the tracking performance during a lane-change maneuver from looking at the rise time, overshoot, settling time and kept inter-vehicle distance of the response. A detailed comparison between many different lateral controllers concluded that no perfect control law exists that can be used in every situation and at all velocities. Choosing a suitable controller for a specific design case is done by considering its simplicity, efficiency, and robustness.

The **APF-method** offers a very promising solution to the path-planning problem for autonomous vehicles. The path is planned from forward simulation of the vehicle motion that is influenced by the artificial force coming from the PFs. The main challenge is how to include the vehicle dynamics into the path-planning process in order to produce feasible paths. The height, slope, and shape of the PF are recognized to play an important role in producing the desired vehicle behavior.

1-3 Thesis outline

The main body of the thesis is organized in the following way. Chapter 2 describes the basic principles of autonomous driving that are relevant for this research. This includes the autonomous system, the employed sensors, the highway environment, and some of the most common vehicle maneuvers. After that, the design of the APF path-planning and -control method is explained in Chapter 3. It treats the equations behind the bicycle model, the construction of the PFs, and the formulation of the Model Predictive Control (MPC) framework. Chapter 4 elaborates on the used simulation environment. It talks about the implementation of the APF MPC controller in MATLAB and Simulink, gives some details about the TNO controller, and discusses the obtained results. Finally, Chapter 5 gives the conclusions of this graduation work, ending with some recommendations for future research.

Several Appendices is found at the back of this thesis, containing material that is too extensive to be included into the main body. It contains the derivation of equations, mathematical theory, several plots and figures, and tables with simulation results. An exact overview of each Appendix is given next:

- A: Bicycle model derivation;
- B: Quadratic Taylor approximation;
- C: MPC matrix derivation;
- D: Simulink model;
- E: Tuned obstacle PF parameters;
- F: Lane-change characteristics results.

The thesis closes off with the bibliography, the glossary, containing the list of acronyms and the list of symbols, and finally the index.

Chapter 2

Autonomous driving

This Chapter serves as a more detailed introduction to the topic of autonomous driving. The outline is as follows. Section 2-1 briefly outlines the complex technology behind the control system of an autonomous vehicle. Next, Section 2-2 elaborates on the things that characterize the human driving behavior. Section 2-3 treats the subject of the highway layout, which is the environment of interest in this research. After that, Section 2-4 gives some details on the sensors that are commonly used by autonomous vehicles. Furthermore, it describes the sensor measurements that are available for the design of the path-planning method. Section 2-5 shortly discusses the two most basic vehicle maneuvers. Next, Section 2-6 describes the different approaches to collision avoidance. Finally, a list is presented with the assumptions that were made in the design of the APF path-planner and -controller.

2-1 Autonomous vehicle system

This Section describes the main characteristics of the autonomous vehicle system and its general differences with robots.

The problem of finding collision-free trajectories does not only exist in the automotive field, but also in nautics, aerospace, and robotics. Although path planning for robots and vehicles is very much alike, there are some crucial differences between the two, which is why the methods for robotics generally cannot be applied directly to autonomous vehicles. For example, geometric methods are unfavorable in scenarios with several obstacles and rule-based methods easily become too complex because it is impossible to predefine all possible traffic situations that can be encountered [10].

Robotic path-planning algorithms generally produce non-smooth paths that are not satisfied by the non-holonomic constraints of a vehicle. Some robots are able to track such paths, because they are omnidirectional. A car, however, would have to stop repeatedly to reorient its front wheels in order to perfectly track the path with discontinuous curvature. A non-holonomic system is subject to kinematic constraints that restricts its motion. Car-like vehicles can only move forward or backward in the direction perpendicular to the rear wheel

axle. Simply put, a car is unable to move straight to the side. Also, the turning radius is bounded due to the mechanical limits of the steering wheel. These non-holonomic constraints make planning paths much harder.

Another significant difference with path planning for robots is that the lateral and longitudinal danger distribution around a driving vehicle is not uniform. That is to say: a vehicle should stay far behind a preceding vehicle, but is allowed to come relatively close to the side of another vehicle. This is related to the fact that the lateral velocity is very small in comparison with the longitudinal velocity, especially on highways. Furthermore, the direction of travel is limited by the road layout.

A global path planner is often used in cooperation with a local planner [19]. Global path planning is a slow and deliberative process in which long-distance paths are planned to reach the final destination. The much faster local path planner, on the other hand, is concerned with short-distance planning, dealing with tasks that are closer at hand, like vehicle stability, obstacle avoidance, and safety. It is much more reactive and runs in real time. Local path planning is the topic of this thesis, aiming at generating paths that are consistent with the kinematic vehicle constraints, on top of having comfortable, safe, and human-driver-like motion.

Figure 2-1 gives a detailed view of the different technologies in the complicated system of an autonomous vehicle and how they are related. Roughly said, the overall autonomous system consists of three different modules, with each their own task to complete. These are:

1. Sensing – To provide data of the environment that can be used for planning;
2. Planning – To generate safe and feasible paths from data from the sensing module; and
3. Control – To operate the vehicle to follow the desired path from the planning module.

The navigation control, which consists of tracking a predefined trajectory, is usually divided into a hierarchical structure of planning and control [20]. In this scheme, the path planner uses the data from the perception module to plan a path. The controller then calculates and applies the required steering, throttle, and/or braking action to follow the desired path. As was mentioned before, the APF-method is capable of unifying the two modules of path planning and tracking.

Passive ADAS systems are designed to give auditory, visual, or haptic feedback in order to inform the driver. Active ADAS approaches are developed to increase safety while working cooperatively with the driver. Both ADAS technologies have their own trade-offs and advantages. Passive systems are easier to implement because the final responsibility still lies at the driver, who is required to respond to the given warning. In the case of active systems, however, the possibility of human errors is eliminated, which comes at the cost of having a much more complicated system because it needs to have a higher degree of robustness. This prevents these type of ADAS solutions from being readily taken into production [8].

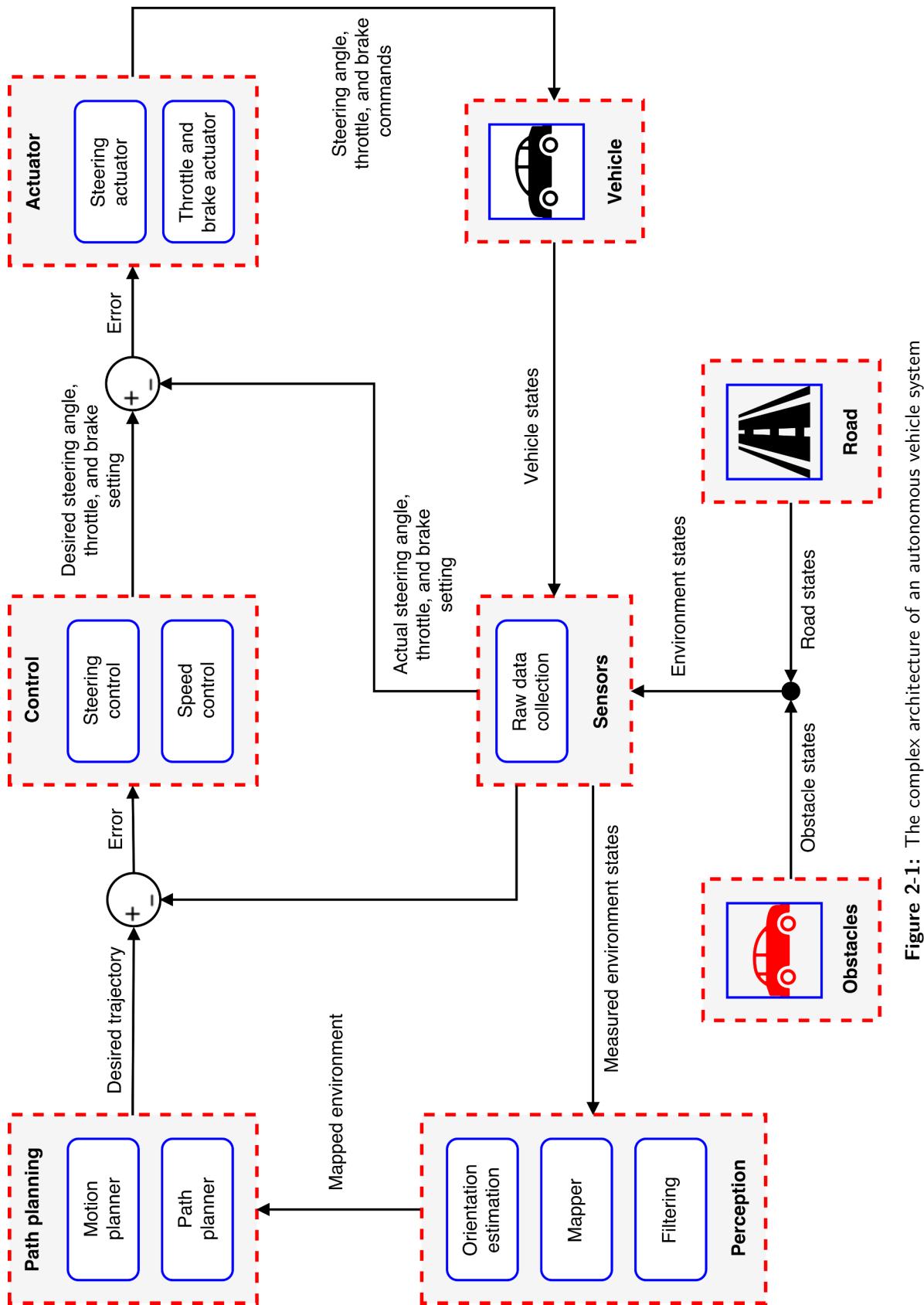


Figure 2-1: The complex architecture of an autonomous vehicle system

2-2 Driver behavior

A study on human lane-change driving, concludes that the steering wheel angle during the maneuver resembles a sine function [21]. To be more precise, four different phases can be discerned in the movement of the steering wheel. In the first phase, the initial steering action is to the left, after which in phase two the steering returns to its central position. This position is reached when the heading angle approximately attains its maximum value. In the third phase, the steering wheel is turned to the right, and the last phase entails the stabilizing part at the end of the maneuver [22].

The steering wheel angle can be controlled in two ways. In the first way, a motor is directly coupled to the steering column, that is used to provide the torque. This configuration is used for steering aid in lane-keeping ADAS systems, for example. The second way is an example of the so-called 'by-wire' actuation systems. In these systems, no mechanical connection exists between the driver input and the actuators, but instead the desired steering angle is commanded by a digital controller. Steer-, brake-, and throttle-by-wire are examples of systems that are currently found in production vehicles [8]. Most steering control modules are developed on the assumption that drivers steer their vehicle in closed-loop fashion. However, it is commonly accepted that driving does not require continuous error correction or permanent visual feedback, because sometimes attention is paid to other aspects, that may not be related to driving [22].

The lateral acceleration and the associated jerk should not exceed $0.12 \text{ g} \approx 1.18 \text{ ms}^{-2}$ and $0.24 \text{ g/s} \approx 2.35 \text{ ms}^{-3}$, respectively, for a comfortable driving experience [21]. The maximum values of these dynamic properties are relevant and good indicators for comfort and safety perceived by the driver [23]. The observed lateral accelerations on highways from [24] were less than 3.5 ms^{-2} , with most lower than 1.8 ms^{-2} .

The total reaction time of a driver varies from 0.5 to 4 seconds. This large range can be explained by realizing that the four contributions to the total reaction time greatly differ from person to person [25]. The four contributions from the so-called PIEV theory are:

- Perception: the time required to transmit the sensation to the brain;
- Intellection: the time required to understand the situation;
- Emotion: the time elapsed during emotional sensation; and
- Volition: the time taken for the final action.

2-3 Highway environment

The structure and relative simplicity of highways offer great opportunity to develop ADAS for vehicles. Complicated scenarios associated with driving in urban environments, like traffic lights, stop signs, crossing traffic, and tight turns, are not found on highways. Furthermore, because of the high vehicle speeds on highways, avoiding collisions is of even greater importance. While many ADAS systems have been realized, the responsibility for avoiding collisions

still remains with the driver [16]. The research from the National Motor Vehicle Crash Causation Survey, conducted from 2005 to 2007 on light vehicles, concluded that driver error was the most critical factor in 94% of the crashes [11]. So the opportunity and need for increased driver assistance on highways is both obvious and desired.

The geometric design of a highway is concerned with the dimensions and layout of visible features such as alignment, sight distance, and radius of curve. The main goal of geometric highway design is to maximize the comfort and safety of the traffic in operation, while minimizing the costs and environmental impacts [25]. Superelevation, also referred to as cant or banking, is an important aspect in the geometric design of road infrastructures. It is the ratio of the height of the outer edge of a curved road with respect to the width of the road. Superelevation is added by raising the outer edge of the road. This is done to reduce or counteract the effect of the centrifugal force, which has the tendency to make the vehicle turn over or skid outwards, when driving along a horizontal curve [25].

Table 2-1 shows the minimum horizontal lane curve radii for different design speeds and cants conform *Richtlijn Ontwerp Autosnelwegen 2014* from Rijkswaterstaat [26]. It should be noted that curves with a radius smaller than 300 m should have a minimum cant of 5.0% to counteract the centrifugal forces, which explains the Not Applicable (N/A) entries in the Table. Furthermore, highway curves are designed without discontinuous road curvatures such that the curvature gradually increases or decreases.

Table 2-1: Minimum horizontal lane curve radii in [m] for different design speeds and cants

Minimum horizontal curve radii in [m]				
Cant in [%]	Different design speeds in kmh^{-1}			
	120	90	70	50
0	4,000	2,000	800	300
2.5	1,500	700	350	N/A
3.0	1,350	630	315	N/A
3.5	1,200	560	N/A	N/A
4.0	1,050	490	N/A	N/A
4.5	900	420	N/A	N/A
5.0	750	350	180	85
5.5		340	175	85
6.0		330	170	85
6.5			165	85
7.0			160	85

2-4 Sensors

Sensors are essential for retrieving meaningful information about the environment. The choice of sensors developed for autonomous vehicles is big, amongst which the most common are infrared sensors, ultrasonic sensors, sonar, LiDAR, laser range finder, and camera [15]. Path

planning in an unknown environment greatly relies on having reliable sensor measurements, since the positions, directions, and velocities of the surroundings are updated using sensor data. The difficulty is that these sensors alone are unable to independently provide the system with all the necessary information. As a result, for example, processing techniques are necessary to determine the distance to an object through triangulation and to predict the future motion of moving obstacles, based on current and past observations.

The sensors in this control framework are most important for lane detection. The approaches that are commonly used to handle this problem are either feature-based or model-based [27]. The former uses image segmentation to detect the lanes. The problem with this method is that it does not impose constraints on the lane shapes, which may lead to wrong detection due to occlusion or noise. The model-based method uses mathematical models to represent the lanes.

TNO has a modified Toyota Prius car at its disposal for real-life experiments on test tracks. The car is equipped with a rapid control prototyping system that controls the steering angle with an electric power steering motor. Furthermore, a Mobileye camera is used to obtain line information from the road. The line representations that are obtained from this are used as a basis for the path-planning problem. The position of other nearby vehicles is obtained from the fused camera and radar data. Figure 2-2 shows an illustration of the sensor measurements of the environment, where measurements 1 to 3 are related to the lane centers and 4 to 6 to the obstacle vehicle.

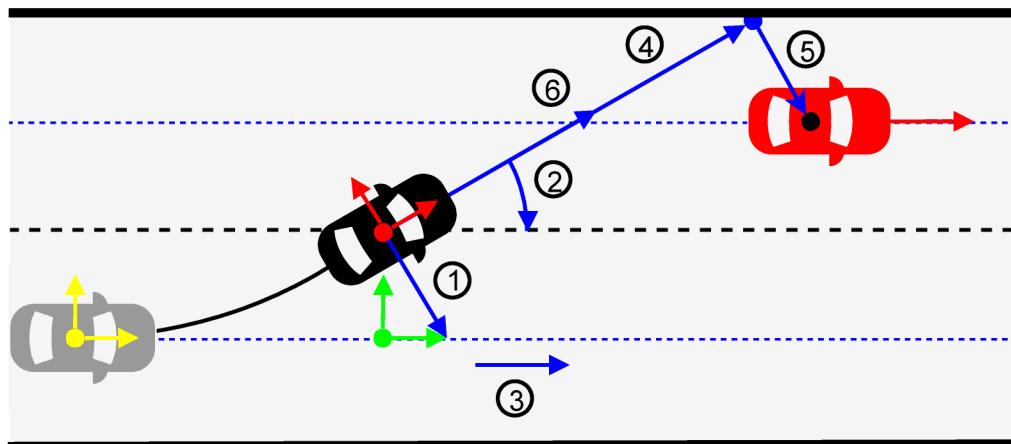


Figure 2-2: Sensor measurements of the environment: 1) lateral lane-center distance, 2) lane heading, 3) lane curvature, 4) longitudinal obstacle distance, 5) lateral obstacle distance, 6) longitudinal velocity difference

The three differently colored vehicles are interpreted as follows. The gray car depicts the host vehicle at its initial position, the black car is the host vehicle in its current position, and the red car is the obstacle vehicle. Furthermore, three different coordinate frames can be seen. The yellow coordinate frame denotes the global road coordinate frame. It is fixed to the road in the right lane center of the starting position, with its x -axis aligned in the longitudinal road direction. The green coordinate frame is the local road coordinate frame. It moves along with the host vehicle, such that the host vehicle is always located at $X_l = 0$ m. Again, the x -axis

is aligned with the road. Finally, the red coordinate frame is the local host vehicle coordinate frame, of which the axes are fixed to the vehicle Center of Gravity (CoG), coinciding with the host vehicle lateral and longitudinal axes. In the remainder of this thesis, the subscripts g , l , and h will be used to indicate that the variables are expressed in the global road, local road, or host vehicle coordinate frame, respectively. The Mobileye sensor measurements from Figure 2-2 are represented as follows:

$$\begin{aligned} \textcircled{1} &\rightarrow \frac{Y_{\ell c,\ell} - Y_{h,\ell}}{\cos(\psi_{\ell c,\ell} - \psi_{h,\ell})}, & \textcircled{4} &\rightarrow \Delta X_{o,h}, \\ \textcircled{2} &\rightarrow \psi_{\ell c,\ell} - \psi_{h,\ell}, & \textcircled{5} &\rightarrow \Delta Y_{o,h}, \\ \textcircled{3} &\rightarrow 1/2\rho, & \textcircled{6} &\rightarrow \Delta V_{o,h}. \end{aligned} \quad (2-1)$$

It should be noted that the measurements are originally expressed in the local host vehicle frame, since the sensors are mounted on the host vehicle. However, the first three measurements from Eq. (2-1) are expressed in the local lane frame, since that is how they are used in the developed algorithm. The last three measurements from Eq. (2-1) are still expressed in the local host vehicle frame. The symbols from Eq. (2-1) are summarized in Table 2-2.

Table 2-2: Symbols from the Mobileye sensor measurements

Description	Symbol	Value	Unit
Lateral lane center position in local road frame	$Y_{\ell c,\ell}$	0 or 3.5	[m]
Lateral host vehicle position in local road frame	$Y_{h,\ell}$		[m]
Lane center heading in local road frame	$\psi_{\ell c,\ell}$	0	[rad]
Host vehicle heading in local road frame	$\psi_{h,\ell}$		[rad]
Road curvature	ρ		$\text{[m}^{-1}\text{]}$
Longitudinal distance to obstacle vehicle in host frame	$\Delta X_{o,h}$		[m]
Lateral distance to obstacle vehicle in host frame	$\Delta Y_{o,h}$		[m]
Velocity difference between obstacle and host vehicle	$\Delta V_{o,h}$		$\text{[kmh}^{-1}\text{]}$

The Mobileye sensor measurements of the road, which are given by number 1 to number 3 from Eq. (2-1), are used to determine the lateral position of the lane center expressed in the local road frame with the following equation:

$$Y_{\ell c,\ell} = c_2 X_\ell^2 + c_1 X_\ell + c_0, \quad (2-2)$$

with c_2 the quadratic coefficient, representing the Mobileye lane curvature through the relation $c_2 = 1/2\rho$, c_1 the linear coefficient, representing the initial lane heading, and c_0 the scalar term, representing the initial lateral lane offset, all expressed in the local road frame. The symbols from Eq. (2-2) are summarized in Table 2-3.

The Mobileye sensor distance measurements of the obstacle vehicle, which are given by number 4 and number 5 from Eq. (2-1), are used to determine the Inter-Vehicle Distance (IVD) expressed in the local road frame as follows:

$$\begin{aligned} X_{o,\ell} &= \cos(\psi_{h,\ell}) \Delta X_{o,h} + \sin(\psi_{h,\ell}) \Delta Y_{o,h}, \\ Y_{o,\ell} &= -\sin(\psi_{h,\ell}) \Delta X_{o,h} + \cos(\psi_{h,\ell}) \Delta Y_{o,h} + Y_{h,\ell}. \end{aligned} \quad (2-3)$$

Table 2-3: Symbols from the lateral lane center position equation

Description	Symbol	Value	Unit
Mobileye lane curvature	c_2	$\frac{1}{2}\rho$	[m ⁻¹]
Initial lane heading in local road frame	c_1	0	-
Initial lateral lane offset in local road frame	c_0	0 or 3.5	[m]
Longitudinal position in local road frame	X_ℓ		[m]

Finally, the velocity difference between the host and obstacle vehicle, which is the last measurement from Eq. (2-1), can be calculated in the longitudinal direction of the local road frame like:

$$\Delta V_{o,h} = V_o \cos(\psi_{o,\ell} - \psi_{h,\ell}) - V_h. \quad (2-4)$$

The symbols used in calculating the obstacle information from the sensor measurements are summarized in Table 2-4.

Table 2-4: Symbols from the obstacle information calculations

Description	Symbol	Value	Unit
Longitudinal host vehicle position in local road frame	$X_{o,l}$		[m]
Lateral obstacle vehicle position in local road frame	$Y_{o,l}$		[m]
Obstacle vehicle heading in local road frame	$\psi_{o,\ell}$		[rad]
Obstacle vehicle velocity	V_o		[kmh ⁻¹]
Host vehicle velocity	V_h		[kmh ⁻¹]

2-5 Vehicle maneuvers

Path planning is one of the most challenging tasks in autonomous driving. However, when driving in a structured environment it can be generalized and simplified by breaking it down into two basic vehicle maneuvers, namely lane keeping and lane changing [28]. With lane keeping, the vehicle follows and stays in its current lane by continuously adjusting its distance and orientation to the lane center.

The single lane change is the most common vehicle maneuver, from which other more complex maneuvers can be composed, like vehicle overtaking, obstacle avoidance, and road departure. Lane changing also has a significant effect on the traffic throughput and traffic safety [29]. However, it is not a safe and straightforward maneuver. Statistics show that many accidents occur during a lane change. This can be explained by the fact the maneuver requires high driver cognitive workload and skill [30]. Besides, there are many different ways of how to realize the corresponding trajectory. It is therefore one of the most extensively researched automated driving operations.

Figure 2-3 shows of a flow chart with the reasoning behind the decision making of which maneuver to perform for different highway-driving scenarios.

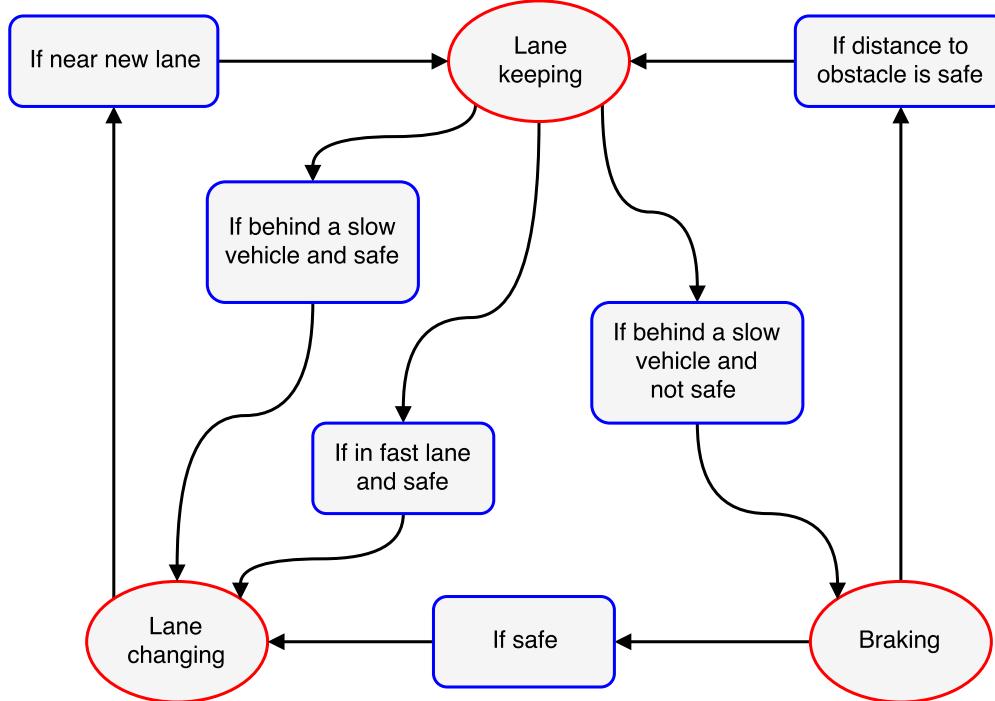


Figure 2-3: Flow chart of vehicle maneuvers for highway driving

2-6 Collision avoidance

Human error is the main cause for traffic accidents, as confirmed by statistics. According to the World Health Organization, with more than 1.3 million road accidents fatalities occurring world-wide every year, it was the 9th leading cause of death in 2004 and expected to be in the top 3 by 2020 [31]. Among the main causes of traffic accidents are: weather and road conditions, dangerous traffic participant behavior, and the inability of human drivers to correctly predict and quickly react to collision threats.

The main requirement for collision avoidance is that the executed maneuver is safe and comfortable, for which there are two options. These options are illustrated in Figure 2-4. When a vehicle encounters an obstacle vehicle it can either find an alternative path or adjust its velocity. The first option is the topic of this research and the second option is handled by a longitudinal controller. These two options of changing lanes or braking in a comfortable way can only be performed if the obstacle is sensed at a large distance. In case of an emergency, when the obstacle is too close, safety considerations predominate over comfort and the vehicle brakes or steers as much as possible. However, in these circumstances, longitudinal control is less effective at preventing collisions. Aggressive lateral control is more appropriate in that case, since a steering maneuver can be completed in a shorter distance than the distance required for coming to a full stop [32]. However, as was previously mentioned, these emergency scenarios are not taken into consideration in this research. Changing lanes is the preferred maneuver, since this will permit the vehicle to continue driving at the same reference speed. It also disturbs the traffic flow the least. In the limiting case, when a safe lane change is no longer possible to avoid collision, the ACC module will take over control by reducing the vehicle speed or by even stopping the vehicle completely in situations of imminent danger.

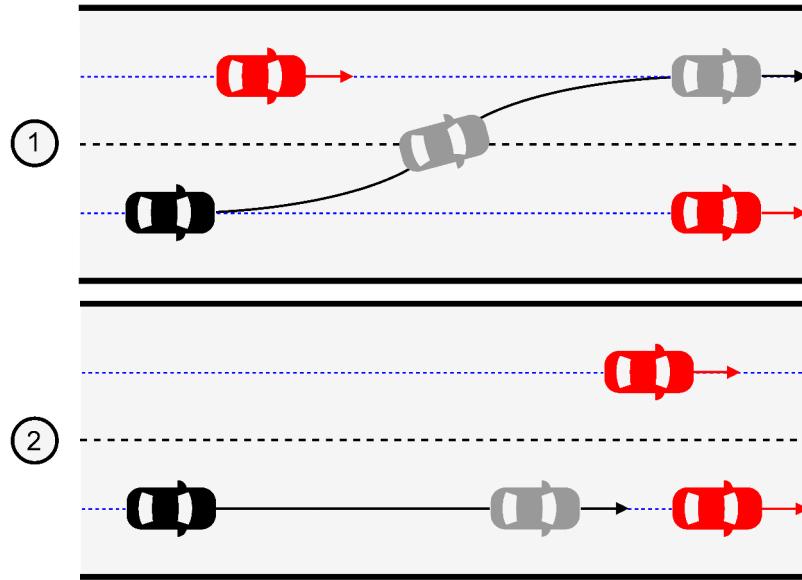


Figure 2-4: Two options for collision avoidance: 1) comfortable lane change, 2) normal braking

There are several methods to estimate the required safety spacing if the leading vehicle slows down or performs a stopping maneuver. If there is no information exchange between the involved cars, a worst-case assumption has to be made by taking a large safety margin. Otherwise, if the vehicle class and braking capabilities are communicated, this can be used to significantly reduce the safety margin. Possibly even to the minimum spacing distance for longitudinal vehicle following [33].

The ability of a vehicle to avoid collisions is predominantly determined by the friction potential of its tires. A collision avoidance system typically operates close to the limits of vehicle handling, which are governed by the nonlinear tire-road friction properties. The environmental conditions that affect these properties may change abruptly, whereas properties like vehicle mass and inertia vary slowly with time and are therefore more easily estimated. The development of a friction estimator that functions properly under a wide range of operating conditions still requires further research [34].

Some simplifying assumptions were made during the design of the APF-algorithm:

- The research is limited to highway scenarios, with low road curvatures, a speed limit of 130 km/h, and lanes of width $\ell_w = 3.5$ m;
- The host vehicle, defined as the vehicle controlled by the developed approach, is equipped with a steering controller capable of following the commanded steering angle input;
- The longitudinal velocity of the host vehicle is assumed to be constant, which in practice is regulated by the ACC;
- Only a single obstacle vehicle appears in front of the host vehicle, with a constant speed, driving in the center of the lane; and
- Reliable host and obstacle vehicle data, like position, speed, heading, etc., is available.

Chapter 3

Controller design

This Chapter describes the design of the developed APF-controller. It has the following outline. Section 3-1 explains about MPC, which is the control method that was chosen to be integrated with the APF-method. After that, Section 3-2 gives the equations behind the used vehicle model, namely the bicycle model, and discusses the benefits and drawbacks of this model. Next, Section 3-3 treats the design of the employed PF functions, consisting of the road PF and the obstacle PF, which are superimposed to yield the final APF. Finally, Section 3-4 talks about how the APF-method is combined with the MPC-method. The governing equations defining the MPC cost function are presented, which form the fundamental base of the developed APF MPC method.

3-1 Model predictive control

Dynamic controllers have the advantage of taking the vehicle dynamics into account, either by directly deriving a control law from the dynamic model or by incorporating some of the dynamic properties. Optimal controllers compute the desired control input by solving a Quadratic Programming (QP) problem, which results in a simple control law. However, most optimal controllers are designed based on a linear model of the system, which limits their operating range. Adaptive controllers have good robustness against disturbances and uncertainties. This is especially desirable in applications with unknown system properties or changing operation conditions. However, their design usually requires a lot of effort, since the intelligent algorithms need to be trained or programmed with the right knowledge. The properties of dynamic, optimal, and adaptive controllers are combined in MPC. The method uses a model-based controller that is involved in the optimization step of the predicted states of the model to generate the optimal control input. It is similar to adaptive control because it is capable of adjusting itself to changing conditions. It handles input and output constraints explicitly by solving an optimization problem at each control interval. These properties make MPC a suitable candidate for the APF path-planning and -tracking method. The proposed APF MPC controller is seen in the autonomous vehicle block diagram of Figure 3-1.

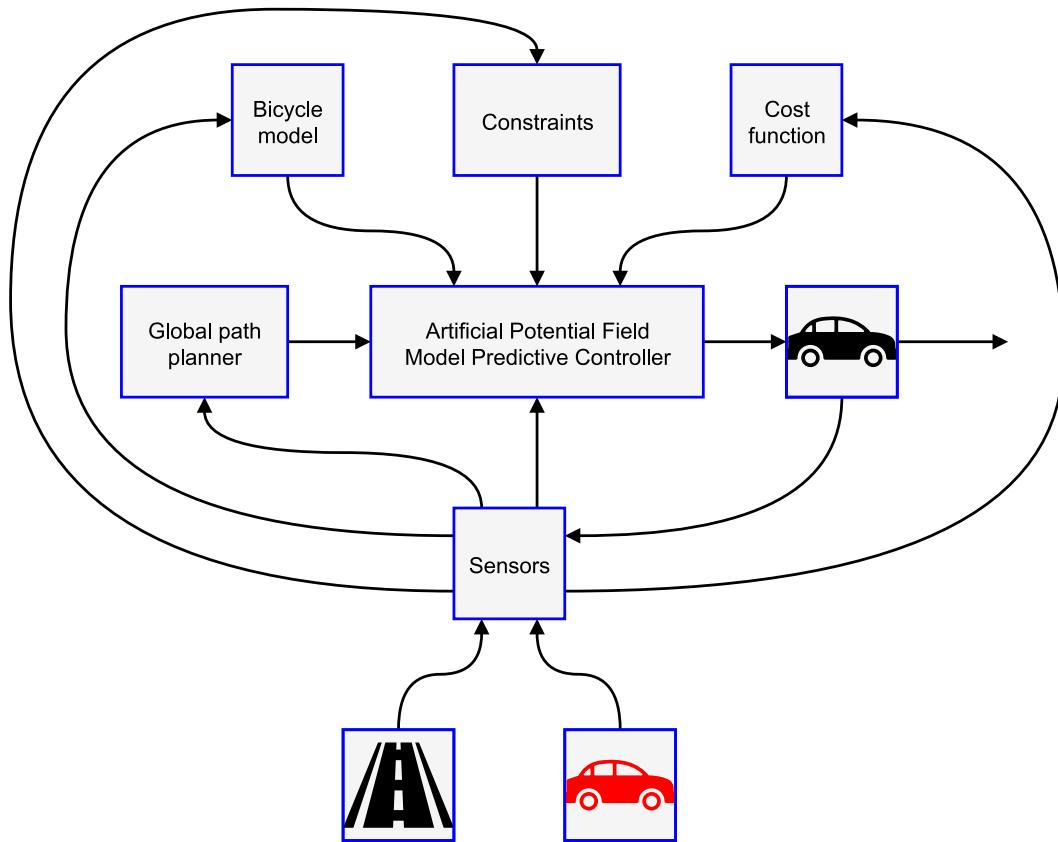


Figure 3-1: Simplified block diagram of autonomous vehicle system with APF MPC controller

In MPC, a model is used to predict the future states of the system. It involves an optimization step in order to determine the optimal control input to the model, based on optimizing a certain cost function subject to some constraints. By using a model of the system and its corresponding constraints, the method guarantees that only those control inputs leading to feasible trajectories will be generated. The first sample of the optimal control input is then applied to the system at time t while keeping it constant over the sampling time interval. The prediction horizon is then shifted, after which a new optimization problem is solved. This concept is known as Receding Horizon Control (RHC), which essentially is the same as a repeated optimal control scheme performed. The RHC approach is very attractive for several reasons. Firstly, the cost function is continuously updated with the latest sensor measurements of the vehicle configuration, road parameters, and obstacle information [32]. This accommodates for the limited range of the sensors. Secondly, updating and solving the problem incrementally introduces a feedback-mechanism, offering the possibility to re-plan the paths to account for sudden changes in the environment [4]. Thirdly, the use of a limited horizon reduces the computational burden in path planning [35]. Lastly, the method based on RHC is similar to how humans drive [36]. MPC-controllers are sometimes seen as adaptive controllers, because of the RHC, which also accommodates for the rejection of disturbances. Adaptability is a desirable property in order to deal with various road conditions. Furthermore, an adaptive controller can also compensate for speed variations, inaccurate vehicle parameters, unmodeled frictions, and other unexpected circumstances. So the RHC principle also provides MPC with some robustness.

The application of MPC has been (and is still being) actively researched since there has been a big increase in the availability of systems that can process complex algorithms [5]. The high computational demand for solving the optimization problem can be decreased by reducing the prediction horizon or by taking larger time steps. However, this comes with a loss in the accuracy and reliability of the controller. A longer prediction horizon, on the other hand, allows for early, anticipatory actions in the present to avoid dangerous situations in the future.

3-2 Vehicle model

The vehicle model plays a crucial role in two aspects of the controller design. The first aspect is the control law derivation, which can be based on the mathematical representation of the vehicle model. The second aspect is the system simulation, where the vehicle model is used to simulate the vehicle behavior and the performance of the proposed controller.

The vehicle dynamics involve coupled lateral and longitudinal dynamics. It is common to assume that these dynamics are decoupled in highway-driving applications, where the road curvature is low, the lateral velocity can be neglected compared to the longitudinal velocity, and the traffic is usually driving steadily. Furthermore, the steering angles are small and the longitudinal velocity changes slowly. The coupling becomes more apparent when driving at high speeds, with high accelerations on tightly curved roads and can therefore no longer be ignored in these situations. The advantage of decoupling the dynamics is that it allows for independent design of the lateral and longitudinal controller. The longitudinal controller, in this case, delivers the speed reference and accelerates or decelerates accordingly, either to limit the centripetal acceleration or to drive at the maximum allowed speed.

It is crucial to include the vehicle dynamics into the path-planning method in order to generate feasible paths that satisfy the non-holonomic constraints and the other system limitations of a vehicle. There exist several vehicle models for simulating different types of driving behavior. The complexity of these models depends on the purpose of the model and the available information of the vehicle. This can range from a simple point-mass model to a very detailed multi-body model. The construction of these models is based on a certain set of assumptions. The validity of these assumptions in the specific case determine whether using a simple linear or a complex non-linear vehicle model gives sufficiently reliable results.

The bicycle model, also called the single-track model, is the most popular model to describe the lateral vehicle dynamics and to design a path-tracking controller. In this model, shown in Figure 3-2, the left and right tires are assumed to behave equally and are therefore merged and represented by a single tire at both the front and rear axle of the vehicle. This approach comes with the weakness of leaving out significant differences in the left and right tire response during sharp cornering [5].

Additionally, the self-aligning moments of the wheels are neglected. Furthermore, the vehicle body is assumed to be a rigid body with concentrated mass at the CoG, without suspension movements. The vertical motion of the vehicle is not of interest and is left out of consideration by assuming a flat road surface [5]. The aerodynamics are also neglected, so the most influencing external forces come from the interaction between the tire and the road surface. The deformation of the tires generates forces in both the lateral and longitudinal direction and are calculated from a non-linear function that is usually linearized.

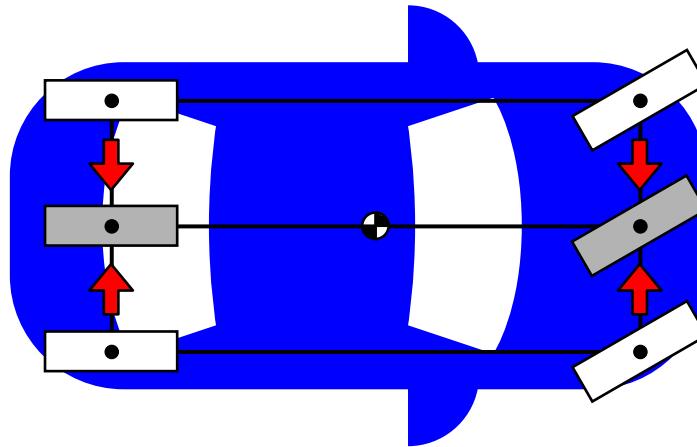


Figure 3-2: Transformation from car to bicycle model

The bicycle model describes the lateral dynamics of the vehicle in the local coordinate frame O^h attached to the host vehicle CoG. The host frame has its axes aligned with the longitudinal and lateral axis of the host vehicle. The kinematic vehicle model is used to express the motion in the global road coordinate frame O^g that is fixed to the right lane center of the road [37]. The rotation matrix that is a function of the vehicle heading angle ψ_h is needed for this. Furthermore, there is the local road coordinate frame O^ℓ that is attached to the right lane center of the road. Both the global and the local road frame have their x -axis aligned along the longitudinal direction of the road. These different coordinate frames can be seen in Figure 3-3.

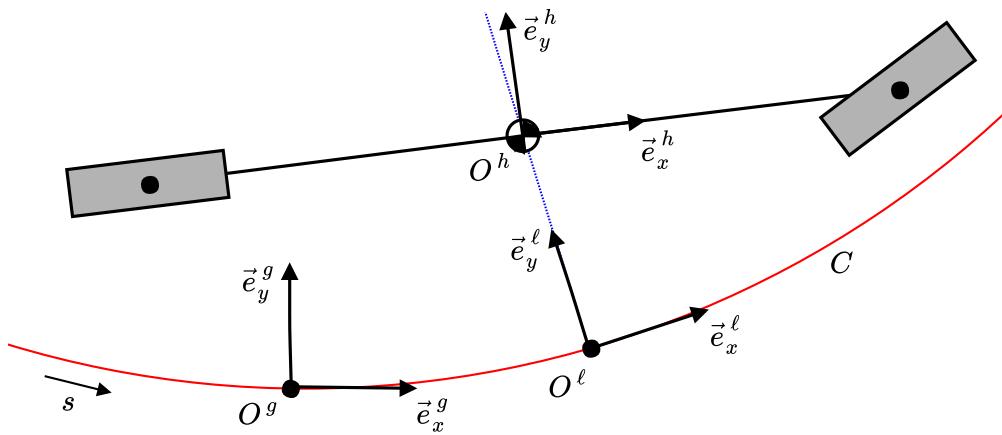


Figure 3-3: Different coordinate frames used in simulation

The governing equations of the bicycle model can be derived by using Newton's laws of motion on the free-body diagram of the bicycle model from Figure 3-4. Appendix A gives a detailed derivation of the linear equations of motion for the lateral dynamics of the bicycle model. The linearized model is based on assuming linear tire response and using small angle approximations, which greatly simplifies the equations. Using a nonlinear model gives higher accuracy at the expense of being more complex.

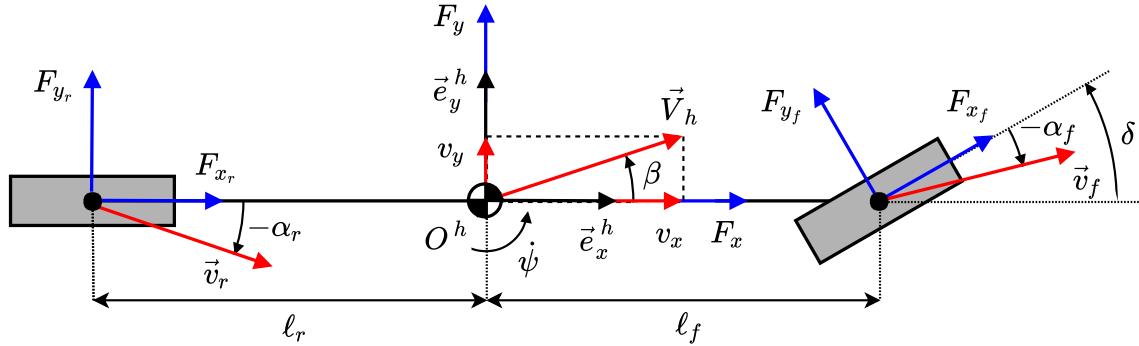


Figure 3-4: Free-body diagram of the bicycle model

The vehicle dynamics of the bicycle model are described by two states: the vehicle yaw rate $\dot{\psi}_h$ and the lateral velocity v_y of the CoG. These are the most important variables of the lateral dynamics that form the Degrees of Freedom (DoF) together with the longitudinal velocity of the vehicle CoG, given by v_x . The total velocity of the host vehicle CoG is represented by V_h . The input to the system is the front wheel steering angle δ . A nonlinear vehicle model can be derived from the free-body diagram of Figure 3-4. This leads to the following equations, where the accelerations of the vehicle CoG in the longitudinal, lateral, and yaw direction, \dot{v}_x , \dot{v}_y , and $\ddot{\psi}_h$, respectively, are nonlinear functions of the tire forces:

$$\begin{aligned} m(\dot{v}_x - \dot{\psi}_h v_y) &= F_{x_f} \cos(\delta) - F_{y_f} \sin(\delta), \\ m(\dot{v}_y + \dot{\psi}_h v_x) &= F_{x_f} \sin(\delta) + F_{y_f} \cos(\delta) + F_{y_r}, \\ I_z \ddot{\psi}_h &= \ell_f (F_{x_f} \sin(\delta) + F_{y_f} \cos(\delta)) - \ell_r F_{y_r}, \end{aligned} \quad (3-1)$$

Here, m is the vehicle mass, I_z is the mass moment of inertia around the vehicle CoG, and ℓ_f and ℓ_r are the distance from the front and rear axle to the vehicle CoG, respectively. Furthermore, F_{x_f} is the longitudinal force at the front tire, and F_{y_f} and F_{y_r} , are the lateral forces at the front and rear tire, respectively. Having a proper model for the tire behavior is important, since the generated tire forces are the main forces acting on the vehicle.

For highway driving, where the speed is high and the road curvature is low, the front wheel steering angle is small. Therefore, it is reasonable to make use of the small-angle approximation. The angles that are assumed to satisfy the small angle approximation are: the vehicle body slip angle β , the front and rear tire slip angles, α_f and α_r , respectively, and the steering angle δ . At high speed cornering, the tires must produce significant lateral forces to counteract the lateral acceleration. The lateral forces from the front and rear tires are generated as a result of the tire slip with the road. The tire slip angle is defined as the angle between the heading of the vehicle and its travel direction, that can be obtained from the velocity vector. The lateral forces increase with the slip angle. The linearized version of the bicycle model assumes a linear tire response, which greatly simplifies the dynamic equations. The linear lateral forces are calculated as follows:

$$\begin{aligned} F_{y_f} &= C_f \alpha_f, \\ F_{y_r} &= C_r \alpha_r. \end{aligned} \quad (3-2)$$

A positive slip angle produces a negative lateral force, so the rear and front cornering stiffness C_r and C_f , respectively, are negative. The front and rear tire slip angles, α_f and α_r , respectively, are defined as the angle between the heading and the velocity vector of the corresponding wheel. The velocity vectors of the rear and front wheel are given by \vec{v}_r and \vec{v}_f , respectively. The tire slip angles are calculated as:

$$\begin{aligned}\alpha_f &= \delta - \beta_f = \delta - \frac{v_y + \ell_f \dot{\psi}_h}{V_h}, \\ \alpha_r &= -\beta_r = -\frac{v_y - \ell_r \dot{\psi}_h}{V_h}.\end{aligned}\quad (3-3)$$

The linearization of the tire response limits the applicability of the model to vehicle accelerations lower than 0.5 g [8] and lateral slip angles lower than 5 degrees [5]. Outside these driving conditions, the linear tire model fails to predict the real tire behavior correctly. The nonlinear empirical Pacejka tire model, that uses tire measurements and identification algorithms, improves the accuracy of the model at high accelerations significantly [38]. In highway driving scenarios, only emergency maneuvers involving hard braking and sharp cornering require the use of a nonlinear model. Since these maneuvers are not considered here, the use of the simplified linear model is justified.

By using a small angle approximation for the vehicle body slip angle β , it is assumed that the lateral velocity is much smaller than the longitudinal velocity. Furthermore, the longitudinal velocity is assumed to be constant, which means that $\dot{v}_x = 0$, causing the dynamical model to only give reasonable approximations when the longitudinal velocity is changing slowly. These assumptions are justified when using the model in steady-state highway-driving scenarios.

The resulting linear vehicle model is given by:

$$\begin{bmatrix} \dot{v}_y \\ \ddot{\psi}_h \end{bmatrix} = \begin{bmatrix} \frac{C_f + C_r}{mv_x} & -v_x + \frac{\ell_f C_f - \ell_r C_r}{mv_x} \\ \frac{\ell_f C_f - \ell_r C_r}{I_z v_x} & \frac{\ell_f^2 C_f + \ell_r^2 C_r}{I_z v_x} \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi}_h \end{bmatrix} + \begin{bmatrix} \frac{-C_f}{m} \\ \frac{-\ell_f C_f}{I_z} \end{bmatrix} \delta. \quad (3-4)$$

The longitudinal velocity v_x , which is approximately equal to the host vehicle speed V_h , is a coefficient in the state matrix of the bicycle model. This vehicle speed is controlled by the vehicle powertrain which generates the rotational speeds of the wheel axles. Since this speed appears in the denominator of the fractions in the equations of motion, the bicycle model becomes invalid at very slow speeds [39]. In this research, it is assumed that the longitudinal velocity is constant.

The influence of using a motion controller based on the linearized bicycle model was evaluated in [37] by comparing the outputs of the nonlinear model to the outputs of the linear model. The simulation results of doing challenging lane-change maneuvers show small differences between the controller outputs. This means that classical control theory can be applied, since that is developed for linear systems.

3-3 Potential field functions

The concept of using APFs for path planning was first introduced by Oussama Khatib [40]. It was originally developed to move some of the slow, high-level planning tasks to the lower, faster level of control in robotics. This was done by directly coupling sensing to actuation, which leads to a fast and instinctive response to immediate hazards, by providing a level of restoring force corresponding to the gradient of the PF [16].

The literature on APFs reports several examples of applications to the control of passenger vehicles. In many cases, however, the vehicle dynamics were not incorporated. The vehicle was rather treated as a particle, thereby assuming the vehicle could perfectly track the resulting trajectories [16]. However, in doing so, the vehicle path became unstable in the high-speed region where the vehicle dynamics cannot be overlooked [41]. In practice, this has a negative impact on the tracking performance.

Analogy The effect of using APFs for path planning, control, and obstacle avoidance resembles the framework of using virtual bumpers around the vehicle for this purpose. An example to show what this looks like can be seen in Figure 3-5. Here the vehicle behaves as if imaginary springs and dampers are attached to its CoG [42]. Lane-keeping is achieved by having the other end of the spring attached to the lane center and obstacle avoidance by having a virtual bumper around the obstacle. If the vehicle were to deviate from the lane center, the potential hazard of the field increases. As a result, the controller will provide a restoring force towards the safe lane center in order to prevent the vehicle from departing the lane.

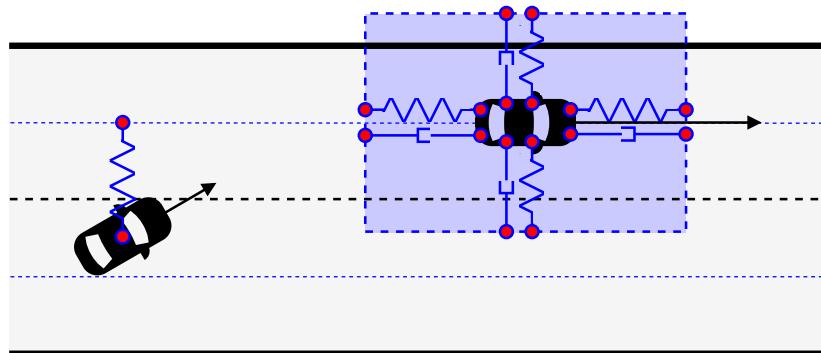


Figure 3-5: Spring and damper analogy for the APF method

Another analogy for the APF method is the principle of how an electron moves through a charged field, where it is being attracted by regions of positive charge and repulsed by negative charges. The virtual control force in these analogies is applied to the vehicle and can be realized by controlling the tire forces. The combination of longitudinal forces and steering can be represented as a virtual force, by creating an equivalent force system that consists of longitudinal and lateral forces on the vehicle [8].

The result is a simple and computationally favorable method. Furthermore, the method is capable of assigning different PF to different types of obstacles and road structures, which can be very useful. Nevertheless, several shortcomings have been identified as inherent problems of the method [43].

Drawbacks The approach suffers from getting trapped in local minima, that arise from the workspace configuration and the selection of weight coefficients for the different PFs [44]. Besides, the method is known to have the disadvantage of yielding oscillatory motion plans in the presence of multiple obstacles or in narrow passages, where the attractive forces cancel the repulsive ones when they are almost equal and collinear but in the opposite direction [45].

Luckily, these drawbacks mostly apply to robotics and do not occur for the approach in this thesis. Robots use the PF to reach a goal location, so getting stuck in a local minimum is a relevant problem. In this research, the APF is specifically constructed for lateral path planning and control, leaving the longitudinal control to the ACC system. Therefore, any minimum in the APF represents an acceptable driving corridor, namely the two lane centers. Whenever the road is entirely blocked by obstacles, the ACC controller makes sure the vehicle brakes.

A lateral controller that steers the vehicle along a fixed path typically exhibits large control effort. Besides, this approach ignores the path smoothness and the importance of strictly following the planned path. One of the benefits of using PFs is that this concept allows a driving corridor with some tolerance, that results in more smoothly steering and a much smaller control effort. This better approximates real driving, since small tracking errors are accepted by human drivers. Moreover, since obstacles can be naturally incorporated into the APF method there is no need for explicit collision avoidance trajectory planning and a connection between planning and control is provided. A controller based on APFs is able to assess how serious a tracking error is and to adapt its response accordingly [46].

One disadvantage of the APF method is that it usually complicates the analysis of the closed-loop control system [46]. This motivates the choice for a simple PF, like a quadratic. However, the gradient of a quadratic function, which is the driving force in the APF method, is equivalent to a proportional gain, which is not sufficient to stabilize the vehicle that can be approximated by a second order system.

Design requirements The individual PFs were designed such that each fulfills a particular role, after adding them together to construct the final APF. The mathematical functions of the PFs should possess the following properties:

- The PFs should have continuous gradient everywhere, to prevent a discontinuous change in the system dynamics;
- The PFs are constructed based on the information from the sensors;
- The shape of the PFs should be easily adjustable to account for changing traffic situations, like varying obstacle vehicle speeds and road curvatures; and
- The calculation of the PF must be efficient.

The mathematical definition of a general PF function is that of a real-valued non-negative function $V(x) : \mathbb{R}^m \rightarrow \mathbb{R}$, with $x \in \mathbb{R}^m$ the state of a dynamical system. The function $V(x)$ represents the artificial energy associated with the dynamical system, where the local minima correspond to desired equilibrium states of the system. As a rule of thumb, the boundaries of the PF function go to infinity when corresponding to restricted regions of the state space. Furthermore, the PF function usually becomes constant for states approaching an irrelevant region of the state space [47].

Morse potential field The following Morse PF function was proposed to be used in CACC by TNO to describe the performance of gap closing, vehicle following, and collision avoidance:

$$U_M(x_i) = k_1 \left(k_3 - e^{-k_2(x_i - c_e)} \right)^2, \quad (3-5)$$

with the following expression for c_e :

$$c_e = \frac{\log(k_3)}{k_2}, \quad (3-6)$$

and with:

$$x_i = e_{1,i} + c_d e_{2,i}, \quad (3-7)$$

where $e_{1,i}$ is the position error, $e_{2,i}$ is the velocity error of vehicle i , and c_d determines the amount of nonlinear damping [47]. Figure 3-6 shows the shape of PF U_M from Eq. (3-5) together with its Quadratic Approximation (QA) plotted for $k_1 = 0.3$, $k_2 = 0.042$, and $k_3 = 15.5$. The influence of the tunable parameters k_1 , k_2 , and k_3 on the shape of the Morse-like PF function can be derived from looking at the original Morse PF function. That function, often used in atom physics to describe inter-atomic interaction, is given by:

$$U_M(x_i) = a_M \left(1 - e^{-b_M(x_i - c_M)} \right)^2. \quad (3-8)$$

The a_M defines the depth of the dip, b_M governs the width of the potential (the smaller b_M , the wider the dip), and c_M equals the x_i where the function should be equal to zero. This global minimum of the function is located at the desired inter-vehicle distance for platooning. The gradient of the PF function (see Figure 3-7) is used to derive the nonlinear applied control gain. The nonlinear control law used for the acceleration input of vehicle i in the platoon is:

$$q_i = \frac{\partial U_M}{\partial x_i} + u_{i-1}, \quad (3-9)$$

with u_{i-1} the (delayed) acceleration of the preceding vehicle. The feedback control from Eq. (3-9) in combination with the PF from Eq. (3-5) achieves the desired vehicle behavior. When the vehicle i approaches the preceding vehicle $i - 1$, the acceleration is gradually increased in order to close the gap between the two vehicles. This gradient of the PF drives the vehicle towards its goal, acting as the attractive part of the function. At the desired inter-vehicle distance, automatic vehicle following, i.e. nominal CACC, should take place. When the controlled vehicle comes too close, the strongly increasing PF pushes back by applying a negative acceleration, which can be seen as the repulsive part of the function that aims at avoiding collisions. For gap closing, the APF controlled vehicle closes the gap less aggressively than the vehicle with the PD controller from TNO. This comes at the expense of a longer maneuver but with more comfortable accelerations. For vehicle following it is concluded that the APF strategy results in a performance equivalent to a PD strategy. For collision avoidance, the APF controlled vehicle shows better performance compared to the PD controlled vehicle. It keeps a safer distance to the leading vehicle and heavily decelerates in a shorter time [47].

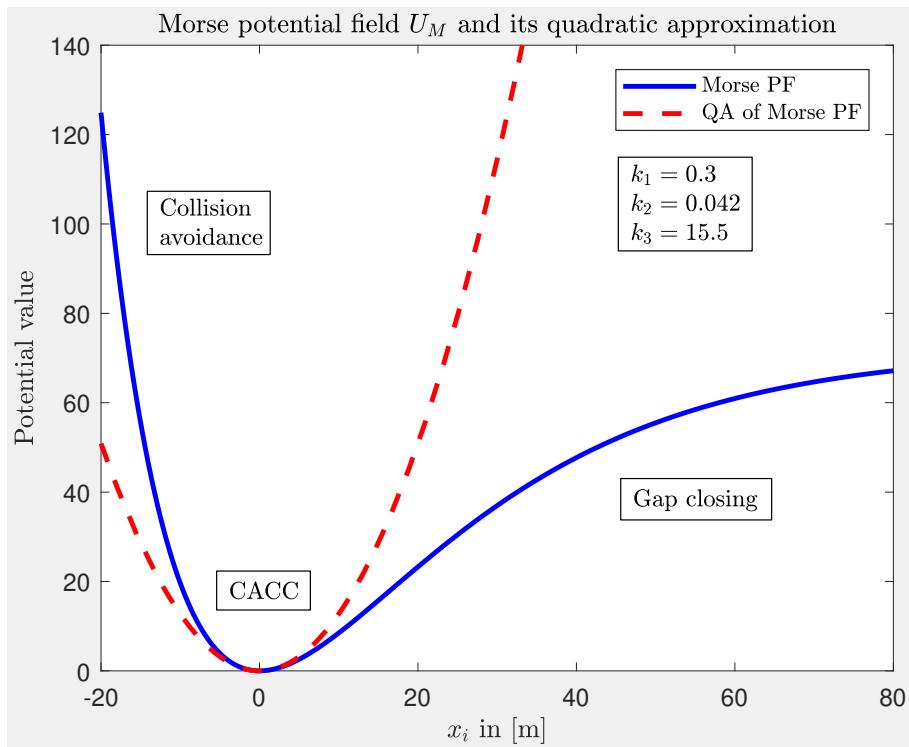


Figure 3-6: Morse PF U_M , with $k_1 = 0.3$, $k_2 = 0.042$, and $k_3 = 15.5$

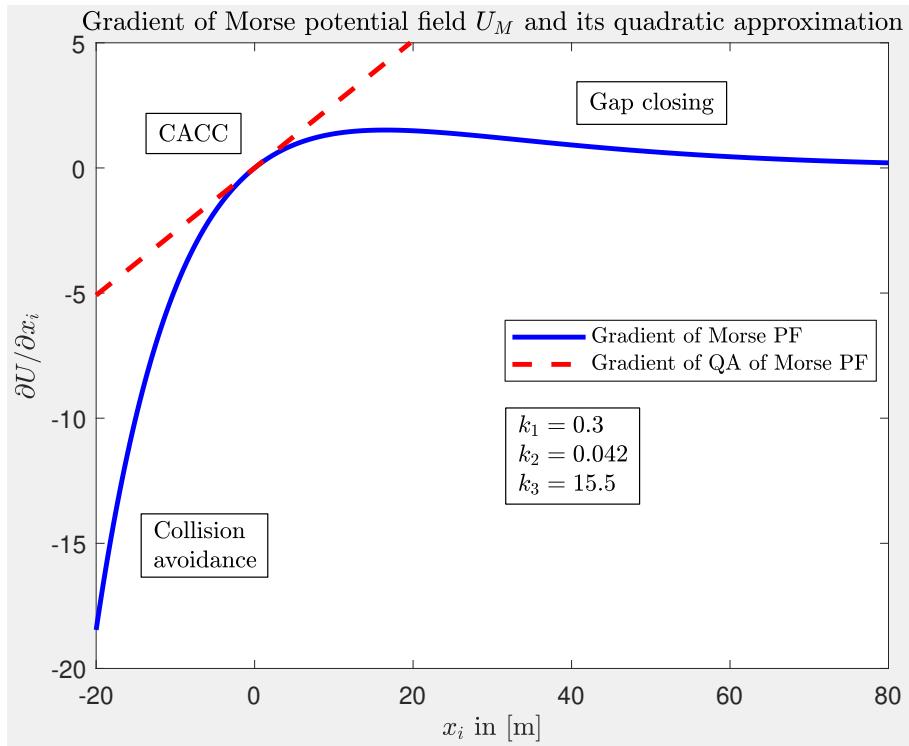


Figure 3-7: Gradient of Morse PF U_M , with $k_1 = 0.3$, $k_2 = 0.042$, and $k_3 = 15.5$

The Morse PF proves to be very suitable for designing a controller that integrates multiple nonlinear control objectives in a single mechanism. These results sparked the original interest of TNO to further investigate the use of APFs in autonomous driving. Furthermore, the described Morse PF was the inspiration for the design of the road PF function that will be treated in the next Section.

3-3-1 Road potential field

The road PF prevents the vehicle from departing the main road and driving too close to the borders, because of the related high accident risk. The road PF therefore has maximum value at the road boundaries. Additionally, the slope arriving at this maximum value is maximum as well, which provides a maximum restoring force, since driving there needs to be avoided at all cost. Furthermore, a resistance to changing lanes is incorporated by having a peak at the location of the lane divider line, which encourages the vehicle to stay in its current lane. However, the vehicle has to be able to overcome this resistance, in case changing lanes is necessary. This places a certain constraint on the height of the peaks. Furthermore, the road PF is equal to zero and locally symmetric at the lane center. This guides the vehicle towards the lane center, which is the preferred position in the absence of other traffic or obstacles.

To summarize: the PF goes to infinity at the locations of the solid lane markings, indicating the road boundaries, with a local maximum at the location of the (crossable) dashed lane marking and one local minimum related to the centerline of each of the two lanes. So the configuration of the road PF is fixed by basing it on the lane center measurements coming from the Mobileye sensor, which was described in Section 2-4.

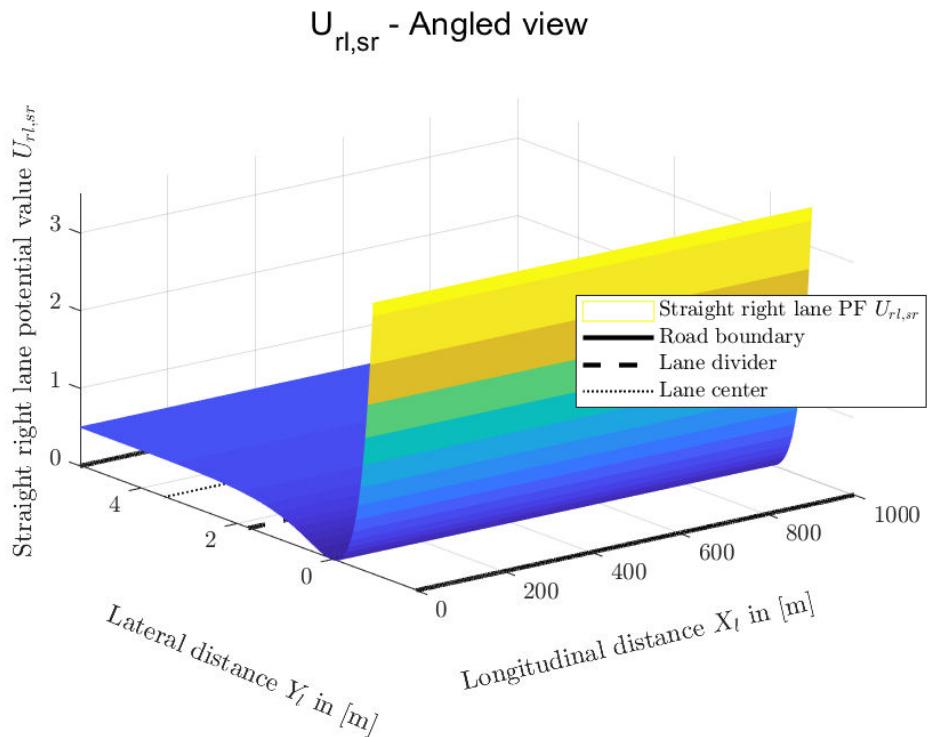
The mathematical representation of the road PF is expressed in the local lane frame and consists of the superposition of two lane PFs, one for each lane. A modified version of the Morse PF from Eq. (3-8) is used to represent the lane and road PFs. The function for the total road PF for straight roads is given by:

$$U_{r,sr} = \underbrace{A_r \left(1 - e^{-b_r(Y_h - Y_{rlc,sr})} \right)^2}_{\text{Right lane PF } U_{rl,sr}} + \underbrace{A_r \left(1 - e^{b_r(Y_h - Y_{llc,sr})} \right)^2}_{\text{Left lane PF } U_{ll,sr}}, \quad (3-10)$$

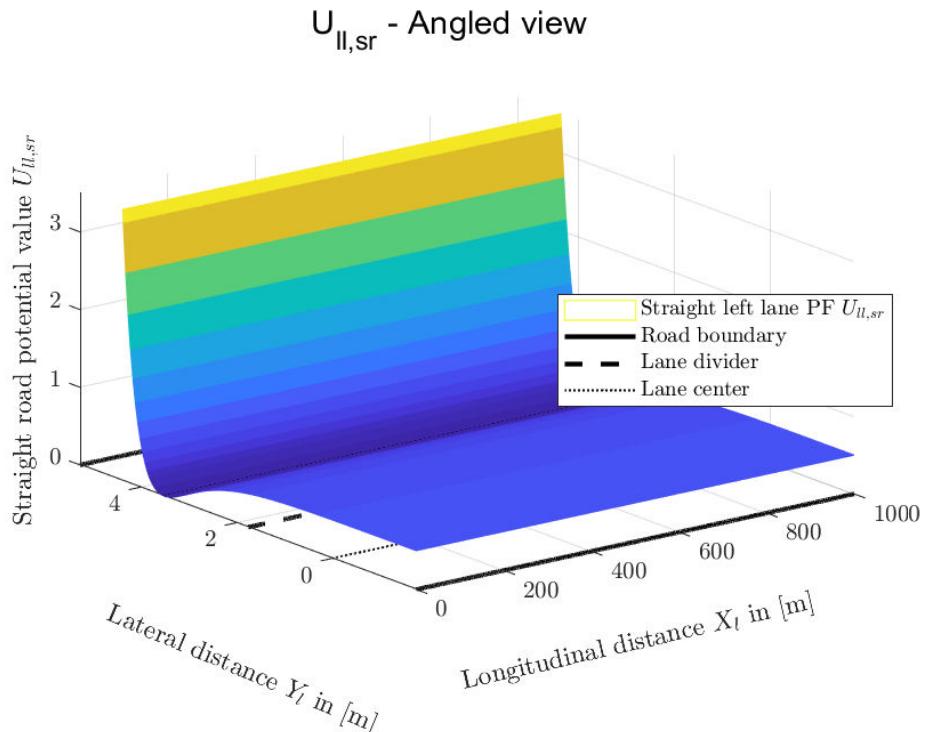
The only difference between the right and left lane PF, $U_{rl,sr}$ and $U_{ll,sr}$, respectively, is the minus sign in front of b_r and the different right and left lateral lane center positions:

$$\begin{aligned} Y_{rlc,sr} &= 0 \text{ m}, \\ Y_{llc,sr} &= 3.5 \text{ m}, \end{aligned} \quad (3-11)$$

respectively. Figure 3-8 shows the resulting 3D plots of the lane PF for a straight road, with the right and left lane from the angled view in Figure 3-8a and Figure 3-8b, respectively. Figure 3-9 shows the resulting 3D plots of the total straight road PF, from the diagonal and top view in Figure 3-9a and Figure 3-9b, respectively. The colors are a measure for the potential value of the functions, so the height of the PF, with dark blue corresponding to the lowest value and yellow corresponding to potential values of 5 and above. This was done to clearly show the shape around the lane divider. These Figures were created using $A_r = 0.5$, and $b_r = 1$. Table 3-1 shows an overview of the symbols from the straight road PF function.



(a) Angled view of right lane PF $U_{rl,sr}$ for a straight road



(b) Angled view of left lane PF $U_{ll,sr}$ for a straight road

Figure 3-8: Right and left lane potential fields $U_{rl,sr}$ and $U_{ll,sr}$ for a straight road, with $A_r = 0.5$ and $b_r = 1$

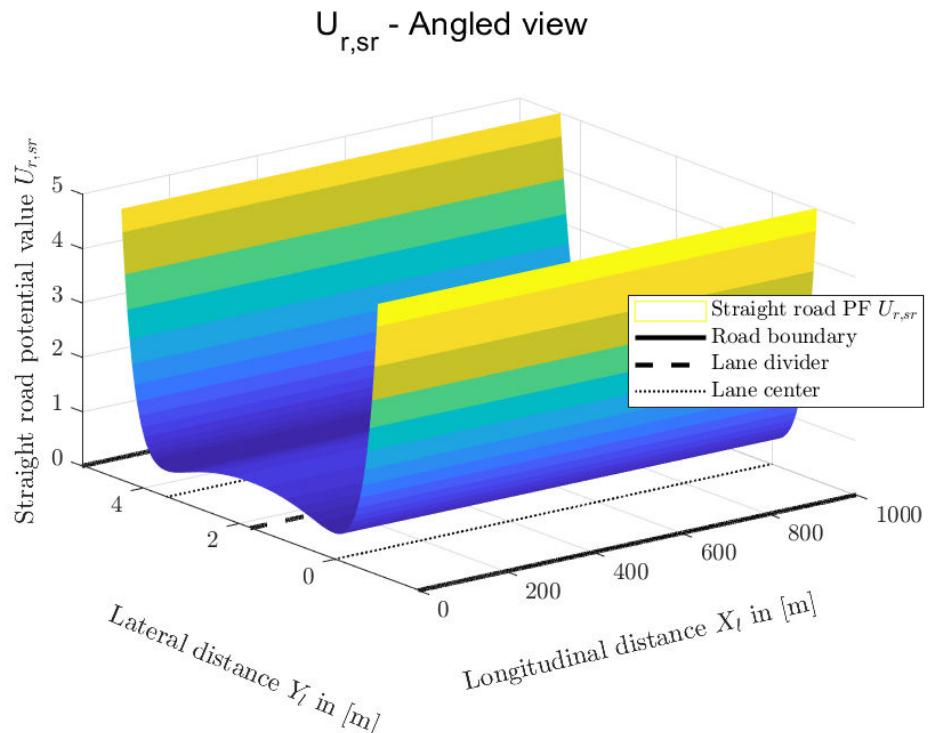
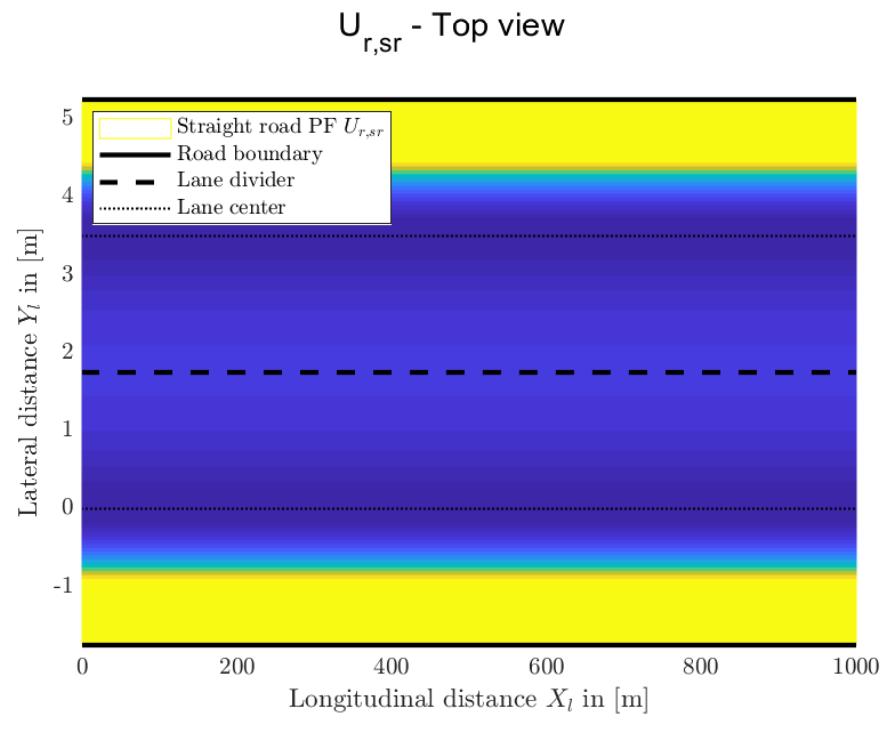
(a) Angled view of straight road PF $U_{r,sr}$ (b) Top view of straight road PF $U_{r,sr}$ **Figure 3-9:** Straight road potential field $U_{r,sr}$, with $A_r = 0.5$ and $b_r = 1$

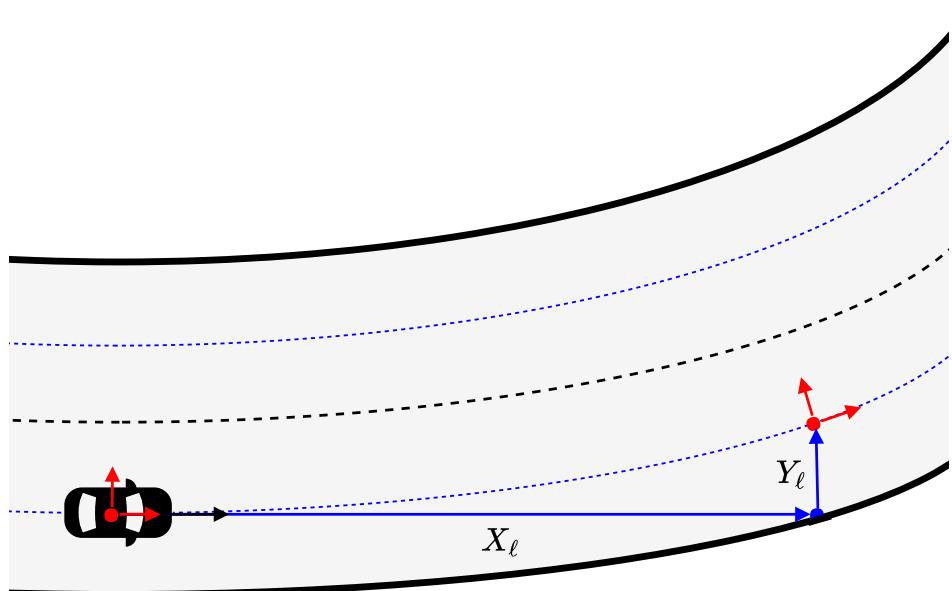
Table 3-1: Symbols from the straight road PF function

Description	Symbol	Value	Unit
Straight road PF value	$U_{r,sr}$	-	
Right lane straight road PF value	$U_{rl,sr}$	-	
Left lane straight road PF value	$U_{\ell\ell,sr}$	-	
Road PF depth	A_r	0.5	-
Parameter controlling road PF width	b_r	1	-
Longitudinal position host vehicle CoG in local road frame	X_h	[m]	
Lateral position host vehicle CoG in local road frame	Y_h	[m]	
Lateral position right lane center straight road	$Y_{rlc,sr}$	[m]	
Lateral position left lane center straight road	$Y_{\ell\ell c,sr}$	[m]	

The equation for the right lane PF for curved roads is different than the right lane PF equation for straight roads, $U_{rl,sr}$ from Eq. (3-10). It is given by the following equation:

$$U_{rl,cr} = A_r \left(\frac{-b_r \operatorname{sign}(Y_h - Y_{rlc,cr})}{1 - e^{-\left(\frac{Y_h - b_y}{m_y} - (X_h + \delta X_h) \right)^2 + (Y_{rlc,cr} - Y_h)^2}} \right)^2. \quad (3-12)$$

The construction of Eq. (3-12) is explained by using Figure 3-10, which shows the future local road frame at (X_l, Y_l) expressed in the current local road frame. The exponential term after b_r in Eq. (3-10) stands for the distance to the right lane center. This distance is also included in the exponential of Eq. (3-12), but now for the curved road.

**Figure 3-10:** Curved road PF computation

The distance to the right lane center of a curved road is calculated as:

$$\sqrt{\left(\frac{Y_h - b_y}{m_y} - (X_h + \delta X_h)\right)^2 + (Y_{rlc,cr} - Y_h)^2}. \quad (3-13)$$

Here, $Y_{rlc,cr}$ is the lateral position of the right lane center of the curved road, given by:

$$Y_{rlc,cr} = c_2 (X_h + \delta X_h)^2 + c_1 (X_h + \delta X_h) + c_{0,rlc}. \quad (3-14)$$

The following change of variable was made for numerical reasons:

$$X_h \rightarrow X_h + \delta X_h = X_{rlc,cr}. \quad (3-15)$$

The reason for this becomes clear from the equation for the slope of the line perpendicular to the lane center, which is equal to the y -axis of the future local road frame from Eq. (??):

$$m_y = -\frac{1}{2c_2 (X_h + \delta X_h) + c_1}. \quad (3-16)$$

It can be seen that the $X_h + \delta X_h$ term is located in the denominator of the fraction used in the calculation of m_y . The addition of δX_h is needed, because otherwise the outcome of m_y would go to infinity for $X_h = 0$. The y -intercept of the line perpendicular to the lane center is obtained as follows:

$$b_y = Y_{rlc,cr} - m_y (X_h + \delta X_h). \quad (3-17)$$

The first term of the first squared term from Eq. (3-13) is obtained by rewriting Eq. (3-17):

$$\frac{Y_{rlc,cr} - b_y}{m_y} = X_h + \delta X, \quad (3-18)$$

but then with Y_h instead of with $Y_{rlc,cr}$ to cover all the lateral positions Y_h . It should be noted that the c_1 -term drops out of Eq. (3-14) and Eq. (3-16), because the lateral positions of the curved lane centers are expressed in the local road frame, which is always aligned with the road, meaning that c_1 is always equal to zero. Finally, the sign $(Y_h - Y_{rlc,cr})$ term is added such that the right lane PF goes to infinity on one side of the right lane center $Y_{rlc,cr}$ and towards its asymptote on the other side. Finally, the total road PF is constructed from:

$$U_{r,cr} = U_{rl,cr} + U_{ll,cr}. \quad (3-19)$$

The left lane PF for curved roads $U_{ll,cr}$ can be obtained from Eq. (3-12) by substituting $-b_r$ for b_r and substituting $Y_{ll,cr}$ for $Y_{rl,cr}$, where $c_{0,llc}$ is used instead of $c_{0,rlc}$. Figure 3-11 shows the resulting 3D plots of the curved road PF, from the diagonal and top view in Figure 3-11a and Figure 3-11b, respectively. Yellow corresponds to potential values of 5 and above. This was done to clearly show the shape around the lane divider. These figures were created using $A_r = 0.5$, $b_r = 1$, $c_2 = 5e-6 \text{ m}^{-1}$, $c_1 = 0$, $c_{0,rrb} = 0 \text{ m}$, $c_{0,lrb} = 3.5 \text{ m}$, and $\delta X_h = 1.0e-10 \text{ m}$. Table 3-2 shows an overview of the symbols from the curved road PF function.

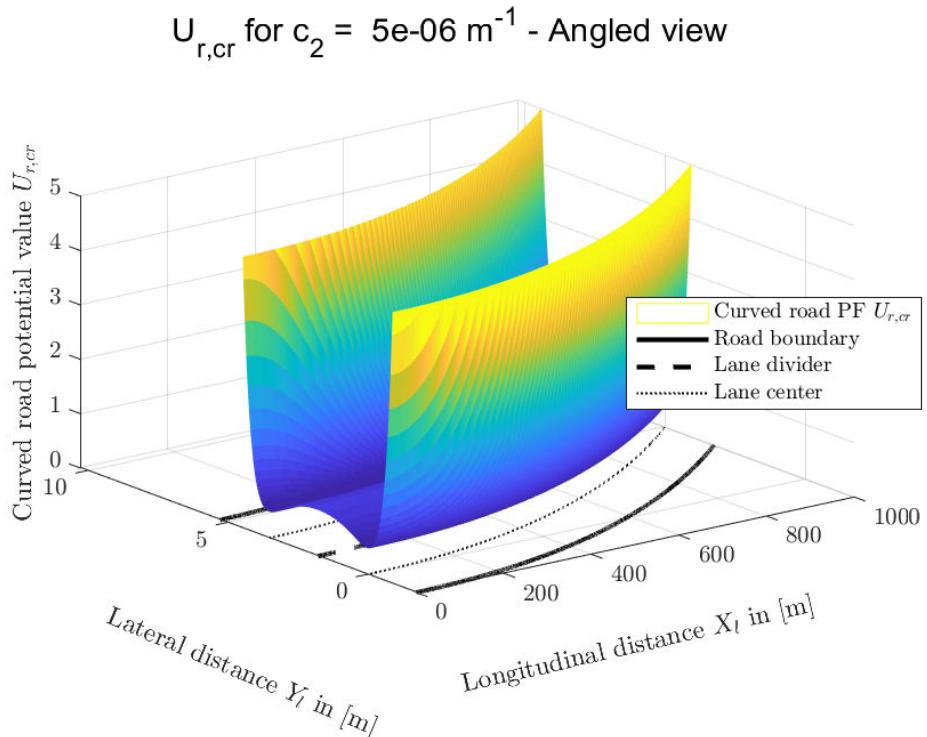
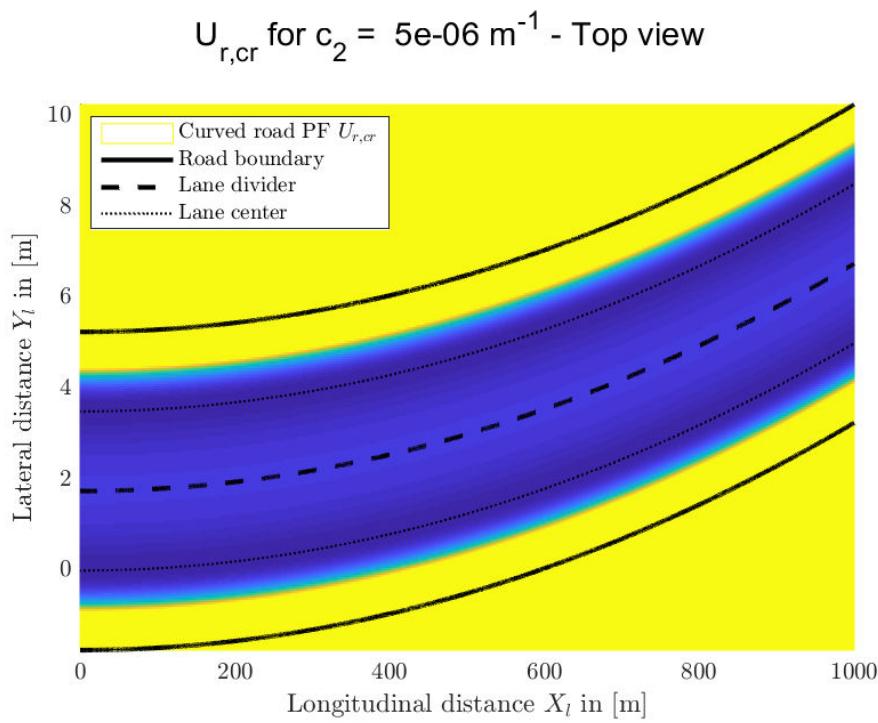
(a) Angled view of curved road PF $U_{r,cr}$ (b) Top view of curved road PF $U_{r,cr}$ **Figure 3-11:** Curved road potential field $U_{r,cr}$, with $A_r = 0.5$, $b_r = 1$, and $c_2 = 5e-06 \text{ m}^1$

Table 3-2: Symbols from the curved road PF function

Name	Symbol	Value	Unit
Right lane potential value of curved road	$U_{rl,cr}$	-	
Left lane potential value of curved road	$U_{ll,cr}$	-	
Curved road potential value	$U_{r,cr}$	-	
Lateral position of right lane center of curved road	$Y_{rlc,cr}$	[m]	
Lateral position of left lane center of curved road	$Y_{llc,cr}$	[m]	
Longitudinal position of right lane center of curved road	$X_{rlc,cr}$	[m]	
Infinitesimal small longitudinal offset	δX_h	1.0e-10	[m]
Slope of line normal to lane center	m_y	-	
y -intercept of line normal to lane center	b_y	-	[m]

3-3-2 Obstacle potential field

The construction of the obstacle PF U_o , but more specifically the tuning of its parameters, is identified as being very important and more complex than the construction of the road PF. The shape of the obstacle PF can namely be used to appropriately encourage the host vehicle to perform a lane-change maneuver if it is nearing in on the obstacle vehicle. This means that the obstacle PF can incorporate the structure and protocol of highway-driving [48]. The vehicle should change to the left in order to overtake slower preceding vehicles. So the obstacle PF should force the vehicle into a lane change whenever the distance to the obstacle vehicle becomes too small. This could also be employed to change lanes to the right in order to let faster trailing vehicles pass, but this is not necessarily desired behavior. To achieve this, the obstacle PF is built around the measured position of the obstacle vehicle. Furthermore, the obstacle PF parameters are a function of the relative and absolute velocity of the host and obstacle vehicle, and the road curvature. This will be explained in more detail in Section 4-3 that talks about how the obstacle PF was to tuned in order to obtain good performance. The position of the obstacle PF is based on the available sensor measurements of the obstacle. However, these measurements do not include the heading of the obstacle vehicle. The only information available is the longitudinal and lateral distance between the obstacle and the host vehicle expressed in the local vehicle frame, given by $X_{o,h}$ and $Y_{o,h}$, respectively, from Section 2-4. So an assumption has to be made regarding the heading of obstacle, which is done by taking the heading angle of the closest lane center. Additionally, if the obstacle vehicle has some deviation from the lane center, the location of the obstacle PF is translated towards that same closest point in one of the lane centers, as illustrated in Figure 3-12. This is done such that the host vehicle always approaches the obstacle PF in the same way, thereby guaranteeing predictable trajectories.

A rectangular shape is probably the best approximation of the outline of an obstacle vehicle. However, an ellipsoid is better suited to describe the repulsive field around any encountered vehicle, because it has a less complicated mathematical representation and still describes the vehicle shape well enough [11]. Furthermore, in order to avoid gradient discontinuities in the APF, continuous functions are needed to represent the potential values. Therefore, rectangles are unsuitable candidates and ellipses are chosen instead. It should be noted that close proximity maneuvers can probably not be performed with ellipses, since the approximated shape

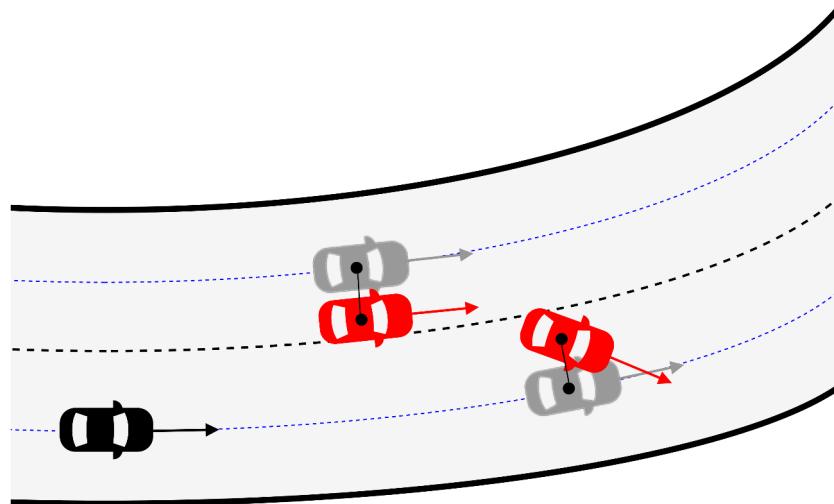


Figure 3-12: Obstacle projection to lane center

lacks accuracy, compared to the true vehicle shape. A two-dimensional Gaussian function, with the longitudinal and lateral coordinates as its variables, is proposed to fulfill the function of the obstacle PF. The power to which the exponential of the Gaussian function is raised is any negative-definite quadratic form. As a consequence, its level sets are ellipses. However, it is not a convex function, since the used Gaussian is pointing upwards. This means that the potential value of the obstacle PF increases exponentially when the distances to the obstacle decreases. When this happens, the optimization will try to find a solution that keeps the vehicle away from the obstacle. The size of the Gaussian function can easily be extended by increasing its standard deviation in either direction of its principal axes. This can be used to model the prediction uncertainty of the moving obstacle. The Gaussian equation for the obstacle PF is given by:

$$U_o = A_o e^{-\left(a_\psi (X_h - X_o)^2 + 2b_\psi (X_h - X_o)(Y_h - Y_o) + c_\psi (Y_h - Y_o)^2\right)}, \quad (3-20)$$

with the rotation parameters defined as follows:

$$\begin{aligned} a_\psi &= \frac{\cos^2 \psi_o}{2\sigma_x^2} + \frac{\sin^2 \psi_o}{2\sigma_y^2}, \\ b_\psi &= -\left(-\frac{\sin(2\psi_o)}{4\sigma_x^2} + \frac{\sin(2\psi_o)}{4\sigma_y^2}\right), \\ c_\psi &= \frac{\sin^2 \psi_o}{2\sigma_x^2} + \frac{\cos^2 \psi_o}{2\sigma_y^2}, \end{aligned} \quad (3-21)$$

and the obstacle vehicle heading angle:

$$\psi_o = \arctan(2c_2 X_o + c_1), \quad (3-22)$$

where the argument of the arctan-function in the last line of Eq. (3-22) is obtained from:

$$\left. \frac{dY_{\ell c}}{dX_\ell} \right|_{X_\ell=X_o} = 2c_2 X_o + c_1, \quad (3-23)$$

which is the derivative of the lateral position $Y_{\ell c,l}$ of the lane center of the current lane, in which the obstacle vehicle is driving, from Eq. (2-2) with respect to X_ℓ , evaluated at $X_\ell = X_o$. Now, the derivation of the longitudinal and lateral standard deviations, σ_x and σ_y , respectively, is given. These are computed from the Gaussian equation without rotation:

$$f(X, Y) = A_o \exp \left(- \left(\frac{(X - X_o)^2}{2\sigma_x^2} + \frac{(Y - Y_o)^2}{2\sigma_y^2} \right) \right) \quad (3-24)$$

Substituting $Y = Y_o$ and into Eq. (3-24) gives the following expression:

$$f(X) = A_o \exp \left(- \frac{(X - X_o)^2}{2\sigma_x^2} \right). \quad (3-25)$$

After that, the next change of variables is applied $x_\sigma = X - X_o$. Furthermore $f(X)$ is set to 0.01. Rewriting Eq. (3-25) gives the expression for σ_x :

$$\sigma_x = \sqrt{-\frac{x_\sigma^2}{2 \ln(0.01/A_o)}}. \quad (3-26)$$

The expression for σ_y is obtained similarly. In the tuning phase it was discovered that x_σ is a function of the host vehicle velocity V_h , the velocity difference between the obstacle and host vehicle $\Delta V_{o,h}$, and the curvature c_2 . The y_σ is chosen as $y_\sigma = 3/4 l_w$, with l_w the lane width of 3.5 m. This is done such that the obstacle PF has some overlap with the next lane if the obstacle vehicle is driving in the lane center, but not too much. The host vehicle should namely not be influenced by the obstacle PF from vehicles passing in the next lane.

Table 3-3 shows an overview of the symbols from the obstacle road PF function. Figure 3-13 shows the resulting 3D plots of the obstacle road PF, from the diagonal and top view in Figure 3-13a and Figure 3-13b, respectively. These figures were created using $A_o = 1$, $(X_o, Y_o) = (375, 0)$ m, $x_\sigma = 375$ m, and $\sigma_y = 0.865$ m. Table 3-4 shows an overview of the symbols from the obstacle road PF rotation coefficients.

Table 3-3: Symbols from the obstacle PF function

Description	Symbol	Value	Unit
Obstacle obstacle potential value	U_o		-
Obstacle PF peak	A_o	1	-
First obstacle PF rotation coefficient	a_ψ		-
Second obstacle PF rotation coefficient	b_ψ		-
Third obstacle PF rotation coefficient	c_ψ		-
Longitudinal position obstacle vehicle CoG in local road frame	X_o		[m]
Lateral position obstacle vehicle CoG in local road frame	Y_o		[m]

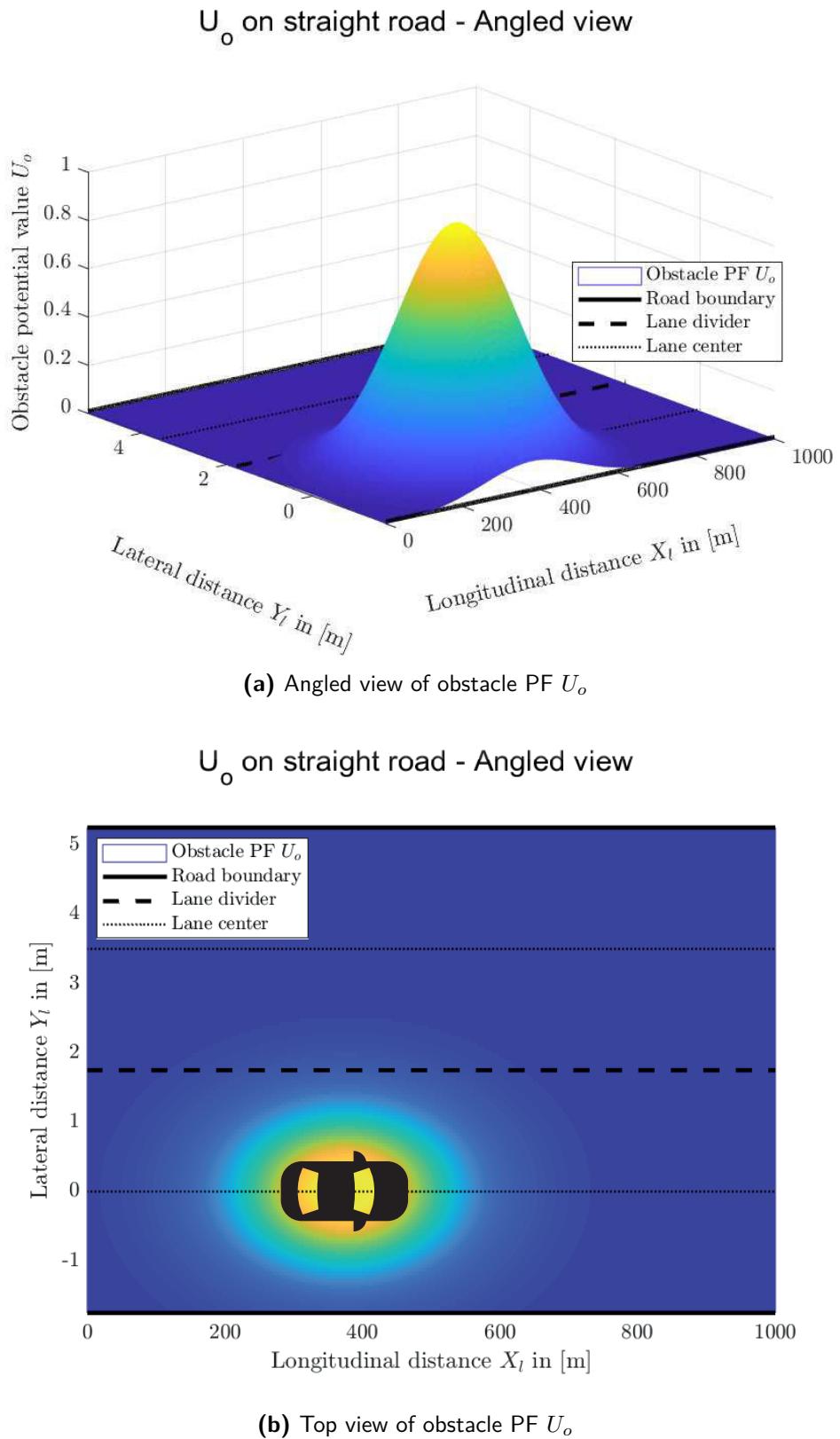


Figure 3-13: Obstacle potential field U_o , with $A_o = 1$, $(X_o, Y_o) = (375, 0)$ m, $x_\sigma = 375$ m, and $\sigma_y = 0.865$ m

Table 3-4: Symbols from the obstacle PF rotation coefficients

Description	Symbol	Value	Unit
Heading angle obstacle in local road frame	ψ_o		[rad]
Parameter controlling obstacle PF σ_x	x_σ		[m]
Obstacle PF longitudinal standard deviation	σ_x		[m]
Lane width	ℓ_w	3.5	[m]
Parameter controlling obstacle PF σ_y	y_σ	2.625	[m]
Obstacle PF lateral standard deviation	σ_y	0.865	[m]

3-3-3 Artificial potential field

The superposition principle of the individual PFs allows for the design of various assistance systems that can easily be integrated with the APF-method [16]. This is also the case with the presented design. The final APF is namely constructed from the superposition of the road PF and the obstacle PF, that take care of lane keeping and road staying, and vehicle avoidance and passing, U_r and U_o , respectively:

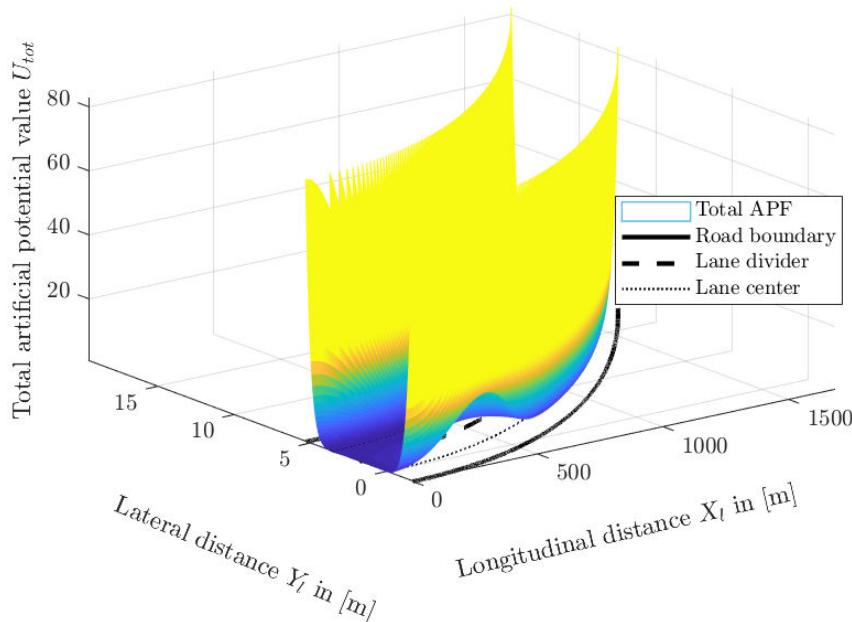
$$U_{tot} = \lambda_r U_r + \lambda_o U_o \quad (3-27)$$

Figure 3-14 shows the resulting 3D plots of the total APF on a curved road with two obstacles, from the diagonal and top view in Figure 3-14a and Figure 3-14b, respectively. Yellow corresponds to potential values of 25 and above. This was done to clearly show the shape around the lane divider. These figures were created using the same values of the previously illustrated PFs. The positions of the two obstacle PF are given by $(X_{o,1}, Y_{o,1}) = (375, 0.7031)$ m and $(X_{o,2}, Y_{o,2}) = (775, 6.5031)$ m, for the first and second obstacle vehicle, respectively. The lateral obstacle positions Y_o were obtained from:

$$Y_o = c_2 X_o^2 + c_1 X_o + c_0. \quad (3-28)$$

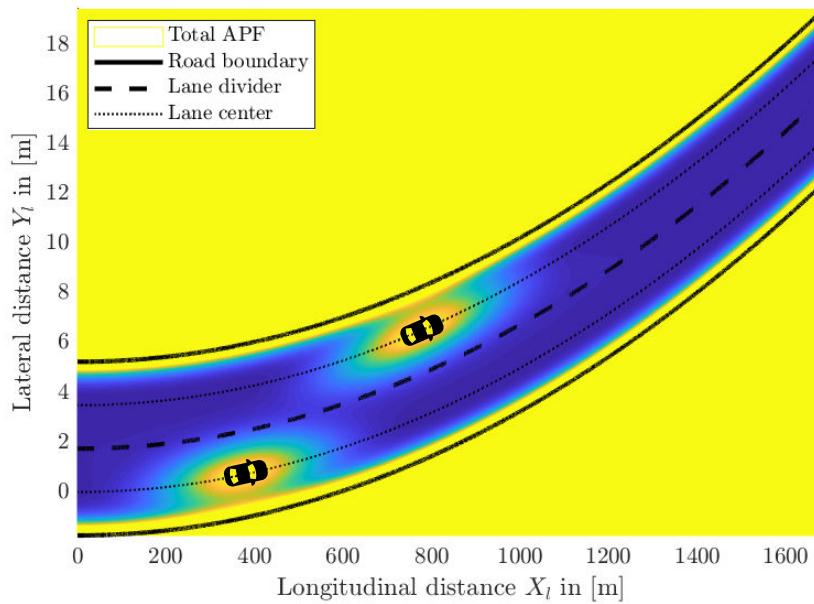
It should be noted from Eq. (3-27) that both PFs from Figure 3-14 are plotted by being scaled with their corresponding weights. As a result, the PF values come to lie in the same range, which is an important instrument in the tuning process of the PF weights, which will be addressed in Section 4-3. The values used for the road and obstacle PF weights are $\lambda_r = 1$ and $\lambda_o = 25$, respectively.

Total APF on curved road with $c_2 = 5e-06 \text{ m}^{-1}$ - Angled view



(a) Angled view of total APF

Total APF on curved road with $c_2 = 5e-06 \text{ m}^{-1}$ - Top view



(b) Top view of total APF

Figure 3-14: Total artificial potential field of curved road with two obstacles, with $\lambda_r = 1$, $A_r = 0.5$, $b_r = 1$, $c_2 = 5e-06 \text{ m}^{-1}$, $\lambda_o = 25$, $A_o = 1$, $(X_{o,1}, Y_{o,1}) = (375, 0.7031) \text{ m}$, $(X_{o,2}, Y_{o,2}) = (775, 6.5031) \text{ m}$, $x_\sigma = 375 \text{ m}$, and $\sigma_y = 0.865 \text{ m}$

3-4 Combining APF with MPC

Model discretization The continuous vehicle model needs to be discretized to obtain a discrete-time model that can be implemented into the MPC-controller. Proper discretization influences the accuracy of the predicted motion. The Zero-Order Hold (ZOH) method assumes the input is constant between subsequent time steps, which typically matches lower-level controllers [20]. This method works well for short time steps, since the constant input signal will then accurately reflect the actual input [49]. The First-Order Hold (FOH) discretization method assumes linear variation between two time steps, which is a more accurate interpolation that gives better results for long time steps. The sampling time should be chosen small enough to accurately capture the system dynamics. However, short time steps in combination with a long prediction horizon can make it difficult to solve the optimization in real-time.

The sampling time and prediction horizon chosen for this MPC-problem are 0.1s and 10 steps, respectively, yielding a 1s prediction horizon.

To have constant throttle and brake input, and therefore constant vehicle longitudinal speed, over the prediction horizon of 1s is moderately reasonable, since the longitudinal dynamics typically have bandwidth less than 1 - 2 Hz [50]. Using a sampling time of 0.1s seems valid, since the total reaction time of a driver varies from 0.5s to 4s, as was explained in Section 2-2 with the PIEV theory [25]. Consequently, the update rate of the controller is equal to the MPC sampling time of 10 Hz, which means that any changes in the states, speed, and input are taken into account within 0.1s. This sampling time leads to good handling of these slowly varying parameters [50].

The prediction horizon was set by using two guide-lines. The first guideline is that the vehicle dynamics are fast enough to capture most of the initial vehicle response in the prediction horizon of 1s [50]. The second guideline is that the prediction horizon cannot be made arbitrarily large, since the prediction errors grow as the model is extended into the future. Because of the simplified bicycle model, the MPC-predictions are most accurate at the beginning of the horizon and less accurate when the states move away from the initial conditions.

The bicycle model given by Eq. (3-4) from Section 3-2 is appended with some extra states, namely \ddot{X}_h , \dot{X}_h and ψ_h . This results in the following continuous-time state equation:

$$\underbrace{\begin{bmatrix} \ddot{X}_h \\ \dot{X}_h \\ \dot{Y}_h \\ \dot{v}_y \\ \ddot{\psi}_h \\ \dot{\psi}_h \end{bmatrix}}_x = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & V_h \\ 0 & 0 & 0 & A_{11} & A_{12} & 0 \\ 0 & 0 & 0 & A_{21} & A_{22} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}_{A_c} \underbrace{\begin{bmatrix} \dot{X}_h \\ X_h \\ Y_h \\ v_y \\ \dot{\psi}_h \\ \psi_h \end{bmatrix}}_x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ B_1 \\ B_2 \\ 0 \end{bmatrix}}_{B_c} u. \quad (3-29)$$

The bicycle model can be recognized to be included in the state matrix A_c from Eq. (3-29) as follows:

$$\begin{bmatrix} \dot{v}_y \\ \ddot{\psi}_h \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi}_h \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u, \quad (3-30)$$

with the entries of the bicycle model state matrix given by:

$$\begin{aligned} A_{11} &= \frac{C_f + C_r}{mV_h}, \\ A_{12} &= -V_h + \frac{\ell_f C_f - \ell_r C_r}{mV_h}, \\ A_{21} &= \frac{\ell_f C_f - \ell_r C_r}{I_z V_h}, \\ A_{22} &= \frac{\ell_f^2 C_f + \ell_r^2 C_r}{I_z V_h}, \end{aligned} \quad (3-31)$$

and the entries of the bicycle model input matrix by:

$$\begin{aligned} B_1 &= \frac{-C_f}{m}, \\ B_2 &= \frac{-\ell_f C_f}{I_z}. \end{aligned} \quad (3-32)$$

The extra states from Eq. (3-29) on top of the bicycle model states v_y and $\dot{\psi}_h$ were added to incorporate the kinematic equations of the vehicle into the state-space equation. These kinematic constraints are represented by the next equations:

$$\begin{aligned} \ddot{X}_h &= 0; \\ \dot{X}_h &= V_h \cos(\psi_h + \beta) \approx V_h = \dot{X}_h, \\ \dot{Y}_h &= V_h \sin(\psi_h + \beta) \approx V_h \psi_h, \\ \dot{\psi}_h &= \dot{\psi}_h, \end{aligned} \quad (3-33)$$

where the small angle approximation is used in the kinematic expressions for \dot{X}_h and \dot{Y}_h :

$$\begin{aligned} \cos(\psi_h + \beta) &\approx 1, \\ \sin(\psi_h + \beta) &\approx \psi_h. \end{aligned} \quad (3-34)$$

The first line of Eq. (3-33) indicates that the host vehicle has zero longitudinal acceleration, because it is driving at constant speed. This constant longitudinal velocity is given by \dot{X}_h , which is simply selected from the state vector x , as can be seen from the second line of Eq. (3-33). It should be noted that the fact that \dot{X}_h is approximately equal to V is reflected in the initial state of \dot{X}_h and not in the state matrix of Eq. (3-29). Furthermore, the expression for \dot{Y}_h is a simple trigonometric relation and $\dot{\psi}_h$ is selected from x , just as \dot{X}_h . It may not appear to be very meaningful to have $\dot{X}_h = \dot{X}_h$ and $\dot{\psi}_h = \dot{\psi}_h$ in the state-space equation. However, these variables are definitely needed in order to include X_h and ψ_h , the latter of which is needed for the expression of \dot{Y}_h and therefore Y_h . Having the longitudinal and lateral host vehicle position, X_h and Y_h , respectively, in the state vector will be necessary to calculate the corresponding PF values. The road and obstacle PFs that were designed in Section 3-3 are namely functions of the new host vehicle state vector entries X_h and Y_h .

The related continuous-time output equation is defined as:

$$\underbrace{\begin{bmatrix} v_y \\ \dot{\psi}_h \\ \psi_h \end{bmatrix}}_y = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{C_c} \underbrace{\begin{bmatrix} \dot{X}_h \\ X_h \\ Y_h \\ v_y \\ \dot{\psi}_h \\ \psi_h \end{bmatrix}}_x, \quad (3-35)$$

where v_y , $\dot{\psi}_h$ and ψ_h were chosen as the output variables, because these variables will have reference values assigned to them in the eventual MPC cost function formulation. Next, the discrete-time matrices A_d , B_d , and C_d are determined by applying the ZOH discretization method on their continuous-time counterparts A_c , B_c , and C_c , respectively, with a sampling time of 0.1s. The discrete-time state-space matrices are calculated as follows:

$$\begin{aligned} A_d &= e^{A_c t_h}, \\ B_d &= \int_0^{t_h} e^{A_c \tau} d\tau \cdot B_c, \\ C_d &= C. \end{aligned} \quad (3-36)$$

For a host vehicle speed of $V_h = 130$ km/h and a sampling time of $t_h = 0.1$ s this gives:

$$\underbrace{\begin{bmatrix} v_X(k+1) \\ X_h(k+1) \\ Y_h(k+1) \\ v_y(k+1) \\ r(k+1) \\ \psi_h(k+1) \end{bmatrix}}_{x(k+1)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.0089 & 0.1423 & 3.6111 \\ 0 & 0 & 0 & 0.4234 & -1.6777 & 0 \\ 0 & 0 & 0 & 0.1027 & 0.3736 & 0 \\ 0 & 0 & 0 & 0.0066 & 0.0682 & 1 \end{bmatrix}}_{A_d} \underbrace{\begin{bmatrix} v_X(k) \\ X_h(k) \\ Y_h(k) \\ v_y(k) \\ r(k) \\ \psi_h(k) \end{bmatrix}}_{x(k)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0.2071 \\ 0.2133 \\ 2.9964 \\ 0.1649 \end{bmatrix}}_{B_d} u(k), \quad (3-37)$$

where the notation for \dot{X}_h and $\dot{\psi}_h$ is changed to their equivalent discrete-time counterparts v_X and r , respectively. The used values for the host vehicle parameters, which correspond to the values of the available test vehicle at TNO, are found in Table 3-5. The following statements can be made about the extended bicycle model from Eq. (3-37):

- The discrete-time open-loop state matrix A_d has five poles: two poles from the vehicle dynamics, $\lambda_{1,2} = 0.3985 + 0.4143i$ and three poles from the added kinematic equations, $\lambda_{3,4,5} = 1$, showing that the vehicle is stable;
- The poles related to the vehicle dynamics change with changing longitudinal velocity, making it a linear parameter-varying system;
- The system (A_c, B_c) is controllable, since the corresponding controllability matrix has full rank; and
- The matrices A_c and B_c depend on the vehicle mass, inertia, and axle cornering stiffness, that change with load, tire wear, and road and environmental conditions, etc.

Table 3-5: Parameters of the host vehicle

Name	Symbol	Value	Unit
Vehicle mass	m	1625	[kg]
Vehicle yaw moment of inertia	I_z	2865.61	[kgm ²]
Distance between front axle and	ℓ_f	1.108	[m]
Distance between rear axle and vehicle CoG	ℓ_r	1.592	[m]
Wheel base	L_{wb}	1.700	[m]
Negative front tire cornering stiffness	C_f	-98 389	[Nrad ⁻¹]
Negative rear tire cornering stiffness	C_r	-198 142	[Nrad ⁻¹]

Cost function The discretized model from Eq. (3-36) is used to construct the MPC-controller, which consists of a cost function that defines an optimization problem of conflicting demands. It is solved by evaluating the future dynamics over a prediction horizon. Unfortunately, due to the increase in dimension with every time step, the method is limited to small-order systems. In order to obtain efficient solutions, the MPC-problem is formulated as a quadratic, convex cost function with linear time-varying constraints.

By ensuring the optimization is a convex programming problem, standard QP-solvers can be used. A convex environment is analogous to a marble inside a bowl, which will always settle at the bottom of the bowl, which can be seen as the global minimum of the convex optimization problem. Numerical approaches for solving nonlinear and non-convex MPC-problems are available, but these techniques are not supported by the standard QP-solvers [51]. These solvers do not need any iterations or initial starting guess, so only a single QP-problem needs to be solved at each time step [18]. This reduces the speed of the optimization dramatically, allowing fast solutions with little computational effort and efficient real-time implementation. These properties make MPC a highly favorable path-planning method with high-frequency re-planning capabilities.

As always, special attention needs to be paid to the size of the sampling time. If the sampling time is too short, the computational power may limit the use of a long prediction horizon, since it will involve too many time steps. Using a small prediction horizon instead, fails to effectively incorporate future information. A too large sampling time, on the other hand, will not capture all of the relevant vehicle dynamics.

The cost function can easily be extended by adding different performance qualifying terms. In this research, it embodies the trade-off between tracking the reference values, ensuring vehicle stability, and avoiding collisions. This means that the optimization is aimed at minimizing the tracking error between future desired and predicted outputs, the inputs, and the input increments. The input increments stand for the rate of change of the steering angle, which is connected to the lateral stability of the vehicle. Abrupt transitions in the generated trajectory lead to rapidly varying steering angles, which may negatively affect the stability [52]. Therefore, it is important to have smooth paths. Weighting factors are introduced to prioritize in the multi-objective cost function, since the cost function objectives are generally conflicting.

The functionality of obstacle avoidance can be included into the MPC-framework in two

different ways [53]. The first is to use additional constraints to the state values that define the safe distance that should be kept with respect to the obstacle. The second way is to simply change the reference trajectory to overtake the obstacle. The approach used in this thesis, is to modify the cost function, by using an additional cost associated to obstacle proximity.

In order to incorporate the contribution of the PFs into the cost function, while keeping the optimization problem quadratic and convex, the nonlinear PFs are converted into quadratic functions by using the second order Taylor approximation. Appendix B contains the mathematical background of this method and shows some figures that relate to the consequences of this approximation. Although the QA of the PFs increases the calculation time, the added time is negligible compared to the time needed for solving a nonlinear optimization problem.

The cost function of the optimization problem over the prediction horizon N_p is given by:

$$\begin{aligned} J(k) = & \sum_{i=1}^{N_p} \underbrace{(y(k+i|k) - y_{\text{ref}}(k+i|k))^T \lambda_y (y(k+i|k) - y_{\text{ref}}(k+i|k))}_{\text{Vehicle dynamics}} + \\ & \underbrace{\lambda_u (u(k+i-1|k) - u_{\text{ref}}(k+i-1|k))^2}_{\text{Input}} + \underbrace{\lambda_{\Delta} \Delta u(k+i-1|k)^2}_{\text{Input increment}} + \\ & \underbrace{\lambda_r U_r(k+1|k)}_{\text{Road potential}} + \underbrace{\lambda_o U_o(k+1|k)}_{\text{Obstacle potential}}, \end{aligned} \quad (3-38)$$

with the following output weight matrix:

$$\lambda_y = \begin{bmatrix} \lambda_{v_y} & 0 & 0 \\ 0 & \lambda_{\dot{\psi}} & 0 \\ 0 & 0 & \lambda_{\psi} \end{bmatrix}. \quad (3-39)$$

The notation $y(k+i|k)$ from Eq. (3-38) is used to indicate the output at future time $k+i$ predicted at current time k . The scalar cost function from Eq. (3-38) has the following equivalent vector notation:

$$\begin{aligned} J(k) = & \underbrace{(\tilde{y} - \tilde{y}_{\text{ref}})^T \lambda_y (\tilde{y} - \tilde{y}_{\text{ref}})}_{\text{Vehicle dynamics}} + \underbrace{(\tilde{u} - \tilde{u}_{\text{ref}})^T \lambda_u (\tilde{u} - \tilde{u}_{\text{ref}})}_{\text{Input}} + \\ & \underbrace{\Delta \tilde{u}^T \lambda_{\Delta} \Delta \tilde{u}}_{\text{Input increment}} + \underbrace{\lambda_r \tilde{U}_r}_{\text{Road potential}} + \underbrace{\lambda_o \tilde{U}_o}_{\text{Obstacle potential}}. \end{aligned} \quad (3-40)$$

The terms with a \sim on top of them are vectors containing predictions of their future values, with the exception of \tilde{U}_r and \tilde{U}_o , which are scalar terms consisting of the summed predicted potential values of the road and obstacle PF, respectively. The standard formulation of the QP-problem is given by:

$$\min_{\tilde{u}} J = \min_{\tilde{u}} \frac{1}{2} \tilde{u}^T H \tilde{u} + c^T \tilde{u} \text{ subject to } \tilde{A}_{\text{ineq}} \tilde{u} \geq \tilde{b}_{\text{ineq}}, \quad (3-41)$$

with the Hessian matrix H and the linear vector c of the cost function. Since there are no equality constraints, the \tilde{A}_{eq} and \tilde{b}_{eq} are defined to be appropriately dimensioned matrices

Table 3-6: Symbols from the cost function

Description	Symbol	Value	Unit
Cost function value	J	-	-
Discrete time instance	k	-	-
Summation index	i	-	-
Prediction horizon	N_p	10	-
Output vector	y	-	-
Input	u	[rad]	[rad]
Input increment	Δu	-	-
Output weight	λ_y	-	-
Lateral velocity weight	λ_{v_y}	0	$[(\text{ms}^{-1})^{-1}]$
Yaw rate weight	$\lambda_{\dot{\psi}}$	1000	$[(\text{rads}^{-1})^{-1}]$
Heading angle weight	λ_{ψ}	1000	$[\text{rad}^{-1}]$
Input weight	λ_u	10	$[\text{rad}^{-1}]$
Input increment weight	λ_{Δ}	10	$[\text{rad}^{-1}]$
Road PF weight	λ_r	1	-
Obstacle PF weight	λ_o	-	-

equal to zero. Table 3-6 shows an overview of the symbols from the cost function and some fixed values, that were already determined or obtained in the tuning phase.

Appendix C shows how Eq. (3-40) is written out in detail. The quadratic terms resulting from that are grouped under H and the linear terms under c :

$$\begin{aligned}
 H = & \underbrace{2 \left(\tilde{C} \tilde{B} \right)^T \lambda_y \left(\tilde{C} \tilde{B} \right)}_{\text{Vehicle dynamics}} + \underbrace{2 \lambda_u I_{N_p}}_{\text{Input}} + \underbrace{2 \tilde{A}_u^T \lambda_{\Delta} \tilde{A}_u}_{\text{Input increment}} + \\
 & \underbrace{\lambda_r \left(\tilde{C}_{XY} \tilde{B} \right)^T \tilde{U}_{r,2} \left(\tilde{C}_{XY} \tilde{B} \right)}_{\text{Road potential}} + \underbrace{\lambda_o \left(\tilde{C}_{XY} \tilde{B} \right)^T \tilde{U}_{o,2} \left(\tilde{C}_{XY} \tilde{B} \right)}_{\text{Obstacle potential}}, \\
 c = & \underbrace{2 \left(\tilde{C} \tilde{A}x(k) - \tilde{y}_{\text{ref}} \right)^T \lambda_y \left(\tilde{C} \tilde{B} \right)}_{\text{Vehicle dynamics}} - \underbrace{2 \lambda_u \tilde{u}_{\text{ref}}^T}_{\text{Input}} + \\
 & \underbrace{\lambda_r \left(\left(\tilde{U}_{r,1} + \left(\tilde{C}_{XY} \tilde{A}x(k) \right)^T \tilde{U}_{r,2} \right) \left(\tilde{C}_{XY} \tilde{B} \right) \right)}_{\text{Road potential}} + \\
 & \underbrace{\lambda_o \left(\left(\tilde{U}_{o,1} + \left(\tilde{C}_{XY} \tilde{A}x(k) \right)^T \tilde{U}_{o,2} \right) \left(\tilde{C}_{XY} \tilde{B} \right) \right)}_{\text{Obstacle potential}}. \tag{3-42}
 \end{aligned}$$

For the derivation of the extended MPC-matrices with the tilde notation from Eq. (3-42), the reader is referred to Appendix C. The extended inequality matrix \tilde{A}_{ineq} and corresponding vector \tilde{b}_{ineq} are also derived in that same Appendix.

Reference values The equations that represent the reference values for the output y and input u from Eq. (3-40) are given next. The output reference at time instance $k + 1$, given by $y_{\text{ref}}(k + 1|k)$, consists of the reference values for the lateral velocity v_y , the yaw rate r , and the heading angle ψ of the host vehicle at time instance $k + 1$. This is denoted by:

$$y_{\text{ref}}(k + 1|k) = \begin{bmatrix} v_{y,\text{ref}}(k + 1|k) \\ r_{\text{ref}}(k + 1|k) \\ \psi_{\text{ref}}(k + 1|k) \end{bmatrix}. \quad (3-43)$$

The output reference extended up to the prediction horizon N_p is given by:

$$\underbrace{\begin{bmatrix} y_{\text{ref}}(k + 1|k) \\ y_{\text{ref}}(k + 2|k) \\ \vdots \\ y_{\text{ref}}(k + N_p|k) \end{bmatrix}}_{\tilde{y}_{\text{ref}}} = \begin{bmatrix} v_{y,\text{ref}}(k + 1|k) \\ r_{\text{ref}}(k + 1|k) \\ \psi_{\text{ref}}(k + 1|k) \\ v_{y,\text{ref}}(k + 2|k) \\ r_{\text{ref}}(k + 2|k) \\ \psi_{\text{ref}}(k + 2|k) \\ \vdots \\ v_{y,\text{ref}}(k + N_p|k) \\ r_{\text{ref}}(k + N_p|k) \\ \psi_{\text{ref}}(k + N_p|k) \end{bmatrix}. \quad (3-44)$$

The lateral velocity reference from Eq. (3-43) is defined in the following way:

$$\underbrace{\begin{bmatrix} v_{y,\text{ref}}(k + 1|k) \\ v_{y,\text{ref}}(k + 2|k) \\ \vdots \\ v_{y,\text{ref}}(k + N_p|k) \end{bmatrix}}_{\tilde{v}_{y,\text{ref}}} = V_h \sin(\beta_{ss}) \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad (3-45)$$

with the steady-state body slip angle for driving along a curvature ρ at velocity V_h [32]:

$$\beta_{ss} = \left(\ell_r - \frac{m\ell_f V_h^2}{L_{wb} C_r} \right) \rho, \quad (3-46)$$

with the path curvature ρ being equal to $2c_2$, which is the lane center curvature, assuming the host vehicle is keeping to its lane. The path curvature is calculated analytically from:

$$\rho = \frac{\frac{d^2 X_h}{d Y_h^2}}{\left(1 + \left(\frac{d X_h}{d Y_h} \right)^2 \right)^{\frac{3}{2}}}. \quad (3-47)$$

The yaw rate reference from Eq. (3-43) for driving along a curvature ρ at velocity V_h is [32]:

$$\underbrace{\begin{bmatrix} r_{\text{ref}}(k + 1|k) \\ r_{\text{ref}}(k + 2|k) \\ \vdots \\ r_{\text{ref}}(k + N_p|k) \end{bmatrix}}_{\tilde{r}_{\text{ref}}} = \frac{\rho V_h}{\cos(\beta_{ss})} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}. \quad (3-48)$$

The vehicle heading angle reference from Eq. (3-43) is calculated as:

$$\underbrace{\begin{bmatrix} \psi_{\text{ref}}(k+1|k) \\ \psi_{\text{ref}}(k+2|k) \\ \vdots \\ \psi_{\text{ref}}(k+N_p|k) \end{bmatrix}}_{\tilde{\psi}_{\text{ref}}} = \arctan(2c_2X_p + c_1), \quad (3-49)$$

with the predicted longitudinal host vehicle positions given by:

$$X_p = X_h \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + V_h \cos(\psi_h + \beta_{ss}) \begin{bmatrix} t_h \\ 2t_h \\ \vdots \\ N_p t_h \end{bmatrix}. \quad (3-50)$$

The input reference vector, representing the steering wheel angle δ , is defined as follows:

$$\underbrace{\begin{bmatrix} u_{\text{ref}}(k|k) \\ u_{\text{ref}}(k+1|k) \\ \vdots \\ u_{\text{ref}}(k+N_p-1|k) \end{bmatrix}}_{\tilde{u}_{\text{ref}}} = \left(\frac{L_{wb}}{R} + \frac{mg}{L_{wb}} \left(\frac{\ell_r}{C_f} - \frac{\ell_f}{C_r} \right) \frac{a_y}{g} \right) \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad (3-51)$$

which is the steady-state cornering solution for the steering wheel angle for driving along a curve with radius R .

Constraint equations The constraints of the optimization problem are set up to consider the vehicle limitations, the aspects of human driving behavior, and the roadside boundaries. These constraints can be easily integrated into the optimization, as long as they are written in terms of the optimization variable, which is the system future input vector \tilde{u} .

The following set of output constraints is used, so for the lateral host vehicle position Y_h , the lateral velocity v_y , the yaw rate r , and the vehicle heading ψ_h :

$$\begin{aligned} Y_{rrb} &\leq Y_h \leq Y_{lrb}, \\ -v_{y,\max} &\leq v_y \leq v_{y,\max}, \\ -r_{\max} &\leq r \leq r_{\max}, \\ \psi_{\min} &\leq \psi_h \leq \psi_{\max}, \end{aligned} \quad (3-52)$$

The lateral positions of the road boundaries, Y_{rrb} and Y_{lrb} from Eq. (3-52), are given by:

$$\begin{aligned} Y_{rrb} &= c_{2,rrb}X_p^2 + c_{1,rrb}X_p + c_{0,rrb}, \\ Y_{lrb} &= c_{2,lrb}X_p^2 + c_{1,lrb}X_p + c_{0,lrb}, \end{aligned} \quad (3-53)$$

with c_2 , c_1 , and c_0 as defined previously in Section 2-4, for both the right and left road boundary, denoted by the subscripts rrb and lrb , respectively.

The predicted longitudinal host vehicle positions X_p are obtained from Eq. (3-50). The maximum lateral velocity from Eq. (3-52) is calculated as:

$$v_{y,\max} = V_h \sin(\beta_{\max}) \quad (3-54)$$

with the maximum body slip angle β_{\max} taken as $3/180 \pi$ rad such that the small angle approximation remains valid. The maximum vehicle yaw rate r_{\max} from Eq. (3-52), representing the lateral stability, is defined from the maximum lateral acceleration $a_{y,\max}$ and the longitudinal host vehicle speed V_h :

$$r_{\max} = \frac{a_{y,\max}}{V_h}, \quad (3-55)$$

with $a_{y,\max} = 4 \text{ ms}^{-2}$. Instead of having $a_{y,\max}$ in the numerator of r_{\max} , other yaw rate constraints expressions use μg , with μ being the tire-road friction coefficient and g the gravity constant. The friction coefficient μ is approximately 0.4 for wet roads and 0.7 for dry roads [54]. Combined with a g of 9.81 ms^{-2} this gives 3.92 ms^{-2} and 6.87 ms^{-2} , respectively. So using $a_{y,\max} = 4 \text{ ms}^{-2}$ is similar to using the friction coefficient $\mu = 0.4$, corresponding to driving on a wet road. This puts a tighter constraint on the maximum allowed yaw rate, by also guaranteeing that the vehicle response is stable for driving on wet roads.

The maximum lane-change angle ψ_{lc} equal to $5/180 \pi$ rad is determined from looking at the attained vehicle heading angles while changing lanes on a straight road. This ψ_{lc} is combined with the heading angle of the current lane at the predicted longitudinal positions X_p to yield the minimum and maximum vehicle heading, ψ_{\min} and ψ_{\max} , respectively, from Eq. (3-52):

$$\begin{aligned} \psi_{\min} &= \arctan(2c_2 X_p + c_1) - \psi_{lc}, \\ \psi_{\max} &= \arctan(2c_2 X_p + c_1) + \psi_{lc}. \end{aligned} \quad (3-56)$$

The steering actuator has mechanical limits that put bounds on the control input and variation of the control input, also called the control increment. The physical limit for the front wheel steering angle of a traditional vehicle is commonly taken as $|\delta_{\max}| = 30^\circ$ [55]. However, the maximum steering wheel angle input is calculated from:

$$u_{\max} = \frac{L_{wb} a_{y,\max}}{V_h^2} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad (3-57)$$

by using the maximum allowable lateral acceleration $a_{y,\max}$. This leads to the following input inequality equation:

$$-u_{\max} \leq u \leq u_{\max}. \quad (3-58)$$

Appendix C describes how these constraint equations are included into the optimization.

It should be remarked that the presented algorithm does not take into account the dynamics of the moving obstacles in the MPC cost function. Only the current position measurements of the obstacle vehicle are substituted into the function of the obstacle PF. If the sampling time is small enough, though, this does not pose any problem. This is similar to the human visual system not being able to perceive separate images as long as the displayed frame rate is high enough [17]. However, it might improve the performance of the algorithm if the position of the obstacle PF is also properly extended into the future.

Initial states The initial host vehicle states defining the initial conditions are given by:

$$\begin{aligned}
 V_{x,0} &= V_0 \cos (\psi_0 + \beta_0), \\
 X_0 &= 0, \\
 Y_0 &= c_2 X_0^2 + c_1 X_0 + c_0, \\
 v_{y,0} &= V_0 \sin (\beta_0), \\
 r_0 &= 2c_2 V_0, \\
 \psi_0 &= \arctan (2c_2 X_0 + c_1),
 \end{aligned} \tag{3-59}$$

with the initial body slip angle β_0 equal to the steady-state body slip angle β_{ss} from Eq. (3-46). These are the initial conditions from which the APF MPC optimization algorithm is started at each time step iteration of the simulation, which is addressed in the next Chapter.

Chapter 4

Simulations

This Chapter analyzes the performance of the developed APF MPC controller. The outline is as follows. Section 4-1 treats the theoretical background of the lateral controller from TNO. Next, Section 4-2 describes the properties of QP-problems and the available MATLAB functions to solve the related quadratic optimization. Section 4-3 gives some insight into the tuning process of the APF MPC controller, which can be done through the cost function weights and the shape of the PFs. After that, Section 4-4 presents the simulation results and discusses some of the encountered challenges and limitations of the proposed APF-method. Finally, Section 4-5 offers some thoughts on future experimental testing. Unfortunately, no real-world tests could be done during this graduation research.

4-1 TNO controller

The lateral controller from TNO is used to compare its performance with the performance of the APF MPC controller in simulation. Reference [56] gives the theoretical analysis and results of simulations and vehicle experiments of the generic lateral controller. It was shown to be able to handle different modes of lane keeping and vehicle following. The controller switches between using a feedback control loop based on either path or single point preview information, depending on the available measurements. For example, if the lanes can be detected by the on-board camera, path information is available. From this information, the lateral and heading error of the vehicle with respect to the center of the lane can be obtained. The single point preview following approach is used in vehicular platooning, when the line of sight of the camera is blocked by the preceding vehicle and lane information is unavailable. Since platooning is not the aim of this thesis, this control method is excluded from the comparison.

The objective of the lateral controller is to let the vehicle CoG track the planned path. This means that the vehicle either keeps to the center of the lane or performs a lane-change maneuver, when driving behind a slower obstacle vehicle. The controller aims to minimize the lateral path deviation and simultaneously tries to align the longitudinal axis of the vehicle

with the tangent of the path. However, it should be noted that this second objective is generally infeasible, since the direction of the vehicle's velocity V_h rarely coincides with its longitudinal axis, due to the body sideslip β_h . Nevertheless, this objective is commonly used, since no good methods exist to accurately obtain the sideslip angle.

The graphical representation of the tracking errors from the two control objectives are depicted in Figure 4-1. The lateral position error y_e is defined as the distance from the vehicle CoG to the path, perpendicular to the vehicle orientation. The heading angle error ψ_e is defined by the difference between the actual heading of the vehicle ψ_h and the desired path heading angle ψ_p , both expressed in the global coordinate frame O^g . A steady-state heading angle error will remain, equal to the steady-state body sideslip angle, because it is the vehicle velocity vector and not its longitudinal axis that should be parallel to the tangent of the path.

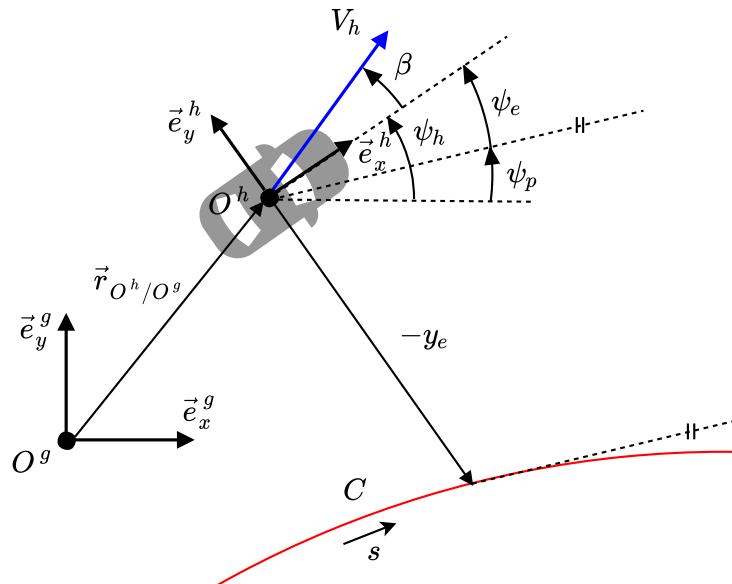


Figure 4-1: Graphical representation of tracking error definitions used by TNO

The control input from the path controller consists of a feedback and feedforward steering angle, δ_{fb} and δ_{ff} , respectively. The system can be stabilized by using feedback control only, but to remove the undesirable steady-state errors in the corners for good tracking performance, appropriate feedforward is applied as well. The control law is given by:

$$\begin{aligned} \delta &= \underbrace{\frac{2(L_{wb} + \eta V_h^2)}{d_{LA}^2} (y_e + x_{LA}\psi_e)}_{\delta_{fb}} + \underbrace{(L_{wb} + \eta V^2)\rho}_{\delta_{ff}}, \\ &= (L_{wb} + \eta V_h^2) \left(\rho + \frac{2}{d_{LA}^2} (y_e + x_{LA}\psi_e) \right), \end{aligned} \quad (4-1)$$

with the understeer gradient η that is calculated as follows:

$$\eta = \frac{mg}{L_{wb}} \left(\frac{\ell_r}{C_f} - \frac{\ell_f}{C_r} \right). \quad (4-2)$$

Both x_{LA} and d_{LA} denote the distance to the virtual look-ahead point, with x_{LA} being measured from the vehicle CoG and d_{LA} from the rear axle. They are related by:

$$d_{LA} = \ell_r + x_{LA} = \ell_r + V_h t_{LA}, \quad (4-3)$$

where t_{LA} is the look-ahead time. The applied feedforward term consists of the steering angle solution of the bicycle model for driving a curve with radius R at a speed V_h in steady-state conditions [56]. The advantage of the path controller is that it uses a clear error definition with a lateral tracking error that is regulated to zero. Furthermore, only one tuning parameter is used, namely the look-ahead time, which limits the tuning effort considerably. The described lateral controller from TNO needs to have a reference curvature ρ from the planned path, as can be seen from Eq. (4-1). Quintic polynomials are used for this, which can appropriately represent the vehicle trajectory with its six DoF. The DoF are given by the six coefficients of the fifth order polynomial function:

$$f(X) = \alpha_0 + \alpha_1 X + \alpha_2 X^2 + \alpha_3 X^3 + \alpha_4 X^4 + \alpha_5 X^5. \quad (4-4)$$

The drawback of using a single quintic polynomial to generate lane-change paths is that it results in a point-symmetric lateral velocity profile. This does not agree with human driver behavior, where the lateral acceleration is higher at the beginning of the lane change when steering out of the initial lane and lower when steering back into the target lane [23]. The coefficients α_i can be determined uniquely from the following six boundary conditions:

$$\begin{aligned} f(X_0) &= Y_0, & f(X_1) &= Y_1, \\ \frac{df}{dX} \Big|_{X_0} &= Y'_0, & \frac{df}{dX} \Big|_{X_1} &= Y'_1, \\ \frac{d^2 f}{dX^2} \Big|_{X_0} &= Y''_0, & \frac{d^2 f}{dX^2} \Big|_{X_1} &= Y''_1, \end{aligned} \quad (4-5)$$

where (X_0, Y_0) is the initial position of the host vehicle and (X_1, Y_1) the desired final position. The first and second order position derivatives make sure that the trajectory has a smooth transition at the initial and final position. The second order derivative boundary condition is directly related to the curvature of the path and therefore also to the centripetal acceleration. Overall, curvature continuity is an important requirement to smoothly follow the planned path. The curvature at the starting position of the path should be equal to zero [52]. The curvature at the end of the trajectory should be equal to the curvature of the destination lane. The boundary equations from Eq. (4-5) are used to formulate the matrix equality constraints:

$$\underbrace{\begin{pmatrix} 1 & X_0 & X_0^2 & X_0^3 & X_0^4 & X_0^5 \\ 0 & 1 & 2X_0 & 3X_0^2 & 4X_0^3 & 5X_0^4 \\ 0 & 0 & 2 & 6X_0 & 12X_0^2 & 20X_0^3 \\ 1 & X_1 & X_1^2 & X_1^3 & X_1^4 & X_1^5 \\ 0 & 1 & 2X_1 & 3X_1^2 & 4X_1^3 & 5X_1^4 \\ 0 & 0 & 2 & 6X_1 & 12X_1^2 & 20X_1^3 \end{pmatrix}}_{A_X} \underbrace{\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{pmatrix}}_{x_\alpha} = \underbrace{\begin{pmatrix} Y_0 \\ Y'_0 \\ Y''_0 \\ Y_1 \\ Y'_1 \\ Y''_1 \end{pmatrix}}_{b_Y}. \quad (4-6)$$

where A_X represents the boundary condition matrix in Eq. (4-6), x_α the polynomial coefficient vector, and b_Y the vector with boundary values. The polynomial coefficients are then obtained, under the assumption that A_X is non-singular, with the following calculation:

$$x_\alpha = A_X^{-1} b_Y, \quad (4-7)$$

which fixes the polynomial trajectory from Eq. (4-4). The curvature of this planned trajectory is then passed to the lateral TNO controller that commands the steering angle to minimize the tracking error, according to Eq. (4-1)

4-2 Quadratic programming

Convex optimization problems are much easier to solve than general nonlinear optimization problems. The explanation for this lies in the fact that convex functions do not contain local minima, but have only one global minimum, whereas nonlinear functions can have both. This special property of convex functions can be employed to efficiently solve QP-problems that are also convex. A quadratic function is only convex if it is positive semi-definite. The same applies to n -dimensional quadratic functions, where a positive semi-definite Hessian matrix is a sufficient (but not necessary) condition to guarantee convexity. Checking this condition can be done by looking at the eigenvalues of the Hessian matrix H from the quadratic function $x^T H x$. The necessary and sufficient condition for H to be positive semi-definite, is that all its eigenvalues are non-negative. Effectively, this means the quadratic function is greater or equal to zero everywhere. The cost function is formulated as a convex quadratic function, since by doing so, it is non-negative for any given input vector \tilde{u} , which is the unique property that allows for efficient solutions.

Every symmetric matrix with real entries has real eigenvalues. The formulation of the Hessian matrix, however, does not guarantee that it is symmetric. Luckily, any asymmetric matrix H from the quadratic equation $x^T H x + x^T c$ can be made symmetric with the next operation:

$$H_{sym} = \frac{1}{2} (H + H^T). \quad (4-8)$$

This means that every Hessian matrix can be converted into a symmetric matrix with real eigenvalues. If additionally these eigenvalues are greater than (or equal to) zero, the Hessian is also positive (semi)definite, which will result in fast optimization solutions.

MATLAB solvers The **quadprog** function from MATLAB [57] is a solver that finds the minimum for the QP-problem specified by the following formulation:

$$\min_x \frac{1}{2} x^T H x + c^T x \text{ such that } \begin{cases} A_{ineq} x \leq b_{ineq}, \\ A_{eq} x = b_{eq}, \\ x_{\min} \leq x \leq x_{\max}. \end{cases} \quad (4-9)$$

The syntax to call the function to solve the preceding problem is:

```
x = quadprog(H, c, Aineq, bineq, Aeq, beq, xmin, xmax, x0, options).
```

The Hessian matrix H represents the quadratic in the cost function and must be positive definite for there to be a finite minimum. The vector c stands for the linear term in the cost function. The quadratic optimization is solved by starting from the initial solution x_0 . The problem is subject to restrictions, where the matrix A_{ineq} and the vector b_{ineq} come from the linear equality constraints, the matrix A_{eq} and b_{eq} come from the linear inequality constraints, and x_{min} and x_{max} are lower and upper bound vectors, respectively. The input argument `option` is used to set specific solver options, for example which of the two algorithms is used in solving the problem.

The interior-point-convex algorithm tries to arrive at the solution by following a path that stays within the constraints. It removes redundancies and solves for the easy components first, in order to simplify the problem. The trust-region-reflective algorithm is a subspace trust-region method based on the interior-reflective Newton method. It employs the method of preconditioned conjugate gradients to approximately solve a large linear system iteratively.

The output argument is the solution vector x that minimizes the problem subject to all bounds and constraints. The `quadprog` can both handle non-convex and convex problems. In the former case, x can be a local minimum, and in the latter case, x is a global minimum.

The `mpcqpsolver` function from MATLAB [58] is another QP-solver. The main difference with `quadprog` is that it uses the so-called KWIK algorithm that was built to solve online QP-problems in a more efficient way. The biggest advantage of using `mpcqpsolver` over `quadprog`, apart from its computational efficiency, is its support of C-code generation with MATLAB Coder. This is very useful since it allows the function to be used inside a MATLAB Function block for simulation in the Simulink environment. The syntax for using this function is:

```
[x,iA] = mpqpssolver(Linv,c,Aineq,bineq,Aeq,beq,iAO,options).
```

The most important difference with `quadprog`, is that instead of using the Hessian matrix H directly, the `mpqpssolver` uses the inverse of the lower-triangular Cholesky decomposition of H , denoted by `Linv`. It is computed with the following MATLAB command [59]:

```
[L,p] = chol(H,'lower').
```

The Cholesky decomposition of a matrix A is given by:

$$A = LL^*, \quad (4-10)$$

where L is the lower triangular matrix with real and positive diagonal entries output from `chol`, and L^* denotes the conjugate transpose of L . For the `chol` function to work, and the KWIK algorithm, consequently, it is required that the Hessian is positive definite. This can be checked by looking at the output p , which is equal to zero when H is positive definite, or equal to a positive integer when it is not. The Cholesky decomposition is approximately twice as efficient as the LU-decomposition, making it very useful for solving systems of linear equations in numerical solutions [60]. Another difference with `quadprog` is that the inequality constraints used here are not in the form of $A_{ineq}x \leq b_{ineq}$, but rather the inverted version of this, namely $A_{ineq}x \geq b_{ineq}$. The former can easily be converted into the latter, by switching the inequality sign by simply multiplying the left and right side of the equation by minus

one. This leads to a different expression for both \mathbf{A} and \mathbf{b} . Furthermore, no lower and upper bounds appear explicitly as input arguments. However, these can still be included into the optimization problem by integrating them into \mathbf{A} and \mathbf{b} , which is done in Appendix C

The input argument \mathbf{iAO} represents the initial active inequalities. Active inequalities are inequalities that have their equality part of the inequality satisfied. To 'cold' start the solver, all inequality constraints are defined as inactive. The active inequalities at the found solution \mathbf{x} are given by the output \mathbf{iA} . The initial guess of KWIK is the analytic solution to the unconstrained MPC-problem. If this solution satisfies the constraints, it is the optimal solution and the solver terminates. Otherwise, KWIK uses an efficient, numerically robust strategy to find the inequality constraints that are satisfied at equality. A 'warm' start is used where the active constraints of the previous step are the initial guess in the next step.

The advantage of using `quadprog` as a solver, is that it is even capable of solving non-convex optimization problems. Since real-time implementation is only possible with the `mpcqpsolver`, due to its superior efficiency and support of C-code generation, the remainder of this report only elaborates on the approach and results of using that specific solver.

4-3 Weight tuning

Initially, tuning the APF MPC controller was a lengthy and complicated process. This can be attributed to the numerous adjustable parameters in the cost function.

First, the **weight factors** that appear directly in the cost function can be tuned. The most important requirement that must be satisfied by all weight factors, is that they must be greater than (or equal to) zero, which will result in a cost function with a Hessian matrix that is positive (semi)definite. Otherwise, the `mpcqpsolver` will not work. Increasing the weights will result in higher contributions to the cost function by the terms associated with the respective weights. These particular terms will be minimized first, if admissible by the constraints. This can be used to set priorities in the controller performance. For example, if tracking the reference value is more important, the corresponding weight should be set to a higher value. The weights that determine the importance of minimizing any deviations from the reference values are given by the yaw rate, heading angle, and input signal weights.

Secondly, the shape of the PFs can be modified by changing the values of the **PF parameters**. It is difficult to identify the parameters of each PF in a systematic way for the dynamic highway environment. The parameters of the road and obstacle PF need to be carefully selected to obtain the appropriate desired vehicle trajectories. The heights of the PFs should be calibrated such that lane keeping is preferred in the absence of an obstacle in the same lane. This means that the obstacle PF cannot extend too far laterally. Otherwise, the host vehicle would also be affected by the obstacle PF if it is driving in the next lane. Whenever the vehicle closes in on an obstacle, the PF in between the lanes should be low enough to allow crossing over to the next lane. The resulting lane-change maneuver should furthermore take place in a manner qualitatively similar to how a human driver would operate the vehicle. Scaling the height of the road or obstacle PF relative to its valleys corresponds to a bigger dominance in the cost function which results in greater lane-keeping or obstacle-avoiding action, respectively [16].

The MPC weight factors were chosen through repeated tuning. The main difficulty was to find an adequate combination of weights. After some time, however, the weights could be set by using the heuristic knowledge gained from experience. The values for the PF weights λ_r and λ_o were chosen such that the range of the road and obstacle PF is comparable, as seen by the diagonal view of the total APF from Figure 3-14a. The values for the weight λ_o were determined at $c_2 = 0 \text{ km}^{-1}$ and were kept fixed for all other values of c_2 to simplify the tuning. Whenever the host vehicle speed V_h , the speed difference between the obstacle and host vehicle $\Delta V_{o,h}$, or the curvature of the road c_2 changes, however, the obstacle PF parameter x_σ needs to be changed accordingly. Higher values for x_σ need to be selected for higher $\Delta V_{o,h}$ because then the obstacle PF needs to have a larger range in the longitudinal direction. This can be used to let the host vehicle initiate its lane change at the right Time To Collision (TTC). The TTC is defined as the time it would take for two vehicles to collide if they would continue to drive at their current speed on their current path. The lane change should be initiated at $\text{TTC} = 5\text{s}$. Five seconds after lane-change initiation, the host vehicle passes the obstacle vehicle, having already reached the center of the left lane. Furthermore, the host vehicle should have crossed the lane divider halfway through the lane-change maneuver. These TTC instances used as indicators in the lane-change tuning process are seen in Figure 4-2.

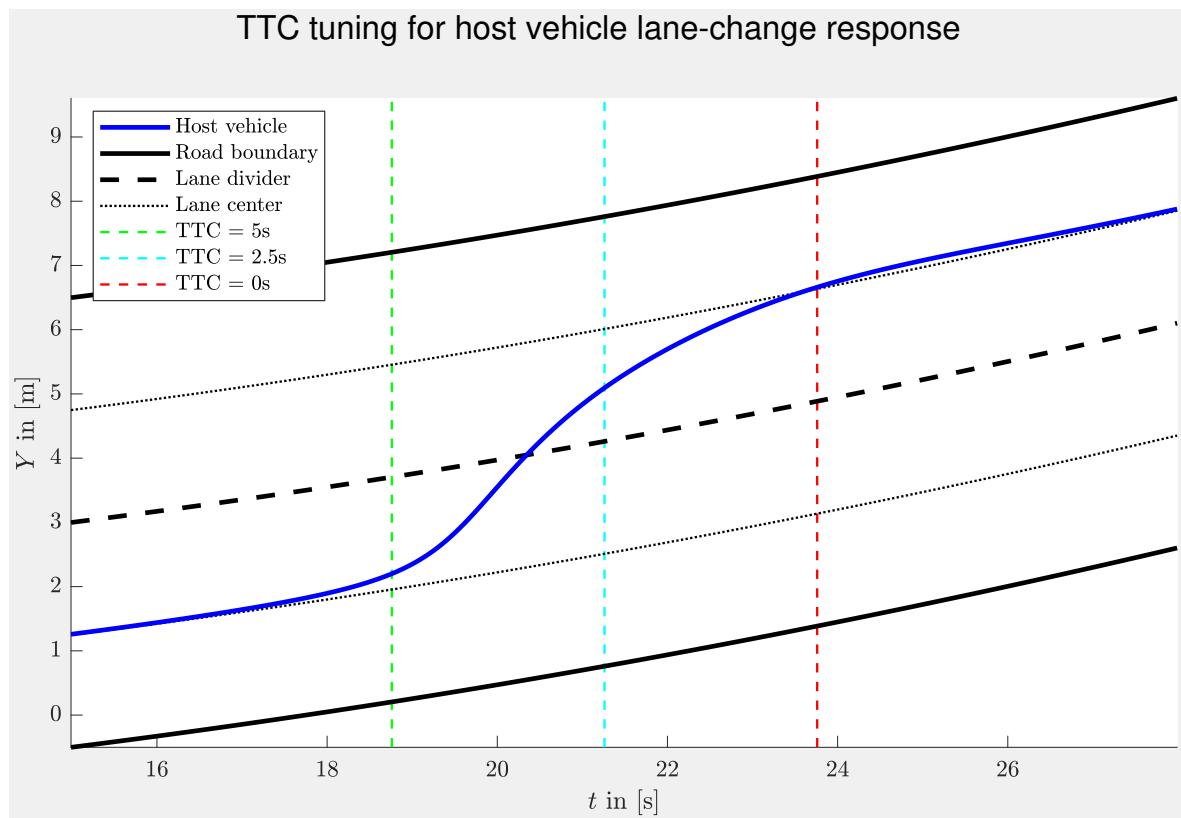


Figure 4-2: TTC-instances for tuning the host vehicle lane change, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$

The tuning results for the obstacle PF weight λ_o and parameter x_σ are seen in Appendix E. The considered values for V_h range from 20 to 130 km/h, for $\Delta V_{o,h}$ from -5 to -20 km/h. The considered values for c_2 are $\pm [0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 2.5e-4] \text{ m}^{-1}$. A positive

c_2 corresponds to a left-curving road and a negative c_2 to a right-curving road. The road curvature of $c_2 = 2.5e-4 \text{ m}^{-1}$ corresponds to the minimum horizontal road curve radius of 4,000 m for the design speed of 120 km/h without cant from Table 2-1. These values were chosen to have good coverage of the road scenarios that are possibly encountered. The obstacle PF λ_o and x_σ were determined off-line and are stored in a Simulink look-up table. The look-up table linearly performs a linear interpolation to determine the required obstacle PF weight and parameter, in case the attained values for V_h , $\Delta V_{o,h}$ and c_2 fall in between the fixed discretization points of the look-up table. It is desirable to perform the tuning process in a more structured way in order to avoid inaccuracies and errors.

The main challenge of tuning the APF MPC is that the rotated obstacle PF can block the other lane, which results in an overtaking maneuver from the wrong side. This becomes more evident at roads with high curvatures when driving at large speed differences with respect to the obstacle vehicle. In these cases, the obstacle PF should have a large longitudinal range, extending far to the back, which is governed by x_σ . However, due to the curving road and the PF rotation angle being equal to the direction of the lane center, this may block the correct lane-change side. It remains to be discussed whether a lane change is even desirable in these circumstances. Figure 4-3 shows the described scenario, with the plotted APF being an exaggerated version of the actual APF, with $x_\sigma = 130$ m instead of 32 m, to better illustrate the scenario. A different rotation angle for the obstacle PF could be used, like the angle between the line connecting the obstacle and host vehicle CoGs. However, this would involve the use of an additional calculation step, so the heading of the current lane is still being used. In that way, the PF points in the direction of the lane, which is the most logical situation.

Total APF on curved road with $c_2 = -0.00025 \text{ m}^{-1}$ - Top view

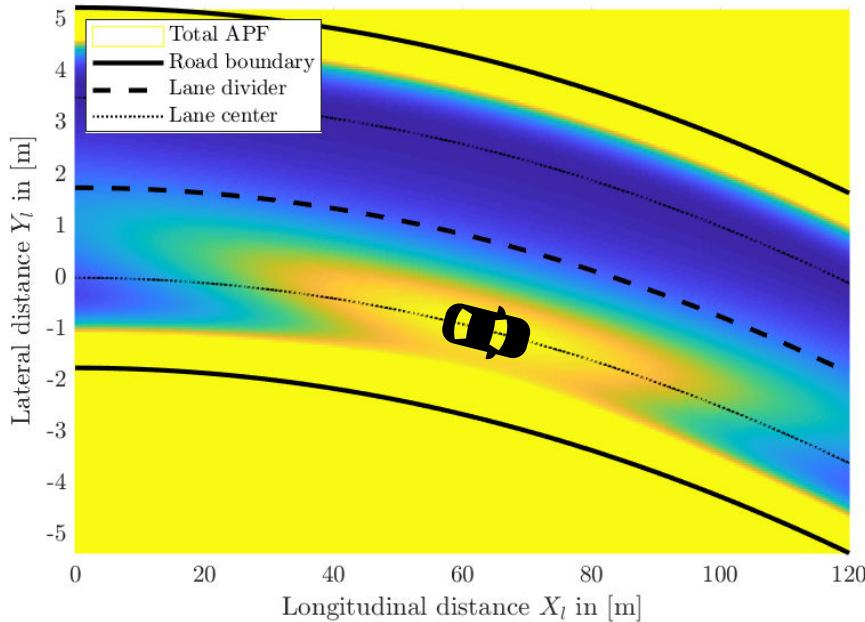


Figure 4-3: Angled view of total APF with obstacle PF blocking the left lane, with $V_h = 120 \text{ km/h}$, $\Delta V_{o,h} = -20 \text{ km/h}$, $c_2 = -0.00025 \text{ m}^{-1}$, $x_\sigma = 130 \text{ m}$, and $\sigma_y = 0.865 \text{ m}$.

4-4 Results

The developed Simulink model is used to analyze the performance of the control algorithm from Chapter 3 in a predefined environment without uncertainties in the vehicle parameters, road layout, or sensor measurements and without any disturbances. TNO already had an existing Simulink model to test their lateral controller, but this model was limited to straight roads only. Since the TNO Simulink model was not able to do simulations on curved roads, it had to be extended in order to also support curved roads. The used simulation model consists of three main subsystems. The **World Model** imitates the vehicle sensors and defines the simulation environment, like the obstacle vehicle dynamics and the road layout. The **Host Vehicle** subsystem contains the host vehicle dynamics, like the drive line dynamics, the bicycle model, and the kinematic coordinate transformations. The **Control System** is where the host vehicle is controlled, either by the APF MPC controller or the TNO controller. The obstacle vehicle is always controlled by the TNO controller and is forced to follow and stay in its lane. The described Simulink model and its subsystems are found in Appendix D.

Next, the lane-change simulation results are discussed. The difficulty with evaluating the quality of a lane change is that there are many possible optimization parameters for the trajectory of a single lane-change maneuver. Moreover, many studies use different definitions for the initiation and completion of the lane change. Reference [61] defines the lane-change initiation as the moment the vehicle begins to move laterally. The maneuver is completed once the vehicle CoG has reached the next lane. Using these definitions, the mean duration of a lane change on the highway was found to be 6.3s with a standard deviation of 2.0s. Once a certain trajectory is driven, its quality can be evaluated on two main aspects: the risk of collision and the comfort level of the maneuver. The first aspect is related to the position, velocity, and heading of the host vehicle with respect to the avoided obstacle vehicle. The second aspect is related to the mean and maximum value of the lateral acceleration and jerk. The characteristics used to evaluate the lane-change maneuvers are given by:

- | | |
|--|---|
| (1) Rise time t_r in [s]; | (5) Lateral IVD d_y in [m]; |
| (2) Lane-change duration t_d in [s]; | (6) Overshoot d_o in [m]; |
| (3) Settling time t_{se} in [s]; | (7) Lateral acceleration a_y in [ms^{-2}]; |
| (4) Longitudinal IVD d_x in [m]; | (8) Lateral jerk j_y in [ms^{-3}]. |

These lane-change characteristics are also shown graphically in Figure 4-4. Other lane-change characteristics that cannot be seen in Figure 4-4 are the simulation time and the control effort. The simulation time is the total time it takes for the simulation to run, which is used as measure for the computational demand of the algorithm. The control effort is represented by the steering wheel input, which involves a trade-off for the lane-change performance. A large steering input causes a large lateral accelerations, associated with discomfort and instability. On the other hand, a small steering input results in a lane change with long duration, which is also undesirable. However, the control effort was eventually left out because its mean value is almost equal to zero in all simulations. This indicates that the vehicle steers equally to the left as to the right.

The lane-change initiation is defined as the first moment the host vehicle deviates from the lane center by 0.1 m. All lane-change time characteristics are determined from this moment. The rise-time instance is when the host vehicle crosses the lane divider. The lane change is completed when the host vehicle passes the center of the new lane, from which the duration is determined. The settling-time instance is defined as the first moment the host vehicle stays within a 0.1 m margin from the new lane center. These time instances are illustrated in Figure 4-5 for a simulation with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$. The longitudinal IVD is equal to the IVD at the lane-change initiation. The lateral IVD is equal to the IVD when both vehicles are passing. The overshoot is the maximum outward deviation from the new lane center. The mean lateral acceleration is calculated from the lateral accelerations during the lane change by using $a_y = \rho V_h^2$. The mean lateral jerk is then obtained from the time derivative of the lateral accelerations.

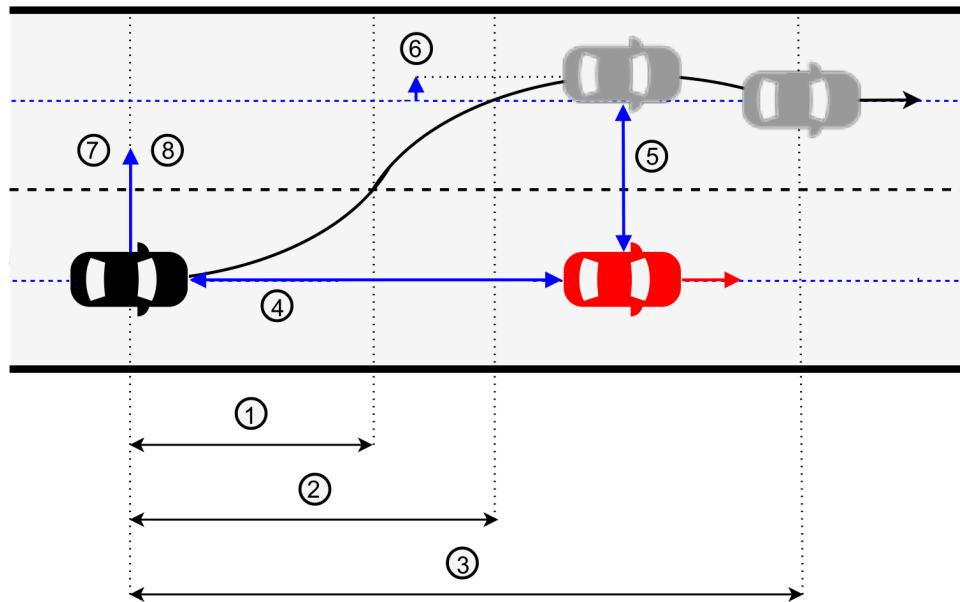


Figure 4-4: Characteristics to evaluate the lane-change performance: 0) lane-change initiation, 1) rise time: t_r , 2) lane-change duration: t_d , 3) settling time: t_{se} , 4) longitudinal IVD: d_{lo} , 5) lateral IVD: d_{la} , 6) overshoot: d_o , 7) lateral acceleration a_y , 8) lateral jerk j_y

Table 4-1 gives an overview of the lane-change simulation results of the APF MPC controller and the TNO controller, only for $c_2 = 0 \text{ m}^{-1}$, $c_2 > 0 \text{ m}^{-1}$, and $c_2 < 0 \text{ m}^{-1}$. The simulation results for each separate value of c_2 are found in Table E-1 to Table E-8 from Appendix F. For all simulations corresponding to black table entries, the host vehicle performs the lane change in a feasible and desirable way. After reaching the center of the next lane, it stays there for the rest of its course, giving also proof of good lane-keeping capabilities.

Looking at the simulation results, there are some noticeable differences between the vehicle response for roads curving to the left and to the right, which is supported by literature. The average value for the duration of lane changes to the left from [61] is higher than when changing lanes to the right, with a difference of about 0.3s. This is explained by noting that for a left lane change, the driver has to merge with traffic driving in the fast lane. This makes the lane change more difficult which is why it takes longer to complete.

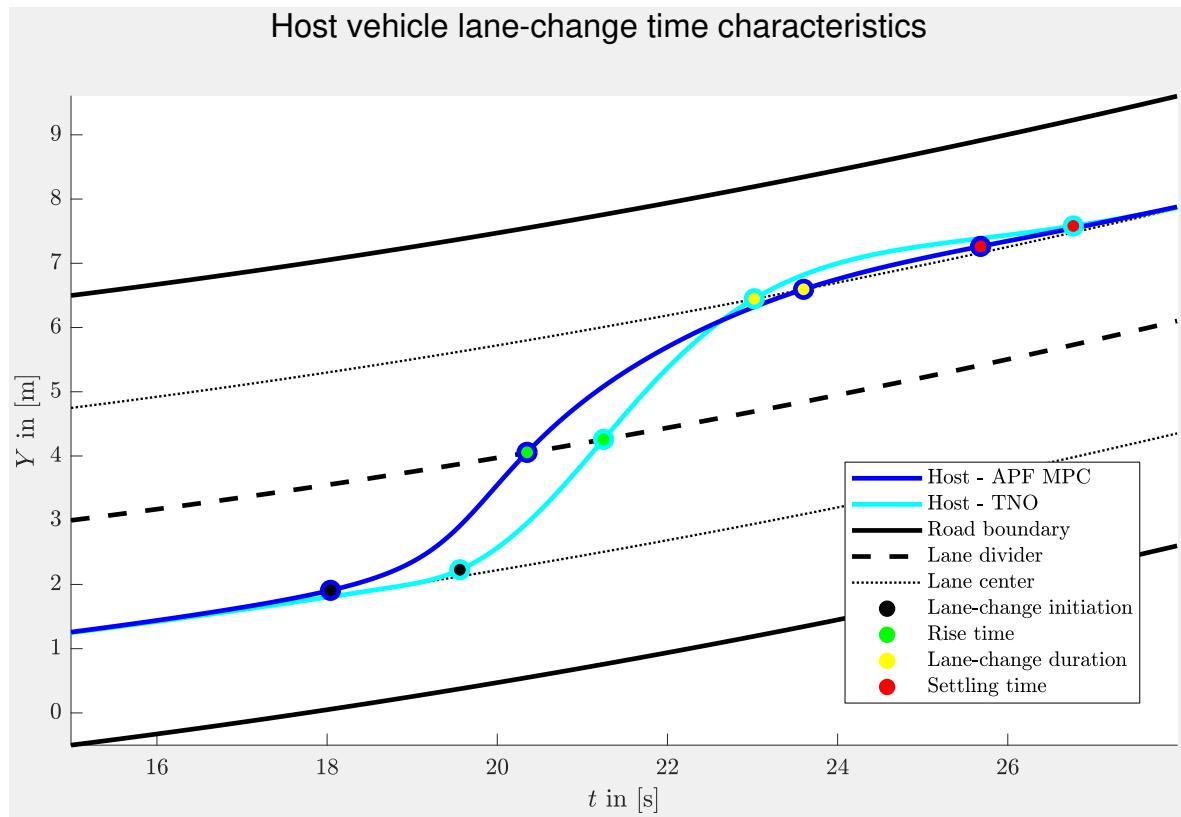


Figure 4-5: Time characteristics of lane-changing host vehicle, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$

Table 4-1: Mean lane-change characteristics of APF MPC and TNO controller

Characteristic	Mean lane-change characteristics for APF MPC and TNO controller					
	$c_2 = 0 \text{ m}^{-1}$		$c_2 > 0 \text{ m}^{-1}$		$c_2 < 0 \text{ m}^{-1}$	
	APF MPC	TNO	APF MPC	TNO	APF MPC	TNO
Rise time in [s]	2.23	1.69	2.63	2.50	2.30	2.14
Duration in [s]	5.34	3.19	5.91	3.99	4.69	3.67
Settling time in [s]	15.10	11.41	23.52	17.73	16.24	14.40
Overshoot in [m]	0.10	0.36	0.10	0.41	0.20	0.35
Long. IVD in [m]	17.74	14.51	20.77	16.40	14.87	16.16
Lat. IVD in [m]	3.39	3.82	3.38	3.81	3.32	3.80
Min. IVD in [m]	3.37	3.78	3.37	3.78	3.28	3.76
a_y in ms^{-2}	-0.01	-0.03	0.06	0.03	-0.09	-0.11
Max. a_y in ms^{-2}	1.20	0.85	1.05	0.91	1.18	0.83
j_y in ms^{-3}	0.00	-0.01	0.00	-0.01	0.00	-0.01
Max. j_y in ms^{-3}	0.32	0.10	0.24	0.11	0.36	0.10
Sim. time in [s]	5.32	1.62	6.30	1.79	5.34	1.81

Another difference between the simulated lane changes is observed. Better lane-change performance results are obtained whenever the lane change is in the same direction as the curve of the road. This can be explained by noting that the vehicle motion can more easily be restored to the lane center if the lane change is towards the lane on the inside of the curve. In that case, in order to overtake the obstacle vehicle, the host vehicle namely steers in the direction of the curving road. If the lane change is towards the lane on the outside of the curving road, the associated steering wheel angle 'conflicts' with the required steering angle to stay on the road. Consequently, a larger lateral acceleration is needed to bring the host vehicle back to the lane center. The less-challenging lane-change maneuvers are illustrated in Figure 4-6 and the more challenging in Figure 4-7. This result is confirmed by the car crash study from [62], which found a higher crash rate on right-curving roads. Overtaking maneuvers on right-curving roads are shown to be more sensitive to the curve radius, whereas on left-curving roads they are more sensitive to the superelevation. Furthermore, the speed and lateral accelerations are higher on right-curving roads.

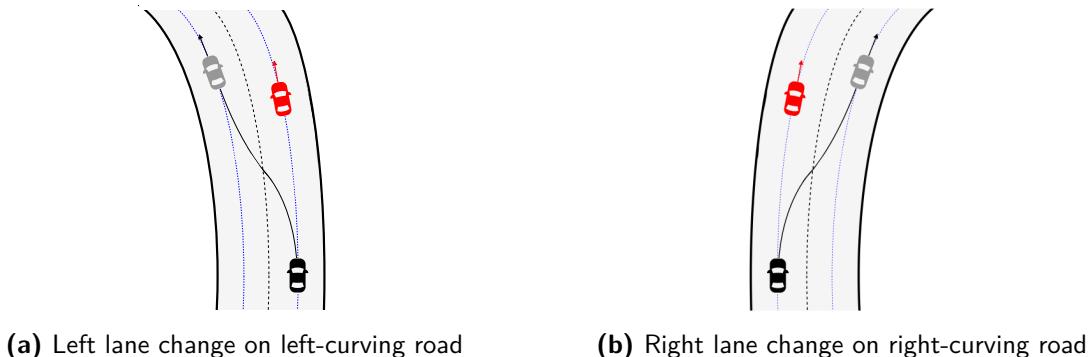


Figure 4-6: Less challenging lane-change maneuvers towards the inside of the curve

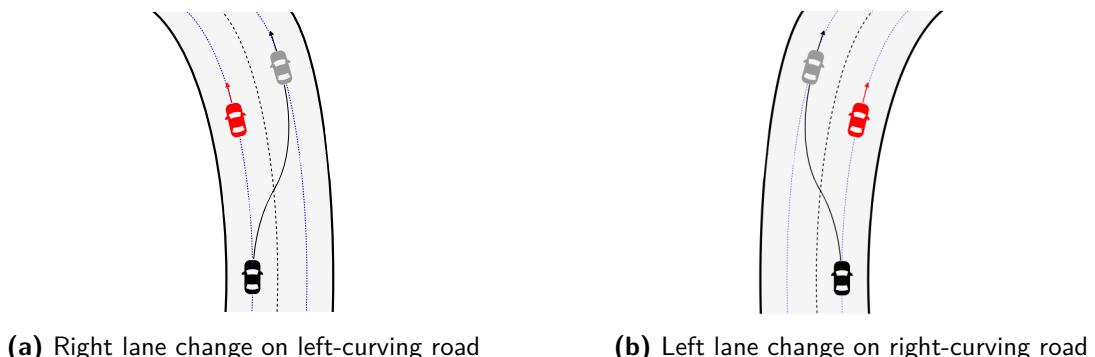


Figure 4-7: More challenging lane-change maneuvers towards the outside of the curve

Figure 4-8 shows an example of the simulated paths driven by the host vehicle controlled by the APF MPC and TNO controller and the obstacle vehicle. Figure 4-10 shows the host vehicle states that are reached while driving along the paths from Figure 4-8, together with their corresponding constraints. Figure 4-9 shows the results for the lateral acceleration a_y and lateral jerk j_y of the host vehicle for both controllers, along the same simulated path. These figures were created using $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5\text{e}{-6} \text{ m}^{-1}$.

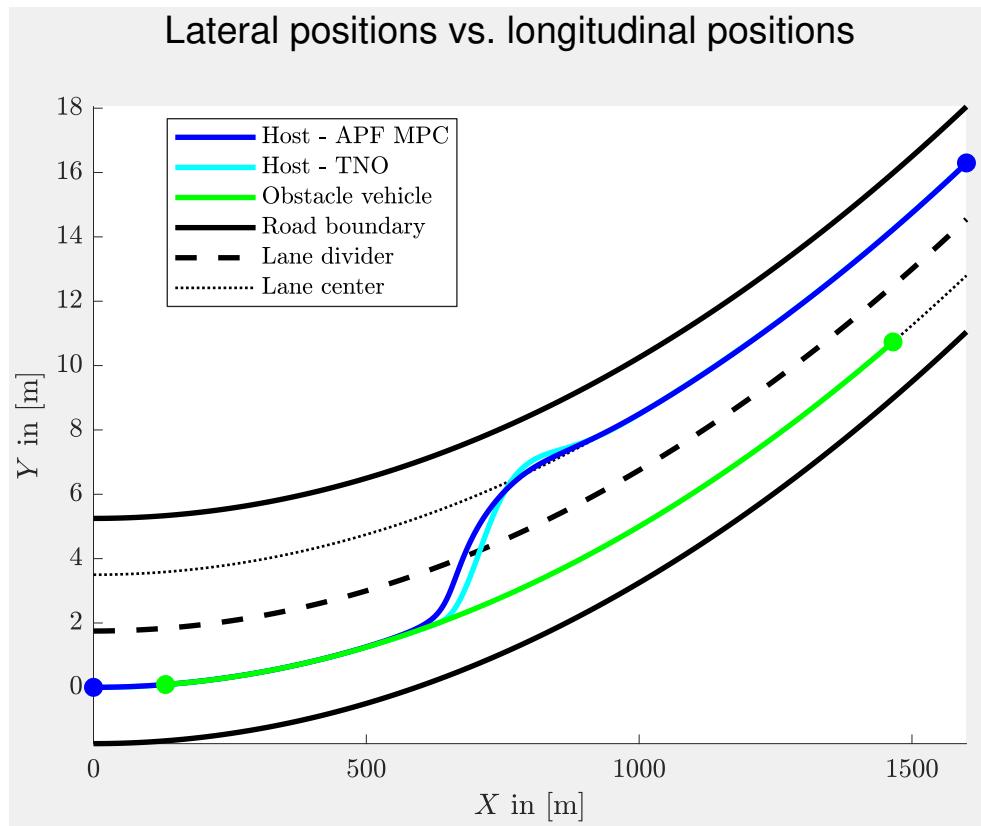


Figure 4-8: Host and obstacle vehicle path results, with $V_h = 120 \text{ kmh}^{-1}$, $\Delta V_{o,h} = -20 \text{ kmh}^{-1}$, and $c_2 = 5e-6 \text{ m}^{-1}$

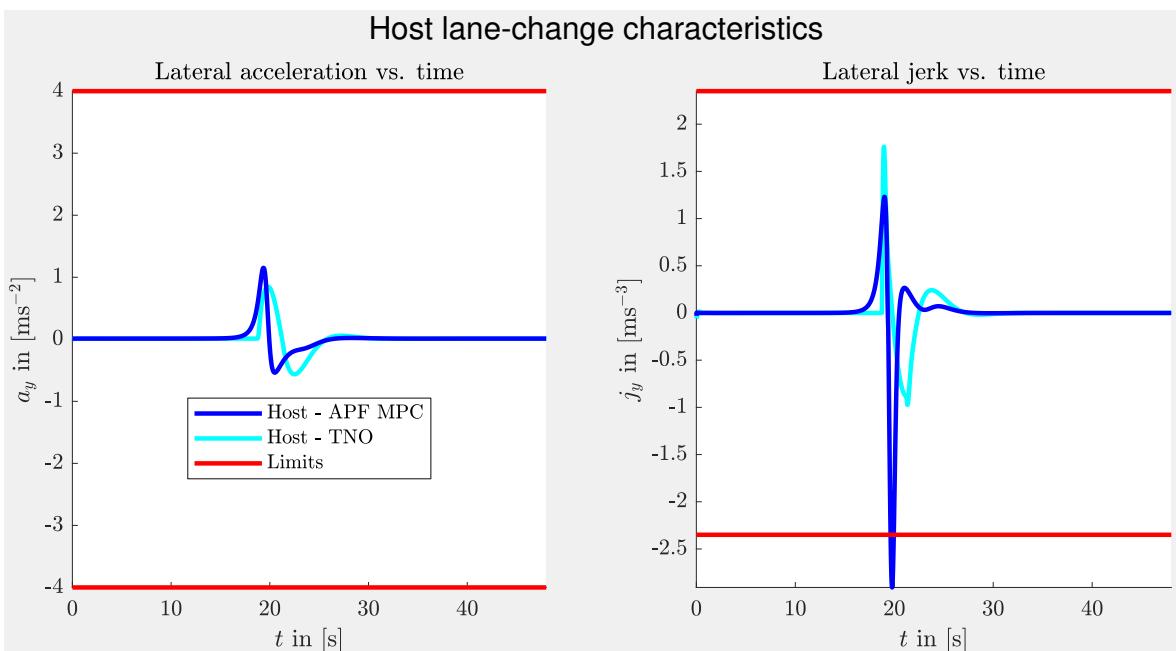


Figure 4-9: Lateral acceleration and jerk for $c_2 = 5e-6 \text{ m}^{-1}$, $V_h = 120 \text{ kmh}^{-1}$, $V_o = 100 \text{ kmh}^{-1}$

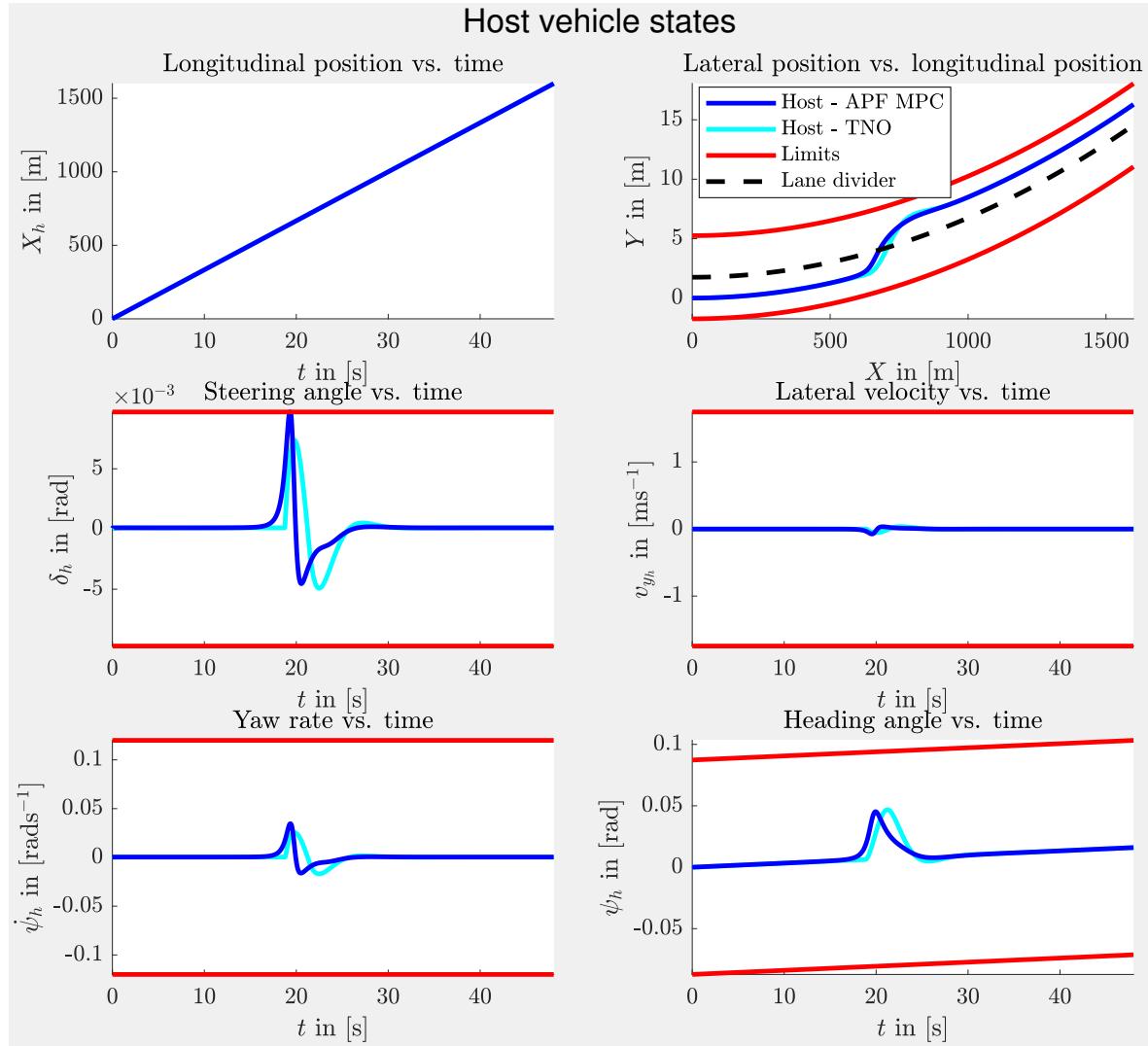


Figure 4-10: Host vehicle states for $c_2 = 5e-6 \text{ m}^{-1}$, $V_h = 120 \text{ kmh}^{-1}$, $V_o = 100 \text{ kmh}^{-1}$

Many of the subplots from Figure 4-9 and Figure 4-10 are related to the steering angle and therefore have the same shape. The initial steer action to the left is larger than the subsequent steering to the right, as can be seen in the corresponding subplot from Figure 4-10. This is desirable steering behavior, since it leads to good vehicle stability. It also means that the first phase of the obstacle avoidance or lane-changing maneuver is performed at higher lateral accelerations, after which a more comfortable return towards the lane center with lower lateral accelerations takes place. It should be remarked that the lateral jerk subplot from Figure 4-9 briefly crosses the comfort boundary of 2.35 ms^{-3} . This is possible since it is not included into the MPC-constraints. All the other constraints are seen to be satisfied, despite the large peak in the steering angle subplot from Figure 4-10.

Based on the performed simulations, it can be concluded that the APF MPC controller has similar performance as the TNO controller. Good lane-changing and lane-keeping results were obtained that showed some minor differences with the results from the TNO controller. The main difference between the two is that the APF MPC controller can more flexibly be

adapted by changing either the MPC-weights or the shape of the PFs. This property can be used to alter the shape of the trajectories in endless ways. Furthermore, the APF MPC controller is more elegant than the TNO controller, because it unifies the path-planning and -tracking processes. The performance of the TNO controller, on the other hand, can more easily be analyzed from a mathematical point of view, since it is based on a clear control law and polynomial trajectories. This is more difficult in the case of the APF MPC controller.

4-5 Experimental considerations

Unfortunately, no real-life experiments could be done. The reason for this, is that the models by TNO have been developed with MATLAB version 2014a, whereas the `mpcqpsolver`, which is at the core of the APF MPC algorithm, was first introduced in the 2015b release of MATLAB. This compatibility issue could not be resolved in time for any real-life experiments to be performed with the available test vehicle from TNO. The back-up plan was to use the miniature vehicle model built at the TU Delft under supervision of dr.ir. M. Alirezaei. However, there were some setbacks with its sensor implementation, causing it to not be finished before the end of this graduation work. So in the end, no experiments took place.

However, after simulation, it remains to be seen whether the controller will also function properly in real life, after implementation on an actual vehicle. So it is really important that the algorithm is tested for different weather conditions, road scenarios, and traffic situations. The main challenge in the field of path-tracking controllers, namely, is the development of a controller that is able to navigate different road curvatures, at different speeds, under different road conditions. Furthermore, the APF MPC control system needs to be robust and guarantee safety in the face of disturbances, model uncertainties, and parameter variations. This was not put to the test in simulation. Next, some other experimental considerations are named.

A vehicle state estimator to estimate the current state x of the vehicle is required for the vehicle model employed by the APF MPC controller. The vehicle speed can then be calculated from the rear wheel velocity, since it is assumed that the rear wheels are free rolling, because the vehicle is front-wheel driven. A Kalman filter should additionally be used to correct the vehicle speed for the accelerometer bias [34].

Furthermore, it is important to have an explicit lane availability check to ensure that the neighboring lane is in fact available, whenever a lane change is about to be initiated. Returning to the rightmost lane center can be achieved by placing a slower virtual vehicle at a certain distance in front of the host vehicle. These functionalities have not yet been implemented into the APF MPC algorithm, but certainly have to be added before doing road testing.

Chapter 5

Conclusions

The aim of this research was to design a unified path-planning and -control method for autonomous vehicles using APFs. The MPC-framework is chosen to formulate the path-planning problem as a quadratic cost function. This choice was motivated by the easy integration of the vehicle dynamics and system constraints into the MPC-optimization. The linear equations from the bicycle model, which give a good representation of the vehicle dynamics for highway-driving, are used by the model-based MPC-controller.

The cost function that is related to the MPC-problem consists of several different terms that each play their own role in controlling the host vehicle motion. The desired vehicle behavior is achieved by including the developed APF into this cost function. Two different PFs were constructed for this purpose, namely the road and obstacle PF, which control the lane-following and lane-changing maneuvers, respectively. The road PF is represented by a modified Morse PF and the obstacle PF by a Gaussian function. Their shapes can be modified to give the desired vehicle response for each encountered traffic situation. The nonlinear PFs are incorporated into the MPC cost function by making use of their quadratic Taylor approximations. The designed APF MPC algorithm can be solved efficiently since the cost function is expressed as a convex QP-problem. This was done to make sure real-time implementation is possible.

The performance of the APF MPC controller is simulated in the Simulink model from TNO, that was extended in order to also support simulations on curved roads. The results were compared with the previously developed path planner and lateral controller from TNO. The main difference is the challenging tuning process for the APF MPC controller. This is attributed to the numerous adjustable parameters in the cost function, namely the MPC weight factors and the parameters that determine the shape of the PFs. On the other hand, these tuning options allow for greater design flexibility than the TNO controller. Other than that, it is concluded that the APF MPC controller shows good performance, comparable to the performance of the TNO controller and resulting in desirable and feasible paths. The differences that were noticed for driving on roads curving in different directions correspond to the results from studies on human driving behavior.

One shortcoming of the presented algorithm is that it assumes constant longitudinal velocity and that it is only capable of planning lateral trajectories. In order to also deliver longitudinal control action, however, the model can be connected to an ACC application. The performed simulations only consider a single leading obstacle vehicle. In theory, the approach will work for any number of obstacles. Unfortunately, no experiments with real vehicles were done. The final conclusion is that the devised approach of using the APF-method in combination with MPC has great potential to simplify the complex system of an autonomous vehicle. Furthermore, the re-planning nature of the APF MPC method adds some robustness and adaptability, which makes it suitable for navigational control in a wide range of operation conditions. However, real-life experiments have to be done in the future to validate the performance of the method.

5-1 Recommendations

A hardware-in-the-loop simulation is the first recommended next step to validate the algorithm. After an extensive simulation phase it is important to do real-world experiments. In that phase, the stability and robustness of the method should be examined thoroughly, where the planned paths should be insensitive to uncertain measurements. On the other hand, the vehicle should be reactive to certain measurements, leading to consistent vehicle behavior without unnatural fluctuations.

In order to combine the proposed lateral control method with longitudinal and vertical control, an integrated control approach should be developed and tested. Only if the steering, throttle and braking, and suspension control inputs are issued in a coordinated fashion, can the autonomous vehicle navigate successfully through a wide range of different environments.

Set up a hierarchical control system, where the higher-level controller selects the cost function weights or modifies the shape of the PF to influence the lower-level APF MPC controller. This could be done by using machine learning techniques.

Another approach would be to find a functional relation between the PF parameters and weights that yields good driving behavior in a great range of driving scenarios. This would be a big improvement, because tuning the controller still requires a lot of work, even though it is very intuitive. This relation can possibly be found by identifying the PF parameters from analyzing driving data. The related identification problem can be formulated as an optimization problem, where the cost function is defined as the sum of the sine of the angle between the observed velocity and the computed gradients of the APF. Minimization of the sine of this angle is desired, because when equal to zero, the direction of the vehicle is aligned with the gradient of the APF. By doing so, it is expected that the identified APF generates paths that are similar to the originally driven trajectories, as was done in [63].

Use analytic functions to define the curve of the trajectory and formulate a simple quadratic PF around that curve, instead of using APFs to generate the paths. The Bézier curve is a good suggested candidate. It has been widely used to generate feasible paths, because of its quick and efficient path-planning solutions [52]. The most important property of Bézier curves is that the starting and ending segments of the curve are tangent to the initial and final heading, respectively. However, online generation and evaluation of analytic lane-change

trajectories increases the computational demand. This can be overcome by using an offline trajectory table that contains detailed trajectory descriptions for many different scenarios.

Introducing slack variables to soften some of the constraints and avoid infeasibility issues. These slack variables are included into the cost function as additional optimization variables. They are assigned with relatively large weights to make them dominate the cost function, even though the slack variables have relatively low values. This method is used to penalize the violation of the constraints. On the other hand, stricter road boundary constraints can be used to prevent the vehicle from performing an overtaking maneuver from the wrong side of the obstacle vehicle, whenever the right side is blocked by the corresponding obstacle PF. Furthermore, the dimensions of the vehicle can be accounted for by reducing the boundary on both sides of the road by half of the vehicle width. The current road boundary constraints prevent the vehicle CoG from crossing the road boundary, but the entire body of the vehicle should not cross these lines.

To deal with the trade-off between model accuracy and computational demand, different-sized time steps can be used. Using a short time step at the beginning of the prediction horizon gives an improved discrete-time model approximation for better tracking and stabilization. A larger time step, on the other hand, better suits the end of the prediction horizon to include the future implications of collision avoidance. To compensate for the far-term modeling inaccuracy, a FOH discretization can be used, instead of a ZOH.

The vehicle dynamics can more accurately be captured by using a nonlinear vehicle model. Therefore, it is recommended to investigate the use of nonlinear MPC. This control method is somewhat understood, with some existing solution algorithms [39]. This could add a slight burden to the computation, but it might also improve the overall performance. An efficient meta-heuristic optimization algorithm like particle swarm optimization may prove to meet the real-time requirements of solving the resulting nonlinear optimization problem [5].

Equip the system with a motion prediction model to better predict the dynamics of the surrounding traffic, such as a simple curve fitting or regression method, a stochastic Markov model, or a Kalman filter. However, equipping the approach with trajectory prediction comes at the expense of increased computational power requirement, since the trajectories need to be calculated at each time step.

Instead of using a controller that commands the steering wheel angle or curvature, it is better to make use of torque-based steering controller, which will lead to smoother steering and less control effort. A position controller, for example, will fight the external forces acting on the steering wheel that have a self-aligning effect, during a turning maneuver. A large control input will be exerted to keep to the exact angles of the trajectory, whereas a human driver would exploit these restoring forces to align the vehicle at the end of the turn. Besides, human drivers deviate from the lane center if this leads to smoother and continuous steering [18]. Furthermore, a steering wheel controller will eventually need to determine the torque that should be applied to turn the wheels. By using torque control, this extra computation step, which only makes it more complicated, is already integrated into the controller.

The last recommendation is motivated by the personal expectation that the first fully autonomous vehicles on the road are still far away. In order to still profit from the advantages of the APF-method, it is recommended to research its application with the driver still in the loop. In that way, a drivable system is created where the high-level tasks remain with the driver, while the APF-controller provides a certain level of steering assistance.

Appendix A

Bicycle model derivation

The dynamic equations of the bicycle model are derived in this Appendix.

First, the kinematic model is derived, which is used to express the vehicle motion in different coordinate frames. Figure A-1 shows the orientation of the bicycle model with respect to a local and a global frame. The free-body diagram of the bicycle model is shown in Figure A-2.

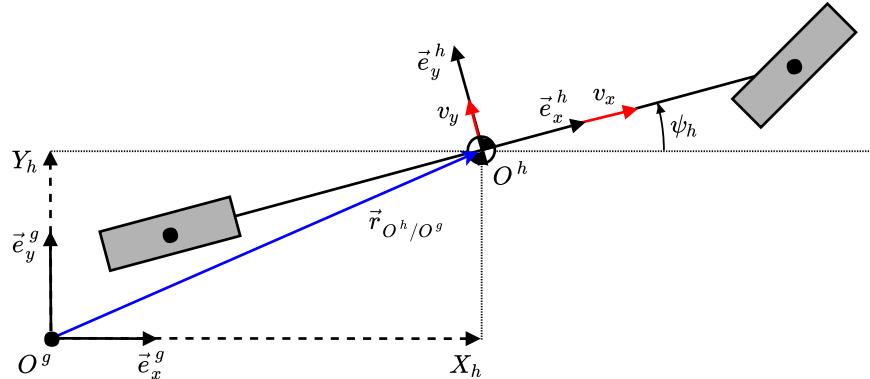


Figure A-1: Bicycle model configuration expressed in global road coordinate frame

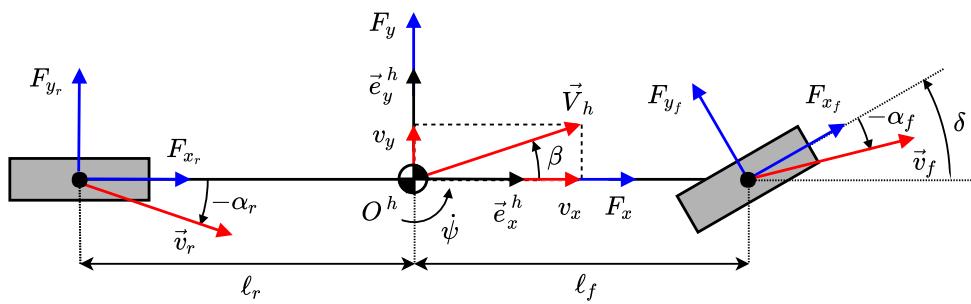


Figure A-2: Free-body diagram of the bicycle model

The stationary global road coordinate frame $S^g = \{O^g, \vec{e}_x^g, \vec{e}_y^g\}$ from Figure A-1 is fixed to the world, with the vector \vec{e}_x^g oriented in the initial driving direction of the road. The body-fixed local host-vehicle coordinate frame $S^h = \{O^h, \vec{e}_x^h, \vec{e}_y^h\}$ is located at the host vehicle CoG, with the vector \vec{e}_x^h directed forward along the longitudinal axis of the vehicle, as can be seen in both Figure A-1 and Figure A-2. The position vector of the origin of the local vehicle frame O^h , expressed in the global road frame O^g using $\underline{\vec{e}}^g$, can then be written as:

$$\vec{r}_{O^h/O^g} = \begin{bmatrix} X_h & Y_h \end{bmatrix} \begin{bmatrix} \vec{e}_x^g \\ \vec{e}_y^g \end{bmatrix} = \begin{bmatrix} X_h & Y_h \end{bmatrix} \underline{\vec{e}}^g. \quad (\text{A-1})$$

Transforming a position vector from being expressed in the global frame to the local frame can be done by using the rotation matrix R_{ψ_h} around the vector $\vec{e}_z^g = \vec{e}_x^g \times \vec{e}_y^g$ as follows:

$$\underline{\vec{e}}^g = R_{\psi_h} \underline{\vec{e}}^h = \begin{bmatrix} \cos(\psi_h) & -\sin(\psi_h) \\ \sin(\psi_h) & \cos(\psi_h) \end{bmatrix} \underline{\vec{e}}^h, \quad (\text{A-2})$$

where ψ_h is the heading angle of the vehicle, defined as the angle between \vec{e}_x^g and \vec{e}_x^h . The transpose of this transformation matrix is equal to its inverse and can be used to transform the position vector from the local frame to the global frame in a similar way:

$$\underline{\vec{e}}^h = R_{\psi_h}^T \underline{\vec{e}}^g = \begin{bmatrix} \cos(\psi_h) & \sin(\psi_h) \\ -\sin(\psi_h) & \cos(\psi_h) \end{bmatrix} \underline{\vec{e}}^g. \quad (\text{A-3})$$

The velocity vector of O^h with respect to O^g is then obtained by differentiating Eq. (A-1) with respect to time, where $\dot{\underline{\vec{e}}}^g = 0$. The result is expressed in the body-fixed frame with $\underline{\vec{e}}^h$, by using the local longitudinal and lateral velocities of the vehicle, v_x and v_y , respectively, instead of their global counterparts \dot{X}_h and \dot{Y}_h . The described steps are seen below:

$$\dot{\vec{r}}_{O^h/O^g} = \begin{bmatrix} \dot{X}_h & \dot{Y}_h \end{bmatrix} \underline{\vec{e}}^g = \begin{bmatrix} v_x & v_y \end{bmatrix} \underline{\vec{e}}^h. \quad (\text{A-4})$$

In order to derive the kinematic expressions for \dot{X}_h and \dot{Y}_h in terms of v_x and v_y , the coordinate frame transformation from Eq. (A-3) is applied to the last line of Eq. (A-4), which yields:

$$\begin{aligned} \dot{\vec{r}}_{O^h/O^g} &= \begin{bmatrix} v_x & v_y \end{bmatrix} \begin{bmatrix} \cos(\psi_h) & \sin(\psi_h) \\ -\sin(\psi_h) & \cos(\psi_h) \end{bmatrix} \underline{\vec{e}}^g, \\ &= \begin{bmatrix} v_x \cos(\psi_h) - v_y \sin(\psi_h) & v_x \sin(\psi_h) + v_y \cos(\psi_h) \end{bmatrix} \underline{\vec{e}}^g. \end{aligned} \quad (\text{A-5})$$

From comparing the first line of Eq. (A-4) with the last line of Eq. (A-5), the following kinematic relations for the motion of the vehicle CoG in the global frame are obtained [2]:

$$\begin{aligned} \dot{X}_h &= v_x \cos(\psi_h) - v_y \sin(\psi_h), \\ \dot{Y}_h &= v_x \sin(\psi_h) + v_y \cos(\psi_h). \end{aligned} \quad (\text{A-6})$$

where:

$$\begin{aligned} v_x &= V_h \cos(\beta), \\ v_y &= V_h \sin(\beta). \end{aligned} \quad (\text{A-7})$$

The derivation of the equations of motion for the bicycle model starts by obtaining the acceleration vector from differentiation of the velocity vector $\dot{\vec{r}}_{O^h/O^g}$ from Eq. (A-4) with respect to time, which results in:

$$\ddot{\vec{r}}_{O^h/O^g} = \begin{bmatrix} v_x & v_y \end{bmatrix} \dot{\underline{\vec{e}}}^h + \begin{bmatrix} \dot{v}_x & \dot{v}_y \end{bmatrix} \underline{\vec{e}}^h. \quad (\text{A-8})$$

The expression for $\dot{\underline{\vec{e}}}^h$ in the above equation is derived from Eq. (A-3) as follows:

$$\begin{aligned} \dot{\underline{\vec{e}}}^h &= \dot{R}_{\psi_h}^T \underline{\vec{e}}^g, \\ &= \dot{\psi}_h \begin{bmatrix} -\sin(\psi_h) & \cos(\psi_h) \\ -\cos(\psi_h) & -\sin(\psi_h) \end{bmatrix} \underline{\vec{e}}^g, \\ &= \dot{\psi}_h \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \cos(\psi_h) & \sin(\psi_h) \\ -\sin(\psi_h) & \cos(\psi_h) \end{bmatrix} \underline{\vec{e}}^g, \\ &= \begin{bmatrix} 0 & \dot{\psi}_h \\ -\dot{\psi}_h & 0 \end{bmatrix} R_{\psi_h}^T \underline{\vec{e}}^g = \begin{bmatrix} 0 & \dot{\psi}_h \\ -\dot{\psi}_h & 0 \end{bmatrix} \underline{\vec{e}}^h. \end{aligned} \quad (\text{A-9})$$

This expression for $\dot{\underline{\vec{e}}}^h$ can then be substituted in Eq. (A-8), giving:

$$\begin{aligned} \ddot{\vec{r}}_{O^h/O^g} &= \begin{bmatrix} v_x & v_y \end{bmatrix} \begin{bmatrix} 0 & \dot{\psi}_h \\ -\dot{\psi}_h & 0 \end{bmatrix} \underline{\vec{e}}^h + \begin{bmatrix} \dot{v}_x & \dot{v}_y \end{bmatrix} \underline{\vec{e}}^h, \\ &= \begin{bmatrix} \dot{v}_x - \dot{\psi}_h v_y & \dot{v}_y + \dot{\psi}_h v_x \end{bmatrix} \underline{\vec{e}}^h. \end{aligned} \quad (\text{A-10})$$

Using this, the equations of motion of the vehicle CoG at O^h are constructed from:

$$m\ddot{\vec{r}}_{O^h/O^g} = \vec{F}_{O^h} = \begin{bmatrix} F_x & F_y \end{bmatrix} \underline{\vec{e}}^h, \quad (\text{A-11})$$

by equating the terms corresponding to the vectors in the longitudinal direction \vec{e}_x^h and the lateral direction \vec{e}_y^h . The lateral tire force F_y is generated as a result of the tire slip with the road surface. The lateral forces from the front and rear tire, F_{y_f} and F_{y_r} , respectively, can be calculated from the linear tire response approximation:

$$F_y = F_{y_f} + F_{y_r} = C_f \alpha_f + C_r \alpha_r. \quad (\text{A-12})$$

with the negative front and rear tire cornering stiffness, C_f and C_r , respectively. These tire forces F_{y_f} and F_{y_r} can directly be added to calculate the total lateral tire force F_y , because the steering angle from Figure A-2 is assumed to small enough to satisfy the small angle approximations:

$$\begin{aligned} \cos(\delta) &\approx 1, \\ \sin(\delta) &\approx \delta. \end{aligned} \quad (\text{A-13})$$

The slip angles α_f and α_r are defined as the angle between the orientation of the wheel and the velocity vector of the wheel. Since both wheels are located on the longitudinal axis of the vehicle in the bicycle model, their velocity component in the direction \vec{e}_x^h equals the

longitudinal velocity v_x of the vehicle CoG. The lateral velocity component of the front and rear wheel, however, consists of a combination of the lateral velocity v_y of the vehicle CoG and the tangential velocity induced by the yaw rate $\dot{\psi}_h$. This is incorporated into the definition of the slip angles, where again the small angle approximation is applied:

$$\begin{aligned}\alpha_f &= \frac{v_y + \ell_f \dot{\psi}_h}{v_x} - \delta, \\ \alpha_r &= \frac{v_y - \ell_r \dot{\psi}_h}{v_x}.\end{aligned}\tag{A-14}$$

The lateral, longitudinal, and rotational equations of motion from Eq. (A-11) are given by:

$$\begin{aligned}m(\dot{v}_x - \dot{\psi}_h v_y) &= F_x, \\ m(\dot{v}_y + \dot{\psi}_h v_x) &= F_{y_f} + F_{y_r}, \\ I_z \ddot{\psi}_h &= \ell_f F_{y_f} - \ell_r F_{y_r},\end{aligned}\tag{A-15}$$

respectively, where the last line is obtained by taking the sum of the moments around the vehicle CoG at O^h , with $\ddot{\psi}_h$ being the yaw acceleration. Furthermore, notice that:

$$\begin{aligned}a_x &= \dot{v}_x - \dot{\psi}_h v_y, \\ a_y &= \dot{v}_y + \dot{\psi}_h v_x.\end{aligned}\tag{A-16}$$

Rewriting Eq. (A-15) after substitution of the expressions from Eq. (A-12) and Eq. (A-14) results in the following dynamic equations of the bicycle model [2]:

$$\begin{aligned}\dot{v}_x &= \dot{\psi}_h v_y + \frac{F_x}{m}, \\ \dot{v}_y &= \left(\frac{C_f + C_r}{mv_x} \right) v_y + \left(-v_x + \frac{\ell_f C_f - \ell_r C_r}{mv_x} \right) \dot{\psi}_h - \left(\frac{C_f}{m} \right) \delta, \\ \ddot{\psi}_h &= \left(\frac{\ell_f C_f - \ell_r C_r}{I_z v_x} \right) v_y + \left(\frac{\ell_f^2 C_f + \ell_r^2 C_r}{I_z v_x} \right) \dot{\psi}_h - \left(\frac{\ell_f C_f}{I_z} \right) \delta.\end{aligned}\tag{A-17}$$

It is assumed that the longitudinal velocity of the vehicle is constant, so that means that $\dot{v}_x = 0$. The end result is the state-space notation of the bicycle model, where the first line of Eq. (A-17) was dropped:

$$\begin{bmatrix} \dot{v}_y \\ \ddot{\psi}_h \end{bmatrix} = \begin{bmatrix} \frac{C_f + C_r}{mv_x} & -v_x + \frac{\ell_f C_f - \ell_r C_r}{mv_x} \\ \frac{\ell_f C_f - \ell_r C_r}{I_z v_x} & \frac{\ell_f^2 C_f + \ell_r^2 C_r}{I_z v_x} \end{bmatrix} \begin{bmatrix} v_y \\ \dot{\psi}_h \end{bmatrix} + \begin{bmatrix} \frac{-C_f}{m} \\ \frac{-\ell_f C_f}{I_z} \end{bmatrix} \delta.\tag{A-18}$$

Appendix B

Quadratic Taylor approximation

This Appendix elaborates on the QA of the nonlinear PFs.

The quadratic Taylor approximation is an extension of local linearization and gives a better approximation since it is constructed to have the same first and second order partial derivatives as the original function at the point of approximation. The general vector form of the quadratic Taylor approximation of a scalar-valued function f with the multidimensional vector input \mathbf{x} around \mathbf{x}_0 is given by:

$$Q_f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T H_f(\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0), \quad (\text{B-1})$$

where $\nabla f(\mathbf{x}_0)$ is the gradient of f evaluated at \mathbf{x}_0 , $H_f(\mathbf{x}_0)$ is the Hessian matrix of f evaluated at \mathbf{x}_0 , and \cdot is the dot product. In scalar notation, the quadratic Taylor approximation of a two-dimensional function $f(x, y)$ around the point (x_0, y_0) looks like this:

$$\begin{aligned} Q_f(x, y) = & f(x_0, y_0) + f_x(x_0, y_0)(x - x_0) + f_y(x_0, y_0)(y - y_0) + \\ & \frac{1}{2} f_{xx}(x_0, y_0)(x - x_0)^2 + f_{xy}(x_0, y_0)(x - x_0)(y - y_0) + \\ & \frac{1}{2} f_{yy}(x_0, y_0)(y - y_0)^2, \end{aligned} \quad (\text{B-2})$$

with the subscripts denoting the partial derivative variables. The vectors from Eq. (B-1) are:

$$\begin{aligned} \nabla f(x_0, y_0) &= \begin{bmatrix} f_x(x_0, y_0) \\ f_y(x_0, y_0) \end{bmatrix}, \\ (\mathbf{x} - \mathbf{x}_0) &= \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix}, \\ H_f(x_0, y_0) &= \begin{bmatrix} f_{xx}(x_0, y_0) & f_{xy}(x_0, y_0) \\ f_{yx}(x_0, y_0) & f_{yy}(x_0, y_0) \end{bmatrix}. \end{aligned} \quad (\text{B-3})$$

The approximation of the PF U around $(X_h, Y_h) = (X_h(k), Y_h(k))$ is given by:

$$U(k+1|k) \approx U_0(k) + U_1(k) \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix} + \frac{1}{2} \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix}^T U_2(k) \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix}. \quad (\text{B-4})$$

With the following expressions for U_0 , U_1 , and U_2 :

$$\begin{aligned} U_0(k) &= U(X_h(k), Y_h(k)) - U_{X_h} X_h(k) - U_{Y_h} Y_h(k) + 1/2 U_{X_h X_h} X_h(k)^2 + \\ &\quad U_{X_h Y_h} X_h(k) Y_h(k) + 1/2 U_{Y_h Y_h} Y_h(k)^2, \\ U_1(k) &= \begin{bmatrix} U_{X_h} - U_{X_h X_h} X_h(k) - U_{X_h Y_h} Y_h(k) \\ U_{Y_h} - U_{X_h Y_h} X_h(k) - U_{Y_h Y_h} Y_h(k) \end{bmatrix}^T, \\ U_2(k) &= \begin{bmatrix} U_{X_h X_h} & U_{X_h Y_h} \\ U_{Y_h X_h} & U_{Y_h Y_h} \end{bmatrix}. \end{aligned} \quad (\text{B-5})$$

The above expressions were obtained symbolically with MATLAB in terms of X_h and Y_h , for both the road and obstacle PFs, U_r and U_o , respectively. So instead of having to compute the QA at each time step, only the updated host vehicle positions have to be inserted into the symbolic expressions to obtain the new QA values.

The selection of $X_h(k+1|k)$ and $Y_h(k+1|k)$ from Eq. (B-4) from the state vector x :

$$\underbrace{\begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix}}_{y_{XY}(k+1|k)} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}}_{C_{XY}} \underbrace{\begin{bmatrix} V_x(k+1|k) \\ X_h(k+1|k) \\ Y_h(k+1|k) \\ v_y(k+1|k) \\ r_h(k+1|k) \\ \psi_h(k+1|k) \end{bmatrix}}_{x(k+1|k)}. \quad (\text{B-6})$$

The end result is the equivalent notation of Eq. (B-4) with substitution of Eq. (B-6):

$$\begin{aligned} U(k+1|k) &\approx U_0(k) + U_1(k) \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix} + 1/2 \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix}^T U_2(k) \begin{bmatrix} X_h(k+1|k) \\ Y_h(k+1|k) \end{bmatrix}, \\ &\approx U_0(k) + U_1(k)y_{XY}(k+1|k) + 1/2 y_{XY}(k+1|k)^T U_2(k)y_{XY}(k+1|k). \end{aligned} \quad (\text{B-7})$$

Figure B-1 to Figure B-4 show some of the QAs of the road and obstacle PFs. These figures can be used to deduce and explain how the host vehicle will react when it encounters these PFs. Black car icons are added to clarify the perspective of the plots and to illustrate the position of the host or obstacle vehicle.

Figure B-1 shows the lateral QAs of the straight road PF at $X_l = 0$ m. The displayed car represents the host vehicle driving in the center of the left lane.

Figure B-2a and Figure B-2b show the longitudinal QAs of the curved road PF at $X_l = 0$ m in the right lane and left lane, respectively. Again, the displayed car represents the host vehicle driving in the center of the lane.

Figure B-3a and Figure B-3b show the lateral QAs of the obstacle PF at $X_l = 375$ m and $Y_l = 0$ m, respectively. Here, the displayed car represents the obstacle vehicle associated with the obstacle PF driving in the center of the right lane at $(X_l, Y_l) = (375, 0)$ m.

Figure B-4a and Figure B-4b show the longitudinal QAs of the obstacle PF at $X_l = 375$ m and $Y_l = 0$ m, respectively. Again, the displayed car represents the obstacle vehicle associated with the obstacle PF driving in the center of the right lane at $(X_l, Y_l) = (375, 0)$ m.

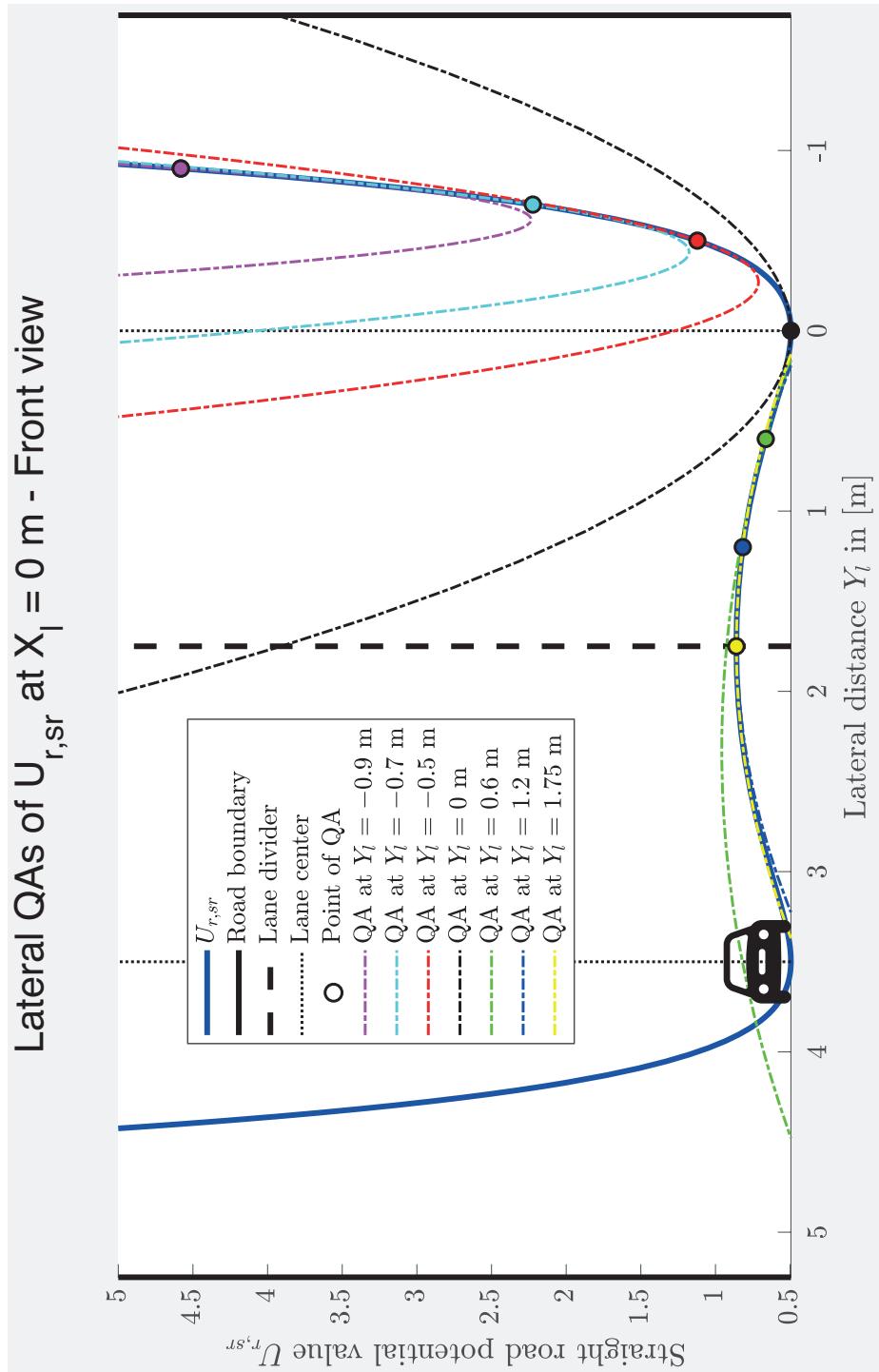


Figure B-1: Lateral QAs of straight road PF $U_{r,sr}$

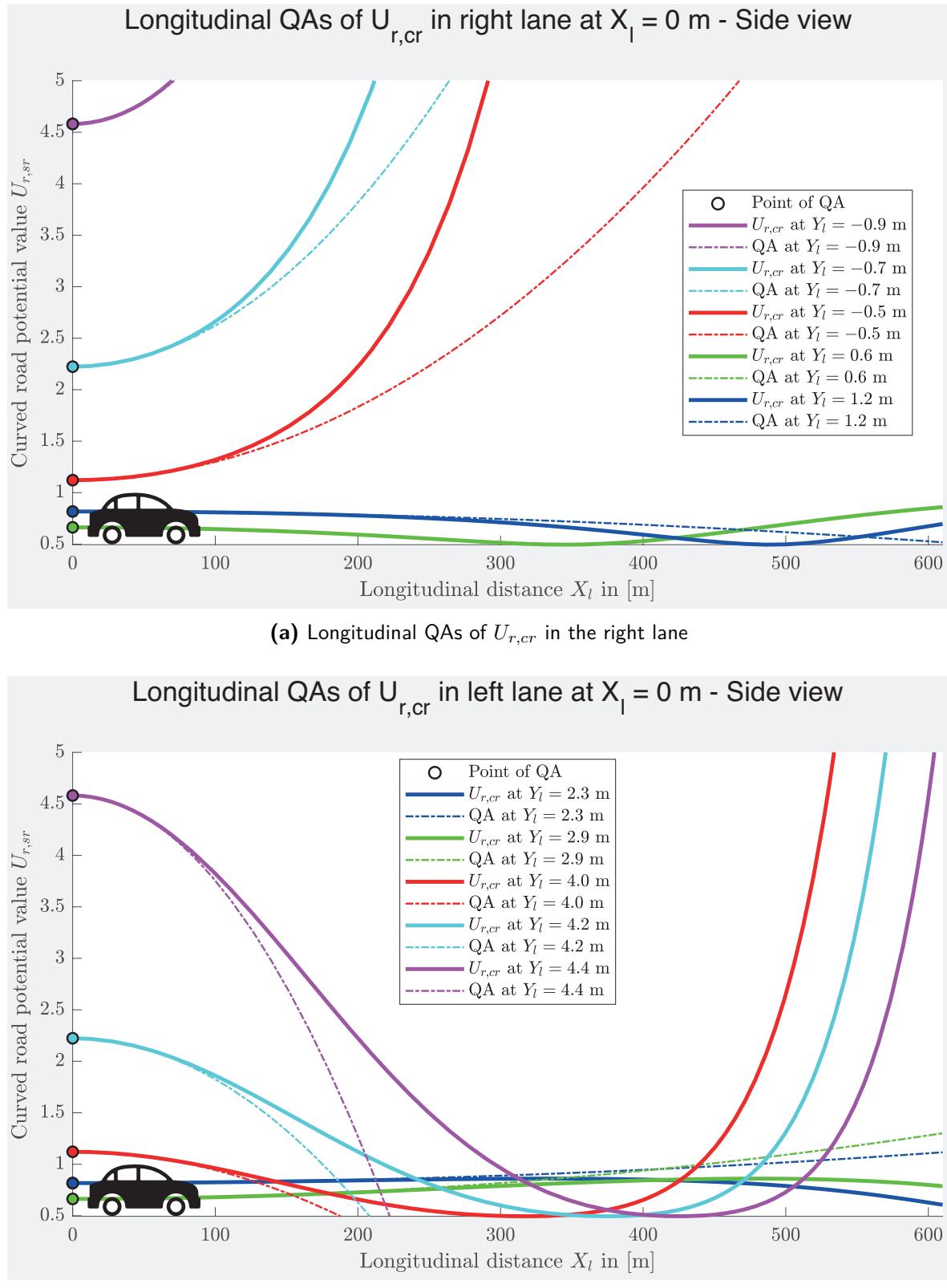
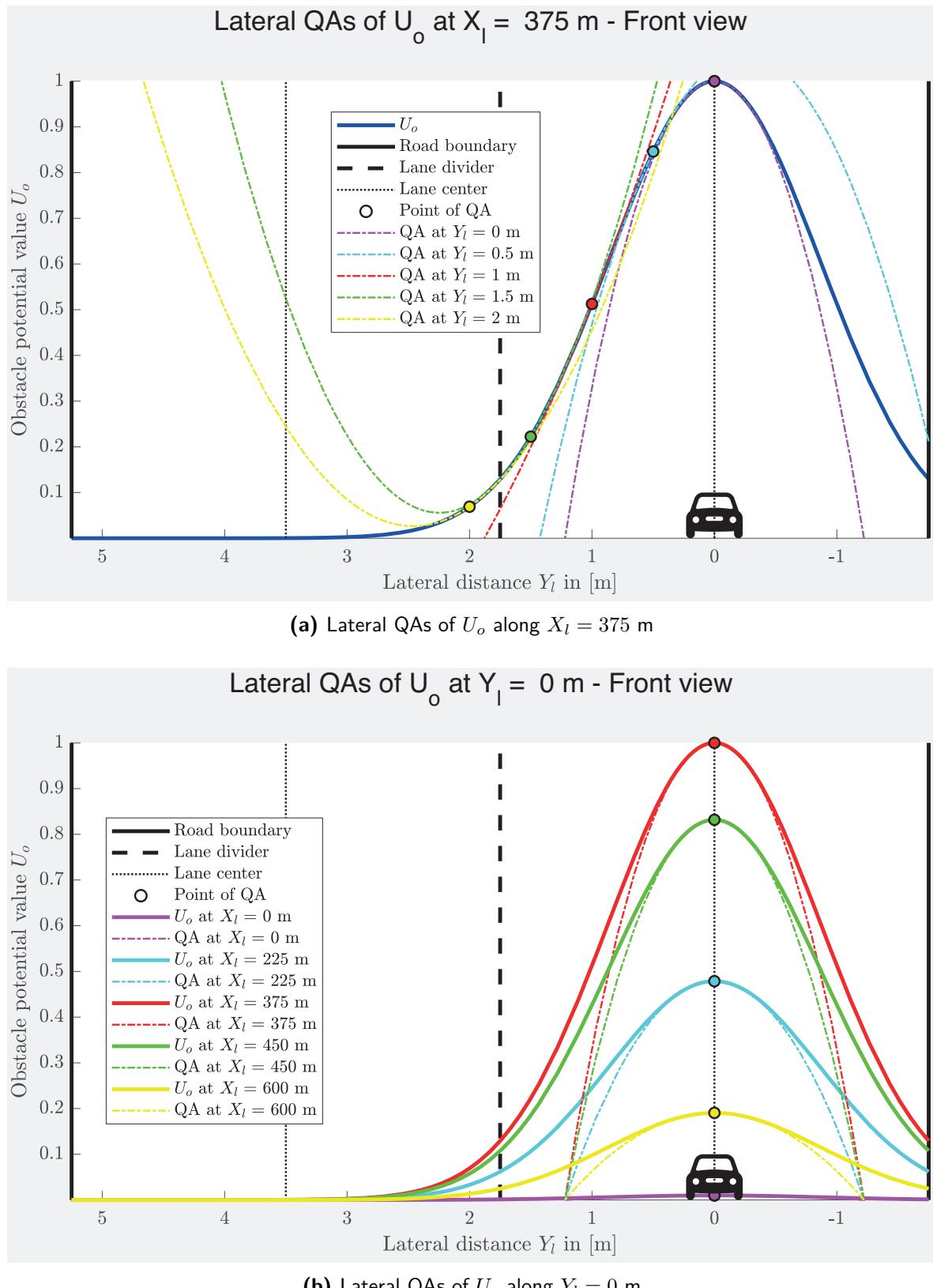
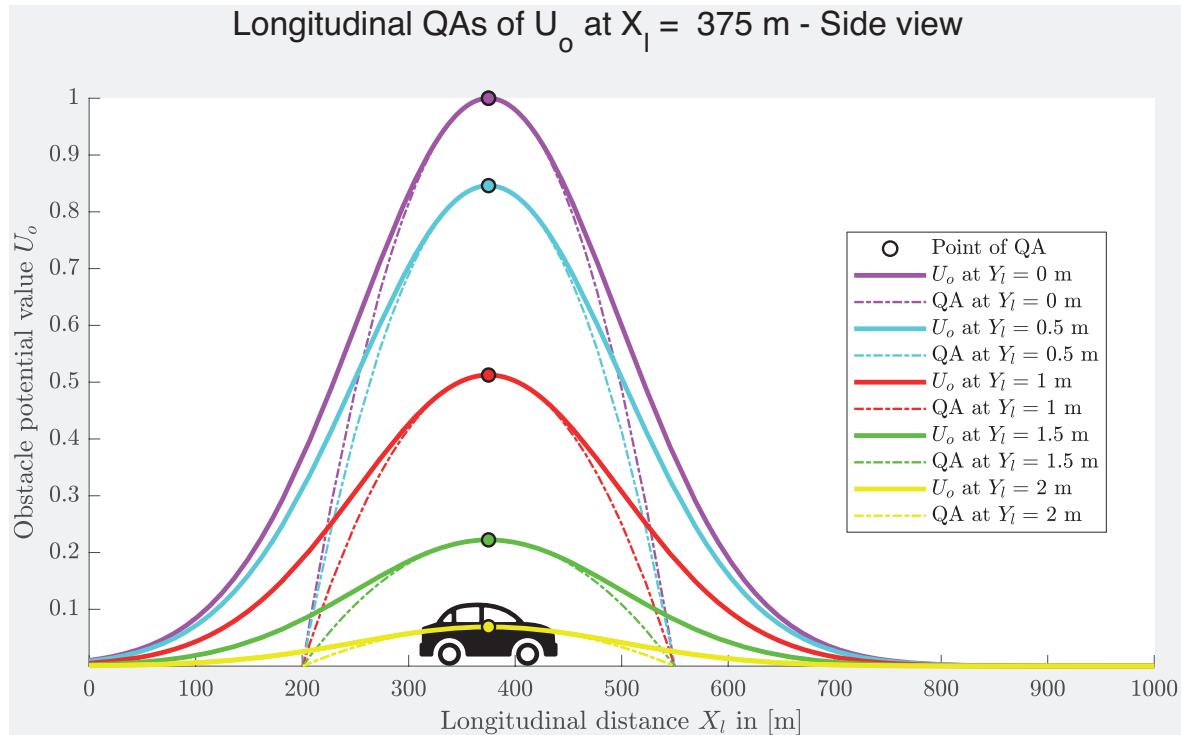
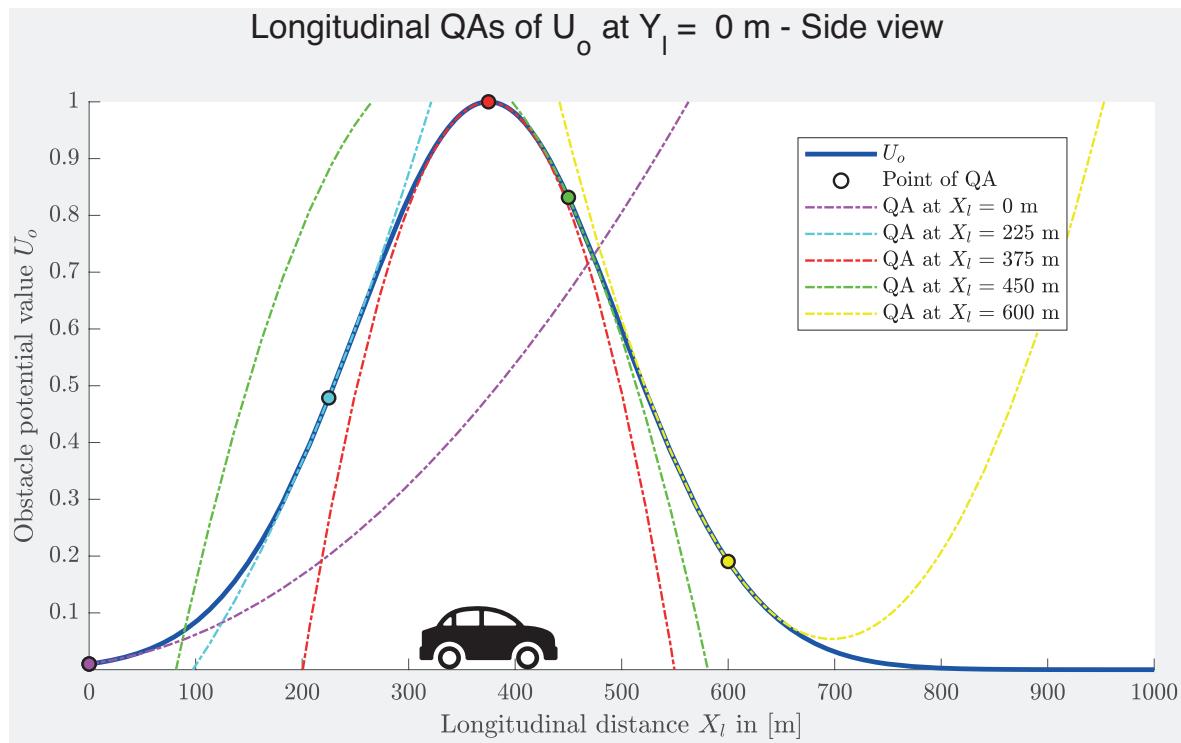


Figure B-2: Longitudinal QA of curved road PF $U_{r,cr}$

**Figure B-3:** Lateral QA of obstacle PF U_o

(a) Longitudinal QAs of U_o along $X_l = 375$ m(b) Longitudinal QAs of U_o along $Y_l = 0$ m**Figure B-4:** Longitudinal QA of obstacle PF U_o

Appendix C

MPC matrix derivation

A detailed derivation of the MPC matrices is given in this Appendix.

The cost function of the optimization problem over the prediction horizon N_p is given by:

$$\begin{aligned}
 J(k) = & \underbrace{\sum_{i=1}^{N_p} (y(k+i|k) - y_{\text{ref}}(k+i|k))^T \lambda_y (y(k+i|k) - y_{\text{ref}}(k+i|k))}_{\text{Vehicle dynamics}} + \\
 & \underbrace{\lambda_u (u(k+i-1|k) - u_{\text{ref}}(k+i-1|k))^2}_{\text{Input}} + \underbrace{\lambda_{\Delta} \Delta u(k+i-1|k)^2}_{\text{Input increment}} + \\
 & \underbrace{\lambda_r U_r(k+1|k)}_{\text{Road potential}} + \underbrace{\lambda_o U_o(k+1|k)}_{\text{Obstacle potential}}
 \end{aligned} \tag{C-1}$$

The summation from Eq. (C-1) has the following equivalent vector notation:

$$\begin{aligned}
 J(k) = & \underbrace{(\tilde{y} - \tilde{y}_{\text{ref}})^T \lambda_y (\tilde{y} - \tilde{y}_{\text{ref}})}_{\text{Vehicle dynamics}} + \underbrace{(\tilde{u} - \tilde{u}_{\text{ref}})^T \lambda_u (\tilde{u} - \tilde{u}_{\text{ref}})}_{\text{Input}} + \\
 & \underbrace{\Delta \tilde{u}^T \lambda_{\Delta} \Delta \tilde{u}}_{\text{Input increment}} + \underbrace{\lambda_r \tilde{U}_r}_{\text{Road potential}} + \underbrace{\lambda_o \tilde{U}_o}_{\text{Obstacle potential}}.
 \end{aligned} \tag{C-2}$$

The terms with a \sim on top of them are vectors containing extended predictions of their future values, with the exception of \tilde{U}_r and \tilde{U}_o , which are scalar terms consisting of the summed predicted potential values of the road and obstacle PF, respectively. The predictions of $x(k)$ up to the prediction horizon N_p are obtained from:

$$\underbrace{\begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{bmatrix}}_{\tilde{x}} = \underbrace{\begin{bmatrix} A_d \\ A_d^2 \\ \vdots \\ A_d^{N_p} \end{bmatrix}}_{\tilde{A}} x(k) + \underbrace{\begin{bmatrix} B_d & 0 & \dots & 0 \\ A_d B_d & B_d & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ A_d^{N_p-1} B_d & \dots & A_d B_d & B_d \end{bmatrix}}_{\tilde{B}} \underbrace{\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N_p-1|k) \end{bmatrix}}_{\tilde{u}}. \tag{C-3}$$

The expressions for matrices \tilde{A} and \tilde{B} in Eq. (C-3) can be derived from repetitive substitution and subsequent expansion of the discrete time state equation:

$$\begin{aligned}
 x(k+1|k) &= A_d x(k) + B_d u(k|k), \\
 x(k+2|k) &= A_d x(k+1|k) + B_d u(k+1|k), \\
 &= A_d (A_d x(k) + B_d u(k|k)) + B_d u(k+1|k), \\
 &= A_d^2 x(k) + A_d B_d u(k|k) + B_d u(k+1|k), \\
 x(k+3|k) &= A_d x(k+2|k) + B_d u(k+2|k), \\
 &= A_d (A_d^2 x(k) + A_d B_d u(k|k) + B_d u(k+1|k)) + B_d u(k+2|k), \\
 &= A_d^3 x(k) + A_d^2 B_d u(k|k) + A_d B_d u(k+1|k) + B_d u(k+2|k), \\
 &\vdots
 \end{aligned} \tag{C-4}$$

Gathering the right terms after continuing this substitution and expansion until the prediction horizon N_p is reached gives the resulting Eq. (C-3). The predictions of $y(k)$ up to the prediction horizon N_p are obtained in a similar way, yielding the following equation:

$$\underbrace{\begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ \vdots \\ y(k+N_p|k) \end{bmatrix}}_{\tilde{y}} = \underbrace{\begin{bmatrix} C_d & 0 & \dots & 0 \\ 0 & C_d & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & C_d \end{bmatrix}}_{\tilde{C}} \underbrace{\begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{bmatrix}}_{\tilde{x}}. \tag{C-5}$$

Considering the optimization problem is solved for the input sequence vector \tilde{u} , the expression for the state vector \tilde{x} from Eq. (C-3) needs to be inserted in Eq. (C-5) in order to write the cost function in terms of \tilde{u} . This gives an equivalent notation of Eq. (C-5), after substitution of Eq. (C-3) for \tilde{x} , like this:

$$\tilde{y} = \tilde{C} \tilde{x} = \tilde{C} (\tilde{A} x(k) + \tilde{B} \tilde{u}) = \tilde{C} \tilde{A} x(k) + \tilde{C} \tilde{B} \tilde{u}. \tag{C-6}$$

The predictions of input increments $\Delta u(k)$ up to the prediction horizon N_p are given by:

$$\underbrace{\begin{bmatrix} u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_p-1|k) \end{bmatrix}}_{\Delta \tilde{u}} = \underbrace{\begin{bmatrix} 1 & 0 & \dots & 0 \\ -1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & -1 & 1 \end{bmatrix}}_{\tilde{A}_u} \underbrace{\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N_p-1|k) \end{bmatrix}}_{\tilde{u}}. \tag{C-7}$$

The vector notation of the PF contribution \tilde{U} to the cost function from Eq. (C-2) can be obtained by substituting the expression for $U(k+i|k)$ from Eq. (B-7) in the following way:

$$\begin{aligned}
 \tilde{U} &= \sum_{i=1}^{N_p} U(k+i|k), \\
 &\approx \sum_{i=1}^{N_p} U_0(k) + U_1(k) y_{XY}(k+i|k) + {}^{1/2} y_{XY}^T(k+i|k) U_2(k) y_{XY}(k+i|k), \\
 &\approx \tilde{U}_0 + \tilde{U}_1 \tilde{y}_{XY} + {}^{1/2} \tilde{y}_{XY}^T \tilde{U}_2 \tilde{y}_{XY},
 \end{aligned} \tag{C-8}$$

The extended notations of the approximated PF contributions from Eq. (C-8), given by \tilde{U}_0 , \tilde{U}_1 , and \tilde{U}_2 , are as follows:

$$\begin{aligned}\tilde{U}_0 &= U_0(k)N_p, \\ \tilde{U}_1 &= \begin{bmatrix} U_1(k) & U_1(k) & \dots & U_1(k) \end{bmatrix}, \\ \tilde{U}_2 &= \begin{bmatrix} U_2(k) & 0 & \dots & 0 \\ 0 & U_2(k) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & U_2(k) \end{bmatrix},\end{aligned}\tag{C-9}$$

with the extension of C_{XY} for the predictions of $y_{XY}(k)$ up to the prediction horizon N_p :

$$\underbrace{\begin{bmatrix} y_{XY}(k+1|k) \\ y_{XY}(k+2|k) \\ \vdots \\ y_{XY}(k+N_p|k) \end{bmatrix}}_{\tilde{y}_{XY}} = \underbrace{\begin{bmatrix} C_{XY} & 0 & \dots & 0 \\ 0 & C_{XY} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & C_{XY} \end{bmatrix}}_{\tilde{C}_{XY}} \underbrace{\begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{bmatrix}}_{\tilde{x}}.\tag{C-10}$$

Eq. (C-10) is rewritten in terms of \tilde{u} , as was done previously for Eq. (C-6), through substitution of Eq. (C-3) for \tilde{x} :

$$\tilde{y}_{XY} = \tilde{C}_{XY}\tilde{x} = \tilde{C}_{XY}(\tilde{A}x(k) + \tilde{B}\tilde{u}) = \tilde{C}_{XY}\tilde{A}x(k) + \tilde{C}_{XY}\tilde{B}\tilde{u}.\tag{C-11}$$

Finally, the approximated PF contribution from the last line of Eq. (C-8) are obtained in terms of \tilde{u} after substitution of Eq. (C-11) for \tilde{y}_{XY} :

$$\begin{aligned}\tilde{U} &\approx \tilde{U}_0 + \tilde{U}_1\tilde{y}_{XY} + 1/2\tilde{y}_{XY}^T\tilde{U}_2\tilde{y}_{XY} \\ &\approx \tilde{U}_0 + \tilde{U}_1(\tilde{C}_{XY}\tilde{A}x(k) + \tilde{C}_{XY}\tilde{B}\tilde{u}) + \\ &\quad 1/2(\tilde{C}_{XY}\tilde{A}x(k) + \tilde{C}_{XY}\tilde{B}\tilde{u})^T\tilde{U}_2(\tilde{C}_{XY}\tilde{A}x(k) + \tilde{C}_{XY}\tilde{B}\tilde{u}), \\ &\approx \underbrace{\tilde{U}_0}_{\text{no } \tilde{u}} + \underbrace{\tilde{U}_1(\tilde{C}_{XY}\tilde{A}x(k))}_{\text{no } \tilde{u}} + \underbrace{\tilde{U}_1(\tilde{C}_{XY}\tilde{B}\tilde{u})}_{\dots \tilde{u}} + \\ &\quad \underbrace{1/2(\tilde{C}_{XY}\tilde{A}x(k))^T\tilde{U}_2(\tilde{C}_{XY}\tilde{A}x(k)}_{\text{no } \tilde{u}} + \underbrace{(\tilde{C}_{XY}\tilde{A}x(k))^T\tilde{U}_2(\tilde{C}_{XY}\tilde{B}\tilde{u})}_{\dots \tilde{u}} + \\ &\quad \underbrace{1/2(\tilde{C}_{XY}\tilde{B}\tilde{u})^T\tilde{U}_2(\tilde{C}_{XY}\tilde{B}\tilde{u})}_{\tilde{u}^T\dots \tilde{u}}.\end{aligned}\tag{C-12}$$

The terms marked in red do not contain \tilde{u} and can therefore not be influenced by the system input \tilde{u} in the eventual expression for the cost function. These terms are excluded from the cost function, since they will not alter the solution \tilde{u} .

Next, Eq. (C-2) is expanded after substitution of Eq. (C-6) for \tilde{y} and Eq. (C-7) for $\Delta\tilde{u}$, giving:

$$\begin{aligned}
 J(k) &= (\tilde{y} - \tilde{y}_{\text{ref}})^T \lambda_y (\tilde{y} - \tilde{y}_{\text{ref}}) + \\
 &\quad (\tilde{u} - \tilde{u}_{\text{ref}})^T \lambda_u (\tilde{u} - \tilde{u}_{\text{ref}}) + \Delta\tilde{u}^T \lambda_{\Delta} \Delta\tilde{u} + \lambda_r \tilde{U}_r + \lambda_o \tilde{U}_o, \\
 &= \left(\tilde{C}\tilde{A}x(k) + \tilde{C}\tilde{B}\tilde{u} - \tilde{y}_{\text{ref}} \right)^T \lambda_y \left(\tilde{C}\tilde{A}x(k) + \tilde{C}\tilde{B}\tilde{u} - \tilde{y}_{\text{ref}} \right) + \\
 &\quad (\tilde{u} - \tilde{u}_{\text{ref}})^T \lambda_u (\tilde{u} - \tilde{u}_{\text{ref}}) + \left(\tilde{A}_u \tilde{u} \right)^T \lambda_{\Delta} \left(\tilde{A}_u \tilde{u} \right) + \lambda_r \tilde{U}_r + \lambda_o \tilde{U}_o, \\
 &= \underbrace{\left(\tilde{C}\tilde{A}x(k) \right)^T \lambda_y \left(\tilde{C}\tilde{A}x(k) \right)}_{\text{no } \tilde{u}} + \underbrace{\left(\tilde{C}\tilde{B}\tilde{u} \right)^T \lambda_y \left(\tilde{C}\tilde{B}\tilde{u} \right)}_{\tilde{u}^T \dots \tilde{u}} + \underbrace{\tilde{y}_{\text{ref}}^T \lambda_y \tilde{y}_{\text{ref}}}_{\text{no } \tilde{u}} + \\
 &\quad \underbrace{2 \left(\tilde{C}\tilde{A}x(k) \right)^T \lambda_y \left(\tilde{C}\tilde{B}\tilde{u} \right)}_{\dots \tilde{u}} - \underbrace{2 \left(\tilde{C}\tilde{A}x(k) \right)^T \lambda_y \tilde{y}_{\text{ref}}}_{\text{no } \tilde{u}} - \\
 &\quad \underbrace{2 \left(\tilde{C}\tilde{B}\tilde{u} \right)^T \lambda_y \tilde{y}_{\text{ref}}}_{\dots \tilde{u}} + \underbrace{\tilde{u}^T \lambda_u \tilde{u}}_{\tilde{u}^T \dots \tilde{u}} + \underbrace{\tilde{u}_{\text{ref}}^T \lambda_u \tilde{u}_{\text{ref}}}_{\text{no } \tilde{u}} - \underbrace{2 \tilde{u}_{\text{ref}}^T \lambda_u \tilde{u}}_{\dots \tilde{u}} + \\
 &\quad \underbrace{\left(\tilde{A}_u \tilde{u} \right)^T \lambda_{\Delta} \left(\tilde{A}_u \tilde{u} \right)}_{\tilde{u}^T \dots \tilde{u}} + \underbrace{\lambda_r \tilde{U}_r}_{\text{Road potential}} + \underbrace{\lambda_o \tilde{U}_o}_{\text{Obstacle potential}}. \tag{C-13}
 \end{aligned}$$

The final step is to gather the right terms, including both the expressions for the road and obstacle PF contribution, denoted with the subscripts r and o , respectively, with the associated weights λ_r and λ_o , from Eq. (C-12). The red terms can again be omitted because they do not contain \tilde{u} . The light-blue quadratic terms from Eq. (C-12) and Eq. (C-13) are grouped under the Hessian matrix H and the dark-blue linear terms under the linear vector c :

$$\begin{aligned}
 H &= \underbrace{2 \left(\tilde{C}\tilde{B} \right)^T \lambda_y \left(\tilde{C}\tilde{B} \right)}_{\text{Vehicle dynamics}} + \underbrace{2 \lambda_u I_{N_p}}_{\text{Input}} + \underbrace{2 \tilde{A}_u^T \lambda_{\Delta} \tilde{A}_u}_{\text{Input increment}} + \\
 &\quad \underbrace{\lambda_r \left(\tilde{C}_{XY} \tilde{B} \right)^T \tilde{U}_{r,2} \left(\tilde{C}_{XY} \tilde{B} \right)}_{\text{Road potential}} + \underbrace{\lambda_o \left(\tilde{C}_{XY} \tilde{B} \right)^T \tilde{U}_{o,2} \left(\tilde{C}_{XY} \tilde{B} \right)}_{\text{Obstacle potential}}, \\
 c &= \underbrace{2 \left(\tilde{C}\tilde{A}x(k) - \tilde{y}_{\text{ref}} \right)^T \lambda_y \left(\tilde{C}\tilde{B} \right)}_{\text{Vehicle dynamics}} - \underbrace{2 \lambda_u \tilde{u}_{\text{ref}}^T}_{\text{Input}} + \\
 &\quad \underbrace{\lambda_r \left(\left(\tilde{U}_{r,1} + \left(\tilde{C}_{XY} \tilde{A}x(k) \right)^T \tilde{U}_{r,2} \right) \left(\tilde{C}_{XY} \tilde{B} \right) \right)}_{\text{Road potential}} + \\
 &\quad \underbrace{\lambda_o \left(\left(\tilde{U}_{o,1} + \left(\tilde{C}_{XY} \tilde{A}x(k) \right)^T \tilde{U}_{o,2} \right) \left(\tilde{C}_{XY} \tilde{B} \right) \right)}_{\text{Obstacle potential}}. \tag{C-14}
 \end{aligned}$$

The results from Eq. (C-14) define the QP problem as follows:

$$\min_{\tilde{u}} J = \min_{\tilde{u}} \frac{1}{2} \tilde{u}^T H \tilde{u} + c^T \tilde{u} \text{ subject to } A_{ineq} \tilde{u} \geq b_{ineq}. \tag{C-15}$$

The optimization problem from Eq. (C-15) is not subject to any equality constraints, so A_{eq} and b_{eq} are defined to be appropriately dimensioned matrices equal to zero. Now, the expressions for A_{ineq} and b_{ineq} from Eq. (C-15) are derived. The output constraints are given by:

$$\begin{aligned} Y_{rrb} &\leq Y_h \leq Y_{\ellrb}, \\ -v_{y,\max} &\leq v_y \leq v_{y,\max} \\ -r_{\max} &\leq r_h \leq r_{\max}, \\ \psi_{\min} &\leq \psi_h \leq \psi_{\max}. \end{aligned} \tag{C-16}$$

For a more detailed description of these constraints, the reader is referred to the main text in Section 3-4. The constraint equations from Eq. (C-16) are rewritten in vector notation, where the equations containing the \geq symbol are multiplied by minus one to rewrite them with the \leq symbol:

$$\underbrace{\begin{bmatrix} Y_h(k+1|k) \\ -Y_h(k+1|k) \\ v_y(k+1|k) \\ -v_y(k+1|k) \\ r_h(k+1|k) \\ -r_h(k+1|k) \\ \psi_h(k+1|k) \\ -\psi_h(k+1|k) \end{bmatrix}}_{y_{\text{ineq}}(k+1|k)} \leq \underbrace{\begin{bmatrix} Y_{\ellrb}(k+1|k) \\ -Y_{rrb}(k+1|k) \\ v_{y,\max}(k+1|k) \\ v_{y,\max}(k+1|k) \\ r_{\max}(k+1|k) \\ r_{\max}(k+1|k) \\ \psi_{\max}(k+1|k) \\ -\psi_{\min}(k+1|k) \end{bmatrix}}_{f_{\text{ineq}}(k+1|k)}, \tag{C-17}$$

The left side of the inequality from Eq. (C-17) is then rewritten with the aid of the state vector $x(k+1|k)$ in the following way:

$$\underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}}_{C_{\text{ineq}}} \underbrace{\begin{bmatrix} V_x(k+1|k) \\ X_h(k+1|k) \\ Y_h(k+1|k) \\ v_y(k+1|k) \\ r_h(k+1|k) \\ \psi_h(k+1|k) \end{bmatrix}}_{x(k+1|k)} \leq \underbrace{\begin{bmatrix} Y_{\ellrb}(k+1|k) \\ -Y_{rrb}(k+1|k) \\ v_{y,\max}(k+1|k) \\ v_{y,\max}(k+1|k) \\ r_{\max}(k+1|k) \\ r_{\max}(k+1|k) \\ \psi_{\max}(k+1|k) \\ -\psi_{\min}(k+1|k) \end{bmatrix}}_{f_{\text{ineq}}(k+1|k)}. \tag{C-18}$$

The predictions of $y_{\text{ineq}}(k+1)$ up to the prediction horizon N_p are then written in the compact form using the tilde-notation:

$$\underbrace{\begin{bmatrix} y_{\text{ineq}}(k+1|k) \\ y_{\text{ineq}}(k+2|k) \\ \vdots \\ y_{\text{ineq}}(k+N_p|k) \end{bmatrix}}_{\tilde{y}_{\text{ineq}}} \geq \underbrace{\begin{bmatrix} f_{\text{ineq}}(k+1|k) \\ f_{\text{ineq}}(k+2|k) \\ \vdots \\ f_{\text{ineq}}(k+N_p|k) \end{bmatrix}}_{\tilde{f}_{\text{ineq}}}. \tag{C-19}$$

Equivalently, for the alternative expression from Eq. (C-18) in terms of \tilde{x} :

$$\underbrace{\begin{bmatrix} C_{\text{ineq}} & 0 & \dots & 0 \\ 0 & C_{\text{ineq}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & C_{\text{ineq}} \end{bmatrix}}_{\tilde{C}_{\text{ineq}}} \underbrace{\begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{bmatrix}}_{\tilde{x}} \geq \underbrace{\begin{bmatrix} f_{\text{ineq}}(k+1|k) \\ f_{\text{ineq}}(k+2|k) \\ \vdots \\ f_{\text{ineq}}(k+N_p|k) \end{bmatrix}}_{\tilde{f}_{\text{ineq}}}. \quad (\text{C-20})$$

As has been done multiple times before already, Eq. (C-20) is worked out with substitution of Eq. (C-3) for \tilde{x} .

$$\begin{aligned} \tilde{C}_{\text{ineq}} \tilde{x} &\leq \tilde{f}_{\text{ineq}}, \\ \tilde{C}_{\text{ineq}} (\tilde{A}x(k) + \tilde{B}\tilde{u}) &\leq \tilde{f}_{\text{ineq}}, \\ \underbrace{\tilde{C}_{\text{ineq}} \tilde{B}}_{\tilde{A}_{x,\text{ineq}}} \tilde{u} &\leq \underbrace{\tilde{f}_{\text{ineq}} - \tilde{C}_{\text{ineq}} \tilde{A}x(k)}_{\tilde{b}_{x,\text{ineq}}}, \end{aligned} \quad (\text{C-21})$$

yielding the state inequality matrix $\tilde{A}_{x,\text{ineq}}$ and the corresponding vector $\tilde{b}_{x,\text{ineq}}$ from the right side of the inequality. Next, the same procedure is repeated, but this time to compute the input inequality matrix $\tilde{A}_{u,\text{ineq}}$ and the corresponding vector $\tilde{b}_{u,\text{ineq}}$. The input inequality constraint is given by:

$$-u_{\max} \leq u \leq u_{\max}. \quad (\text{C-22})$$

Again, the constraint equations from Eq. (C-22) are rewritten in vector notation, where the equations containing the \geq symbol are multiplied by minus one to rewrite them with the \leq symbol:

$$\begin{aligned} \begin{bmatrix} u(k) \\ -u(k) \end{bmatrix} &\leq \begin{bmatrix} u_{\max} \\ u_{\max} \end{bmatrix}, \\ \underbrace{\begin{bmatrix} 1 \\ -1 \end{bmatrix}}_{A_{u,\text{ineq}}} u(k) &\leq \underbrace{\begin{bmatrix} u_{\max} \\ u_{\max} \end{bmatrix}}_{b_{u,\text{ineq}}}. \end{aligned} \quad (\text{C-23})$$

Extending Eq. (C-23) by including the predictions of $u(k)$ up to the prediction horizon N_p :

$$\underbrace{\begin{bmatrix} A_{u,\text{ineq}} & 0 & \dots & 0 \\ 0 & A_{u,\text{ineq}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & A_{u,\text{ineq}} \end{bmatrix}}_{\tilde{A}_{u,\text{ineq}}} \underbrace{\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N_p-1|k) \end{bmatrix}}_{\tilde{u}} \leq \underbrace{\begin{bmatrix} b_{u,\text{ineq}} \\ b_{u,\text{ineq}} \\ \vdots \\ b_{u,\text{ineq}} \end{bmatrix}}_{\tilde{b}_{u,\text{ineq}}}. \quad (\text{C-24})$$

Finally, Eq. (C-21) and Eq. (C-24) are combined into the total inequality equation that is in the correct form to be used by the QA solver from MATLAB:

$$\underbrace{\begin{bmatrix} \tilde{A}_{x,\text{ineq}} \\ \tilde{A}_{u,\text{ineq}} \end{bmatrix}}_{\tilde{A}_{\text{ineq}}} \tilde{u} \leq \underbrace{\begin{bmatrix} \tilde{b}_{x,\text{ineq}} \\ \tilde{b}_{u,\text{ineq}} \end{bmatrix}}_{\tilde{b}_{\text{ineq}}}. \quad (\text{C-25})$$

Appendix D

Simulink model

This Appendix shows the used Simulink model and its different subsystems.

Figure D-1 gives the overview of the Simulink model, showing the different subsystems of the World Model, the Control System, and the Host Vehicle. Figure D-2 shows the inside of the World Model, where the vehicle sensors are imitated and the environment is defined, like the obstacle vehicle dynamics and the road layout. Furthermore, it is shown how the relevant data is logged. Figure D-3 gives an insight into the Host Vehicle block, with the vehicle dynamic equations as its contents, like the drive line dynamics, the bicycle model, and the kinematic coordinate transformations. Figure D-4 and Figure D-5 display the Control System architecture, where the host vehicle is controlled by the TNO and APF MPC controller, respectively, whereas the obstacle vehicle is always controlled by the TNO controller. Figure D-6 shows the main contribution of this research: the APF MPC controller structure.

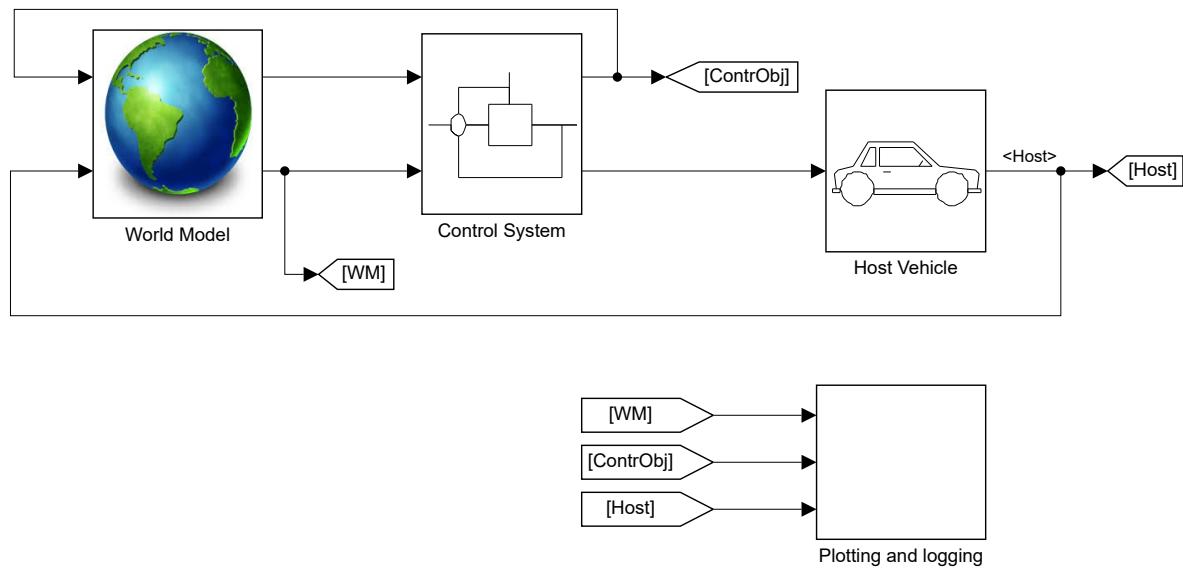


Figure D-1: Simulink model

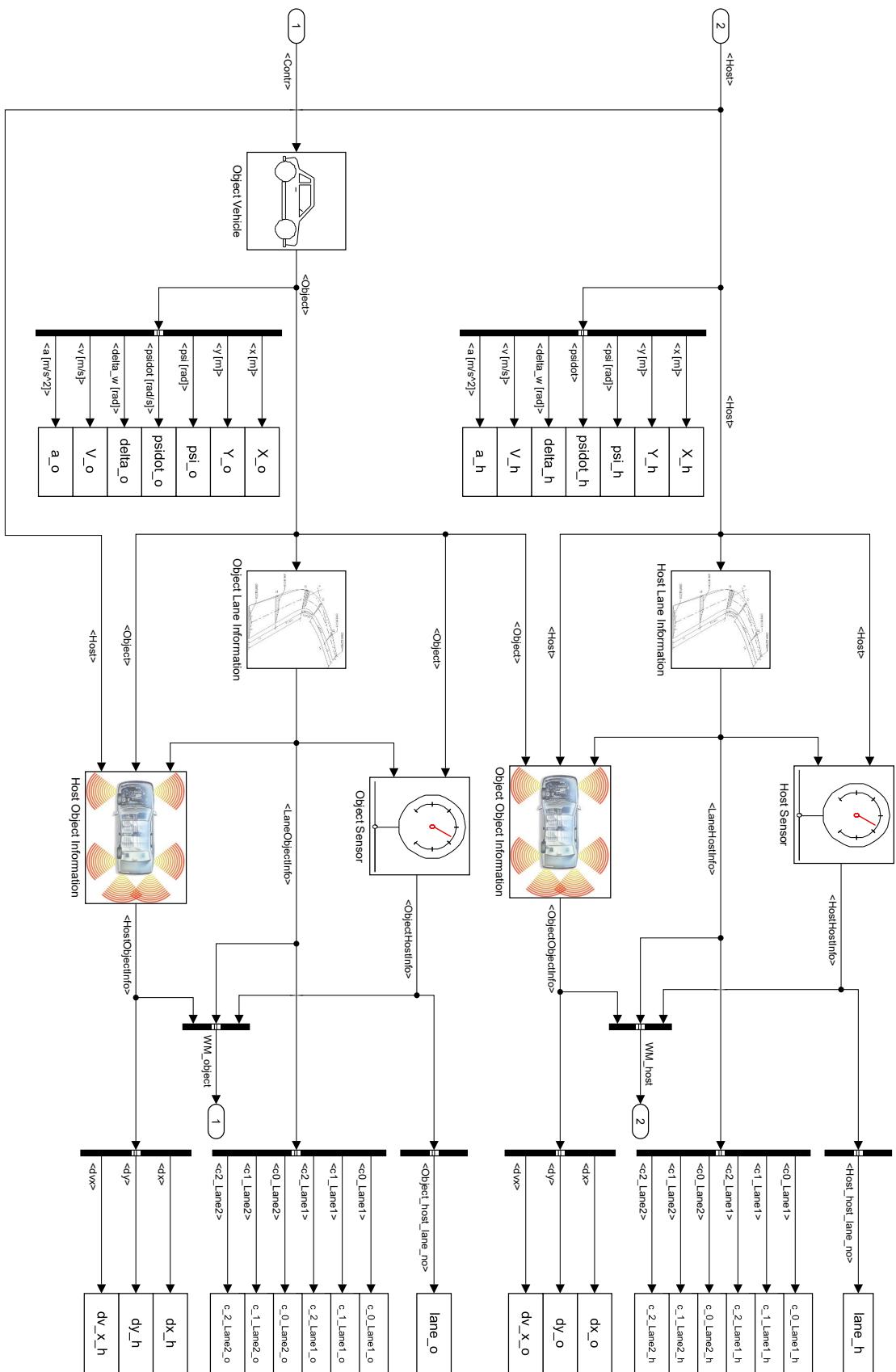


Figure D-2: World model

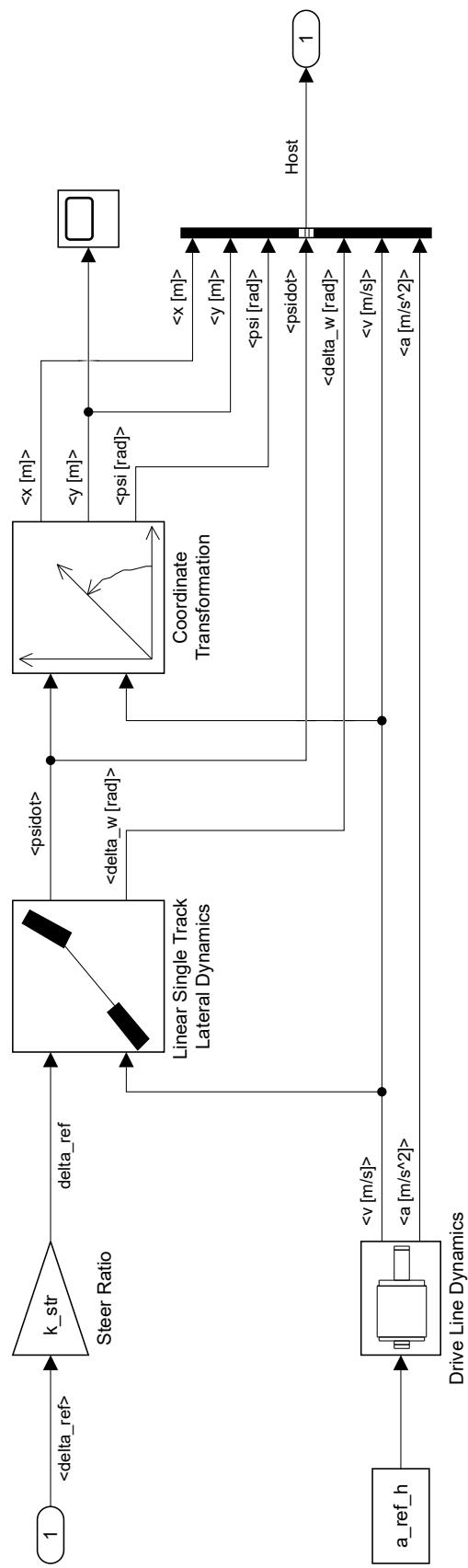


Figure D-3: Host vehicle model

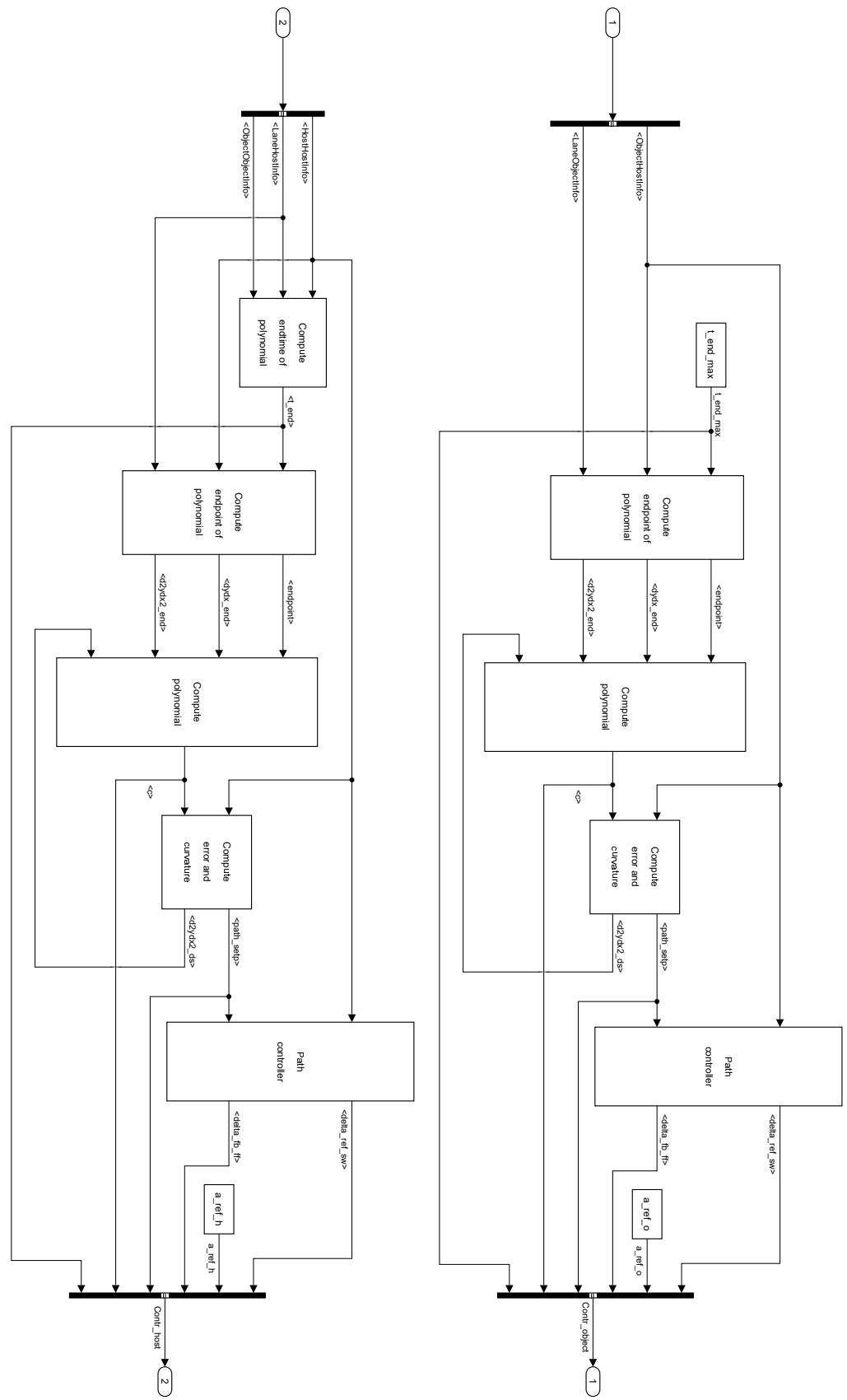


Figure D-4: APF MPC control system

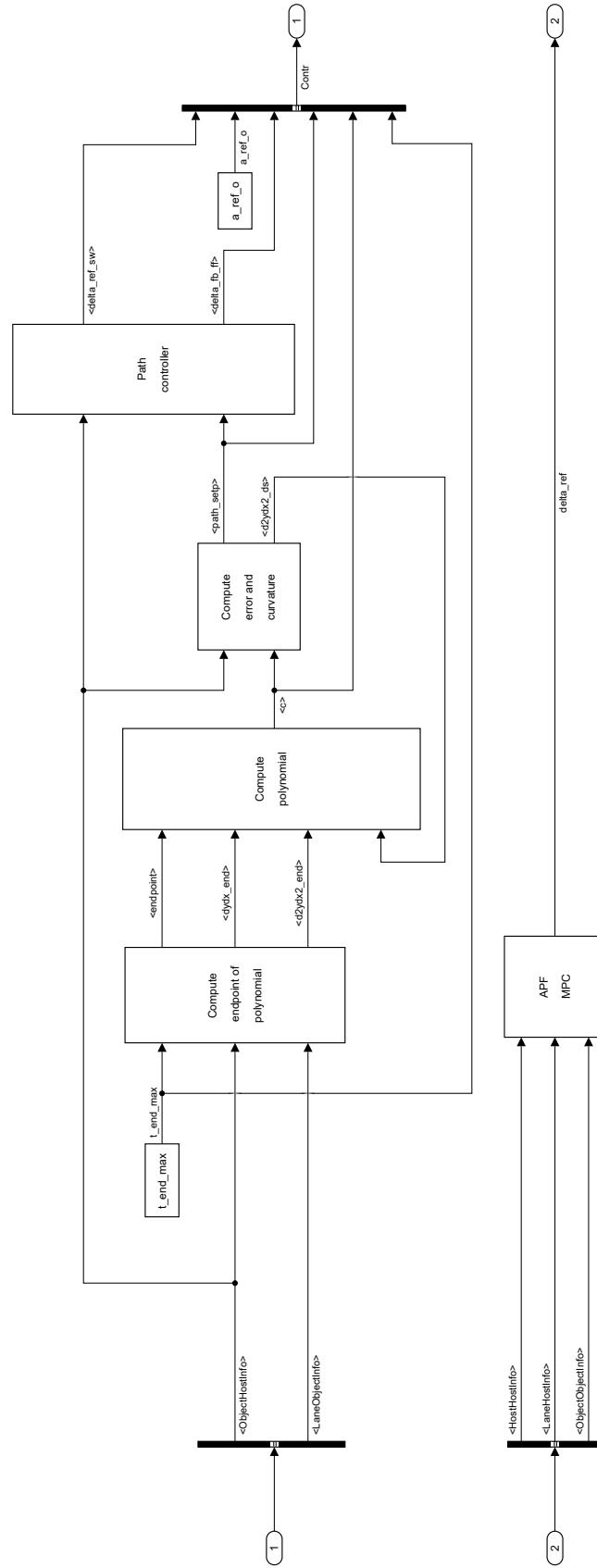


Figure D-5: TNO control system

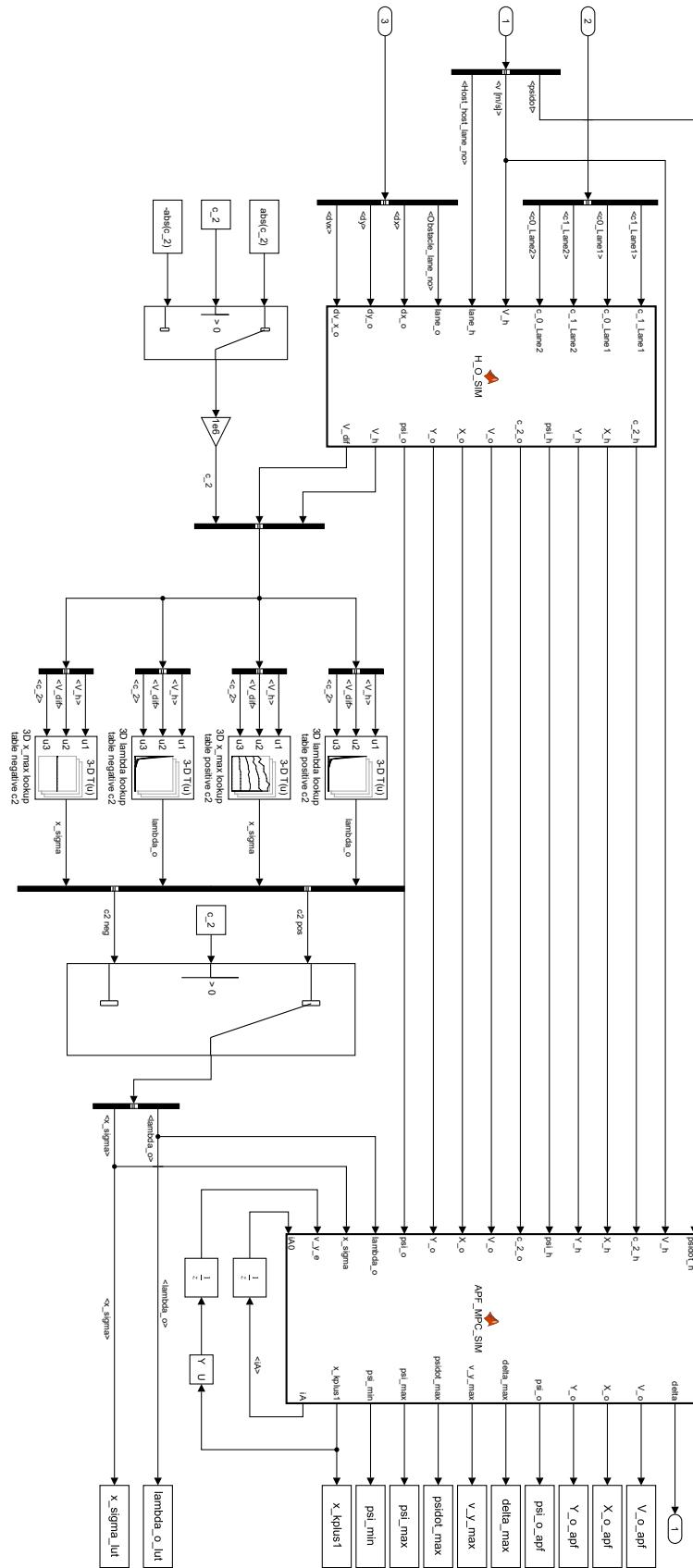


Figure D-6: APF MPC control system details

Appendix E

Tuned obstacle PF parameters

The tuned obstacle PF parameters λ_o and x_σ are reported in this Appendix.

Figure E-1 shows the tuned values for the obstacle PF weight λ_o against V_h for different $\Delta V_{o,h}$, where $\Delta V_{o,h}$ stands for the velocity difference between the host and object vehicle in $[\text{kmh}^{-1}]$. Clearly, the plotted values of λ_o and V_h are seen to be following a certain functional relationship, which is left to be determined in future research. The plotted values are displayed in Table E-1. It should be noted that these values only change for changing $\Delta V_{o,h}$ and stay the same for different values of c_2 .

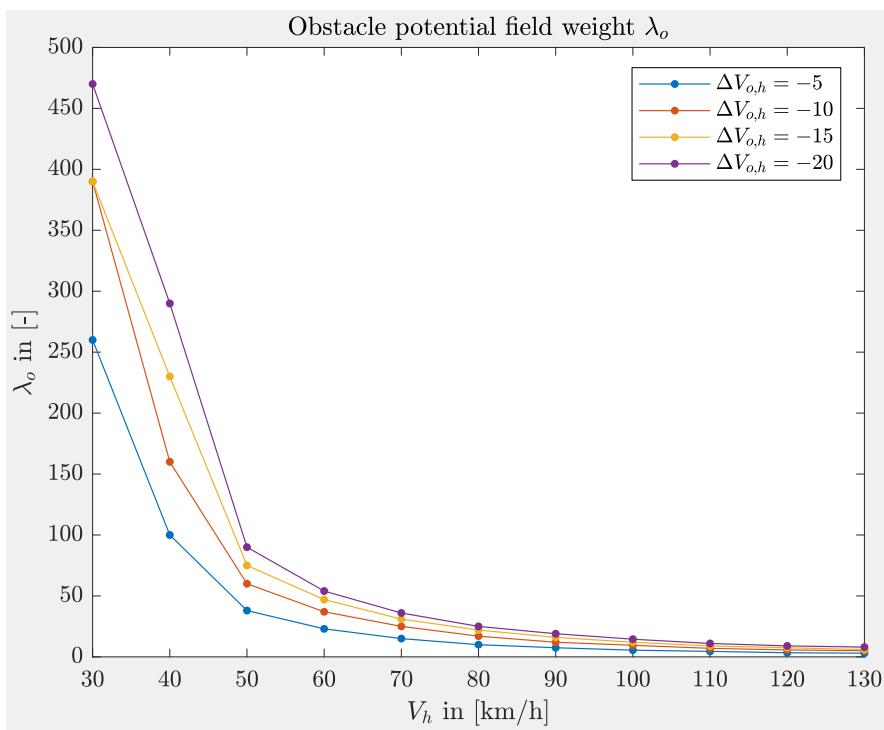


Figure E-1: Obstacle PF weight λ_o versus V_h for different $\Delta V_{o,h}$ for all c_2

Table E-1: λ_o in [-] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for all c_2

	Obstacle PF weight λ_o			
	$\Delta V_{o,h}$			
V_h	-5	-10	-15	-20
	4100	3000	3200	5000
20	260	390	390	470
30	100	160	230	290
40	38	60	75	90
50	23	37	47	54
60	15	25	31	36
70	10	17	22	25
80	7.5	12	16	19
90	5.5	9.5	12	14.5
100	4.5	7	9	11
110	3.4	5.7	7.5	9
120	3	5	6	8
130				

Table E-2 contains the values of x_σ for $c_2 = 0 \text{ m}^{-1}$. Figure E-2 and Figure E-3 show the results of the tuned values of x_σ against V_h , for different $\Delta V_{o,h}$ and the inspected positive and negative values of c_2 , respectively. Again, the plotted values indicate the existence of some analytic relation, this time between x_σ and V_h . Table E-3 to Table E-8 list the rest of the plotted x_σ data for the positive and negative values of c_2 . Red table entries indicate that even though the lane change is successfully performed for the corresponding σ_x , it somehow did not satisfy the tuning requirements completely. Empty table entries mean that no values for σ_x leading to good lane-changing behavior could be found at all.

Table E-2: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 0 \text{ m}^{-1}$

	Obstacle PF σ_x measure x_σ			
	$\Delta V_{o,h}$			
V_h	-5	-10	-15	-20
20	12	26	39	49
30	14	27	42	54
40	14	28	40	51
50	15	30	45	60
60	15	30	45	60
70	16	32	47	60
80	16	31	48	63
90	16	33	50	65
100	16	33	50	65
110	16	35	50	70
120	17	35	50	67
130	17	33	55	66

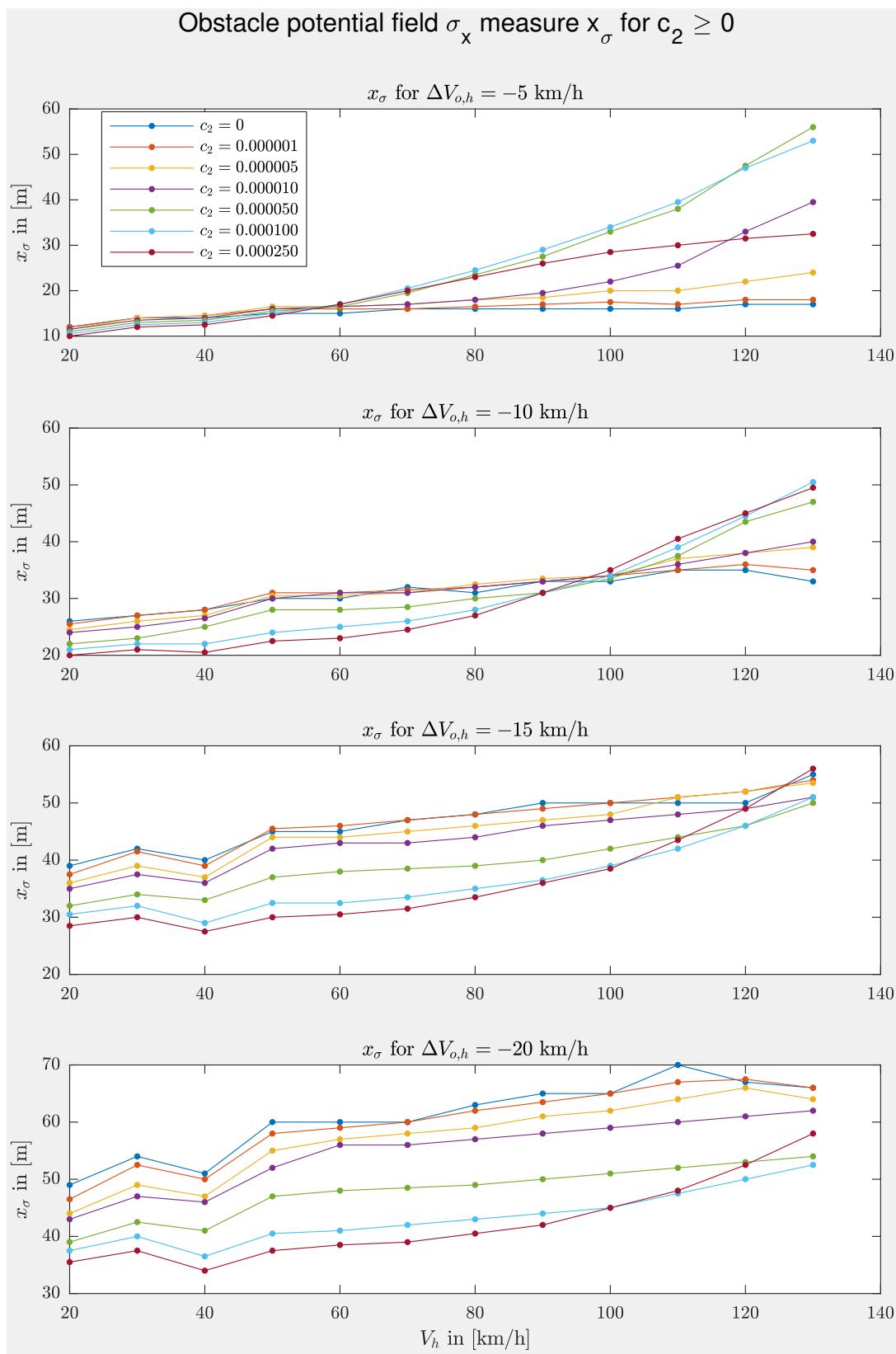


Figure E-2: Obstacle PF σ_x -measure x_σ versus V_h for different $\Delta V_{o,h}$ with $c_2 \geq 0$

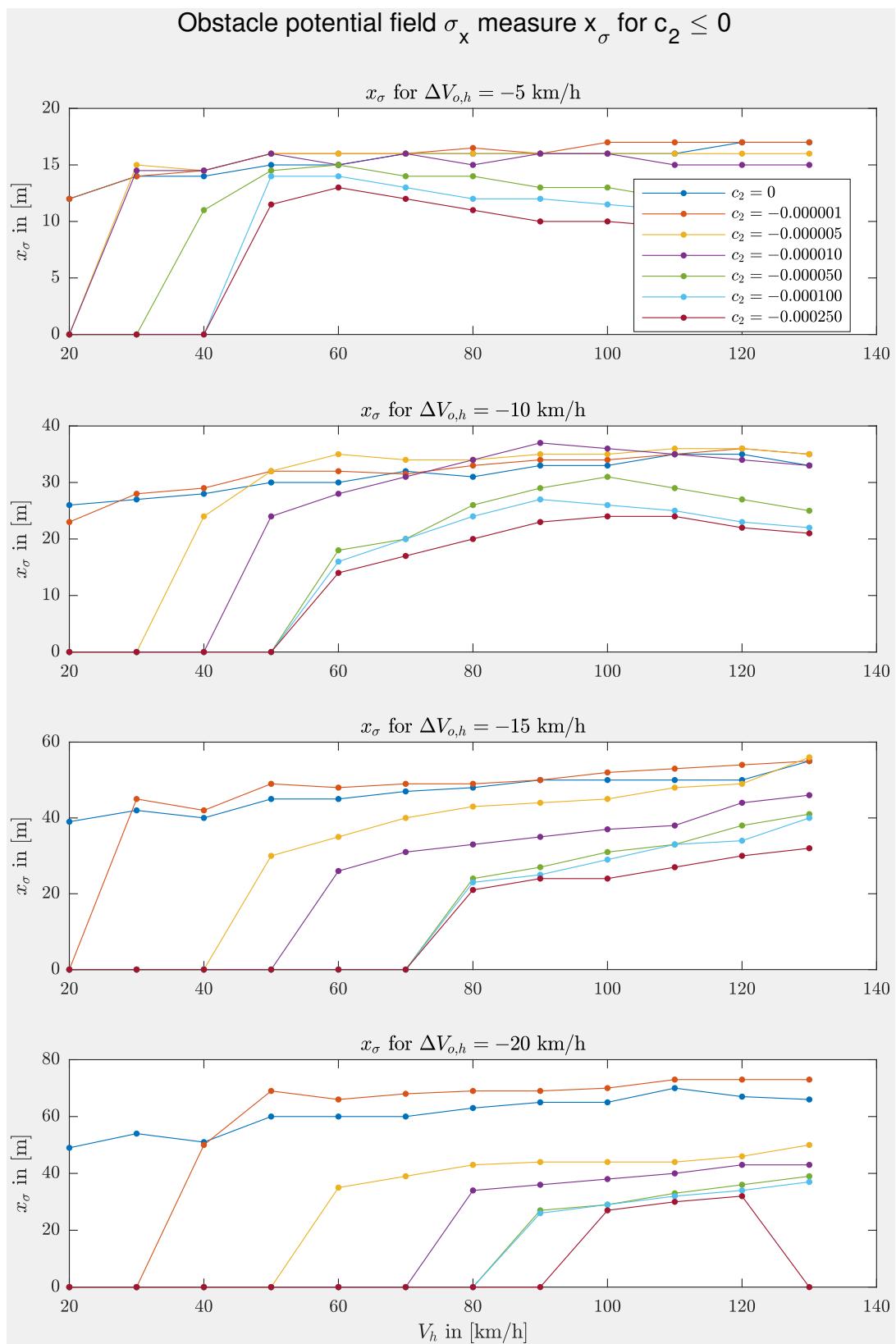


Figure E-3: Obstacle PF σ_x -measure x_σ versus V_h for different $\Delta V_{o,h}$ with $c_2 \leq 0$

Table E-3: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0\text{e}-6$ and $5.0\text{e}-6 \text{ m}^{-1}$

(a) For $c_2 = 1.0\text{e}-6 \text{ m}^{-1}$					(b) For $c_2 = 5.0\text{e}-6 \text{ m}^{-1}$				
Obstacle PF σ_x measure x_σ					Obstacle PF σ_x measure x_σ				
$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h	-5	-10	-15	-20	V_h	-5	-10	-15	-20
20	12	25.5	37.5	46.5	20	11.5	24.5	36	44
30	14	27	41.5	52.5	30	14	26	39	49
40	14.5	28	39	50	40	14.5	27	37	47
50	16	31	45.5	58	50	16.5	30.5	44	55
60	16	31	46	59	60	16.5	30.5	44	57
70	16	31.5	47	60	70	17	31	45	58
80	16.5	32	48	62	80	18	32.5	46	59
90	17	33	49	63.5	90	18.5	33.5	47	61
100	17.5	34	50	65	100	20	34	48	62
110	17	35	51	67	110	20	37	51	64
120	18	36	52	67.5	120	22	38	52	66
130	18	35	54	66	130	24	39	53.5	64

Table E-4: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0\text{e}-5$ and $5.0\text{e}-5 \text{ m}^{-1}$

(a) For $c_2 = 1.0\text{e}-5 \text{ m}^{-1}$					(b) For $c_2 = 5.0\text{e}-5 \text{ m}^{-1}$				
Obstacle PF σ_x measure x_σ					Obstacle PF σ_x measure x_σ				
$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h	-5	-10	-15	-20	V_h	-5	-10	-15	-20
20	11.5	24	35	43	20	11	22	32	39
30	13.5	25	37.5	47	30	13	23	34	42.5
40	14	26.5	36	46	40	13.5	25	33	41
50	16	30	42	52	50	15.5	28	37	47
60	16.5	31	43	56	60	16.5	28	38	48
70	17	31	43	56	70	19.5	28.5	38.5	48.5
80	18	32	44	57	80	23.5	30	39	49
90	19.5	33	46	58	90	27.5	31	40	50
100	22	34	47	59	100	33	33.5	42	51
110	25.5	36	48	60	110	38	37.5	44	52
120	33	38	49	61	120	47.5	43.5	46	53
130	29.5	40	51	62	130	56	47	50	54

Table E-5: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = 1.0\text{e-}4$ and $2.5\text{e-}4 \text{ m}^{-1}$

(a) For $c_2 = 1.0\text{e-}4 \text{ m}^{-1}$					(b) For $c_2 = 2.5\text{e-}4 \text{ m}^{-1}$				
Obstacle PF σ_x measure x_σ					Obstacle PF σ_x measure x_σ				
$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h	-5	-10	-15	-20	V_h	-5	-10	-15	-20
20	10.5	21	30.5	37.5	20	10	20	28.5	35.5
30	12.5	22	32	40	30	12	21	30	37.5
40	13	22	29	36.5	40	12.5	20.5	27.5	34
50	15	24	32.5	40.5	50	14.5	22.5	30	37.5
60	17	25	32.5	41	60	17	23	30.5	38.5
70	20.5	26	33.5	42	70	20	24.5	31.5	39
80	24.5	28	35	43	80	23	27	33.5	40.5
90	29	31	36.5	44	90	26	31	36	42
100	34	34	39	45	100	28.5	35	38.5	45
110	39.5	39	42	47.5	110	30	40.5	43.5	48
120	47	44.5	46	50	120	31.5	45	49	52.5
130	53	50.5	51	52.5	130	32.5	49.5	56	58

Table E-6: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0\text{e-}6$ and $-5.0\text{e-}6 \text{ m}^{-1}$

(a) For $c_2 = -1.0\text{e-}6 \text{ m}^{-1}$					(b) For $c_2 = -5.0\text{e-}6 \text{ m}^{-1}$				
Obstacle PF σ_x measure x_σ					Obstacle PF σ_x measure x_σ				
$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h	-5	-10	-15	-20	V_h	-5	-10	-15	-20
20	12	23			20				
30	14	28	45		30	15			
40	14.5	29	42	50	40	14.5	24		
50	16	32	49	69	50	16	32	30	
60	16	32	48	66	60	16	35	35	35
70	16	31.5	49	68	70	16	34	40	39
80	16.5	33	49	69	80	16	34	43	43
90	16	34	50	69	90	16	35	44	44
100	17	34	52	70	100	16	35	45	44
110	17	35	53	73	110	16	36	48	44
120	17	36	54	73	120	16	36	49	46
130	17	35	55	73	130	16	35	56	50

Table E-7: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0\text{e}-5$ and $-5.0\text{e}-5 \text{ m}^{-1}$

(a) For $c_2 = -1.0\text{e}-5 \text{ m}^{-1}$					(b) For $c_2 = -5.0\text{e}-5 \text{ m}^{-1}$						
Obstacle PF		σ_x	measure	x_σ	Obstacle PF		σ_x	measure	x_σ		
		$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h		-5	-10	-15	-20	V_h		-5	-10	-15	-20
20						20					
30		14.5				30					
40		14.5				40	11				
50	16	24				50	14.5				
60	15	28	26			60	15	18			
70	16	31	31			70	14	20			
80	15	34	33	34		80	14	26	24		
90	16	37	35	36		90	13	29	27	27	
100	16	36	37	38		100	13	31	31	29	
110	15	35	38	40		110	12	29	33	33	
120	15	34	44	43		120	12	27	38	36	
130	15	33	46	43		130	12	25	41	39	

Table E-8: x_σ in [m] against V_h and $\Delta V_{o,h}$ in [kmh^{-1}] for $c_2 = -1.0\text{e}-4$ and $-2.5\text{e}-4 \text{ m}^{-1}$

(a) For $c_2 = -1.0\text{e}-4 \text{ m}^{-1}$					(b) For $c_2 = -2.5\text{e}-4 \text{ m}^{-1}$						
Obstacle PF		σ_x	measure	x_σ	Obstacle PF		σ_x	measure	x_σ		
		$\Delta V_{o,h}$					$\Delta V_{o,h}$				
V_h		-5	-10	-15	-20	V_h		-5	-10	-15	-20
20						20					
30						30					
40						40					
50	14					50	11.5				
60	14	16				60	13	14			
70	13	20				70	12	17			
80	12	24	23			80	11	20	21		
90	12	27	25	26		90	10	23	24		
100	11.5	26	29	29		100	10	24	24	27	
110	11	25	33	32		110	9.5	24	27	30	
120	11	23	34	34		120	9.5	22	30	32	
130	11	22	40	37		130	9.5	21	32		

Appendix F

Lane-change characteristics results

This Appendix presents the simulation results for the lane-change characteristics.

Figure F-1 is repeated as a reminder of how the used lane-change characteristics are defined. Table F-1 contain the mean values of the lane-change characteristics for the simulations at $c_2 = 0 \text{ m}^{-1}$, with the corresponding data displayed in the box plots of Figure F-2. Furthermore, Table F-2 to Table F-7 list the mean values of the lane-change characteristics for the simulations at all other values for c_2 . Figure F-3 to Figure F-14 show the respective box plots.

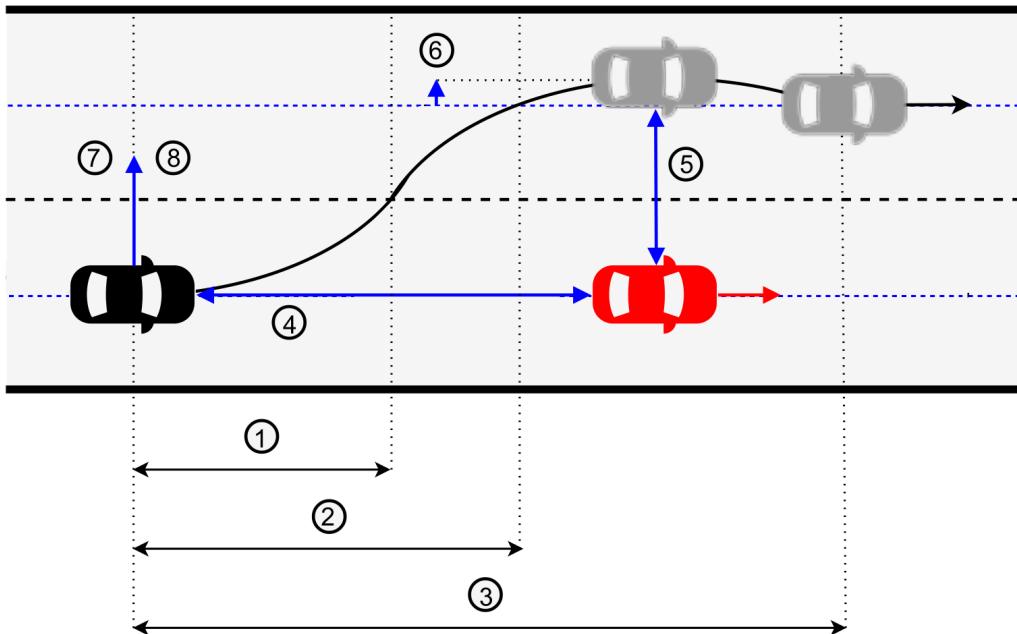


Figure F-1: Characteristics to evaluate the lane-change performance: 0) lane-change initiation, 1) rise time: t_r , 2) duration: t_d , 3) settling time: t_{se} , 4) longitudinal IVD: d_{lo} , 5) lateral IVD: d_{la} , 6) overshoot: d_o

Table F-1: Mean lane-change characteristics for $c_2 = 0 \text{ m}^{-1}$

Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	2.23	1.69
Duration in [s]	5.34	3.19
Settling time in [s]	15.10	11.41
Overshoot in [m]	0.10	0.36
Long. IVD in [m]	17.74	14.51
Lat. IVD in [m]	3.39	3.82
Min. IVD in [m]	3.37	3.78
a_y in ms^{-2}	-0.01	-0.03
Max. a_y in ms^{-2}	1.20	0.85
j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.32	0.10
Sim. time in [s]	5.32	1.62

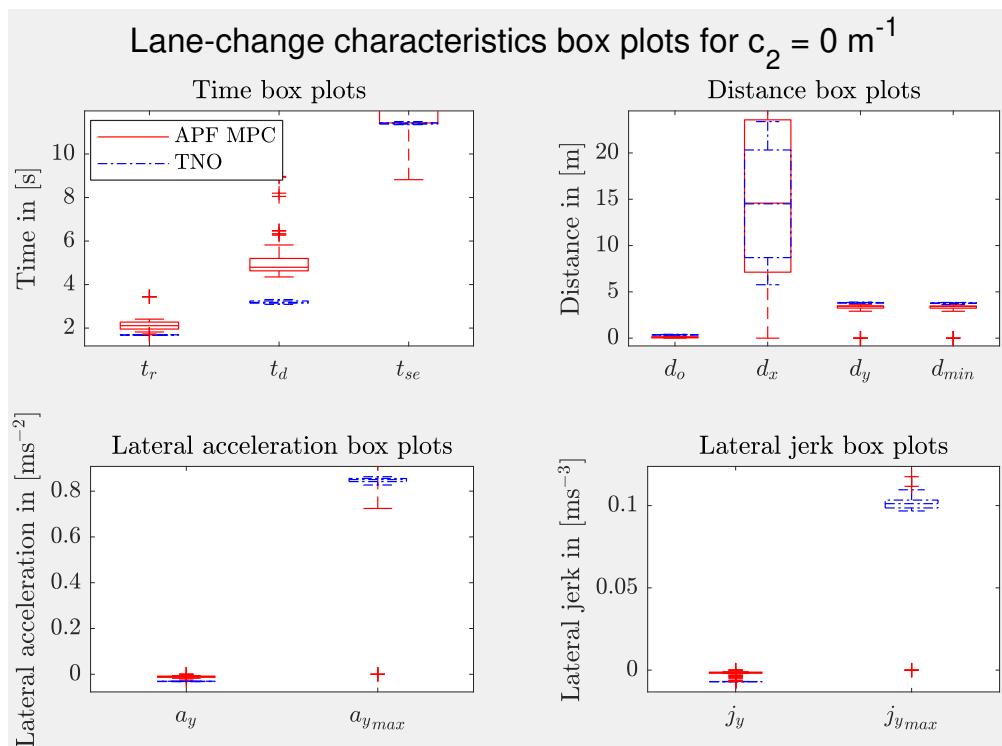
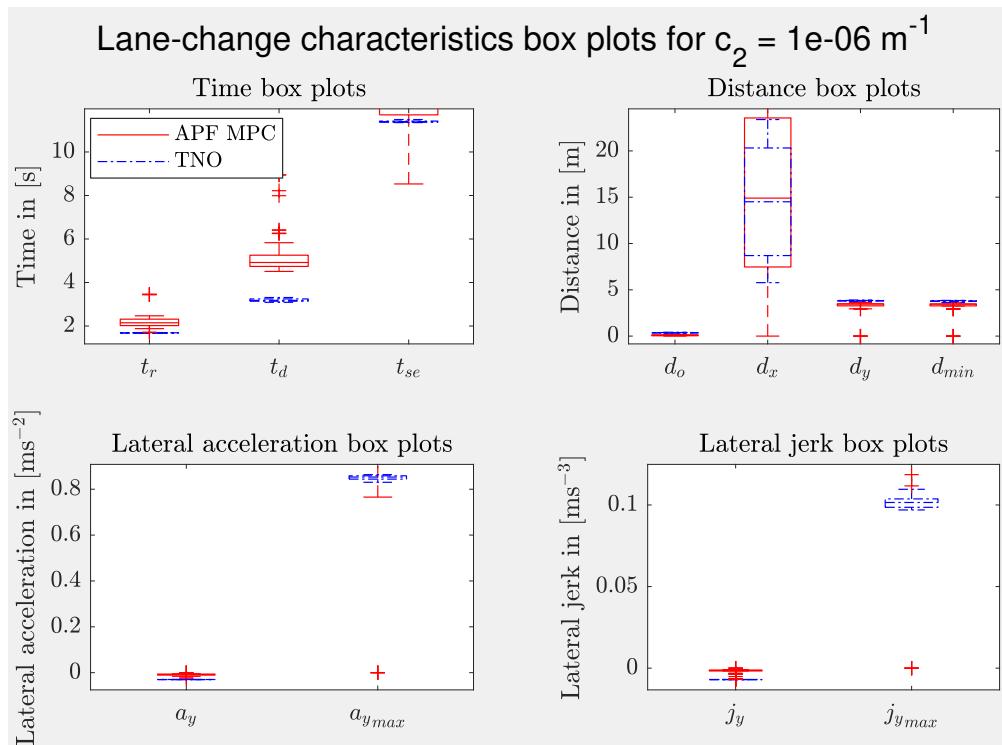
**Figure F-2:** Lane-change characteristics box plots for $c_2 = 0 \text{ m}^{-1}$

Table F-2: Mean lane-change characteristics for $c_2 = 1.0\text{e}-6$ and $5.0\text{e}-6 \text{ m}^{-1}$

(a) For $c_2 = 1.0\text{e}-6 \text{ m}^{-1}$			(b) For $c_2 = 5.0\text{e}-6 \text{ m}^{-1}$		
Mean lane-change characteristics			Mean lane-change characteristics		
Characteristic	Controller		Characteristic	Controller	
Characteristic	APF MPC	TNO	Characteristic	APF MPC	TNO
Rise time in [s]	2.27	1.68	Rise time in [s]	2.37	1.68
Duration in [s]	5.42	3.19	Duration in [s]	5.59	3.19
Settling time in [s]	15.43	11.40	Settling time in [s]	16.68	11.36
Overshoot in [m]	0.10	0.36	Overshoot in [m]	0.09	0.36
Long. IVD in [m]	18.30	14.49	Long. IVD in [m]	19.13	14.50
Lat. IVD in [m]	3.40	3.82	Lat. IVD in [m]	3.41	3.82
Min. IVD in [m]	3.39	3.78	Min. IVD in [m]	3.40	3.78
a_y in ms^{-2}	-0.01	-0.03	a_y in ms^{-2}	0.00	-0.03
Max. a_y in ms^{-2}	1.16	0.85	Max. a_y in ms^{-2}	1.10	0.85
j_y in ms^{-3}	0.00	-0.01	j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.30	0.10	Max. j_y in ms^{-3}	0.27	0.10
Sim. time in [s]	5.58	1.59	Sim. time in [s]	5.91	1.69

**Figure F-3:** Lane-change characteristics box plots for $c_2 = 1.0\text{e}-6 \text{ m}^{-1}$

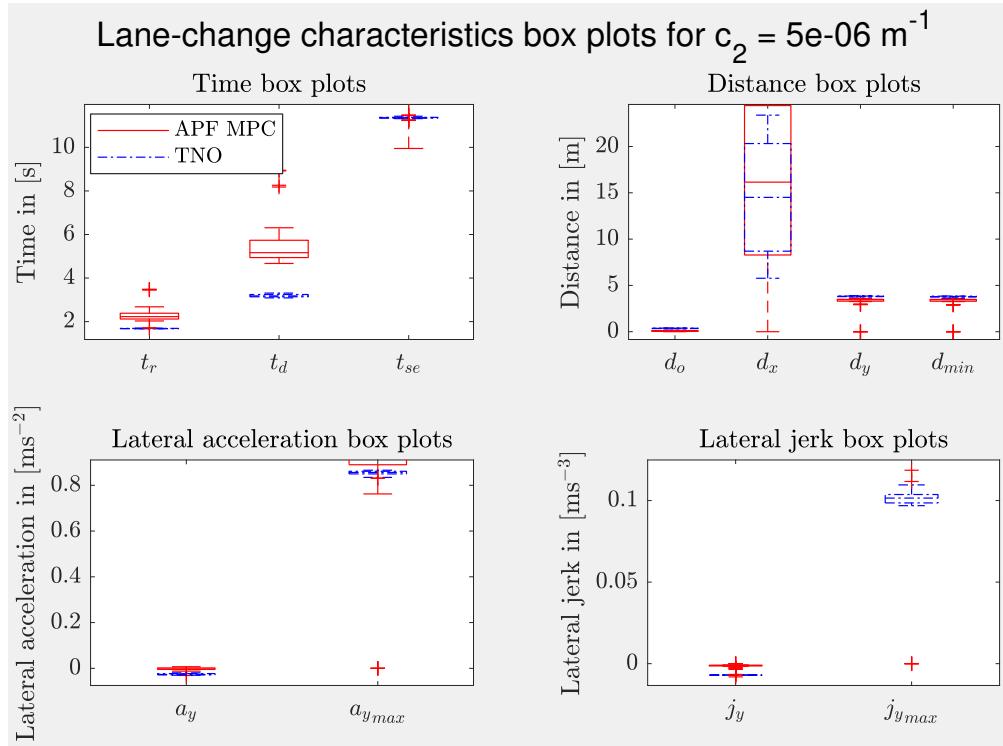


Figure F-4: Lane-change characteristics box plots for $c_2 = 5.0 \times 10^{-6} \text{ m}^{-1}$

Table F-3: Mean lane-change characteristics for $c_2 = 1.0 \times 10^{-5}$ and $5.0 \times 10^{-5} \text{ m}^{-1}$

(a) For $c_2 = 1.0 \times 10^{-5} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	2.45	1.69
Duration in [s]	5.77	3.19
Settling time in [s]	20.61	11.32
Overshoot in [m]	0.08	0.36
Long. IVD in [m]	19.58	14.51
Lat. IVD in [m]	3.41	3.82
Min. IVD in [m]	3.40	3.78
a_y in ms^{-2}	0.00	-0.02
Max. a_y in ms^{-2}	1.06	0.86
j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.26	0.10
Sim. time in [s]	6.75	1.93

(b) For $c_2 = 5.0 \times 10^{-5} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	2.74	1.69
Duration in [s]	6.19	3.20
Settling time in [s]	28.76	20.08
Overshoot in [m]	0.08	0.37
Long. IVD in [m]	21.98	14.54
Lat. IVD in [m]	3.42	3.82
Min. IVD in [m]	3.41	3.78
a_y in ms^{-2}	0.05	0.03
Max. a_y in ms^{-2}	0.99	0.90
j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.23	0.10
Sim. time in [s]	7.14	2.17

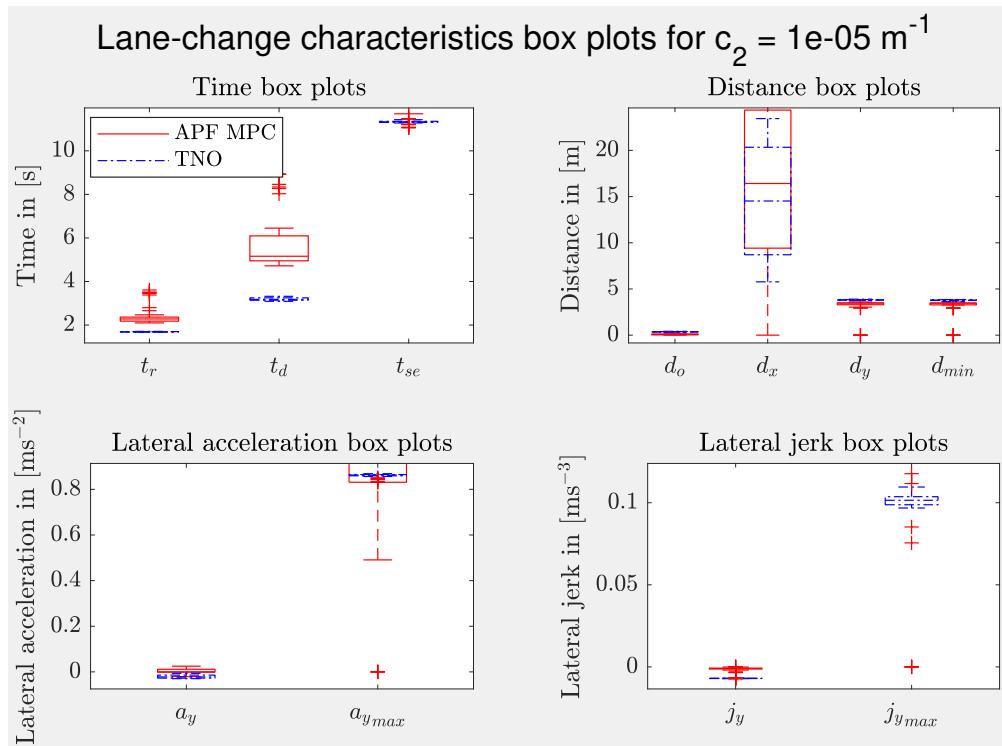


Figure F-5: Lane-change characteristics box plots for $c_2 = 1.0\text{e}-05 \text{ m}^{-1}$

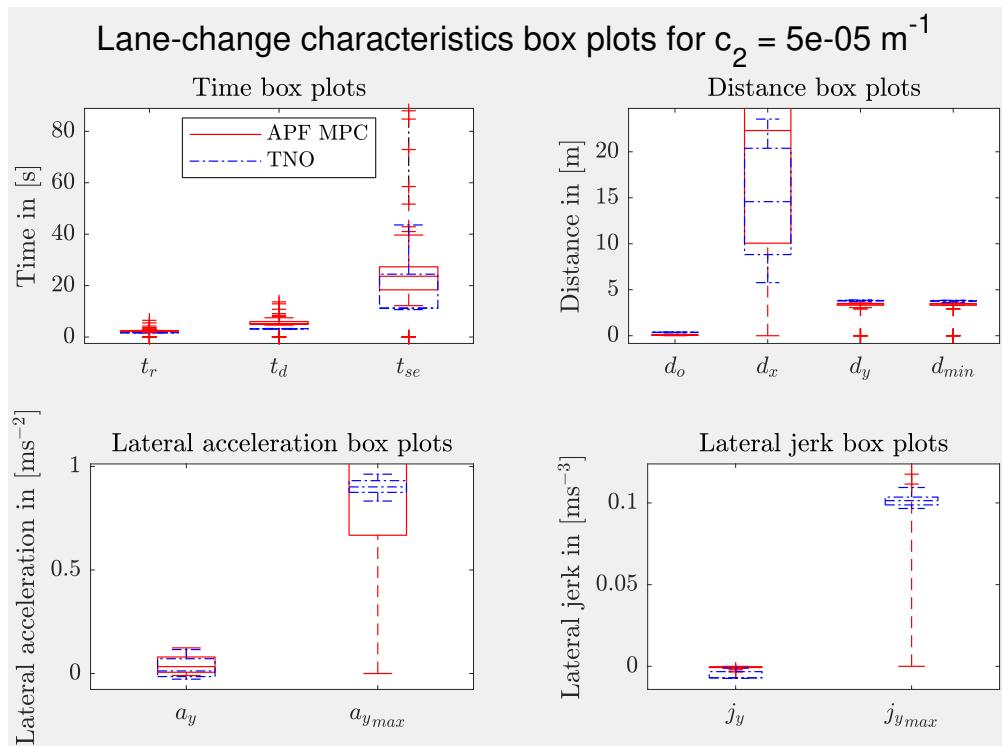
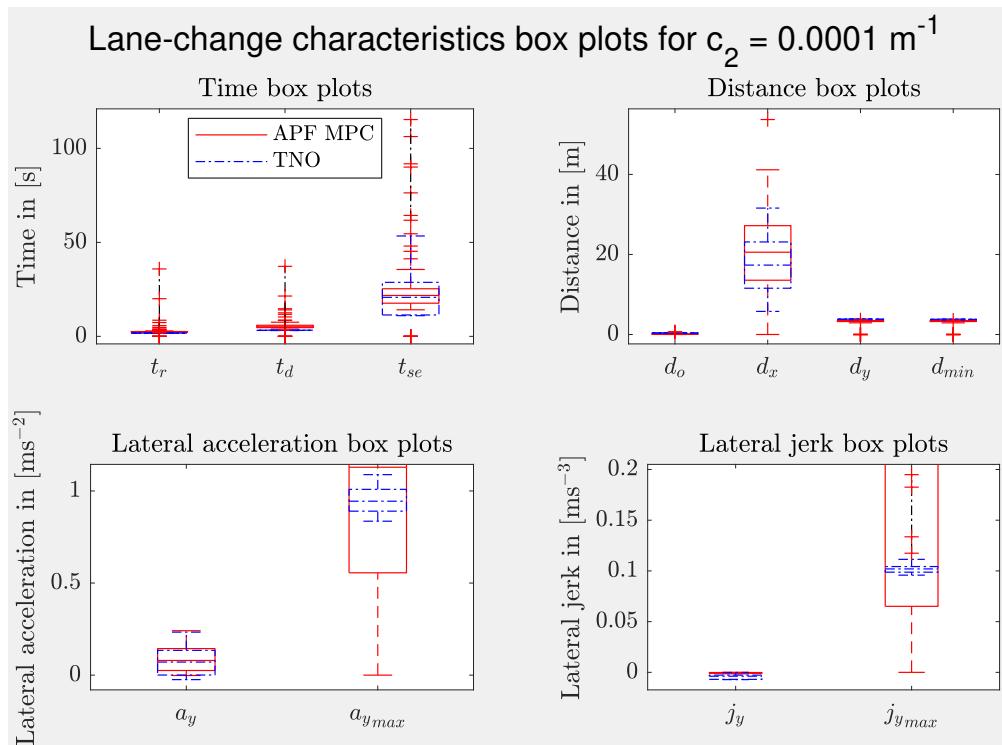


Figure F-6: Lane-change characteristics box plots for $c_2 = 5.0\text{e}-05 \text{ m}^{-1}$

Table F-4: Mean lane-change characteristics for $c_2 = 1.0e-4$ and $2.5e-4 \text{ m}^{-1}$

(a) For $c_2 = 1.0e-4 \text{ m}^{-1}$			(b) For $c_2 = 2.5e-4 \text{ m}^{-1}$		
Mean lane-change characteristics			Mean lane-change characteristics		
Characteristic	Controller		Characteristic	Controller	
Characteristic	APF MPC	TNO	Characteristic	APF MPC	TNO
Rise time in [s]	2.91	2.81	Rise time in [s]	3.10	5.50
Duration in [s]	6.33	4.30	Duration in [s]	6.22	6.94
Settling time in [s]	30.42	24.94	Settling time in [s]	30.50	27.51
Overshoot in [m]	0.11	0.40	Overshoot in [m]	0.17	0.59
Long. IVD in [m]	21.98	16.16	Long. IVD in [m]	24.04	24.34
Lat. IVD in [m]	3.40	3.81	Lat. IVD in [m]	3.27	3.78
Min. IVD in [m]	3.39	3.77	Min. IVD in [m]	3.26	3.75
a_y in ms^{-2}	0.11	0.08	a_y in ms^{-2}	0.22	0.18
Max. a_y in ms^{-2}	1.02	0.95	Max. a_y in ms^{-2}	0.98	1.07
j_y in ms^{-3}	0.00	0.00	j_y in ms^{-3}	-0.00	-0.00
Max. j_y in ms^{-3}	0.22	0.11	Max. j_y in ms^{-3}	0.17	0.13
Sim. time in [s]	6.70	1.75	Sim. time in [s]	5.79	1.60

**Figure F-7:** Lane-change characteristics box plots for $c_2 = 1.0e-4 \text{ m}^{-1}$

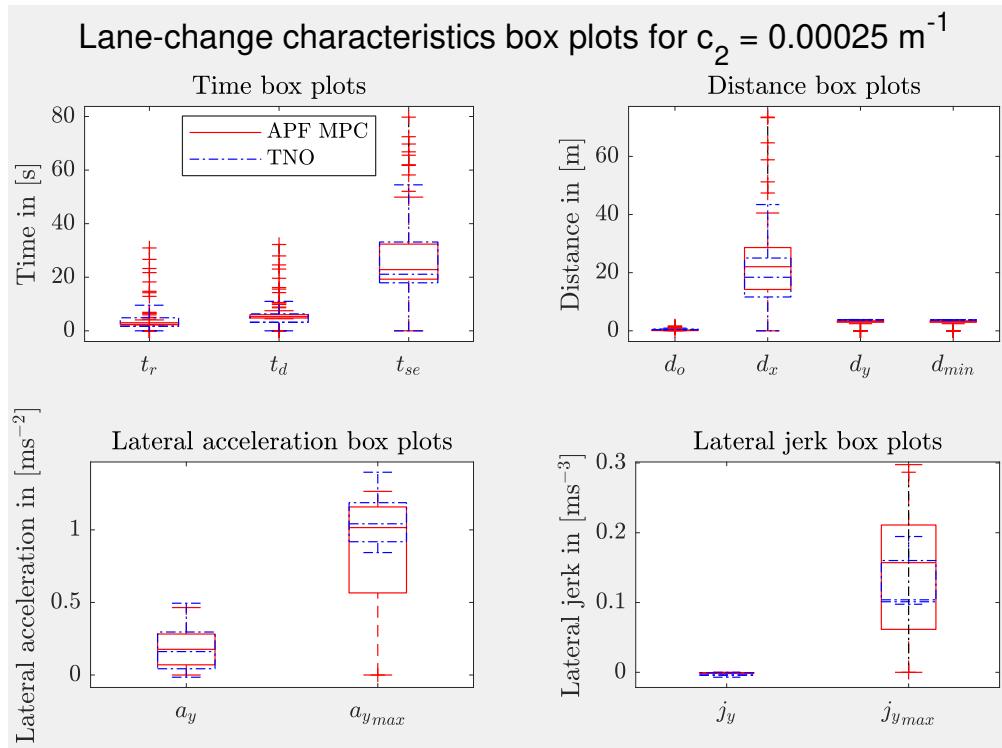


Figure F-8: Lane-change characteristics box plots for $c_2 = 2.5 \times 10^{-4} \text{ m}^{-1}$

Table F-5: Mean lane-change characteristics for $c_2 = -1.0 \times 10^{-6}$ and $-5.0 \times 10^{-6} \text{ m}^{-1}$

(a) For $c_2 = -1.0 \times 10^{-6} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	2.15	1.68
Duration in [s]	5.06	3.19
Settling time in [s]	14.40	11.41
Overshoot in [m]	0.11	0.36
Long. IVD in [m]	17.11	14.05
Lat. IVD in [m]	3.45	3.82
Min. IVD in [m]	3.43	3.78
a_y in ms^{-2}	-0.01	-0.03
Max. a_y in ms^{-2}	1.21	0.85
j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.31	0.10
Sim. time in [s]	6.61	1.88

(b) For $c_2 = -5.0 \times 10^{-6} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	1.99	1.68
Duration in [s]	4.62	3.19
Settling time in [s]	13.92	11.45
Overshoot in [m]	0.13	0.36
Long. IVD in [m]	12.20	13.74
Lat. IVD in [m]	3.22	3.82
Min. IVD in [m]	3.20	3.77
a_y in ms^{-2}	-0.02	-0.04
Max. a_y in ms^{-2}	1.31	0.84
j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.40	0.10
Sim. time in [s]	6.59	1.98

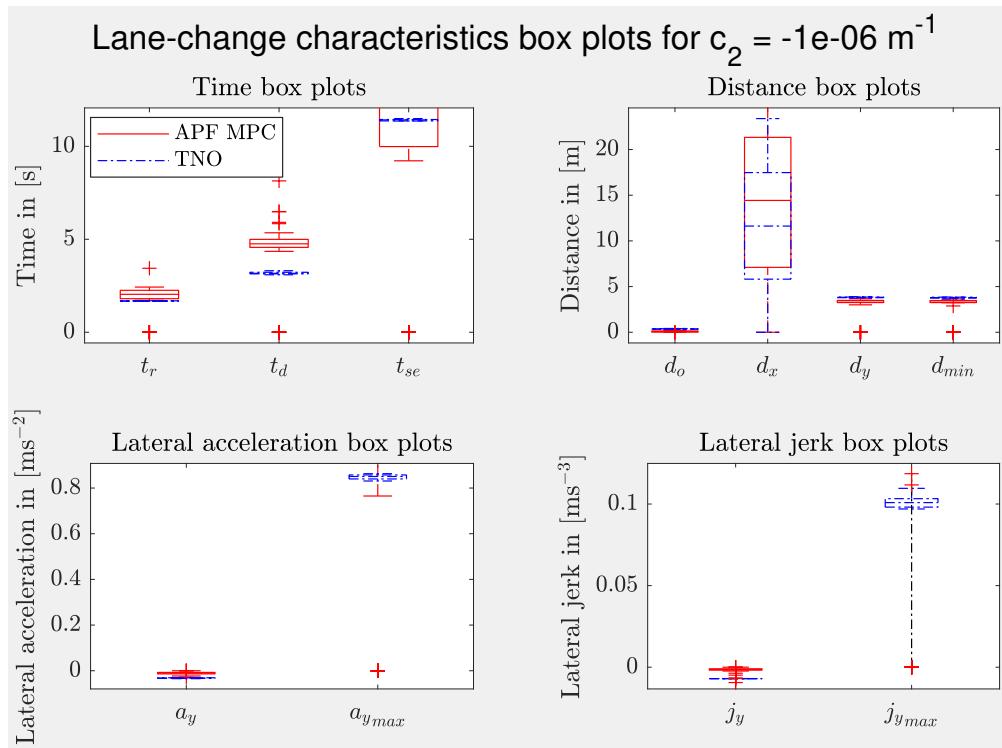


Figure F-9: Lane-change characteristics box plots for $c_2 = -1.0\text{e}-6 \text{ m}^{-1}$

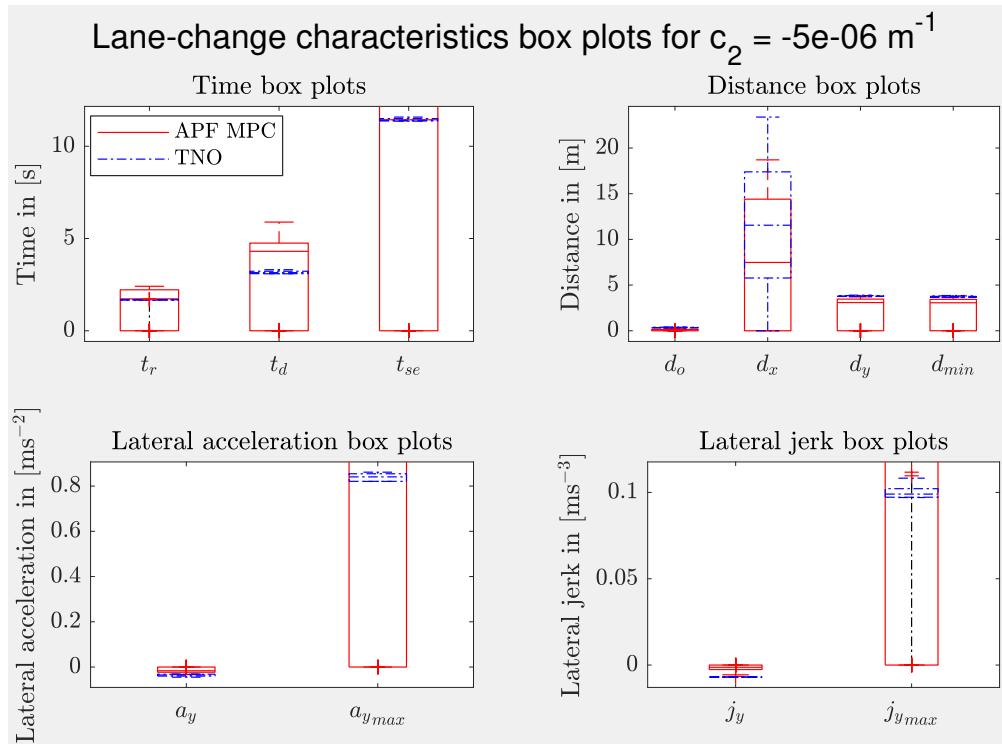
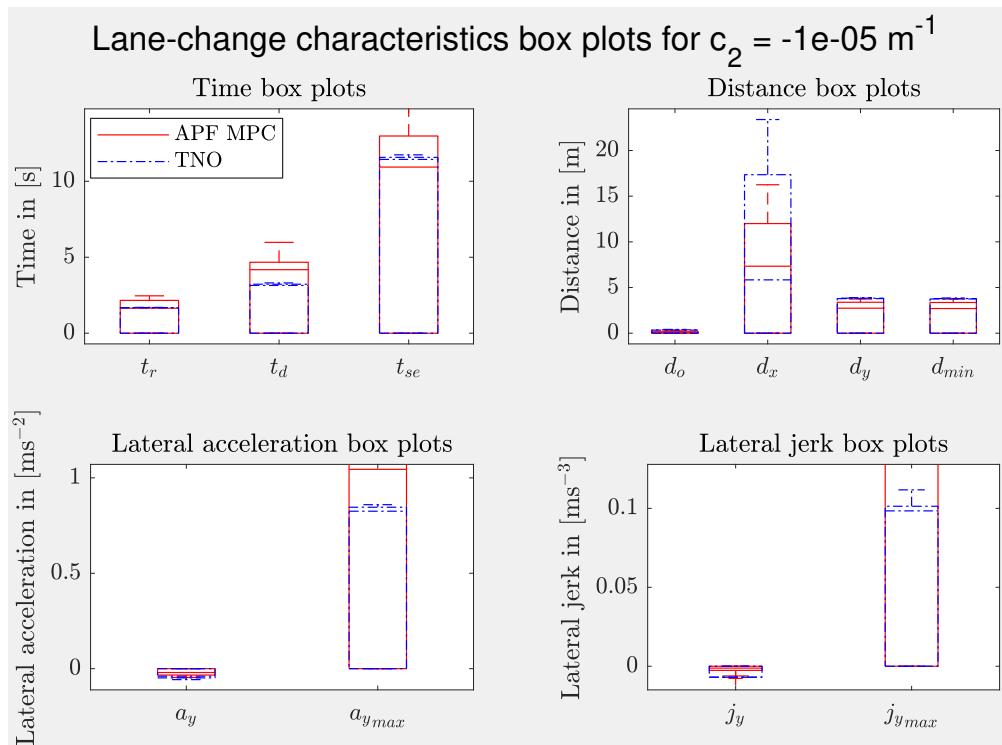


Figure F-10: Lane-change characteristics box plots for $c_2 = -5.0\text{e}-6 \text{ m}^{-1}$

Table F-6: Mean lane-change characteristics for $c_2 = -1.0e-5$ and $-5.0e-5 \text{ m}^{-1}$

(a) For $c_2 = -1.0e-5 \text{ m}^{-1}$			(b) For $c_2 = -5.0e-5 \text{ m}^{-1}$		
Mean lane-change characteristics			Mean lane-change characteristics		
Characteristic	Controller		Characteristic	Controller	
Characteristic	APF MPC	TNO	Characteristic	APF MPC	TNO
Rise time in [s]	1.96	1.68	Rise time in [s]	2.00	1.67
Duration in [s]	4.56	3.20	Duration in [s]	4.35	3.20
Settling time in [s]	13.44	11.52	Settling time in [s]	19.98	16.09
Overshoot in [m]	0.14	0.36	Overshoot in [m]	0.19	0.35
Long. IVD in [m]	10.60	13.15	Long. IVD in [m]	9.59	12.80
Lat. IVD in [m]	3.10	3.82	Lat. IVD in [m]	3.02	3.81
Min. IVD in [m]	3.06	3.77	Min. IVD in [m]	2.97	3.76
a_y in ms^{-2}	-0.03	-0.04	a_y in ms^{-2}	-0.09	-0.10
Max. a_y in ms^{-2}	1.30	0.84	Max. a_y in ms^{-2}	1.28	0.77
j_y in ms^{-3}	0.00	-0.01	j_y in ms^{-3}	0.00	-0.01
Max. j_y in ms^{-3}	0.43	0.10	Max. j_y in ms^{-3}	0.47	0.10
Sim. time in [s]	5.96	1.83	Sim. time in [s]	5.46	1.78

**Figure F-11:** Lane-change characteristics box plots for $c_2 = -1.0e-5 \text{ m}^{-1}$

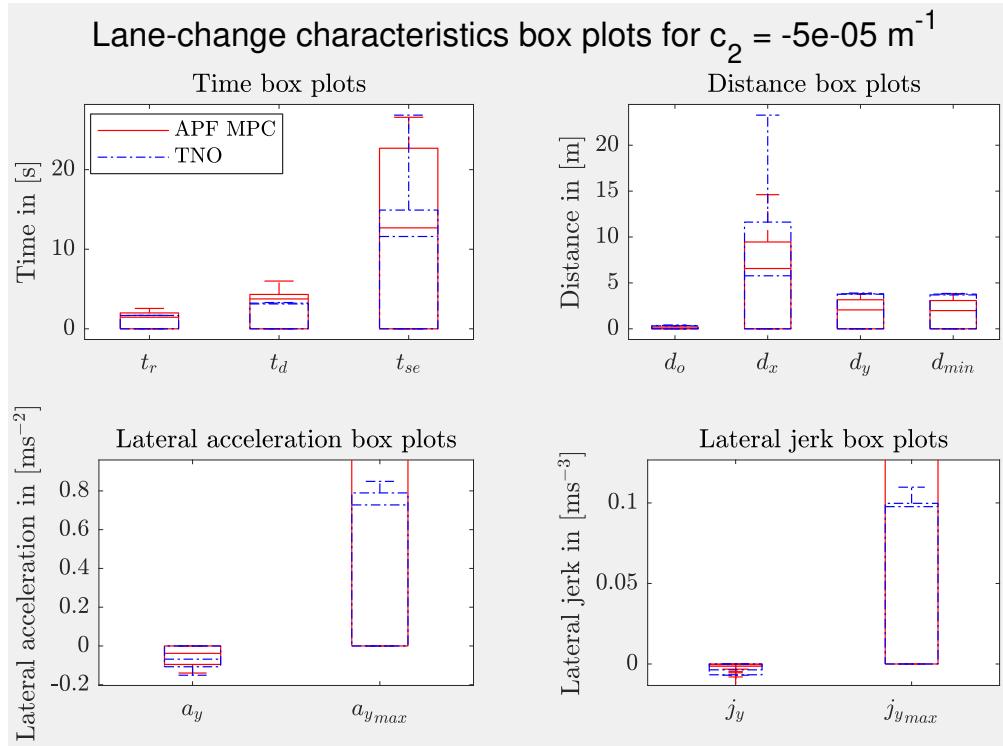


Figure F-12: Lane-change characteristics box plots for $c_2 = -5.0 \times 10^{-5} \text{ m}^{-1}$

Table F-7: Mean lane-change characteristics for $c_2 = -1.0 \times 10^{-4}$ and $-2.5 \times 10^{-4} \text{ m}^{-1}$

(a) For $c_2 = -1.0 \times 10^{-4} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	1.68	1.68
Duration in [s]	3.30	3.30
Settling time in [s]	18.15	18.15
Overshoot in [m]	0.31	0.31
Long. IVD in [m]	23.28	23.28
Lat. IVD in [m]	3.75	3.75
Min. IVD in [m]	3.75	3.75
a_y in ms^{-2}	-0.15	-0.15
Max. a_y in ms^{-2}	0.70	0.70
j_y in ms^{-3}	0.00	0.00
Max. j_y in ms^{-3}	0.10	0.10
Sim. time in [s]	1.59	1.59

(b) For $c_2 = -2.5 \times 10^{-4} \text{ m}^{-1}$

Mean lane-change characteristics		
Characteristic	Controller	
	APF MPC	TNO
Rise time in [s]	5.15	5.23
Duration in [s]	6.87	6.74
Settling time in [s]	20.86	21.75
Overshoot in [m]	0.44	0.32
Long. IVD in [m]	16.89	23.36
Lat. IVD in [m]	3.33	3.77
Min. IVD in [m]	3.21	3.73
a_y in ms^{-2}	-0.36	-0.37
Max. a_y in ms^{-2}	1.30	0.99
j_y in ms^{-3}	0.00	0.00
Max. j_y in ms^{-3}	0.48	0.13
Sim. time in [s]	4.95	1.66

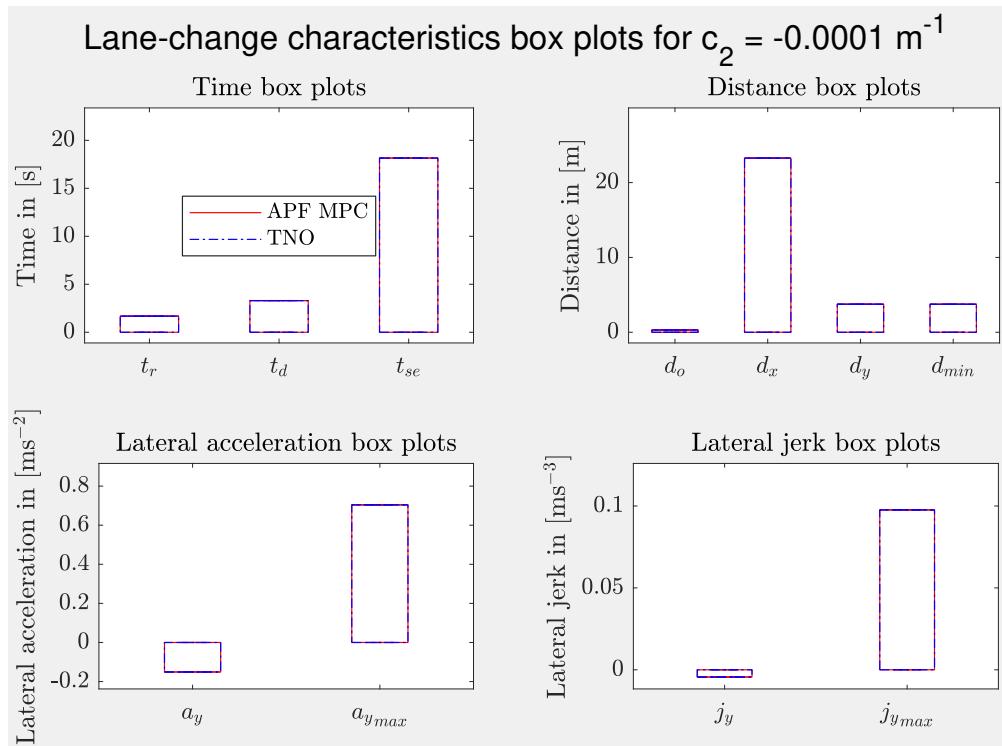


Figure F-13: Lane-change characteristics box plots for $c_2 = -1.0e-4 \text{ m}^{-1}$

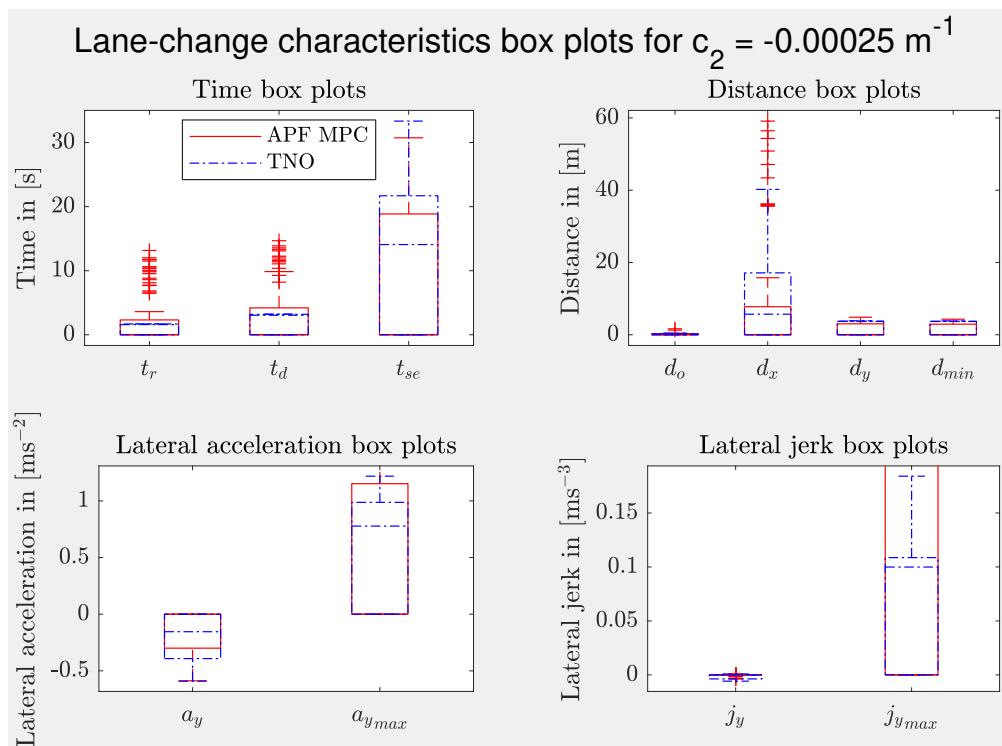


Figure F-14: Lane-change characteristics box plots for $c_2 = -2.5e-4 \text{ m}^{-1}$

Bibliography

- [1] Centraal Bureau voor de Statistiek, “Statistics Netherlands: More cars, fewer freight vehicles,” <https://www.cbs.nl/en-gb/news/2015/26/statistics-netherlands-more-cars-fewer-freight-vehicles>, 2015.
- [2] W. Jansen, “Lateral path-following control for automated vehicle platoons,” *MSc Thesis, Delft University of Technology*, 2016.
- [3] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, “Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing,” *American Control Conference*, vol. 1-13, p. 3910, 2007.
- [4] Z. Huang, Q. Wu, J. Ma, and S. Fan, “An APF and MPC combined collaborative driving controller using vehicular communication technologies,” *Chaos, Solitons and Fractals*, vol. 89, pp. 232–242, 2015.
- [5] N. H. Amer, H. Zamzuri, K. Hudha, and Z. A. Kadir, “Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, pp. 1–30, 2016.
- [6] S. Dominguez, A. Ali, G. Garcia, and P. Martinet, “Comparison of lateral controllers for autonomous vehicle: Experimental results,” *International Conference on Intelligent Transportation Systems*, pp. 1418–1423, 2016.
- [7] S. Magdici and M. Althoff, “Fail-safe motion planning of autonomous vehicles,” *International Conference on Intelligent Transportation Systems*, pp. 452–458, 2016.
- [8] E. J. Rossetter, “A potential field framework for active vehicle lanekeeping assistance,” *PhD Thesis, Stanford University*, 2003.
- [9] S. Arrigoni, “MPC-based framework for autonomous ground vehicles in a complex environment,” *The Dynamics of Vehicles on Roads and Tracks*, pp. 669–678, 2013.

- [10] E. Bauer, F. Lotz, M. Pfromm, M. Schreier, B. Abendroth, S. Cieler, A. Eckert, A. Hohm, S. Lüke, P. Rieth, V. Willert, and J. Adamy, "PRORETA 3: An integrated approach to collision avoidance and vehicle automation," *Automatisierungstechnik*, vol. 60, no. 12, pp. 755–765, 2012.
- [11] R. Dariani, S. Schmidt, and R. Kasper, "Optimal control based approach for autonomous driving," *International Conference on Emerging Technologies and Factory Automation*, 2016.
- [12] New Atlas, "Self-driving vehicles: What are the six levels of autonomy?," <https://newatlas.com/sae-autonomous-levels-definition-self-driving/49947/>, 2017.
- [13] TNO, "Research on integrated vehicle safety," <https://goo.gl/hFmvge>, 2017.
- [14] Helmond, "Automotive Campus," <http://www.automotivcampus.com>, 2017.
- [15] S. G. Anavatti, S. L. Francis, and M. Garratt, "Path-planning modules for autonomous vehicles: Current status and challenges," *International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation*, pp. 205–214, 2015.
- [16] J. C. Gerdes and E. J. Rossetter, "A unified approach to driver assistance systems based on artificial potential fields," *Journal of Dynamic Systems, Measurement, and Control*, vol. 123, no. 3, pp. 431–438, 2001.
- [17] H. Jiang, Z. Wang, Q. Chen, and J. Zhu, "Obstacle avoidance of autonomous vehicles with CQP-based model predictive control," *International Conference on Systems, Man, and Cybernetics*, pp. 1668–1673, 2016.
- [18] B. Gutjahr, L. Groll, and M. Werling, "Lateral vehicle trajectory optimization using constrained linear time-varying MPC," *Transactions on Intelligent Transportation Systems*, pp. 1–10, 2016.
- [19] J. Giesbrecht, "Global path planning for unmanned ground vehicles," *Defense Technical Information Center*, 2004.
- [20] M. Brown, J. Funke, S. Erlien, and J. C. Gerdes, "Safe driving envelopes for path tracking in autonomous vehicles," *Control Engineering Practice*, vol. 61, pp. 307–316, 2017.
- [21] W. Chee and M. Tomizuka, "Vehicle lane change maneuver in automated highway systems," *California Partners for Advanced Transit and Highways*, 1994.
- [22] J. Godthelp, "Precognitive control: Open- and closed-loop steering in a lane-change manoeuvre," *Ergonomics*, vol. 28, no. 10, pp. 1419–1438, 1985.
- [23] T. Heil, A. Lange, and S. Cramer, "Adaptive and efficient lane change path planning for automated vehicles," *International Conference on Intelligent Transportation Systems*, pp. 479–484, 2016.
- [24] J. Xu, K. Yang, Y. Shao, and G. Lu, "An experimental study on lateral acceleration of cars in different environments in sichuan, southwest china," *Discrete Dynamics in Nature and Society*, 2015.

- [25] S. Khanna and C. Justo, "Highway engineering," *Book from Nem Chand & Bros*, 1991.
- [26] Rijkswaterstaat, "Richtlijn ontwerp autosnelwegen," <https://goo.gl/3NRyvx>, 2014.
- [27] K. Osman, J. Ghommam, and M. Saad, "Combined road following control and automatic lane keeping for automated guided vehicles," *International Conference on Control, Automation, Robotics and Vision*, 2016.
- [28] S. Zhang, W. Deng, Q. Zhao, H. Sun, and B. Litkouhi, "Dynamic trajectory planning for vehicle autonomous driving," *International Conference on Systems, Man, and Cybernetics*, pp. 4161–4166, 2013.
- [29] M. Atagoziyev, K. W. Schmidt, and E. G. Schmidt, "Lane change scheduling for autonomous vehicles," *IFAC-PapersOnLine*, vol. 49, no. 3, pp. 61–66, 2016.
- [30] C. Huang, F. Naghy, and H. Du, "Model predictive control-based lane change control system for an autonomous vehicle," *Region 10 Conference*, pp. 3349–3354, 2016.
- [31] L. Iftekhar and R. Olfati-Saber, "Autonomous driving for vehicular networks with nonlinear dynamics," *Intelligent Vehicles Symposium*, pp. 723–729, 2012.
- [32] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 952–964, 2017.
- [33] A. Kanaris, E. B. Kosmatopoulos, and P. A. Ioannou, "Strategies and spacing requirements for lane changing and merging in automated highway systems," *Transactions on Vehicular Technology*, vol. 50, no. 6, pp. 1568–1581, 2001.
- [34] M. Alirezaei, S. Jansen, A. Schmeitz, and A. Madhusudhanan, "Collision avoidance system using state dependent riccati equation technique: An experimental robustness evaluation," *International Symposium on Advanced Vehicle Control*, p. 127, 2016.
- [35] C. Katrakazas, M. Quddus, W. H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015.
- [36] A. Ollero and O. Amidi, "Predictive path tracking of mobile robots. application to the CMU NavLab," *International Conference on Advanced Robotics*, vol. 2, pp. 1081–1086, 1991.
- [37] M. Gidlewski and D. Źardecki, "Investigation of vehicle motion control process due to the linearization of the lateral dynamics reference model used in the controller," *Mechanics Research Communications*, 2016.
- [38] M. Alirezaei and J. Zegers, "Lateral vehicle control (overtake)," *TNO report*, 2016.
- [39] J. M. Snider, "Automatic steering methods for autonomous automobile path tracking," *Robotics Institute Pittsburgh Technical Report CMU-RITR-09-08*, 2009.

- [40] J. Agirrebeitia, R. Avilés, I. F. de Bustos, and G. Ajuria, “A new apf strategy for path planning in environments with obstacles,” *Mechanism and Machine Theory*, vol. 40, no. 6, pp. 645–658, 2005.
- [41] P. Raksincharoensak, T. Ehira, K. Shimono, and Y. Taga, “Autonomous vehicle trajectory planning and control based on virtual disturbance via simulation of feedback control systems,” 2016.
- [42] J. C. Gerdes, E. J. Rossetter, and U. Saur, “Combining lanekeeping and vehicle following with hazard maps,” *Vehicle System Dynamics*, vol. 36, no. 4-5, pp. 391–411, 2010.
- [43] Y. Koren and J. Borenstein, “Potential field methods and their inherent limitations for mobile robot navigation,” *International Conference on Robotics and Automation*, vol. 2, pp. 1398–1404, 1991.
- [44] F. Bounini, D. Gingras, H. Pollart, and D. Gruyer, “Modified artificial potential field method for online path planning applications,” *Intelligent Vehicles Symposium*, pp. 180–185, 2017.
- [45] J. Amiryani and M. Jamzad, “Adaptive motion planning with artificial potential fields using a prior path,” *International Conference on Robotics and Mechatronics*, pp. 731–736, 2015.
- [46] E. Galceran, R. M. Eustice, and E. Olson, “Toward integrated motion planning and control using potential fields and torque-based steering actuation for autonomous driving,” *Intelligent Vehicles Symposium*, pp. 304–309, 2015.
- [47] E. Semsar-Kazerooni, J. Verhaegh, J. Ploeg, and M. Alirezaei, “Cooperative adaptive cruise control: An artificial potential field approach,” *Intelligent Vehicles Symposium*, pp. 361–367, 2016.
- [48] M. T. Wolf and J. W. Burdick, “Artificial potential functions for highway driving with collision avoidance,” *International Conference on Robotics and Automation*, pp. 3731–3736, 2008.
- [49] J. Funke, M. Brown, S. M. Erlien, and J. C. Gerdes, “Collision avoidance and stabilization for autonomous vehicles in emergency scenarios,” *Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1204–1216, 2017.
- [50] C. Beal, “Applications of model predictive control to vehicle dynamics for active safety and stability,” *Stanford University PhD dissertation*, 2011.
- [51] M. Jalalmaab, B. Fidan, S. Jeon, and P. Falcone, “Model predictive path planning with time-varying safety constraints for highway autonomous driving,” *International Conference on Advanced Robotics*, pp. 213–217, 2015.
- [52] I. Bae, J. Moon, H. Park, J. H. Kim, and S. Kim, “Path generation and tracking based on a b  zier curve for a steering rate controller of autonomous vehicles,” *International Conference on Intelligent Transportation Systems*, pp. 436–441, 2013.

-
- [53] J. Backman, T. Oksanen, and A. Visala, “Collision avoidance method with nonlinear model predictive trajectory control,” *IFAC Proceedings Volumes*, vol. 46, no. 18, pp. 35–40, 2013.
 - [54] E. Jones and R. Childers, “Contemporary college physics,” *Addison-Wesley*, 1990.
 - [55] T. Xu and H. Yuan, “Autonomous vehicle active safety system based on path planning and predictive control,” *Chinese Control Conference*, pp. 8889–8895, 2016.
 - [56] A. Schmeitz, J. Zegers, J. Ploeg, and M. Alirezaei, “Towards a generic lateral control concept for cooperative automated driving theoretical and experimental evaluation,” *International Conference on Models and Technologies for Intelligent Transportation Systems*, pp. 134–139, 2017.
 - [57] Mathworks, “quadprog,” <https://nl.mathworks.com/help/optim/ug/quadprog.html>, 2017.
 - [58] Mathworks, “mpcqpsolver,” <https://nl.mathworks.com/help/mpc/ref/mpcqpsolver.html>, 2017.
 - [59] Mathworks, “chol,” <https://nl.mathworks.com/help/matlab/ref/chol.html>, 2017.
 - [60] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, “Numerical recipes in c: The art of scientific computing second,” 1992.
 - [61] T. Toledo and D. Zohar, “Modeling duration of lane changes,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1999, pp. 71–78, 2007.
 - [62] S. Othman, R. Thomson, and G. Lannér, “Are driving and overtaking on right curves more dangerous than on left curves?,” *Annals of Advances in Automotive Medicine*, vol. 54, p. 253, 2010.
 - [63] N. Noto, H. Okuda, Y. Tazaki, and T. Suzuki, “Steering assisting system for obstacle avoidance based on personalized potential field,” *International Conference on Intelligent Transportation Systems*, pp. 1702–1707, 2012.

Glossary

List of Acronyms

3mE	Mechanical, Maritime and Materials Engineering
ABS	Anti-Lock Braking Systems
ACC	Adaptive Cruise Control
ADAS	Advanced Driver-Assistance Systems
AEB	Autonomous Emergency Braking
APF	Artificial Potential Field
CACC	Cooperative Adaptive Cruise Control
CoG	Centre of Gravity
CCS	Cooperative Control Systems
DCSC	Delft Center for Systems and Control
DoF	Degrees of Freedom
ESC	Electronic Stability Control
FOH	First-Order Hold
IVD	Inter-Vehicle Distance
IVS	Integrated Vehicle Safety
MPC	Model Predictive Control
N/A	Not Applicable
PD	Proportional and Derivative
PF	Potential Field

PIEV	Perception, Intellection, Emotion, and Volition
QA	Quadratic Approximation
QP	Quadratic Programming
RHC	Receding Horizon Control
SAE	Society of Automotive Engineers
TNO	Nederlandse organisatie voor Toegepast Natuurwetenschappelijk Onderzoek
TTC	Time To Collision
TUD	Delft University of Technology
ZOH	Zero-Order Hold

List of Symbols

α_f	Front tire slip angle in [rad]
α_i	Coefficient of polynomial function
α_r	Rear tire slip angle in [rad]
β	Vehicle body slip angle in local host-vehicle coordinate frame in [rad]
β_f	Front tire slip angle in local host-vehicle coordinate frame in [rad]
β_r	Rear tire slip angle in local host-vehicle coordinate frame in [rad]
δ	Steering angle in local host-vehicle coordinate frame in [rad]
Δu	Input increment in [rad]
$\Delta V_{o,h}$	Velocity difference between host and object vehicle in [kmh^{-1}]
δX_h	Infinitesimal small longitudinal offset in [m]
η	Understeer coefficient
λ_i	Eigenvalue
λ_y	Output weight
λ_Δ	Input increment weight in $[\text{rad}^{-1}]$
λ_ψ	Yaw rate weight in $[(\text{rads}^{-1})^{-1}]$
λ_ψ	Heading angle weight in $[\text{rad}^{-1}]$
λ_o	Obstacle potential field weight
λ_r	Road potential field weight
λ_u	Input weight in $[\text{rad}^{-1}]$
λ_{v_y}	Lateral velocity weight in $[(\text{ms}^{-1})^{-1}]$
∇f	Gradient of f
ψ	Heading angle in local road coordinate frame in [rad]

ψ_e	Heading angle error of vehicle in [m]
ψ_{lc}	Lane change heading angle in local road frame in [rad]
ρ	Curvature in $[m^{-1}]$
σ_x	Obstacle potential field longitudinal standard deviation in [m]
σ_y	Obstacle potential field lateral standard deviation in [m]
τ	Time integration variable
ℓ_f	Distance from vehicle CoG to front axle in [m]
ℓ_r	Distance from vehicle CoG to rear axle in [m]
ℓ_w	Lane width of 3.5 m
ψ_p	Path heading in [rad]
A	State matrix
a	Acceleration of vehicle CoG in $[ms^{-2}]$
a_M	Parameter defining Morse potential field depth
A_o	Obstacle potential field peak
A_r	Road potential field depth
A_X	Boundary condition matrix
A_{eq}	Matrix with linear equality constraint coefficients in terms of \tilde{u}
A_{ineq}	Matrix with linear inequality constraint coefficients in terms of \tilde{u}
a_ψ	First obstacle potential field rotation coefficient
$A_{u,ineq}$	Matrix with linear inequality constraint coefficients of input u in terms of \tilde{u}
$A_{x,ineq}$	Matrix with linear inequality constraint coefficients of state x in terms of \tilde{u}
B	Input matrix
b_M	Parameter controlling Morse potential field width
b_r	Parameter controlling road potential field width
b_Y	Boundary value vector
b_y	y -intercept of line normal to lane center in [m]
b_{eq}	Vector with linear equality constraint coefficients in terms of \tilde{u}
b_{ineq}	Vector with linear inequality constraints in terms of \tilde{u}
b_ψ	Second obstacle potential field rotation coefficient
$b_{u,ineq}$	Vector with linear inequality constraints of input u in terms of \tilde{u}
$b_{x,ineq}$	Vector with linear inequality constraints of state x in terms of \tilde{u}
C	Output matrix
c	Gradient vector of cost function J
c_0	Mobileye lane offset in [m]
c_1	Mobileye lane heading in [-]
c_2	Mobileye lane curvature in $[m^{-1}]$
c_d	Nonlinear damping in [x]
c_e	Morse potential field parameter
C_f	Negative front tire cornering stiffness in $[Nrad^{-1}]$
c_M	Parameter defining Morse potential field zero

C_r	Negative rear tire cornering stiffness in $\left[\text{Nrad}^{-1}\right]$
C_{ineq}	Matrix to obtain left-hand side of inequality constraint equation from x
c_ψ	Third obstacle potential field rotation coefficient
C_{XY}	Matrix with position coordinate selection coefficients
d_o	Overshoot in [m]
d_x	Longitudinal inter-vehicle distance in [m]
d_y	Lateral inter-vehicle distance in [m]
d_{i+1}	Communicated distance to leading vehicle of vehicle $i + 1$ in [m]
d_{LA}	Distance to look-ahead point measured from vehicle rear axle in [m]
F	Artificial force in [N]
f	Function
F_x	Sum of longitudinal tire forces in [N]
f_x	Partial derivative of f with respect to x
F_y	Sum of lateral tire forces in [N]
f_y	Partial derivative of f with respect to y
f_{ineq}	Vector with linear inequality constraint values of state x
F_{x_f}	Longitudinal force at front tire in [N]
F_{x_r}	Longitudinal force at rear tire in [N]
F_{y_f}	Lateral force at front tire in [N]
F_{y_r}	Lateral force at rear tire in [N]
g	Standard gravity of 9.81 ms^{-2}
H	Hessian matrix
H_f	Hessian matrix of f
i	Index
I_z	Vehicle mass moment of inertia around vehicle CoG in $\left[\text{kgm}^2\right]$
J	Cost function
j	Jerk in $\left[\text{ms}^{-3}\right]$
k	Discrete time instance
k_a	Positive gain of attractive potential field
k_i	Morse potential field tunable parameters
k_r	Positive gain of repulsive potential field
L	Lower triangular matrix from Cholesky decomposition
L_{wb}	Wheel base in [m]
m	Vehicle mass in [kg]
m_y	Slope of line normal to lane center
M_z	Sum of moments around vertical z -axis in [Nm]
N_p	Prediction horizon
O^g	Global road coordinate frame attached to road
O^h	Local host-vehicle coordinate frame attached to vehicle CoG
O^l	Local road coordinate frame attached to right lane center of road

Q_f	Quadratic Taylor approximation of f
q_i	Input of platoon vehicle i in [ms^{-1}]
R	Curve radius in [m]
r	Vehicle yaw rate in [rads^{-1}]
R_ψ	Rotation matrix around angle ψ
t	Time in [s]
t_d	Duration in [s]
t_h	Sampling time in [s]
t_r	Rise time in [s]
t_{LA}	Look-ahead time in [s]
t_{se}	Settling time in [s]
U	Potential field value
u	Steering wheel angle input in [rad]
U_0	Constant term of potential field quadratic approximation
U_1	Linear coefficient vector of potential field quadratic approximation
U_2	Quadratic coefficient matrix of potential field quadratic approximation
U_m	Morse potential field
U_o	Obstacle obstacle potential field value
u_{i-1}	Communicated acceleration of preceding vehicle in [ms^{-1}]
U_{tot}	Total artificial potential field
V	Velocity of vehicle CoG in [ms^{-1}]
V_x	Lateral velocity of vehicle CoG in global road frame in [ms^{-1}]
v_x	Longitudinal velocity of vehicle CoG in local host-vehicle coordinate frame in [ms^{-1}]
v_y	Lateral velocity of vehicle CoG in local host-vehicle coordinate frame in [ms^{-1}]
v_{i+1}	Communicated velocity of vehicle $i + 1$ in [ms^{-1}]
X	Longitudinal position in local road coordinate frame in [m]
x	State vector
x_i	Morse potential field variable
X_p	Predicted longitudinal position of vehicle CoG in local road frame in [m]
x_α	Polynomial coefficient vector
x_σ	Parameter controlling obstacle potential field σ_x in [m]
x_{LA}	Distance to look-ahead point measured from vehicle CoG in [m]
Y	Lateral position in local road coordinate frame in [m]
y	Output vector
y_e	Lateral position error of vehicle CoG in [m]
y_{ineq}	Vector with left-hand side of inequality constraint equation
y_σ	Parameter controlling obstacle potential field σ_y in [m]
y_{XY}	Vector with longitudinal and lateral host position coordinates in [m]
0	Initial

<i>c</i>	Continuous time
<i>d</i>	Discrete time
<i>e</i>	Error
<i>g</i>	Indicating variable expressed in global road coordinate frame
<i>h</i>	Indicating host vehicle parameter or variable expressed in local host-vehicle coordinate frame
<i>o</i>	Indicating obstacle parameter or variable
<i>p</i>	Predicted
<i>x</i>	Longitudinal
<i>y</i>	Lateral
<i>llc</i>	Left lane center
<i>lc</i>	Lane center
<i>lrb</i>	Left road boundary
<i>ℓ</i>	Indicating variable expressed in local road coordinate frame
<i>eq</i>	Equality
<i>ineq</i>	Inequality
<i>max</i>	Maximum
<i>min</i>	Minimum
<i>ref</i>	Reference
<i>at</i>	Attractive
<i>cr</i>	Curved road
<i>fb</i>	Feedback
<i>ff</i>	Feedforward
<i>rlc</i>	Right lane center
<i>rb</i>	Road boundary
<i>re</i>	Repulsive
<i>rrb</i>	Right road boundary
<i>sr</i>	Straight road
<i>ss</i>	Steady-state
X_h	Partial derivative with respect to X_h
Y_h	Partial derivative with respect to Y_h
0	Initial
Δ	Input increment
<i>u</i>	Input
<i>y</i>	Output
<i>x</i>	Vector input to f
x_0	Point of quadratic Taylor approximation
$\ddot{\psi}$	Vehicle yaw acceleration in $\text{[rads}^{-2}\text{]}$
$\dot{\psi}$	Vehicle yaw rate in $\text{[rads}^{-1}\text{]}$

\dot{X}_h	Longitudinal velocity of host vehicle CoG in global road coordinate frame in [ms ⁻¹]
\dot{Y}_h	Lateral velocity of host vehicle CoG in global road coordinate frame in [ms ⁻¹]
\cdot	Time derivative of variable
\sim	Indicating extended model predictive control notation
\vec{e}^g	Unit vector defining directions of global road coordinate frame
\vec{e}^h	Unit vector defining directions of local host-vehicle coordinate frame
\vec{F}_{O^h}	Force vector with components in directions of local host-vehicle coordinate frame
\vec{r}_{O^h/O^g}	Position vector of local host-vehicle coordinate frame expressed in global road coordinate frame
\vec{v}_f	Velocity vector of front wheel in [ms ⁻¹]
\vec{V}_h	Velocity vector of host vehicle CoG
\vec{v}_r	Velocity vector of rear wheel in [ms ⁻¹]

Index

A		P	
ADAS	1, 8	Path planning	5
APF MPC controller	39	Problem statement	3
Artificial potential field (APF)	6, 37		
Autonomous vehicles	7		
B		Q	
Bicycle model.....	19, 69	quadprog.....	52
		Quadratic approximation.....	73
		Quadratic programming.....	52
C		R	
Collision avoidance	15	Recommendations	66
Conclusions	65	Reference values	45
Constraint equations	46	Road potential field	27
Coordinate frames	19		
Cost function.....	42		
E		S	
Experiments.....	63	Sensors	11
		Simulations	49
		Simulink	85
H		T	
Highway environment	10	TNO	2, 49
I		Topic background.....	1
Initial states	48	Trajectory tracking	6
L		Tuning	54
Lane changing	14, 57		
Lane keeping	14, 58		
M			
Model predictive control (MPC)	17		
Morse potential field.....	25		
mpcqpsolver	53		
O			
Obstacle potential field	33		

