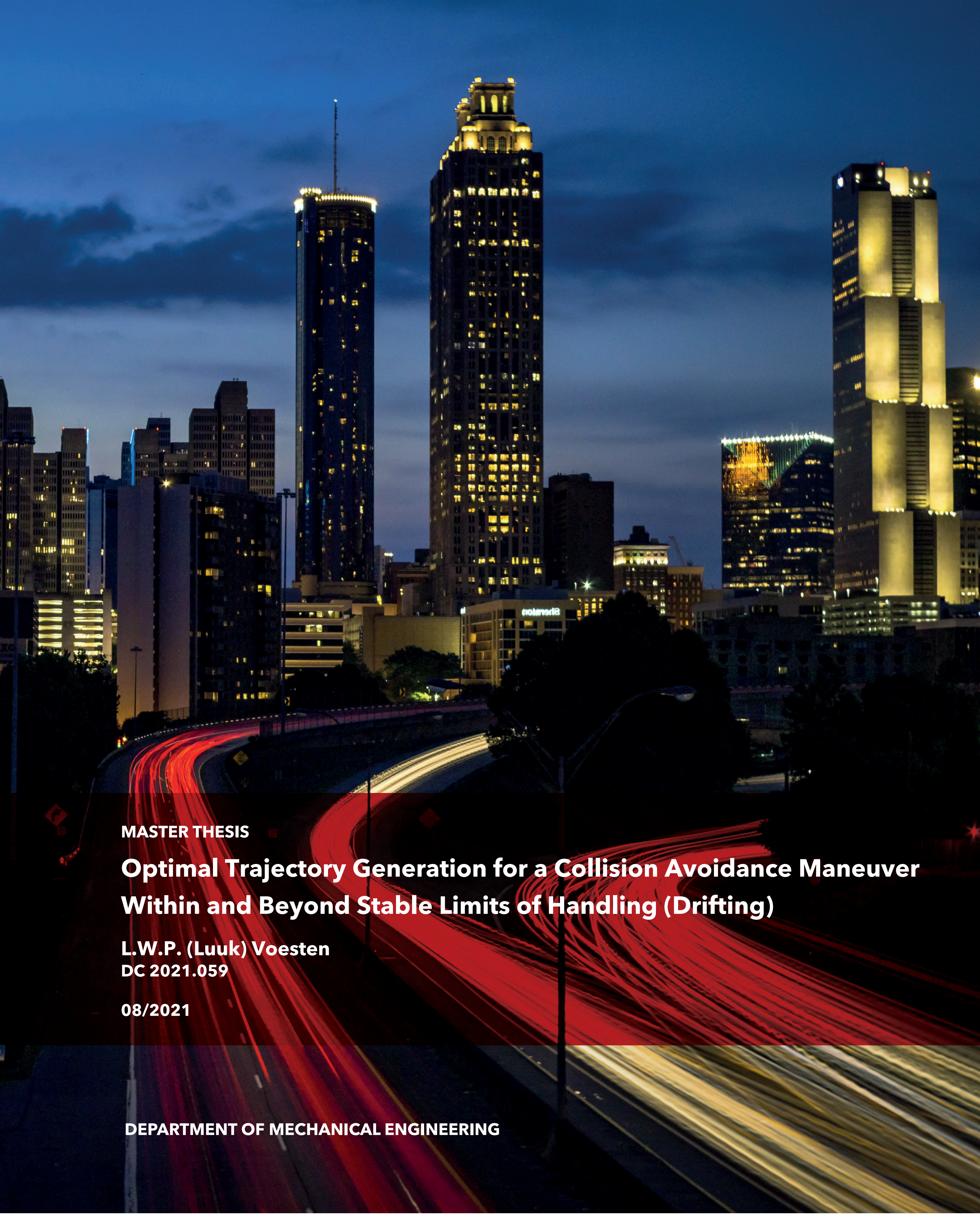
MASTER

Optimal Trajectory Generation for a Collision Avoidance Maneuver Within and Beyond Stable Limits of Handling (Drifting)

Voesten, L.W.P.

Award date:
2021



MASTER THESIS

Optimal Trajectory Generation for a Collision Avoidance Maneuver Within and Beyond Stable Limits of Handling (Drifting)

L.W.P. (Luuk) Voesten
DC 2021.059

08/2021

DEPARTMENT OF MECHANICAL ENGINEERING

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Master Automotive Technology

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Dynamics and Control Research Group
Department of Mechanical Engineering

Optimal Trajectory Generation for a Collision Avoidance Maneuver Within and Beyond Stable Limits of Handling (Drifting)

Master of Science Thesis

L.W.P. (Luuk) Voesten
1270893

Supervisors:
Dr. M. Alirezaei
Prof. dr. H. Nijmeijer

Report ID: DC 2021.059

Eindhoven, August 2021

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

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

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

Date 8/20/21

Name L.W.P. (Luuk) Voesten

ID-number 1270893

Signature

Insert this document in your Master Thesis report (2nd page) and submit it on Sharepoint

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

Abstract

Professional drivers use advanced driving techniques in different driving conditions to utilize a vehicle's capability within and beyond the stable limits of handling. Fully autonomous systems should also be able to control a vehicle in all driving conditions, even in extreme situations where maneuvering beyond the stable limits is required. A collision avoidance maneuver is one of these extreme situations in which it is essential to use the maximum capability of the vehicle in the longitudinal and lateral direction. For these evasive maneuvers, planning optimal trajectories is deemed to play an important role since it could ensure safe operation of an autonomous vehicle in the whole driving envelope.

The main objective of this research is to generate optimal trajectories for evasive maneuvers based on the vehicle's capabilities in the lateral and longitudinal direction. Racing-inspired objective functions, minimum time or maximum exit velocity, are employed in the optimization problem to utilize the vehicle's maximum capabilities. Evasive maneuvers beyond the stable limits of handling are often characterized by counter-steering front wheels, large vehicle sideslip angels, and saturated rear tires. Therefore, setpoint generation for the steering angle δ, the vehicle sideslip angle β, and the longitudinal tire slips λ_f and λ_r of the optimal solution are of importance for the future development of a tracking controller. Vehicle dynamics and road geometry constraints must be enforced on the optimal evasive trajectory to ensure realistic results. Additionally, different road surfaces that influence vehicle behavior have to be taken into account.

The system dynamics in the optimization problem are composed of a combined vehicle and road model. For this research, a nonlinear single-track vehicle model combined with a nonlinear combined slip tire model is used. The implemented tire model provides possibilities to describe the differences in tire behavior for driving on different road surfaces, e.g., asphalt or gravel. The road geometry is modeled using a curvilinear coordinate system. Since the optimization problem is mainly parameterized in terms of position, the decision is made to use the vehicle position as the independent variable instead of time. As a result, the problem changes from a free-horizon problem to a fixed-horizon problem, and the state dimension is reduced by one. The trapezoidal collocation direct method and the IPOPT NLP solver are combined with the ICLOCS2 software to solve the optimization problem. Different optimization models are created to generate evasive emergency maneuvers and determine the critical braking and steering distance.

Utilizing the proposed trajectory generator makes it possible to find optimal trajectories based on vehicle, tire, and road geometry data (with and without obstacles). The trajectory generator is benchmarked with the kinematic last-point-to-brake and last-point-to-steer models. Simulation analysis of both the optimal trajectory generator and the kinematic models show significant similarities between the found critical braking and steering distances. It is shown that using an optimization-based trajectory generator, however, has two main advantages. Firstly, it provides a mathematical approach based on the vehicle dynamics characteristics to combine steering and braking inputs and potentially results in shorter critical distances. Secondly, the vehicle states can be generated along with the generated evasive trajectory, which would be required by an autonomous vehicle controller to execute the maneuver.

Acknowledgments

First and foremost, I would like to express my special thanks of gratitude to my supervisor Dr. Mohsen Alirezaei for his continuous support during my graduation project. He provided me with this awesome project to apply optimization in order to simulate drifting vehicles. Furthermore, I would like to thank him for his guidance, support, and patience during the project. Meetings have always been very positive, and he has always been able to provide me with that extra bit of motivation to get to the results presented in this thesis.

I would also like to thank Prof. dr. Henk Nijmeijer, for the support and feedback provided during the monthly meetings. His expertise has been invaluable and pushed me to critically review my work to bring it to a higher level.

In addition, I would like to thank my fellow students and friends, who have always been willing to listen to the problems or difficulties that I encountered during the execution of the project. Finally, I could not have completed this thesis without the tremendous support of my parents Erik and Petra, sister Aniek, and girlfriend Jessey. Therefore, I would like to thank them all for their time, support, and love over the past ten months.

I hope you will enjoy reading.

Luuk Voesten

Eindhoven, August 20, 2021

Table of Contents

List of Figures

List of Tables

List of symbols

Symbol	Description	Unit
e	Absolute local discretization error	[-]
V	Absolute velocity	[m/s]
u_i	Control input i in $\mathbb{R}^m$	[-]
$C_{F\alpha}$	Cornering stiffness	[N/°]
L_{brake}	Critical braking distance	[m]
L_{steer}	Critical steering distance	[m]
h_k	Discretization step	[m]
a	Distance CG to the front axle	[m]
b	Distance CG to the rear axle	[m]
Δy	Distance perpendicular to centerline	[m]
s	Distance traveled along the road centerline	[m]
μ	Friction coefficient	[-]
g	Gravity	[m/s^2]
h_{cg}	Height CG	[m]
v_{x0}	Initial velocity	[m/s]
S_f	Inverse velocity along centerline	[s/m]
LW	Lane width	[m]
a_y	Lateral acceleration	[m/s^2]
d	Lateral offset required to avoid the obstacle	[m]
F_y	Lateral tire force	[N]
v	Lateral velocity	[m/s]
a_x	Longitudinal acceleration	[m/s^2]
ΔF_z	Longitudinal load transfer	[N]
λ	Longitudinal slip	[-]
λ_f	Longitudinal slip front wheel	[-]
λ_r	Longitudinal slip rear wheel	[-]
F_x	Longitudinal tire force	[N]
u	Longitudinal velocity	[m/s]
m	Mass	[kg]
P_{max}	Maximum available engine power	[Nm]
x_i	Model state i in $\mathbb{R}^n$	[-]
F_z	Normal load on the tire	[N]
N	Number of segments after discretization	[-]
OBS_{side}	Obstacle side	[-]
OBS_{width}	Obstacle width	[m]
F_{res}	Resulting tire force	[N]
κ	Road curvature	[1/m]
ψ_s	Road heading angle	[rad]
R	Road radius of curvature	[m]
α	Slip angle	[°]
δ	Steering angle	[rad]
t	Time	[s]
T	Torque applied to wheel	[Nm]
η	Understeer gradiant	[-]
β	Vehicle sideslip angle	[°]
ψ	Vehicle yaw angle	[rad]
$\Delta\psi$	Vehicle's relative heading angle	[rad]
WD	Weight distribution	[%]
R_w	Wheel radius	[m]
ω	Wheel rotational velocity	[m]
I_z	Yaw moment of inertia	[kgm^2]
r	Yaw rate	[rad/s]

Abbreviations

Abbreviation	Definition
ABS	Anti-lock Brake System
ADAS	Advanced Driver Assistance Systems
CAS	Collision Avoidance Systems
CG	Center of Gravity
CRG	Curved Regular Grid
ESC	Electronic Stability Control
EU	European Union
FWD	Front-wheel drive
ICLOCS	Imperial College London Optimal Control Software
IP	Interior-Point
LPTB	Last-point-to-brake
LPTS	Last-point-to-steer
MEVM	Maximum Exit Velocity Maneuver
MPC	Model Predictive Control
MTM	Minimum Time of travel Maneuver
NLP	Nonlinear Programming
NMPC	Nonlinear Model Predictive Control
OP	Optimization Problem
RC	Radio Controlled
RRT	Rapidly-exploring Random Trees
RWD	Rear-wheel drive
SQP	Sequential Quadratic Programming

Chapter 1

Introduction

The vast majority of road traffic injuries can be assigned to human error, and globally it is the leading cause of death for children and young adults in the age of 5-29 years old [1]. Singh [2] concluded that human error contributes to over 90% of all vehicle accidents. For this reason, manufacturers are implementing safety systems or driver assistance systems in their vehicles. Some good examples are the Anti-lock Brake System (ABS) and Electronic Stability Control (ESC) system to reduce the number of traffic injuries. Driver assistance systems like ABS and ESC are designed to restrict the vehicle to operate below the stability limits and to ensure average drivers are able to operate the vehicle safely [3]. According to the National Highway Traffic Safety Administration [4], the ESC safety system is estimated to reduce the fatal run-off-road crashes by 36% and all fatal crashes by 14%. The introduction of these driver assistance systems has proved to effectively reduce the number of road traffic injuries since these systems are nowadays mandatory in all new passenger cars sold in the European Union (EU). A possible next step in reducing the number of road traffic injuries even further is introducing level five full autonomous vehicles. According to the five levels of autonomous vehicles, developed by SAE International [5], a level five autonomous vehicle should be able to take over all driving tasks from the driver and is able to monitor and maneuver in all driving conditions without human interventions.

As mentioned before, the current safety systems, e.g., ABS and ESC, are designed to restrict the vehicle to operate within the stable region in which the average driver can still safely operate the vehicle. Professional drivers engaged in motorsports regularly use agile drifting maneuvers to utilize the maximum capabilities of a vehicle in the longitudinal and lateral direction. Frequently, these maneuvers are used to negotiate sharp turns, hairpins, and to avoid critical driving scenarios in which a vehicle has to change direction quickly without losing control [6]. Fully autonomous vehicles should also be able to safely control the vehicle beyond stable limit handling conditions, mimicking professional drivers. A vehicle's lateral motion is only affected by the steering angle in typical cornering conditions, and the longitudinal motion is controlled by a throttle or braking input. During maneuvers close to or beyond the stable limits of handling (drifting), the lateral motion is affected by both coupled inputs. When the rear tires start to saturate, operating at their friction limits, the two control inputs become mutually dependent and thus affect both the lateral and longitudinal motion of the vehicle [3]. Mimicking the maneuvers of professional drivers in an autonomous vehicle is thus challenging because drifting is an unstable cornering condition; often characterized by counter-steering front wheels, large vehicle sideslip angels, and saturated rear tires. However, in motorsport competitions like the World Rally Championship, drivers regularly apply this drifting technique for precise control of both the vehicle's position and its sideslip angle to overcome the most challenging driving scenarios and road conditions.

In literature, it is commonly assumed that racing is equivalent to driving a path as fast as possible, i.e., drive the track in minimum time by using the maximum capabilities of the vehicle. This assumption is especially true for circuit racing, where drivers drive the same path repeatedly, trying to optimize their racing line to minimize lap time. The same applies to off-road rally races. The only difference is that a driver is trying to minimize the stage time without extensive practice on that particular stage. The driver has to react to an unpredictable and varying environment while driving on the limits. In rally racing, it is prevalent to negotiate a corner with a large vehicle sideslip angle to align the vehicle with the next road segment. Experimental evidence [3, 7] shows that rally drivers regularly use this aggressive maneuver. It suggests that it might be more efficient under certain road conditions and vehicle layouts than a low vehicle sideslip maneuver. Furthermore, it is important for fully autonomous systems to control the vehicle in the whole vehicle dynamics envelope by utilizing its maximum capacity for safe operation even beyond the stable limits of handling.

1.1 Background and motivation

According to Goh [8], tracking a reference trajectory while driving in the unstable region of the state-space results in two challenges. First, the problem becomes underactuated due to the additional reference state. Second, a strong coupling between longitudinal and lateral dynamics exists because the rear tires are saturated. This coupling is not present in typical driving conditions. Several research projects have devised control methods for automated drifting [9, 10, 11, 12, 13], able to track steady-state drift equilibria and solve the two challenges presented by Goh [8]. Tracking steady-state drift equilibria is one aspect of the transition to a level five autonomous vehicle. The autonomous vehicle should also be able to plan feasible maneuvers beyond the stable limits of handling to guarantee reliability and safety, even for the most challenging road conditions. Planning maneuvers that utilize the vehicle's maximum capabilities in the longitudinal and lateral direction could be beneficial, especially for planning evasive maneuvers. For tracking these maneuvers, setpoints must describe the system states along with the generated maneuver.

A goal for Collision Avoidance Systems (CAS) would be to find the optimal combination of control inputs, steering and braking, to overcome critical driving scenarios, e.g., evasive maneuvers. Generally, it is assumed that the preferred control input is either braking for low velocities or steering for higher velocities. These assumptions are based on constant accelerations in the kinematic equations of the last-point-to-brake and last-point-to-steer models [14]. The equations for the last-point-to-brake L_{brake} and last-point-to-steer L_{steer} are derived in Appendix A and defined as follows:

$$L_{brake} = \frac{v_{x0}^2}{2a_x}, \tag{1.1}$$

$$L_{steer} = v_{x0}\sqrt{\frac{2d}{a_y}}, \tag{1.2}$$

where v_{x0} is the initial velocity of the vehicle, d is the lateral offset required to avoid the obstacle, a_x and a_y the maximum longitudinal and lateral acceleration determined by the friction coefficients in these directions, respectively. In Figure 1.1, the critical distance is shown for varying initial velocities and a required lateral offset of 2.5 $[m]$. The maximum accelerations in the longitudinal and lateral direction are determined using the vehicle parameters and the dry asphalt tire coefficients in Appendix B. From this figure, it can be concluded that a cross-over velocity is present from which it is better to steer instead of brake to avoid the obstacle. These basic kinematic models can be used to get a general idea of what control inputs are required for collision avoidance. One drawback of these models is that it is assumed that using only one of two control inputs is best to avoid a collision. Additionally, the kinematic models can only predict the most suitable control input but cannot determine setpoints for the model states and control inputs. This information will be required by an autonomous vehicle to execute the maneuver using a trajectory tracking controller.

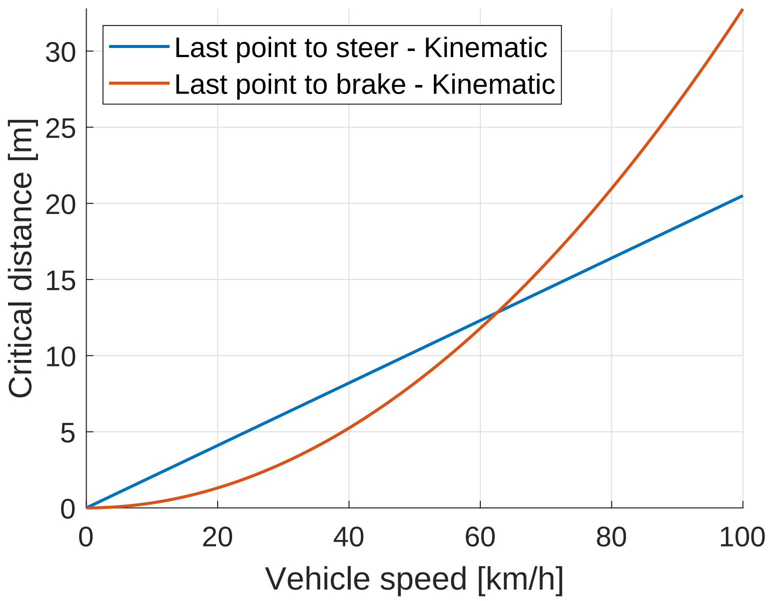


Figure 1.1: Last-point-to-brake L_{brake} and last-point-to-steer L_{steer} for varying initial velocities and a required lateral offset d of 2.5 [m]. Figure adapted from [14].

There is only limited information available about the usage of optimization to generate evasive maneuvers based on a vehicle's capabilities in the lateral and longitudinal direction. However, most studies used discrete optimization with a finite set of predetermined feasible trajectories [15, 16]. Each trajectory from this set is evaluated using a cost function, and from this evaluation, the optimal solution emerges. To the authors' best knowledge, utilizing an optimization algorithm with racing-inspired objective functions to generate optimal evasive maneuvers within and beyond the stable limits of handling is not yet covered in existing research. The evasive maneuvers resulting from the trajectory generator can be benchmarked with the kinematic collision avoidance models. It should be possible to reproduce the critical distance results of the kinematic models using an optimization algorithm. Additionally, such an algorithm can be used to overcome the drawbacks of the kinematic models. The optimization algorithm is not limited to the usage of only one control input. Thus the assumption that using either braking or steering is best for avoiding a collision can be validated. In addition, this method is able to output the model state and control input setpoints required for tracking a generated maneuver.

1.2 Research objectives and contributions

Understanding vehicle behavior up to and beyond the stable limits of handling is of great importance for creating fully autonomous vehicles. These vehicles should control a vehicle in all driving conditions, even in extreme situations where maneuvering beyond the stable limits is required. A collision avoidance maneuver is one of these extreme situations in which it is essential to use the maximum capabilities of the vehicle in the longitudinal and lateral direction. For these evasive maneuvers, time-dependent trajectory planning is deemed to play an important role. The problem addressed in the graduation project is stated as follows.

This thesis seeks to generate optimal trajectories for evasive maneuvers based on a vehicle's capabilities in the lateral and longitudinal direction. The goal is to find an optimal trajectory for an evasive maneuver to avoid a potential collision within and beyond the stable limits of handling.

The optimization algorithm will generate an optimal reference trajectory based on the vehicle topology, tire data, and road geometry data (with and without obstacles). The optimal solution will be determined by minimizing a racing-inspired cost function, e.g., minimum time or maximum exit velocity. Especially, benchmarking the results found by an optimization-based trajectory generator with the results of the kinematic models is a topic that is rarely discussed in the literature. All of this translates to the following research objectives:

- Find the optimal trajectory to negotiate a corner, including the model states and control inputs corresponding to this trajectory, e.g., the steering angle δ, the vehicle sideslip angle β, and the longitudinal tire slips λ_f and λ_r.
- Conduct a sensitivity analysis to exploit the influence of changing the vehicle topology or varying vehicle and road parameters.
- Develop a trajectory generator that can include obstacles to optimize evasive maneuvers within and beyond the stable limits of handling.
- Evaluate the optimized trajectory for collision avoidance and compare the results with the kinematic last-point-brake and last-point-to-steer models.

1.3 Delimitations

This research will focus on finding an optimal trajectory for an evasive maneuver within and beyond the stable limits of handling. To summarize, the delimitations of the research are:

- Vehicle topologies to be tested are FWD or RWD.
- The longitudinal tire slips are used as a control input, and it is assumed that the resulting tire forces are generated instantly. For clarification, motor and tire dynamics are neglected in this research.
- The wheels of the vehicle are always in contact with the road surface.
- Only static obstacles will be considered.

- Road surfaces to be tested are (dry) asphalt or gravel. Thus, the tire-road interaction curves remain constant during the entire maneuver.

These road conditions will also be encountered by autonomous vehicles, and, as mentioned before, autonomous level five vehicles should be able to handle all driving scenarios and conditions without losing control. In the end, the goal is to generate an optimal reference trajectory for an evasive maneuver, including the relevant model state and control input profiles. This information can then be used in a trajectory tracking controller to track an evasive trajectory to prevent a collision and make future vehicles safer.

1.4 Thesis outline

The content of this report is split into seven chapters, including the introduction. Chapter 2 provides an extensive overview of the conducted literature review. The topics of drifting, trajectory generation, collision avoidance, and optimization are covered. Chapter 3 describes the coordinate system used for this research and the vehicle and tire models that are later implemented in the developed optimization algorithm. In Chapter 4, an overview of the Optimization Problem is provided in combination with the transcription method used to convert the continuous-time problem to a Nonlinear Programming problem. Next, Chapter 5 presents the results of the simulations conducted for this research. In this chapter, first, the created optimization algorithm will be validated. Second, the influence of the different objective functions is tested. The last part of the chapter provides the results of the sensitivity analysis. Here, the vehicle topology is varied, combined with varying vehicle parameters or tire curves to assess how it influences the optimal trajectory and the optimal vehicle sideslip angle β corresponding to this trajectory. The results of utilizing the algorithm for the generation of evasive maneuvers are documented in Chapter 6. Consecutively, the results are compared to the last-point-to-brake and last-point-to-steer kinematic models. As the last step, simultaneous braking and steering is implemented to see how this affects the results. Finally, in Chapter 7, the conclusions are drawn about the optimal reference trajectory for evasive maneuvers, and recommendations for future research are formulated.

Chapter 2

Literature review

This chapter will provide a summary of all the relevant information that has been gathered during this research. The literature review is divided into five sections. In Section 2.1, the vehicle dynamics envelope will be explained. In addition, a general introduction of the drifting maneuver will be provided, including phase portraits that show why drifting is considered an unstable equilibrium. Section 2.2 reviews and explains the state-of-the-art results of various research institutes that showed capabilities of generating and tracking trajectories within and beyond the stable limits of handling. Section 2.3 addresses the existing literature on the topic of collision avoidance and collision mitigation. An overview of existing methods and solvers for solving an Optimization Problem is provided in Section 2.4. Finally, Section 2.5 discusses all gathered information and presents decisions for the following chapters.

2.1 Vehicle dynamics envelope

Professional drivers use advanced driving techniques in different driving conditions to utilize a vehicle's capability within and beyond the stable limits of handling. Average drivers are often unable to recover the vehicle when it enters the unstable region of the driving envelope. In these situations, the control inputs become mutually dependent and thus require advanced driving techniques for safe operation of the vehicle. A collision avoidance maneuver is one of these extreme situations in which it is essential to use the maximum capability of the vehicle in the lateral and longitudinal direction. Evasive maneuvers beyond the stable limits of handling are often characterized by counter-steering front wheels, large vehicle sideslip angles, and saturated rear tires. Similar characteristics are found for drifting maneuvers, hence why this topic is part of the literature review.

In normal driving conditions, the inputs of a vehicle are intuitive because steering affects the lateral motion of the vehicle and throttle or braking the longitudinal motion. As mentioned before, the two control inputs become mutually dependent when the rear tires start to saturate. Thus affect both the lateral and longitudinal motion of the vehicle [3]. This effect can be explained using the Friction Circle, where the boundary of this circle describes the maximum combined tire force in the lateral and longitudinal direction. In combined slip scenarios, the lateral capabilities of the tire are compromised by the longitudinal slip, and vice versa [9]. This relationship can also be seen in the commonly known 'friction circle' equation [17] defined by,

$$\mu F_z \geq \sqrt{F_x^2 + F_y^2}, \tag{2.1}$$

where μ is the friction coefficient, F_z is the normal load on the tire, F_x is the longitudinal tire force, and F_y is the lateral tire force. This nonlinear tire behavior is considerably different in comparison to the behavior in normal driving conditions.

Several authors, including Ono et al. [18], Hindiyeh [3], and Bobier et al. [19] have demonstrated that three equilibrium points exist for cornering using phase portraits. Figure 2.1 presents a schematic overview of the three equilibria, one stable (middle situation) and two unstable (left and right situation) which are called drifting equilibria. A phase portrait can be created using the following two-state vehicle model with nonlinear tire force characteristics to describe the lateral vehicle dynamics [18],

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \dfrac{F_{y,f} + F_{y,r}}{mu} - r \\ \dfrac{aF_{y,f} - bF_{y,r}}{I_z} \end{bmatrix}, \tag{2.2}$$

where $F_{y,f}$ and $F_{y,r}$ are the lateral forces of the front and rear tire, m is the vehicle mass, u is the longitudinal velocity, a and b are the distance from the Center of Gravity (CG) to the front and rear axle respectively, I_z is the yaw moment of inertia, β is the vehicle sideslip angle, and r is the yaw rate of the vehicle.

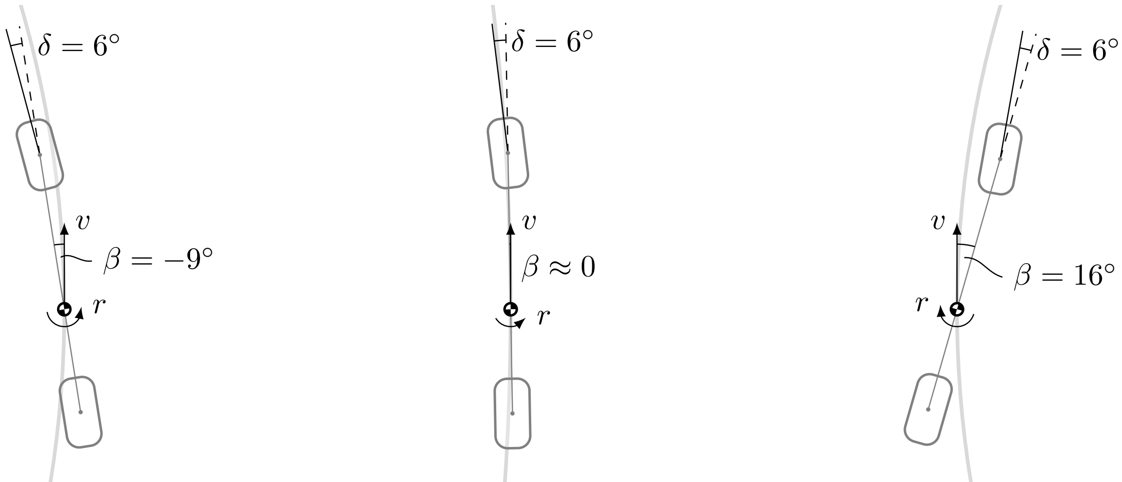


Figure 2.1: Representation of the three equilibria for a steering angle $\delta = 6°$ and a velocity $v = 1.5[m/s]$. Figure adopted from [10].

Phase portraits have been used to analyze the open-loop vehicle dynamics and study the vehicle's cornering equilibria. A phase portrait visualizes the vehicle's trajectories in a prespecified phase plane. In most cases, the vehicle sideslip angle-yaw rate plane (β-r). For example, Figure 2.2 shows a phase portrait in which red dots denote the three equilibria. Here, the middle dot represents the stable cornering equilibrium, and the two dots left, and right are the unstable drifting equilibria. This figure suggests that a stable handling envelope exists because the phase trajectories flow toward the stable equilibrium for a broad region of the phase portrait. However, the system will diverge from the unstable (drifting) equilibria for small perturbations because the phase trajectories nearby flow away.

The information in this section suggests that a nonlinear vehicle and tire model are required for an accurate representation of evasive maneuver characteristics.

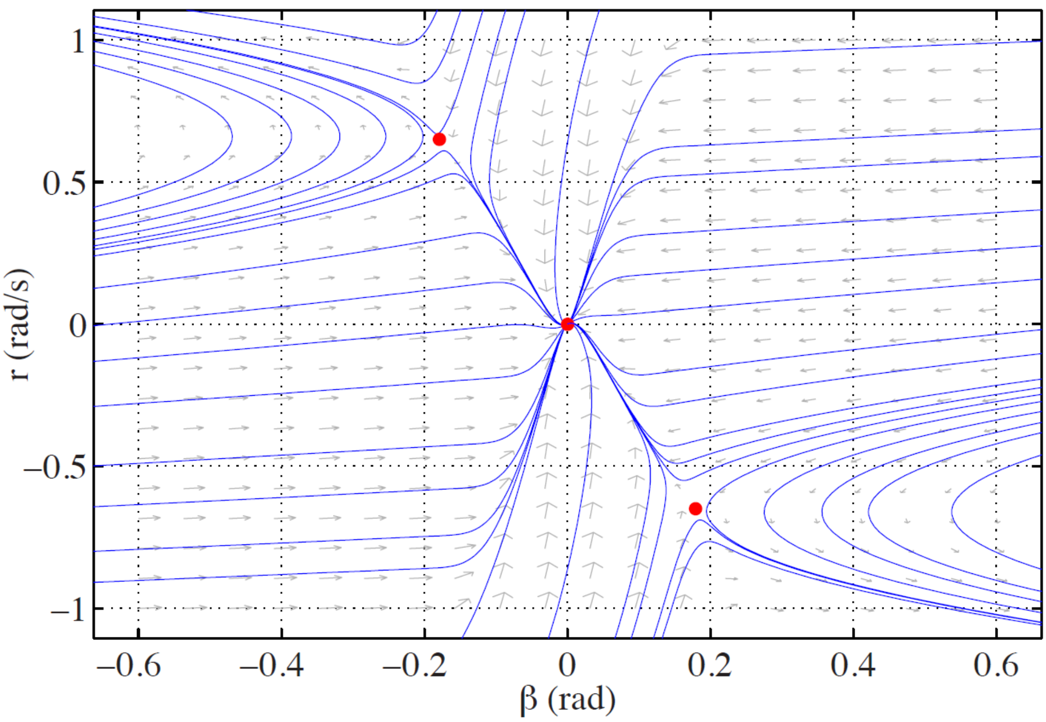


Figure 2.2: Phase portrait of state trajectories for the two-state bicycle model with an equilibrium longitudinal velocity $U_x^{eq} = 8$ [m/s] and an equilibrium steering angle $\delta^{eq} = 0°$. Figure adopted from [3]. Parameters: $m = 1724$ [kg], $I_z = 1300$ [kgm^2], $a = 1.35$ [m], and $b = 1.15$ [m].

2.2 Trajectory generation

Two essential functions of autonomous vehicles are trajectory planning and trajectory tracking. The first function, trajectory planning, attempts to generate a feasible path based on the equipped sensors, creating environmental perception. The latter function, trajectory tracking, aims to control an autonomous vehicle to follow this feasible trajectory. The function of trajectory tracking is not within the scope of this research. Therefore no further elaboration will be provided on this topic.

In the field of automation and robotics, a clear distinction is made between the two main parts of the general motion planning problem [20]. These two main parts are path planning and trajectory planning. The main difference between the two is that trajectory planning is time-dependent, while path planning describes a geometric path from an initial point to a final point. Usually, trajectory planning algorithms take the path planner's path as an input in combination with the constraints applicable to the vehicle. However, these two phases are not necessarily distinct. For example, if only the initial and final positions are known, the two problems can be solved simultaneously to generate a feasible trajectory [20].

This section will discuss two methods applied in previous research projects to find the optimal trajectory. First, a common path-finding algorithm called the Rapidly-exploring Random Trees is discussed. Second, the numerical optimization technique will be addressed.

2.2.1 Rapidly-exploring Random Trees

The Rapidly-exploring Random Trees (RRT) algorithm is an incremental sampling-based motion planning algorithm. One drawback of the RRT algorithm is that optimality cannot be guaranteed according to Jeon et al. [21]. An improvement of the RRT algorithm is the RRT*, introduced by Karaman and Frazzoli in [22], which provably returns a solution that converges to an optimum. In their research, Jeon et al. [21] used the RRT* algorithm to study time-optimal maneuvers of an off-road vehicle on a gravel surface. It has been shown that the aggressive trail-braking maneuver automatically emerges from the algorithm for cornering on surfaces with a low friction coefficient. The vehicle dynamics are included in the simulation using a single-track vehicle model, including longitudinal load transfer and Pacejka's Magic Formula tire model. Using an incremental sampling-based motion planning algorithm has two mentionable drawbacks. The first is that the solution of the RRT* algorithm only approaches the optimal solution with an infinite amount of nodes. Thus, with an increasing number of nodes, the RRT* algorithm will continue to find a different, more accurate minimum-time optimal solution. The second drawback is that both the RRT and RRT* approaches eventually cannot generate a smooth trajectory.

2.2.2 Numerical optimization

Another technique to find a feasible and optimal trajectory is by using nonlinear optimal control techniques. Researchers commonly apply this technique to find the minimum time-optimal solution for a predetermined road geometry regarding vehicle states, vehicle characteristics, and road conditions. In the past, various research projects have applied the optimal control techniques to find the optimal laptime for F1 circuits, like Casanova et al. in [23] for the Suzuka and Catalunya circuits. Heilmeier et al. [24] used an optimization algorithm to study the differences in a trajectory for shortest path optimization, minimum curvature optimization, and minimum time optimization for a section of the Berlin Formula E track. Their algorithm has been tested successfully in a real world environment using the Roborace DevBot during the Formula E event in 2018. To the author's best knowledge, one of the first to demonstrate that it is possible to reproduce race-inspired maneuvers, like the pendulum-turn, using a numerical optimization scheme are Velenis and Tsiotras [25]. There is only one drawback to the method used in their research: the user must provide an initial guess for the optimal control inputs. The convergence of the optimization algorithm heavily depends on the accuracy of this initial guess.

Where Velenis and Tsiotras implemented an all-wheel-drive vehicle in the previously mentioned project, Velenis et al. [26, 27, 28] used a front-wheel driven model to simulate a trail-braking and pendulum-turn maneuver. Both of these maneuvers are created using the same approach as in the research of Velenis and Tsiotras [25]. The only differences are the drivetrain layout and the initial conditions. Olofsson et al. [29], and Tavernini et al. [30] have solved a similar Optimization Problem (OP), and in addition, the road conditions and transmission layout are varied. Compared to the research of Velenis and Tsiotras [25], it suffices to provide the initial and final conditions instead of an accurate initial guess for the control inputs. Their goals are to find the minimum time cornering maneuver and the related input history of a vehicle running on a predefined road surface using a single-track vehicle model (without pitch and roll rotations) combined with the combined slip Magic Formula tire model. According to Tavernini et al.[30], the result of the optimal control problem fully describes both the dynamics of the vehicle during the maneuver and the driver inputs that produce it. One of the main advantages of this method is that no driving rules have to be predefined. The research of Tavernini et al. is later extended to find the optimal trajectory for a handbrake turn, with an FWD vehicle, in a tight hairpin corner [31]. The results are compared to measured data of an FWD rally vehicle performing a handbrake cornering maneuver and showed significant similarities.

Berntorp et al. [32] studied the effect of different vehicle and tire models on the outcome of the optimization algorithm. The trajectory outcome differences are minor over the complete maneuver when looking at simulations with the same tire model. However, a difference in trajectory is noticeable when comparing simulations with different tire models, which is possibly a result of the minor differences between the tire characteristics of the models. A different approach is applied by Verschueren et al. [33], where Nonlinear Model Predictive Control (NMPC) is used in order to control a setup of miniature cars traveling under a minimum time objective. The road centerline is used as a reference to compute the control inputs to satisfy the minimum time objective. Although this method can compute a minimum time maneuver, it tends to be more of a trajectory tracking method than a trajectory planning method.

2.3 Collision avoidance

The previous two sections provided a summary of the information in the literature regarding drifting and trajectory generation. In this section, the existing literature on combining the topics of drifting, trajectory generation, and collision mitigation is reviewed.

In Chu et al. [15] and Hu et al. [16], discrete optimization is used as a method to avoid obstacles. A finite set of feasible paths is used, combined with a cost function, that penalizes running over obstacles and distance from the current road center to determine the optimal path. However, it is hard to determine the safest path using an optimization algorithm if it only checks for collisions since this only provides binary information. Therefore, a risk assessment for a potential collision is used for every path candidate. In addition, both research projects added the path smoothness and a consistency part to the objective function to prevent excessive control effort to follow the path. According to Chu et al., the used method is adopted from a vehicle called “Stanley”, developed by the Stanford University.

Nonlinear Model Predictive Control (NMPC) has successfully been applied by Frash et al. [34] to control a vehicle on a low-friction road with obstacles. A dual-track vehicle model is used in their research, where longitudinal and lateral load transfer is included, and the tire forces are computed using Pacejka’s combined slip tire model. In addition, a reference velocity is included in the objective function to ensure that the initial velocity is tracked while avoiding the obstacles. Since a reference velocity is tracked, the vehicle does not utilize the vehicle’s maximum capabilities during the entire maneuver.

Choi and Kang [35] developed a front-end collision avoidance strategy using simultaneous steering and braking. A kinematic model is used in their research, and NMPC is employed to determine the control inputs necessary to perform a lane change to avoid obstacles. The results of NMPC are compared with the kinematic models for last-point-to-brake and last-point-to-steer. The conclusion is that shorter critical distances could be achieved using simultaneous braking and steering compared to using only one of the two inputs. Singh and Nishihara [36] did draw similar conclusions, using a point mass vehicle model and the friction circle tire model to study the influence of combined braking and steering on the critical distance.

2.4 Optimization

In Section 2.2.2, the literature using numerical optimization techniques for trajectory generation has been discussed. The goal of this research is to find an optimal solution for evasive maneuvering. Therefore, this section will review available techniques to generate these optimal solutions. Here, an overview of existing methods and solvers for the Optimization Problem (OP) will be presented. A significant amount of literature is available for solving OPs, two commonly referenced books are written by Kirk [37], and Betts [38]. For a detailed explanation of the process for solving OPs, the reader is referred to these two books. First, the similarities and differences between the direct and indirect methods are examined.

Next, the transcription process and the most common methods used for this process are covered. In the last section, some available Nonlinear Programming problem (NLP) solvers are discussed, combined with the results acquired using these solvers in previous research projects.

2.4.1 Direct and indirect methods

One drawback of 'real world' problems is that these are often too complex to be solved analytically. According to Perantoni [39], the solution for this drawback is using an alternative method to solve OPs through finite-difference approximations. The idea is to use a numerical approximation based on a quadrature rule to transcribe the infinite-dimensional problem into a finite-dimensional problem. The continuous-time system dynamics are discretized with respect to the independent variable of the OP. The order of the optimization and discretization phases determines the category in which the method can be placed. The problem is first discretized and then optimized for the direct method, while optimization is the first step for the indirect method. When transcription is 'directly' applied to the OP, such a method is a direct method. The alternative indirect method uses Pontryagin's minimum (or maximum) principle to derive necessary conditions for optimality. The resulting boundary value problem is then transcribed in order to solve it. Since the transcription method is not directly applied to the OP, such methods are labeled as indirect. In essence, the goal of an indirect method is to locate a root of the necessary conditions for optimality, while the goal of a direct method is minimizing the objective function of the OP [38].

Both methods have been used in literature to solve optimal trajectory generation problems, where the direct method is used in [39, 40], and the indirect method in [41]. For a detailed overview of the various numerical methods used for solving OPs, the reader may consult the book of Betts [38]. For this research, the decision is made to use the direct method, which has two advantages according to Perantoni [42]. First, for direct methods, it is not required to derive the necessary conditions for optimality. Second, multiple ready-to-use software packages are publicly available for direct methods or are open-source. Therefore, the next section will provide some more in-depth information regarding this method and the transcription process to convert the OP into an NLP.

2.4.2 Transcription

Direct methods convert an OP into an NLP using a process called transcription. There are numerous transcription methods, but all methods can be categorized into two classes: shooting and collocation methods. The difference between the two is mainly dictated by how the constraints on the system dynamics are satisfied. For example, a simulation is performed using specified initial conditions for shooting methods, and then the error is evaluated at the boundary conditions. On the other hand, collocation methods enforce the system dynamics at all points along the discretized trajectory. In the following two paragraphs, the shooting methods and collocation methods will be shortly explained.

Shooting methods
Shooting methods are relatively simple methods to transcribe an OP. The single-shooting method is similar to the way humans conduct experiments to accomplish a specific goal. For example, consider a human is trying to hit a target using a cannon. An untrained operator will most likely miss the target when using the equipment for the first time. However, when iteratively improving the shots, by adjusting the angle of attack of the cannon and the amount of black powder used to fire the cannon, the target will most probably be hit at some point. Single-shooting methods work similarly, but a simulation replaces the experiment to determine the error between the target and the actual outcome. Thus, it has some common ground with trial and error methods. The single-shooting method is visualized in Figure 2.3a.

Then there is the multiple-shooting method which breaks up the trajectory into several segments, and each segment is then solved using the single-shooting method. Although the single-shooting method has one defect constraint between the target and the simulation outcome, the multiple-shooting methods have as many defect constraints as segments. Minimizing the defects is a root-finding problem, and that is the whole idea behind shooting methods. According to Kelly [43], the multiple shooting method makes low-level OPs easier, even though the number of decision variables and constraints increases. The multiple-shooting method is visualized in Figure 2.3b.

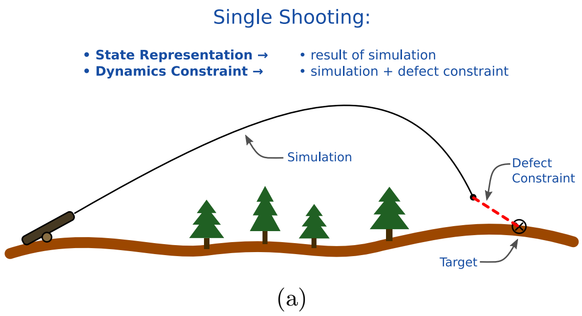


Figure 2.3: Shooting methods, single-shooting and multiple-shooting, for transcribing an optimal control problem. Figures adopted from [44].

Both shooting methods discussed in this subsection have one major drawback. The quality of the initial guess has a significant effect on the final result. A small change in this initial condition can result in a significant difference in the final result. According to Betts [38], this effect called the 'tail wagging the dog' can result in very nonlinear constraints that are hard to solve. The sensitivity for this effect is reduced by using the multiple-shooting method since the problem is divided into smaller steps. A deficiency of the multiple-shooting method is that when path constraints are implemented, the resulting NLP becomes relatively difficult to solve.

Direct collocation methods

Direct collocation methods work by discretizing an OP into N segments, where the continuous state and control trajectories are approximated using piecewise polynomials. The direct collocation method is visualized in Figure 2.4. In this figure, the model states are denoted by p and the system dynamics by $\dot{p} = f(p)$. However, the model states will be denoted by x in this research since this figure is adopted from [44]. First, the trajectory is discretized into a finite set of decision variables. The N segments result in $N + 1$ collocation points (or knot points):

$$\begin{aligned} t &\rightarrow t_0, ..., t_k, ..., t_N, \\ \boldsymbol{x} &\rightarrow \boldsymbol{x}_0, ..., \boldsymbol{x}_k, ..., \boldsymbol{x}_N, \\ \boldsymbol{u} &\rightarrow \boldsymbol{u}_0, ..., \boldsymbol{u}_k, ..., \boldsymbol{u}_N, \end{aligned} \tag{2.3}$$

where t represents the independent variable of the OP which is often time, $\boldsymbol{x}$ are the model states, and $\boldsymbol{u}$ the control inputs specified in the OP. See Chapter 4 for more details.

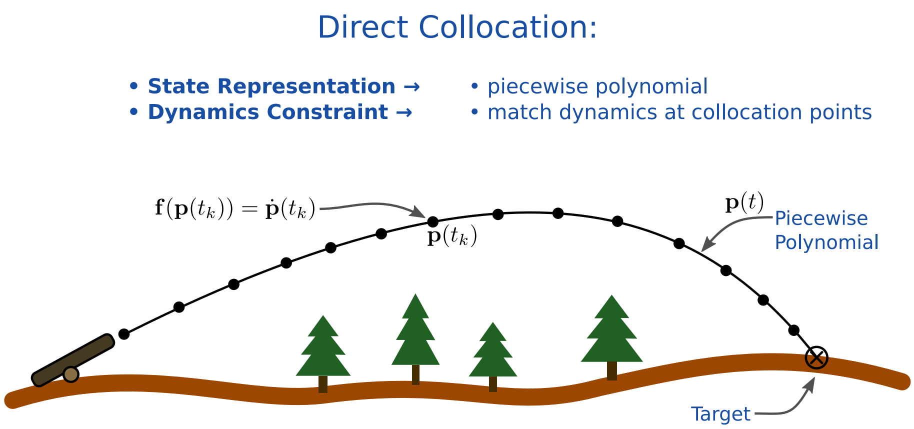


Figure 2.4: Direct collocation. Figure adopted from [44].

The order of the approximating piecewise polynomials is determined by the quadrature rule selected for the transcription process. This quadrature rule determines a couple of aspects of the transcription process [45]: the formulation of the dynamics constraints, weights applied on the objective function, and the interpolation functions. There are two commonly used quadrature schemes as reported by Betts

[38], the trapezoidal method and the Hermite-Simpson quadrature method. Both schemes are Lobatto methods, which means that the endpoints of the interval are collocation points as well. The trapezoidal method is based on a quadratic interpolation polynomial, while the Hermite–Simpson method is based on a cubic interpolation polynomial. Both methods are relatively similar, but the Hermite–Simpson method provides a higher-order accurate solution, but at the cost of an increased problem size due to the extra collocation points. Both methods are mentioned here for completion, but the transcription process in this research is performed using the trapezoidal method. Other research projects have proved to get sufficiently accurate results using this method [40, 42].

2.4.3 NLP solvers

According to Rodrigues et al. [46], optimization algorithms can be categorized into two types: specific optimal control algorithms and nonlinear optimization algorithms. The first category is capable of both the transcription process and solving the resulting NLP to solve the original OP. The second category contains dedicated solver algorithms for NLPs. Most applications for trajectory optimization in the first category incorporate an NLP solver from the second category to prevent reinventing the wheel. A variety of methods exists to solve NLPs of the general form. The general formulation of the NLP is as follows:

$$\begin{aligned} \underset{\boldsymbol{z}}{\text{minimize}} \quad & J(\boldsymbol{z}) \\ \text{subject to} \quad & \boldsymbol{f}(\boldsymbol{z}) \; = \; 0, \\ & \boldsymbol{g}(\boldsymbol{z}) \; \leq \; 0, \\ & \boldsymbol{z}_{low} \; \leq \boldsymbol{z} \leq \boldsymbol{z}_{upp}, \end{aligned} \tag{2.4}$$

where $\boldsymbol{z}$ is the state variable, $J(\boldsymbol{z})$ represents the objective function, $\boldsymbol{f}(\boldsymbol{z})$ the collection of collocation constraints, $\boldsymbol{g}(\boldsymbol{z})$ the general nonlinear constraints, and $\boldsymbol{z}_{low}$ and $\boldsymbol{z}_{upp}$ the bounds on the augmented state.

According to Jackson [45], the state variable $\boldsymbol{z}$ is the augmented state of the initial and final time, and the values of the model states $\boldsymbol{x}$, and control inputs $\boldsymbol{u}$ at each of the collocation points. The augmented state $\boldsymbol{z}$ is defined as follows:

$$\boldsymbol{z} = [t_0, t_f, \boldsymbol{x}_0, \boldsymbol{u}_0, \ldots, \boldsymbol{x}_k, \boldsymbol{u}_k, \ldots, \boldsymbol{x}_N, \boldsymbol{u}_N]. \tag{2.5}$$

Leyffer and Mahajan [47], presented an overview of solvers summarized by their main characteristics in order to solve an NLP. The solvers are differentiated by factors such as the convergence mechanism, licensing (commercial or open-source), software interfaces, and the programming language. Van Koutrik [40] tested two different methods, Sequential Quadratic Programming (SQP) and Interior-Point (IP), capable of solving large-scale NLPs. Since it is difficult to distinguish a superior method, i.e., outperforming the other method in every aspect, both methods are implemented to compare their performance. Therefore, two different solvers have been compared [40], SNOPT, which is an SQP-based solver, and IPOPT, which is based on the IP method. From the comparison, it could be concluded that both solvers show similar performance for small problems. However, the IPOPT solver performed considerately better for larger problems, with many decision variables and collocation points. The outcome of this comparison supports the claim in the book of Nocedal and Wright [48] that IP methods for solving large NLPs often outperform SQP methods. For a detailed description of the methods, the reader is referred to the book of Nocedal and Wright [48].

2.5 Discussion

For evasive maneuvers, it is essential to use the maximum capability of the vehicle in the longitudinal and lateral direction. Therefore, some more in-depth knowledge has been provided about drifting, which is an unstable cornering condition. It has been proved that three corning equilibria exist using a nonlinear single-track vehicle model combined with a nonlinear tire model, The vehicle and tire models implemented to generate an optimal trajectory should also capture this unstable behavior.

Two of the most commonly used trajectory generating approaches have been discussed that can generate an optimal trajectory for various types of corners. The Rapidly-exploring Random Trees and Numerical optimization methods have been considered. Both are capable of generating a trajectory to negotiate a corner while using a completely different approach. The RRT and RRT* algorithms connect randomly generated points until it reaches the goal. The numerical optimization technique uses a cost function and predefined constraints to determine the optimal trajectory. A drawback of the RRT* algorithm is that the number of nodes must go to infinity before the optimal trajectory is found. In the research of Tavernini et al. [30], it is shown that the numerical optimization method is capable of computing the optimal trajectory and the corresponding vehicle sideslip angle β to negotiate a corner for various vehicle configurations and road surfaces. Therefore, this method will be used to generate the optimal evasive trajectories and corresponding vehicle state profiles.

Various research projects on the topic of collision avoidance and collision mitigation have been examined. However, none of the projects discussed on collision avoidance focused on utilizing the maximum capability of the vehicle in the lateral and longitudinal direction. In addition, almost all of these researches employed a kinematic vehicle model, which is not capable of capturing the evasive maneuver characteristics presented in Section 2.1. Utilizing these characteristics is thus still a topic open for investigation. To the author's best knowledge, only limited information is available on the topic of collision avoidance using trajectory optimization techniques. Therefore, this topic has been added as a research objective to evaluate an optimized trajectory for collision avoidance using a nonlinear single-track vehicle model and a nonlinear tire model.

The order of the optimization and discretization phases determines the category a method falls in, i.e., direct methods discretize the problem first, while optimization is the first phase for indirect methods. Using the direct method has some advantages, and therefore the decision has been made to use this method. The first step in solving the OP using a direct method is transcribing the problem. All methods for the transcription process can be categorized into two classes: shooting methods and collocation methods. Shooting methods become relatively difficult to solve when path constraints are implemented. Thus, the decision is made to transcribe the problem using the trapezoidal collocation method, which is better suited for complex problems with path constraints. The next step is solving the NLP resulting from the transcription process. In previous research projects, various NLP solvers have been compared and based on their performances, the IP method is considered for this research.

Chapter 3

Maneuver and vehicle modeling

This research aims to generate the optimal reference trajectory for an evasive maneuver to avoid a potential collision based on a vehicle's capabilities in the lateral and longitudinal direction. In the literature review, it has already been concluded that a nonlinear vehicle model would be required for an accurate representation of evasive maneuver (drifting) characteristics. In this chapter, the track, vehicle, and tire models will be highlighted that are used to formulate the OP for trajectory generation.

In Section 3.1, an explanation of the maneuver model will be provided, including the reasoning behind using a curvilinear coordinate system. This section will also explain how transitioning segments between a straight and corner road segment are constructed using a clothoid curve. Thereafter, the nonlinear vehicle model is defined in Section 3.2, including the equations of motion, the decision to implement longitudinal load transfer, and the combined slip nonlinear tire model. Later, the vehicle model will be reformulated from the time domain to a spatial domain since the OP for trajectory planning is mainly formulated in terms of position.

3.1 Track modeling - Curvilinear coordinate system

In this section, the derivation of the kinematic equations that describe the vehicle's progress along the road centerline and a detailed explanation of the curvilinear coordinate system will be provided. In addition, the advantages of this coordinate system will be discussed. The subsequent subsection will explain the implementation of a clothoid curve, which is a transitioning curve that is commonly used to connect two road segments with different radii of curvature.

3.1.1 Curvilinear coordinate system

In the curvilinear coordinate system, the vehicle position and orientation are described using three curvilinear coordinates: the distance traveled along the road centerline s, the perpendicular distance between the vehicle's Center of Gravity (CG) and the road centerline Δy, and the angle relative to the heading angle of the road $\Delta\psi$. The track model represented in the curvilinear coordinate system is shown in Figure 3.1. Here, the radius of curvature is given by R and the corresponding curvature $\kappa = 1/R$ at any point s along the track. The vehicle's heading angle (ψ) relative to the heading angle of the road (ψ_s) is defined by $\Delta\psi = \psi - \psi_s$.

The dynamics are derived from a relation between the local vehicle frame and the curvilinear coordinate system. Three kinematic equations are derived using the relations shown in Figure 3.1, to describe the vehicle's progress:

$$\dot{s} = \frac{V\cos(\Delta\psi + \beta)}{1 - \Delta y \cdot \kappa}, \tag{3.1}$$

$$\dot{\Delta y} = V\sin(\Delta\psi + \beta), \tag{3.2}$$

$$\dot{\Delta\psi} = r - \kappa \cdot \dot{s}, \tag{3.3}$$

where V is the absolute velocity, β the vehicle sideslip angle, r the yaw rate of the vehicle. For a complete derivation of the equations related to the curvilinear coordinate system, the reader may consult Appendix C.

The curvilinear coordinate system has a couple of advantages over the cartesian coordinates for optimization purposes:

- The road boundary constraints can now easily be defined using two inequality constraints to ensure that Δy is always smaller than or equal to half the road width.

- A curvilinear coordinate system makes it easy to deal with a track that crosses itself since the track is described by the distance traveled along the road centerline s [42].
- The initial and final conditions relative to the road centerline can easily be formulated in the curvilinear coordinate system.

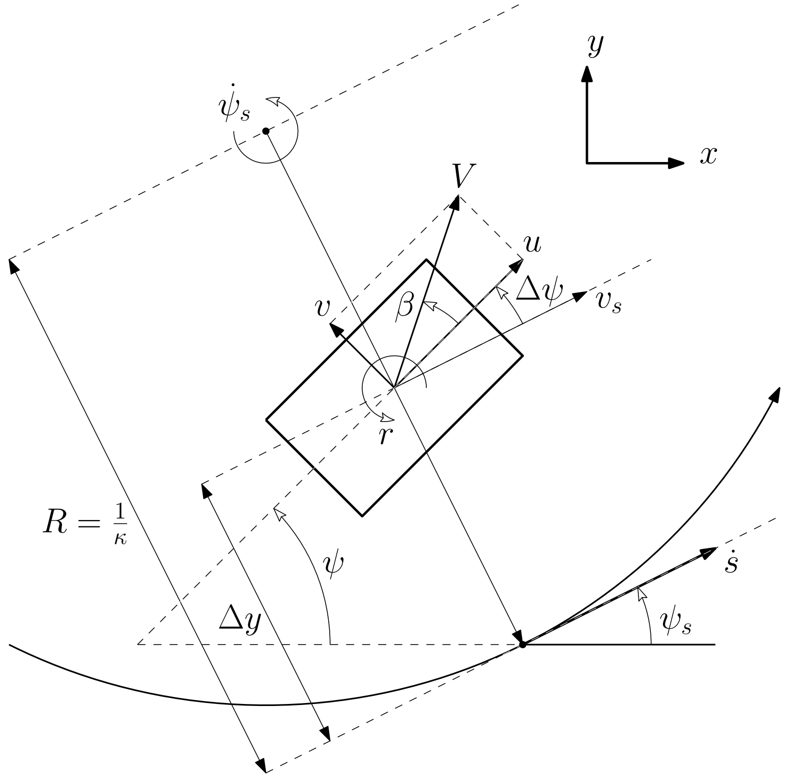


Figure 3.1: Curvilinear coordinate system, the coordinate s defines the distance travelled along the curve in combination with the two relative spatial coordinates Δy and $\Delta\psi$

3.1.2 Clothoid curve

A clothoid (or Euler spiral) is a curve with a curvature proportional to its length, i.e., the curvature is linearly changing with the length. Because of this property, a clothoid is often used as a transitioning curve in roads or railway tracks. The smooth change in the curvature, linearly increasing or decreasing, results in a progressive increase or decrease of the centrifugal forces to prevent that passengers experience undesirable jerk.

In this research, a transitioning clothoid curve may be added to the road geometry, and it is implemented using the alternative method proposed by Vázquez-Méndez and Casal [49]. The option is implemented to have the possibility to create all sorts of road geometries. The numerical method proposed by Vázquez-Méndez and Casal is an alternative to the classical method based on the Tayler expansion of the sine and cosine functions. For the alternative method, the forward Euler method is used to approximate the clothoid parametrization. The alternative method is defined as follows:

$$\begin{aligned} x^n &= x^{n+1} - \Delta s \cos\left(\lambda \frac{(s^n)^2}{2A^2} + \Phi_0\right), \quad n \in 0, ..., (N-1), \\ y^n &= y^{n+1} - \Delta s \sin\left(\lambda \frac{(s^n)^2}{2A^2} + \Phi_0\right), \quad n \in 0, ..., (N-1). \end{aligned} \tag{3.4}$$

where $A = \sqrt{RL}$, $N = \frac{L}{\Delta s}$, R is the minimum corner radius (R_{curve}), Φ_0 is the heading angle at the start of the transition segment, λ is the circle orientation (-1 (CW) and 1 (CCW)), Δs is the discretization step, and L is an arbitrary chosen length. This length L determines the slope of the gradually increasing curvature, since this is length required to go from $\kappa_{straight} \rightarrow \kappa_{curve}$.

According to Vázquez-Méndez and Casal, the obtained values for x^n and y^n are good approximations for the points along the transitioning curve. Moreover, the alternative method can provide good approximations for settings where the classical method fails. Therefore, this relatively simple yet effective method is used to construct the transitioning phases in this research. In order to construct the road geometry the open-source OpenCRG tool is used (see Appendix D). In Figure 3.2, two example road geometries are presented: one with the transitioning segment and one without it.

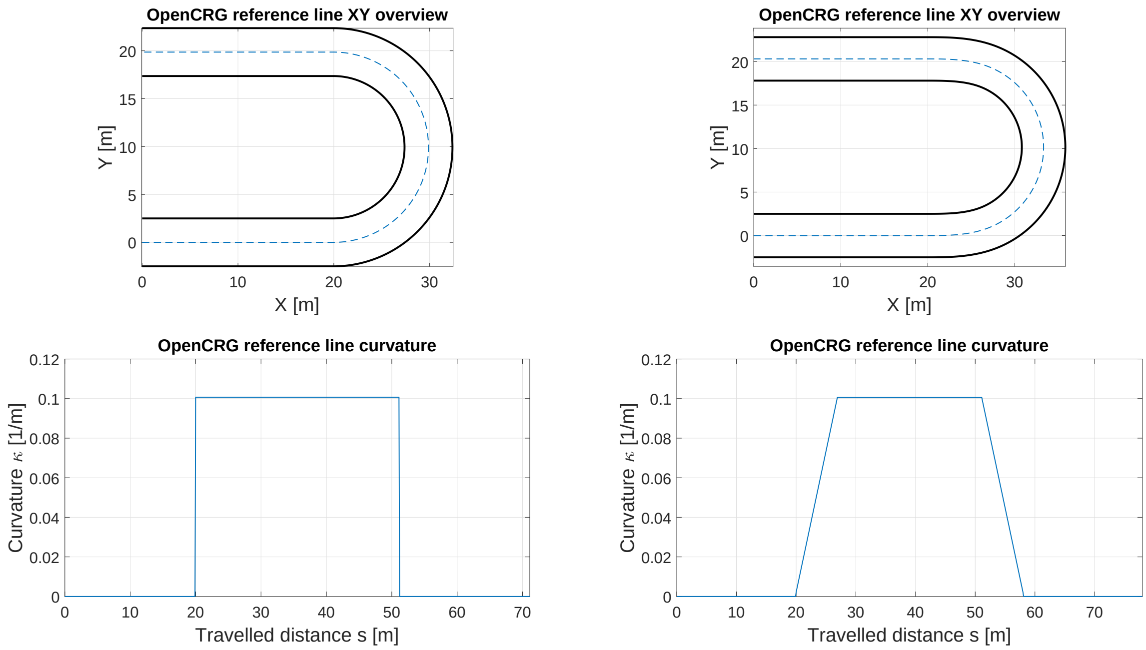


Figure 3.2: Road geometry generated using the OpenCRG tool with and without a transition segment: Clothoid exluded (left) and clothoid included (right).

3.2 Vehicle modeling

In the field of vehicle modeling, many different models with different purposes exist. Examples are the quarter-car and half-car models, which are vertical dynamics models and can be used to study ride comfort. In this research project, the vertical dynamics of the vehicle are of less importance. Therefore, longitudinal and lateral dynamics models will be presented in this section, neglecting some vehicle dynamics related to ride comfort. Lugner and Edelmann [50] categorized the longitudinal and lateral dynamics models into three categories:

1. Planar (linear) vehicle models
2. Extended Nonlinear Models
3. Near-Reality Vehicle Model (multi-body model)

In this section, only the first category will be considered because, in previous research projects, it appeared that a planar vehicle model in combination with a nonlinear tire model is capable of capturing the required characteristics for an evasive or drifting maneuver [3, 9, 10, 11]. The second category can be used to include suspension dynamics, the interaction between the tire and road, and the 3D motion of the vehicle body. As the name suggests, the third category is used for detailed vehicle simulations and can validate control algorithms based on the more basic models.

Three models can be derived from the first category, which have been used extensively in the literature on drifting and have been described by Baars [10] and Hindiyeh [3]: a two-state single-track model, a three-state single-track model, and a double-track model with load transfer. The single-track vehicle model, or bicycle model, is named after the fact that the left and right wheel are lumped into one wheel at each axle with twice the cornering stiffness and force capability [3]. The two-state bicycle model focuses on the lateral dynamics only, whereas the three-state model also incorporates the longitudinal dynamics, which results in an extra state. In addition, the two-state vehicle model uses only one input, the steering angle δ, while the three-state model uses two inputs, the steering angle δ and a longitudinal drive force F_x.

For this research, the three-state single-track vehicle model will be used, with three control inputs. The longitudinal drive force F_x will be substituted for the longitudinal slips λ_f and λ_r. This provides possibilities to consider different vehicle topologies. Therefore, the longitudinal slip values of the front and rear wheel, λ_f and λ_r, are used as a control input in the OP. This model appeared to adequately capture the vehicle states during a maneuver at which a vehicle is operating close to or beyond the stable handling limits. Tavernini et al. [30, 41] have used it to study vehicle behavior for navigating a 180° corner with and without the utilization of the handbrake. In addition, the influence of road conditions and transmission layouts has been tested. The capabilities of the single-track vehicle model are also highlighted in the research conducted by Goh [11]. Based on this vehicle model, a controller is developed capable of tracking a drift course with a predefined vehicle sideslip angle using a custom build all-electric Delorean. Therefore, the single-track vehicle model will be employed in this research, including longitudinal load transfer and a nonlinear tire model.

3.2.1 Single-track vehicle model

The Equations of Motion of the bicycle model can be derived using Newton's second law and the force and moment balance at the CG [3]:

$$m(\dot{v} + r \cdot u) = F_{x,f} \sin\delta + F_{y,f} \cos\delta + F_{y,r}, \tag{3.5a}$$

$$I_z \dot{r} = a(F_{y,f} \cos\delta + F_{x,f} \sin\delta) - bF_{y,r}, \tag{3.5b}$$

where v is the lateral velocity of the vehicle, δ is the steering angle, $F_{x,f}$ and $F_{x,r}$ the tire forces of the front and rear wheel in the longitudinal direction, and $F_{y,f}$ and $F_{y,r}$ the tire forces of the front and rear wheel in the lateral direction. The longitudinal dynamics of the vehicle can again be derived using the same principle. This derivation results in the third Equation of Motion which is expressed in terms of the third vehicle state, the longitudinal velocity u:

$$\dot{u} = \frac{F_{x,r} + F_{x,f} \cos\delta - F_{y,f} \sin\delta}{m} + r \cdot u \cdot \beta. \tag{3.6}$$

In Section 2.1, it has been explained that large sideslip angles characterize drifting and evasive maneuvers. Therefore it makes sense to express the dynamics in terms of the vehicle sideslip angle β. In addition, there can be a substantial difference between the longitudinal velocity u and absolute velocity V during drifting [11]. Therefore, the absolute velocity V and vehicle sideslip angle β are used instead of the longitudinal velocity u and lateral velocity v, which are used in (3.5) and (3.6). The rewritten equations of motion are as follows:

$$\dot{V} = \frac{F_{x,f} \cos(\delta - \beta) - F_{y,f} \sin(\delta - \beta) + F_{x,r} \cos\beta + F_{y,r} \sin\beta}{m}, \tag{3.7}$$

$$\dot{\beta} = \frac{F_{x,f} \sin(\delta - \beta) + F_{y,f} \cos(\delta - \beta) - F_{x,r} \sin\beta + F_{y,r} \cos\beta}{mV} - r, \tag{3.8}$$

$$\dot{r} = \frac{a(F_{y,f} \cos\delta + F_{x,f} \sin\delta) - bF_{y,r}}{I_z}, \tag{3.9}$$

where V is the absolute velocity, β the vehicle sideslip angle, r the yaw rate, F_{xf} and F_{xr} the longitudinal tire forces, F_{yf} and F_{yr} the lateral tire forces, and δ the steering angle. More details regarding the single-track vehicle model can be found in Appendix E.

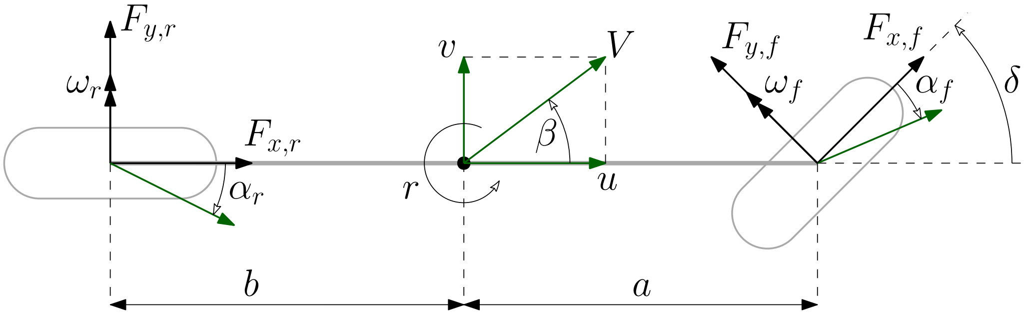


Figure 3.3: Single-track vehicle model.

3.2.2 Longitudinal load transfer

In the research of Velenis and Tsiotras [25] it has indicated that longitudinal load transfer could have a significant impact on the optimal trajectory. Therefore, the decision is made to add it to the vehicle model for more accurate results. The wheel loads on the front and rear wheels are composed of the static vertical load and the load transfer due to acceleration in the longitudinal direction. Therefore, the wheel loads are defined by:

$$\begin{aligned} F_{z,f} &= \frac{b}{a+b}mg - \Delta F_z, \\ F_{z,r} &= \frac{a}{a+b}mg + \Delta F_z, \end{aligned} \tag{3.10}$$

where ΔF_z represents the load transfer between the front and rear wheel. The longitudinal load transfer ΔF_z is defined as follows

$$\Delta F_z = \frac{h_{cg}}{a+b}(F_{x,f}\cos\delta - F_{y,f}\sin\delta + F_{x,r}), \tag{3.11}$$

where h_{cg} is the height of the center of gravity.

3.2.3 Tires

This subsection provides an overview of the tire characteristics on different road surfaces and compares the different tire models used in literature to study the characteristics of maneuvering beyond the stable handling limits.

On-road versus off-road

Tires behave significantly different on gravel in comparison to asphalt since the peak tire-road friction coefficient is occurring at a much lower longitudinal slip λ or lateral slip α [30]. This effect is visualized in Figure 3.4, which shows two tire-road interaction curves. The curves are obtained by varying the Magic Formula coefficients while the same values for the normal tire loads $F_{z,x}$ and $F_{z,y}$ are used, so basically, an equivalent tire is considered on two different road types: dry asphalt (1) and an off-road surface like gravel (2).

Tavernini et al.[30] used similar tire-road interaction curves to investigate how the road surface affects a minimum-time maneuver strategy for a so-called U-turn. It is concluded that the minimum-time maneuver on-road is characterized by low vehicle sideslip angles, like in circuit racing. On the contrary, the minimum-time maneuver off-road is characterized by large vehicle sideslip angles, like in rally racing. Acosta et al. [51] found a similar outcome using a different optimization objective. Instead of looking for a minimum-time solution, their objective has been to find an equilibrium solution while maximizing the centripetal acceleration on gravel and asphalt. On asphalt, the corresponding slip value required to achieve the maximum centripetal acceleration remains close to the location of the peak friction, with a relatively small vehicle sideslip angle β. On the other hand, the slip value required to achieve the maximum centripetal acceleration on gravel is significantly higher and has a significantly higher vehicle sideslip angle β.

Tire models

The available tire models can be divided into two basic categories: (semi-) empirical and physical (see [52]). The empirical model uses a mathematical formulation to represent a measured tire curve, and the physical tire model is based on a detailed or straightforward model of the tire's characteristics. Different tire models used in the literature on automated drifting include the Linear tire model, the (Combined slip) Brush tire model, and the Modified Dugoff tire model. A short derivation of these models is added in Appendix F. Additionally, this appendix compares the combined slip Magic Formula model and the Magic Formula similarity method to see the difference between a non-isotropic and an isotropic tire model. Implementation of a tire model depends on the application, e.g., a tire model implemented in a real-time control algorithm should be fast and accurately represent the tire characteristics.

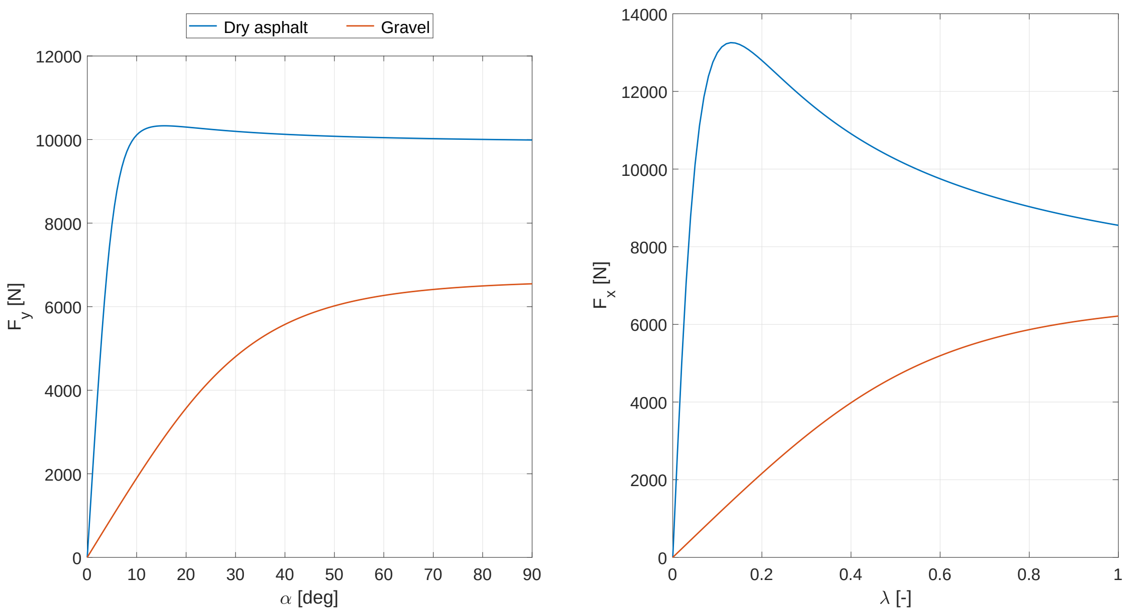


Figure 3.4: Lateral tire force F_y and longitudinal tire force F_x characteristics as function of the lateral slip angle α and longitudinal slip κ.

In this research, a combined slip Magic Formula tire model is used. For a detailed description of the other tire models, the reader may consult the book of Hans B. Pacejka [52]. One of the main reasons for choosing the Magic Formula tire model over the Brush tire model and the Modified Dugoff tire model is that it does not contain piecewise functions. The absence of these types of functions could prove beneficial in solving the OP at a later stage. Additionally, the coefficients for constructing the tire curves are available for dry asphalt and gravel. More information regarding this will follow after the description of the combined slip Magic Formula tire model.

The Magic Formula is widely used in the industry for vehicle simulations because the output represents the data of a measurement on a physical tire. The nominal tire forces under pure slip conditions, in the lateral or longitudinal direction, are computed using the general form of the Magic Formula tire model [52]. The base form of the Magic Formula,

$$F = D\ \sin[C\ \mathrm{atan}\{(1-E)Bx + E\ \mathrm{atan}(Bx)\}], \tag{3.12}$$

can be used to fit the longitudinal force F_x and lateral force F_y based on the input variable x, representing α or λ. The coefficients in the base formula determine the shape of the fit. It has to be noted that these coefficients have no physical quantity but are only curve fitting parameters. The meanings of the parameters are indicated here:

- D determines the peak value, $D = \mu F_z$
- C determines the limit value when $x \to \infty$

- BCD determines the slope near the origin
- B, E, and C determine the location of the peak

In Subsection 2.1 it has been explained that in combined slip scenarios, the lateral capabilities of the tire are compromised by the longitudinal slip and the other way around. According to Pacejka [52], Michelin developed a purely empiric method to describe the effect of combined slip on the force characteristics in the lateral and longitudinal direction. A weighting function G is introduced, which can be used to determine the effect of longitudinal slip λ on the lateral force F_y and lateral slip α on the longitudinal force F_x. The combined slip equations are defined as follows:

$$B_{x\alpha} = B_{x1}\cos(\mathrm{atan}(B_{x2}\lambda)), \tag{3.13a}$$
$$G_{x\alpha} = \cos(C_{x\alpha}\mathrm{atan}(B_{x\alpha}\alpha)), \tag{3.13b}$$
$$F_x = F_{x0}G_{x\alpha}, \tag{3.13c}$$

$$B_{y\lambda} = B_{y1}\cos(\mathrm{atan}(B_{y2}\alpha)), \tag{3.14a}$$
$$G_{y\lambda} = \cos(C_{y\lambda}\mathrm{atan}(B_{y\lambda}\lambda)), \tag{3.14b}$$
$$F_y = F_{y0}G_{y\lambda}, \tag{3.14c}$$

where F_{x0} and F_{y0} are the nominal tire forces under pure slip conditions in the lateral or longitudinal direction. Figure 3.5 is created, using the tire parameters in Appendix B, to visualize the effect of combined slip on the force characteristics in the lateral and longitudinal direction.

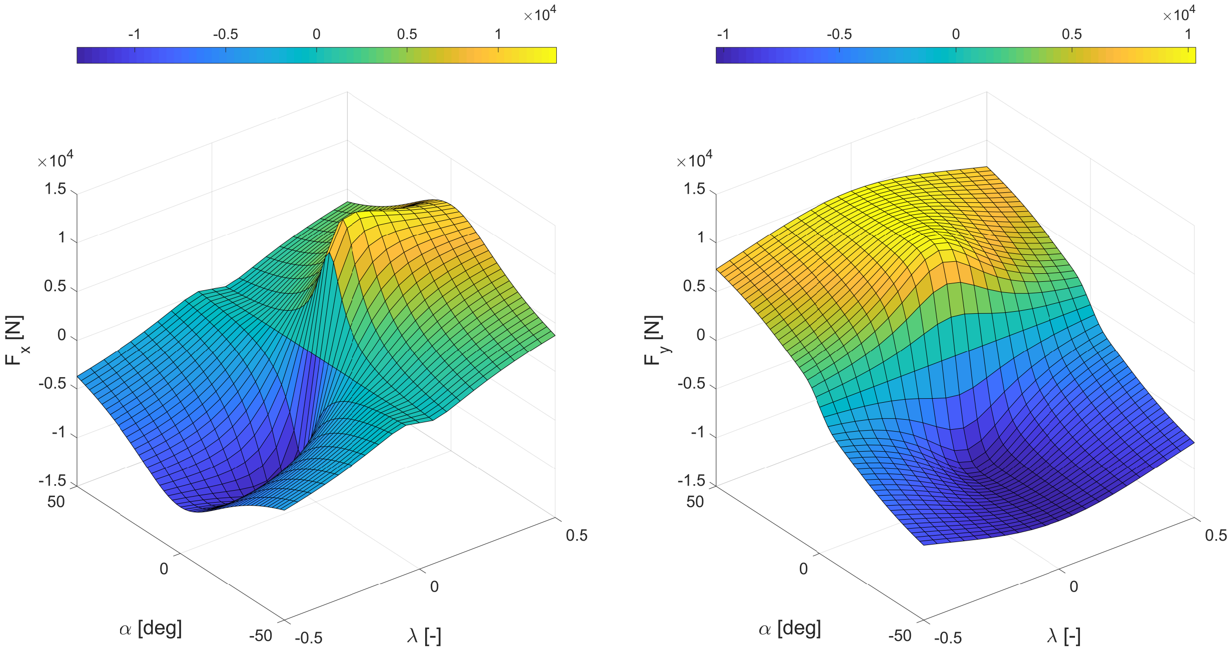


Figure 3.5: Combined slip tire forces in the longitudinal and lateral direction for the dry asphalt tire represented in Figure 3.4.

The Magic Formula coefficients for the combined slip model are adopted from Olofsson et al. [29]. The model coefficients are available for dry asphalt, wet asphalt, snow, and smooth ice in their research. As indicated in Figure 3.4, only the coefficients for the dry asphalt tire are used in this research. Since it is challenging to find Magic Formula coefficients for off-road tires, especially for tires used on, e.g., rally cars (confidential data), the gravel coefficients are adopted and reconstructed from a paper written by Tavernini et al. [30]. The Magic Formula coefficients for the two different types of tires are presented in Appendix B.

3.3 Conclusion

A detailed description of the track, vehicle, and tire model used in this research has been provided. The curvilinear coordinate system is selected for the track model since it has a couple of advantages over the Cartesian coordinate system. As a result, the track is represented by the road curvature κ at any point s along the track. This information is generated using functions of the OpenCRG tool in combination with the nine predefined road variables presented in Appendix D.

For this research, the three-state nonlinear single-track vehicle model is used, including longitudinal load transfer. The model utilizes three control inputs, the steering angle δ and a longitudinal drive force F_x resulting from the longitudinal slips λ_f and λ_r. In addition, a nonlinear combined slip Magic Formula tire model is implemented to determine the tire forces acting on the front and rear wheel. The conclusion has been drawn that tires behave significantly different on asphalt and gravel due to the shape of the tire-road interaction curves. Therefore, a dry asphalt and gravel tire will be employed to see how the choice for a specific tire influences the optimization algorithm results.

Chapter 4

Trajectory generation and optimization

Professional drivers aim to apply the optimal control inputs to navigate a prespecified track as fast as possible. These drivers are capable of driving the vehicle at or near the physical limits without losing control. Fully autonomous systems should also be able to safely control the vehicle within and beyond stable limit handling conditions. An important step to reach such a goal is to find the optimal trajectory for evasive maneuvers. The optimization algorithm should generate an optimal reference trajectory based on vehicle dynamics, tire forces, road boundaries, and obstacles information.

Section 4.1 will provide the general formulation for an Optimization Problem (OP). In Section 4.2, all elements present in the general formulation for an OP will be discussed and formulated for the specific optimization problem solved in this research. Since the optimization problem is mainly parameterized in terms of the vehicle position, a change of the conventional independent variable (time) is proposed in Section 4.3. This section will also provide a summary of the previous sections in the form of the final optimal control problem for motion planning.

In the second half of this chapter, solving the formulated OP will be covered. For this research, the trapezoidal collocation method is used to transcribe the OP. More details regarding the transcription process will be provided in Section 4.4. This process converts the OP into a nonlinear program (NLP).

The optimization software used for this research is covered in Section 4.5. First, the ICLOCS2 software, which is used for the transcription of the OP, will be addressed. Next, the interior-point NLP solver IPOPT is discussed. For optimization, a trade-off decision always has to be made between accuracy and efficiency. Therefore, the influence of the discretization step will be discussed in Section 4.6. Finally, Section 4.7 will summarize the formulated OP and the process of solving this OP for the generation and optimization of a vehicle trajectory on a predefined road geometry.

4.1 General formulation of the optimization problem

This section will introduce the general formulation of the OP, which is used for trajectory optimization in this research. The focus here will be restricted to single-phase continuous-time OPs, i.e., the system dynamics are continuous.

The general formulation to describe the OP in this section is defined in the *Bolza form*, more information on this can be found in Subsection 4.2.2. The definition is adopted from Kelly [53]:

$$\min_{t_0, t_f, \boldsymbol{x}(t), \boldsymbol{u}(t)} \quad J = \underbrace{\Phi(\boldsymbol{x}(t_0), t_0, \boldsymbol{x}(t_f), t_f)}_{\text{Mayer Term}} + \underbrace{\int_{t_0}^{t_f} L(\boldsymbol{x}(t), \boldsymbol{u}(t), t)dt}_{\text{Lagrange Term}}, \tag{4.1}$$

$$\text{s.t.} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{4.2}$$

$$\boldsymbol{h}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)) = \boldsymbol{0}, \tag{4.3}$$

$$\boldsymbol{g}(t, \boldsymbol{x}(t), \boldsymbol{u}(t)) \leq \boldsymbol{0}, \tag{4.4}$$

where J is the objective function, Φ and L are the Mayer and Lagrange terms of the objective function, $\boldsymbol{x}$ are the model states, $\boldsymbol{u}$ are the control inputs, and t_0 and t_f are the initial and final time. The most important constraints are the system dynamics in (4.2). The nonlinear equality and inequality constraints, or path constraints, are denoted by (4.3) and (4.4). Path constraints are used to enforce a restriction on the solution of the OP over the entire trajectory, e.g., the road boundaries, obstacles, and limitations on the inputs.

4.2 Optimization problem for motion planning

In this section, the specific case of the OP for motion planning will be formulated. Subsection 4.2.1 will discuss the system dynamics, the model states, and control inputs. Two types of objective functions are used in this research, the Mayer form and the Lagrange form. The two types will be covered in Subsection

4.2.2. The following subsection will introduce the equality and inequality constraints imposed on the solution of the OP. The initial conditions of the OP are described in Subsection 4.2.3. Finally, in Subsection 4.2.5 additional constraints are introduced to implement obstacles within the road boundaries.

4.2.1 System dynamics

The system dynamics imposed on the OP for motion planning are a combination of the track model introduced in Section 3.1, and the vehicle model described in Section 3.2. A complete overview of the model states and control inputs that belong to the defined OP used in this research is presented in Table 4.1.

Table 4.1: Overview of the model states and control inputs

Label	Type	Description	Symbol	Unit
x_1	Model state	Absolute velocity	V	m/s
x_2	Model state	Vehicle sideslip angle	β	rad
x_3	Model state	Yaw rate	r	rad/s
x_4	Model state	Distance along the centerline of the path	s	m
x_5	Model state	Distance perpendicular to the centerline	Δy	m
x_6	Model state	Angle relative to the centerline of the path	$\Delta\psi$	rad
u_1	Control input	Steering angle	δ	rad
u_2	Control input	Longitudinal slip front wheel	λ_f	$-$
u_3	Control input	Longitudinal slip rear wheel	λ_r	$-$
u_4	Control input	Normalized longitudinal load transfer	ΔF_z	$-$

State equations The state equations of the OP for motion planning are a combination of the Equations of Motion of both the track model and vehicle model. The complete model used in this research consists of six state variables:

$$x = \begin{pmatrix} V & \beta & r & s & \Delta y & \Delta\psi \end{pmatrix}^\top. \tag{4.5}$$

The complete model is defined in 4.6,

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{V} \\ \dot{\beta} \\ \dot{r} \\ \dot{s} \\ \dot{\Delta y} \\ \dot{\Delta\psi} \end{bmatrix} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} \dfrac{F_{x,f}\cos(\delta-\beta) - F_{y,f}\sin(\delta-\beta) + F_{x,r}\cos\beta + F_{y,r}\sin\beta}{m} \\ \dfrac{F_{x,f}\sin(\delta-\beta) + F_{y,f}\cos(\delta-\beta) - F_{x,r}\sin\beta + F_{y,r}\cos\beta}{mV} - r \\ \dfrac{a(F_{y,f}\cos\delta + F_{x,f}\sin\delta) - bF_{y,r}}{I_z} \\ \dfrac{V\cos(\Delta\psi+\beta)}{1-\Delta y\cdot\kappa} \\ V\sin(\Delta\psi+\beta) \\ r - \kappa\cdot\dot{s} \end{bmatrix}, \tag{4.6}$$

where all model states and control inputs are a function of time, but for simplicity, this notation is omitted.

Control inputs The goal of solving an OP is to find a combination of control inputs that minimizes/maximizes an objective function. For a driver, these control inputs are the steering wheel angle, accelerator and brake pedal position. In this research the longitudinal slip values, λ_f, and λ_r, of the front and rear wheels are selected as the control inputs. This selection isolates the problem from decisions on the engine type and powertrain system. For future research, the slip could be controlled using a low-level controller, dependent on the vehicle's powertrain topology and brake system layout. The controller may be similar to how ABS works, e.g., preventing wheel lock during braking.

In addition to the longitudinal slip values, there are two extra control inputs, the steering angle δ and the normalized longitudinal load transfer ΔF_z. The last one represents the load transfer between the front and rear axle and is introduced as an additional algebraic variable. It is included as a control input to solve the algebraic loop present in the OP to compute the tire forces. This is necessary due to the way the longitudinal load transfer is modeled. The longitudinal and lateral tire forces are a function of the wheel loads, see (3.10), but these forces are also required to compute the load transfer, see (3.11). Therefore, the tire forces cannot be computed explicitly from the model states. By introducing the longitudinal load transfer ΔF_z as a control input, including an equality constraint to satisfy (3.11), the algebraic loop is solved.

4.2.2 Objective functions

The general formulation to describe the objective function of an OP is the *Bolza form*, as shown in (4.1). In the Bolza form, two terms are included in the objective function, the boundary term Φ and the integral term L. When only the boundary term Φ is included, the objective function is said to be in the *Mayer form*, and an objective function with only the integral term L is defined in the *Lagrange form*. The Mayer and Lagrange form are both used in this research to change the behavior of the optimal solution.

Two objective functions are used to formulate the problem for a Minimum Time of travel Maneuver (MTM) and a Maximum Exit Velocity Maneuver (MEVM). These two cost functions represent common cornering maneuvers encountered in circuit racing and rally racing. In both situations the final time t_f is unknown and follows from the solution of the OP. The objective function for the MTM is defined in the Lagrange form

$$J = \int_{t_0}^{t_f} dt, \tag{4.7}$$

while the objective function for the MEVM is defined in the Mayer form

$$J = -x_1(t_f) = -V_f. \tag{4.8}$$

The minus sign in (4.8) is added in order to maximize the velocity at the final position of the maneuver. The effect of both objective functions on the optimal solution is studied, and the results of this will be discussed in Chapter 5.

4.2.3 Boundary conditions

The boundary conditions are constraints that put restrictions on the initial and final states of the trajectory. Table 4.2 provides an overview of the boundary conditions imposed on the OP. Here, fixed means that the state has a predefined value, and free means that the optimization algorithm determines the value according to the imposed constraints, system dynamics, and objective function.

Table 4.2: Boundary conditions (BC)

Variable	**Initial BC**	**Final BC**
V	Free/Fixed	Free
β	0	0
r	0	0
s	0	Fixed
Δy	0	Free
$\Delta\psi$	0	0
t	0	Free

4.2.4 Constraints and bounds

In contradiction to the boundary conditions, the path constraints enforce restrictions on the optimal solution along the entire trajectory, and the bounds are used to limit the model states and control inputs. A bound is used to limit the steering angle of the vehicle to a certain maximum angle δ_{max} as follows:

$$\delta \leq |\delta_{max}|. \tag{4.9}$$

The longitudinal slip values λ_f and λ_r of the front and rear wheel are also bounded. The lower bound of the slip values is indicated by wheel lock during a braking action,

$$-1 \leq \lambda, \tag{4.10}$$

while the upper bound is determined by a constraint on the maximum available engine power,

$$\frac{T_f \cdot \omega_f + T_r \cdot \omega_r}{1000} - P_{max} \leq 0, \tag{4.11}$$

where T_f and T_r are the torques applied to the front and rear wheel, ω_f and ω_r are the front and rear wheel rotational velocities, and P_{max} is the maximum engine power in $[kW]$. The derivation of the equations to compute the wheel torques and wheel rotational velocities can be found in Appendix E.

As mentioned in Subsection 4.2.1, an additional control input and constraint are necessary to solve the algebraic loop present in the OP. An equality constraint is used to ensure that the longitudinal load transfer ΔF_z represents the load transfer between the front and rear axle. Since the optimal solution has to satisfy the constraints, this solves the algebraic loop. The equality constraint for the longitudinal load transfer ΔF_z is defined as follows:

$$\frac{h_{cg} \cdot F_x}{l} - \Delta F_z = 0 \quad \Rightarrow \quad \frac{h_{cg}(F_{x,r} + F_{x,f} \cos\delta - F_{y,f} \sin\delta)}{a+b} - \Delta F_z = 0, \tag{4.12}$$

where l is the wheelbase of the vehicle, and h_{cg} denotes the height of the CG. Additionally, the wheel loads on the front and rear wheel are limited by a constraint since these are not states of the system dynamics. The wheel loads are bounded to prevent that negative forces can occur as follows:

$$-F_z \leq 0. \tag{4.13}$$

The last constraints imposed on the system dynamics are necessary to limit the feasible path by the edges of the road. For the sake of simplicity, only the location of the midpoint of the front and rear wheel are constrained to prevent it from exceeding the road boundaries. As indicated in Section 3.1, the advantage of the curvilinear coordinate system is that the road boundary constraints can now easily be defined using two inequality constraints:

$$|\Delta y + a \cdot \sin\Delta\psi| - \frac{LW_{new}}{2} \leq 0, \tag{4.14}$$

$$|\Delta y - b \cdot \sin\Delta\psi| - \frac{LW_{new}}{2} \leq 0, \tag{4.15}$$

where LW_{new} denotes the reduced width of the road, and the absolute value part represents the midpoint of the front wheel in (4.14), and the midpoint of the rear wheel in (4.15). The implementation of the road boundary constraints is visualized in Figure 4.1. The green area represents the feasible domain, the light and dark red area indicate the infeasible domain, and the thick solid black lines represent the road boundaries. Additionally, the road width LW is reduced by half of the vehicle width W on both sides of the road to ensure that the vehicle bodywork also stays within the road boundaries.

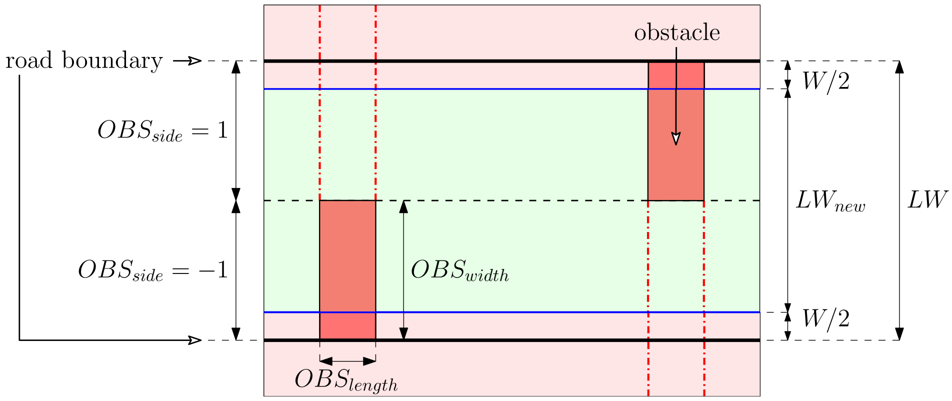


Figure 4.1: Road boundary and obstacle constraints implementation

4.2.5 Obstacle implementation

One of the research objectives is to develop a trajectory generator that has the possibility to include obstacles in the road geometry. Obstacles are implemented using the same method as for the road boundaries. One difference is that the obstacle constraints are only active at the positions of the obstacle, i.e., between the red dash dotted lines (OBS_{length}). In Figure 4.1, the obstacles are visualized on both sides of the road with a width of half the road width. It is just a parameter in the constraint equation, so the width of the obstacle can easily be changed. Since the constraints are only active at the longitudinal positions of the obstacle, the green area still represents the feasible domain. The obstacle constraints are applied to the midpoints of the front and rear wheel to prevent that the vehicle hits the obstacle. The constraints are defined as:

$$OBS_{side} \cdot (\Delta y + a \cdot \sin \Delta\psi) - \left(\frac{LW}{2} - OBS_{width}\right) \leq 0, \tag{4.16}$$

$$OBS_{side} \cdot (\Delta y - b \cdot \sin \Delta\psi) - \left(\frac{LW}{2} - OBS_{width}\right) \leq 0, \tag{4.17}$$

where OBS_{side} denotes whether the obstacle is located on the left (1) or right (-1) side of the road, and OBS_{width} is the obstacle width, i.e., the dimension of the obstacle in the lateral direction of the road.

4.3 Change of the independent variable

The track model is based on the vehicle position along the centerline and is thus based on a curvilinear coordinate system, as explained in Section 3.1. The system dynamics for OP uses time as the independent variable. However, the OP is mainly parameterized in terms of position because the entire road geometry is described as the distance along the path centerline s. Using the distance traveled along the path as the independent variable, instead of time, has the advantage that the state dimension is reduced by one because the dynamics w.r.t. the independent variable s become redundant.

Another advantage of using s as the independent variable is that problem changes from a free-horizon problem to a fixed-horizon problem. According to Velenis and Tsiotras [25], this fixed-horizon problem has the advantage that it easier converges (numerically) to a solution. The transformation of the independent variable, from time to the distance traveled, is also applied in [33], [40], and [42].

Time and distance traveled are closely related independent variables, and in order to make a change between them in the formulation of the OP, there has to be a one-to-one correspondence between them. It has to be ensured that the vehicle's velocity in the track-tangent position is positive, i.e., the traveled distance is a strictly increasing function of time. Using the relationship that follows from the curvilinear coordinate system and the fact that

$$ds = \dot{s}dt, \tag{4.18}$$

it is possible to calculate the derivative of time t with respect to the distance traveled s as follows

$$dt = \frac{dt}{ds}ds. \tag{4.19}$$

The term $\frac{dt}{ds}$ in (4.18) will be indicated by S_f from now on. According to Perantoni and Limebeer [42], the quantity S_f represents the reciprocal of the vehicle's velocity along the track centerline. Therefore, the quantity S_f is the inverse of (3.1), and it is defined as follows:

$$S_f = \left(\frac{ds}{dt}\right)^{-1} = \frac{1 - \Delta y \cdot \kappa}{V\cos(\Delta\psi + \beta)}. \tag{4.20}$$

The spatial dynamics can now easily be obtained by multiplying the system dynamics by S_f, which results in:

$$\frac{dx}{ds} = S_f \frac{dx}{dt} = \boldsymbol{f}(\boldsymbol{x}(s), \boldsymbol{u}(s)). \tag{4.21}$$

The dot symbol $(\cdot)$ is used to denote the derivatives with respect to time t and the prime symbol $(')$ is used to denote the derivatives with respect to the traveled distance s to make a clear distinction in the difference in the dynamics equations. The state derivatives with respect to the distance traveled s are given by:

$$\boldsymbol{x}' = \begin{bmatrix} V' \\ \beta' \\ r' \\ \Delta y' \\ \Delta\psi' \end{bmatrix} = S_f \cdot \begin{bmatrix} \dfrac{F_{x,f}\cos(\delta-\beta) - F_{y,f}\sin(\delta-\beta) + F_{x,r}\cos\beta + F_{y,r}\sin\beta}{m} \\ \dfrac{F_{x,f}\sin(\delta-\beta) + F_{y,f}\cos(\delta-\beta) - F_{x,r}\sin\beta + F_{y,r}\cos\beta}{mV} - r \\ \dfrac{a(F_{y,f}\cos\delta + F_{x,f}\sin\delta) - bF_{y,r}}{I_z} \\ V\sin(\Delta\psi + \beta) \\ r - \kappa \cdot \dot{s} \end{bmatrix}. \tag{4.22}$$

The objective function in the Lagrange form (MTM), as indicated in (4.7), has been reformulated as well to express it as a function of the distance traveled s instead of time t:

$$J = \int_{t_0}^{t_f} dt = \int_{s_0}^{s_f} \frac{dt}{ds}ds = \int_{s_0}^{s_f} \frac{1 - \Delta y \cdot \kappa}{V\cos(\Delta\psi + \beta)}ds, \tag{4.23}$$

where s_0 and s_f are the start and end positions of the track. For the objective function in the Mayer form (MEVM) this reformulation is not required, see (4.8). This form uses a boundary term, and in the case of MEVM, it maximizes the final velocity, which is not specifically dependent on one of two independent variables.

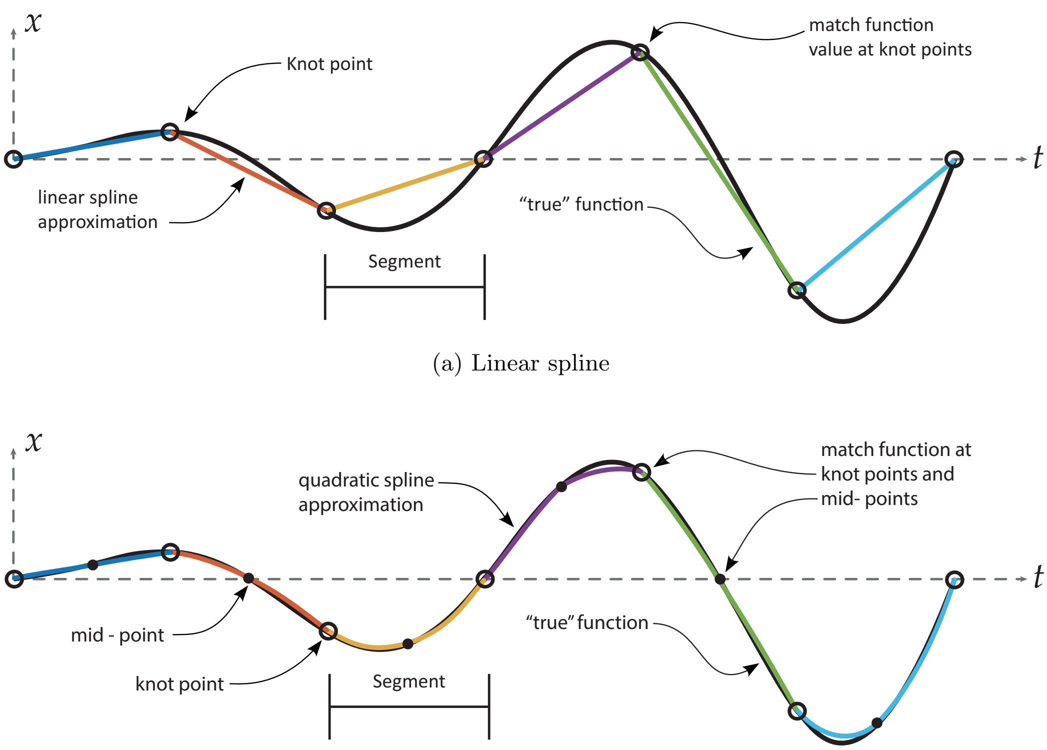
The final optimal control problem consists of the dynamics in (4.22), and (4.23) or (4.8) representing one of the two objective functions, and the constraintes and bounds presented in the previous sections.

4.4 Trapezoidal collocation

In this research, the trapezoidal method transcribes the continuous-time OP, discussed in the previous section, into a discretized NLP. The trapezoidal method uses the trapezoid rule for integration to convert the continuous functions into a discrete approximation. The procedure of applying the trapezoidal method is adopted from [53]. According to Jackson [45], the transcription process can be summarized in six steps:

1. Discretize the problem
2. Define an initial guess
3. Create the augmented state $\boldsymbol{z}$
4. Calculate the constraints
5. Solve the NLP
6. Interpolate the results

The written out transcription process using trapezoidal collocation can be found in Appendix G. Since the trapezoidal method is used, the interpolated control trajectory is constructed by a linear spline, while a quadratic spline represents the interpolated state trajectory. This construction follows directly from the derivation of the equations shown in the appendix. The difference between the two approximations is shown in Figure 4.2, where it is visible that the quadratic approximation is more accurate compared to the linear approximation. The model states, particularly the vehicle sideslip angle β, which is of interest for this research, are approximated using the more accurate quadratic spline. Based on the findings from other research projects [40, 42], it has been assumed that this approximation would be accurate enough for this research. If later it turns out to be too inaccurate for practical implementation, a higher-order quadrature scheme can be used like the Hermite-Simpson method.



(a) Linear spline

(b) Quadratic spline

Figure 4.2: Function approximation using splines. Figures adopted from [53].

4.5 Optimization software

The implemented optimization software and NLP solver utilized for solving the OP are discussed in this section. First, the ICLOCS2 optimization software will be reviewed, including the options implemented in the software. According to the developers of the ICLOCS2 software [54], it is designed to enable a flexible trade-off between accuracy and efficiency. Second, the IPOPT NLP solver will briefly be explained. IPOPT is the default and recommended NLP solver to be used in combination with the ICLOCS2 software. The last subsection discusses the gathered information and provides a conclusion on the decisions made.

4.5.1 ICLOCS2

In this research, the ICLOCS2 software is used, which is the updated version of the ICLOCS software used by Perantoni et al. [42] to solve a minimum-lap-time OP. The software offers a wide range of possibilities, i.e., different transcription methods and NLP solvers. Various tools are implemented in the software that could improve the efficiency to enable this flexibility. One of these tools is the automatic transcription of the OP, e.g., using the trapezoidal or Hermite-Simpson method. For more information regarding the transcription methods the reader is referred to Subsection 2.4.2 and Section 4.4. The advantage of automatic transcription is that the OP can be altered without the inconvenience of changing the code of the entire optimization algorithm. Another tool is Adigator [55], which can be used for algorithmic differentiation to supply derivative information to the NLP solver. In ICLOCS2, it is also possible for the user to provide the analytical derivatives to speed up the computation speed. For this research, the decision has been made to calculate the Jacobian and Hessian information using finite difference approximations. ICLOCS2 provides the user with the option to calculate this information automatically since every NLP solver requires first-order and second-order derivative information.

A reduction in accuracy could occur when using finite approximations due to numerical errors. However, no information is available on the possible degradation of the accuracy due to using finite difference approximations. Furthermore, checking the difference in accuracy between algorithmic differentiation, analytic differentiation, and numeric differentiation is not within the scope of this research. Therefore, this possible degradation in accuracy is taken for granted since the main objective of this research is to generate optimal trajectories for evasive maneuvers based on a vehicle's capabilities in the lateral and longitudinal direction.

4.5.2 IPOPT

The choice for a suitable NLP solver is dependent on multiple characteristics, e.g., the problem size and the number of constraints. In Subsection 2.4.3 it has already been concluded that IP methods for solving large NLPs often outperform SQP methods. IP methods, or barrier methods, approximate the constraints of the NLP as a set of boundaries surrounding a feasible region. According to Bianco et al. [56], IPOPT is the solver that is used in most projects that solve OPs for motion planning. This NLP solver is used successfully in similar research projects conducted by Van Koutrik [40], Perantoni et al. [42], and Bianco et al. [56]. Therefore, the proven solver, IPOPT (short for Interior Point OPTimizer), is selected for this research. In addition, it is advantageous that IPOPT is selected as the default NLP solver for the ICLOCS2 software. IPOPT is a primal-dual IP algorithm that utilizes a filter line-search method for solving NLPs [57]. For the mathematical details of the IPOPT solver, the reader is referred to a paper written by the developers of the algorithm, Wächter and Biegler [57].

4.5.3 Discussion

The software/algorithms selected in this section are based on the results obtained in previous research projects. The selected NLP solver and optimization software, implemented in MATLAB, are primarily tools used to find the optimal trajectory for this research. It is not a goal to find the most accurate or most efficient software available. Therefore, the default NLP solver of the ICLOCS2 software is used. For details regarding the solver and optimization software, the reader is referred to the corresponding references in the previous subsections.

4.6 Influence of discretization step

As mentioned in the introduction of this chapter, a trade-off must be made between accuracy and efficiency when solving OPs. The problem size has a significant influence on this trade-off. A reduction in the discretization step results in a more accurate approximation of the differential equations. Although, a reduction also increases the problem size resulting in longer computing times, and thus it compromises the efficiency.

For this research, the decision is made to use a fixed discretization step because the same step is used to generate the road geometry using the OpenCRG algorithm. In order to determine a suitable discretization step, a small test with five test cases is conducted to see what effect the discretization step has on the value of the objective function (cost). The settings for the five test cases are presented in Table 4.3. In addition, both objective functions, Minimum Time of travel Maneuver (MTM) and a Maximum Exit Velocity Maneuver (MEVM), are tested in combination with two types of tires and two road geometries.

Table 4.3: Test overview - varying the discretization step

Objective function	**Corner**	**Tire**	**Drivetrain**	**Color in Figure 4.3**
MTM	90°	2 (Gravel)	FWD	Yellow
MTM	90°	2 (Gravel)	RWD	Purple
MTM	180°	1 (Asphalt)	FWD	Green
MEVM	180°	1 (Asphalt)	FWD	Blue
MEVM	180°	1 (Asphalt)	RWD	Orange

The most suitable step size is determined by running an optimization with the same overall settings while varying the discretization step. The steps tested are: 0.05, 0.1, 0.2, 0.25, 0.5, and 1 $[m]$. In order to check the influence for both objective functions, as discussed in Section 4.2.2, the relative difference is determined between the results. The smallest step (0.05 $[m]$) is used as a reference and therefore has a relative difference of 0%, as can be seen in Figure 4.3.

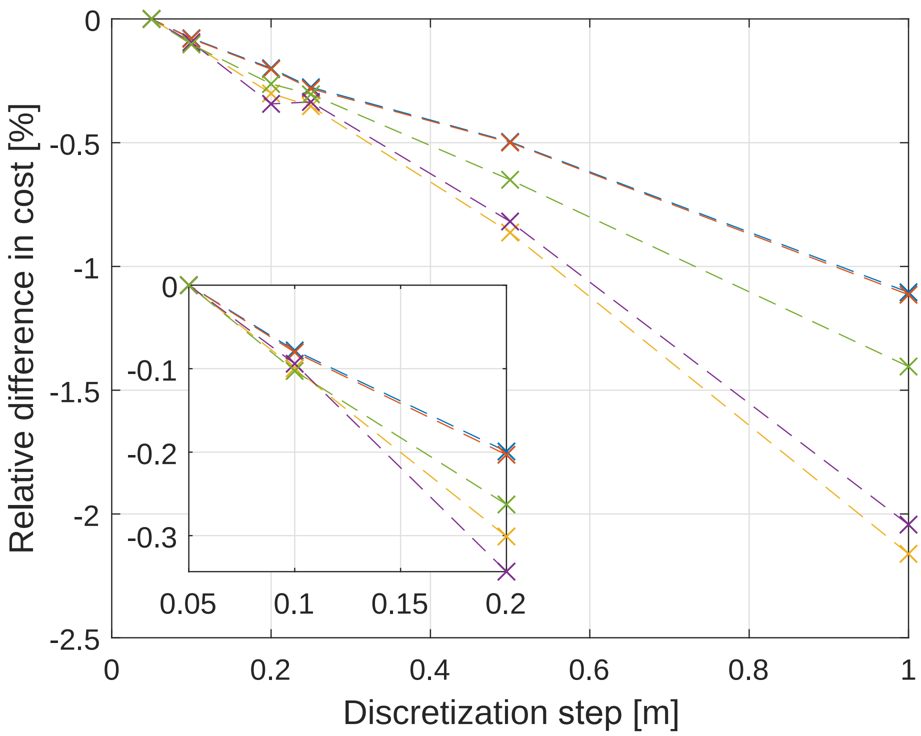


Figure 4.3: Relative difference in the objective function value for different discretization steps.

The results of the test are shown in Figure 4.3. One remark has to be made; the orange and blue results are almost on top of each other, and therefore the blue line is hardly visible. As can be seen, the relative difference increases for an increasing step size, and the results become less consistent. From the zoomed section of the figure, it can be concluded that the relative differences for an step of 0.1 $[m]$ are small, and the results are relatively consistent for the five test cases shown in Table 4.3. The conclusion has been drawn that a smaller discretization step than 0.1 $[m]$ is unnecessary to satisfy the trade-off between accuracy and efficiency. Therefore, the discretization step of 0.1 $[m]$ is selected.

4.7 Conclusion

A general formulation for the OP has been provided in this chapter. This general formulation is then used to define the OP for motion planning, including the system dynamics, the objective function, equality and inequality constraints, and the applicable boundary conditions. The constraints for adding an obstacle in the road geometry are explained as well. The OP for motion planning is mainly parameterized in terms of the vehicle position. Therefore, the decision has been made to change the independent variable from time t to the distance traveled along the road centerline s. This change has two advantages: the state dimension is reduced by one, and the problem changes from a free-horizon problem to a fixed horizon problem.

After formulating the final OP for motion planning, the next step is transcribing the continuous-time OP into a discretized NLP using trapezoidal collocation. This method uses the trapezoid rule for integration. The next step is solving the NLP, and thereafter the results are interpolated to construct a continuous trajectory again. Since trapezoidal collocation is used, the control inputs and system dynamics are approximated using linear splines. The more accurate quadratic splines represent the model states. From the results of previous research, it has been assumed that the trapezoidal collocation method is accurate enough for this research. This assumption is validated in Section 5.1, by checking the estimation of the absolute local discretization error.

The optimization software ICLOCS2 is used in this research because it offers a wide range of possibilities. For example, it offers the option to use finite difference approximations to calculate the Jacobian and Hessian automatically. This is an advantage because every NLP solver requires first- and second-order derivative information. Additionally, the selected NLP solver, IPOPT, is implemented in ICLOCS2 as the default solver.

In the last section, the influence of the discretization step has been checked. This is an essential step because a trade-off must be made between accuracy and efficiency when solving OPs. A fixed discretization step is used for this research because the same step is used in combination with the OpenCRG tool to construct the road geometry. The decision is made to use an step of 0.1 $[m]$ because of the consistent results for the tested cases.

The next chapter will present the accomplished results with the proposed trajectory generator using the information provided in the previous chapters.

Chapter 5

Algorithm assessment

All information provided in the previous chapters is combined to develop an optimization algorithm capable of finding the optimal trajectory for highly dynamic vehicle maneuvers. In addition, the algorithm is able to handle different vehicle topologies, road geometries with or without obstacles, and varying tire-road interaction curves. This chapter will discuss the simulation results for road geometries without obstacles. The results for roads with obstacles are discussed in the next chapter. The optimization algorithm is developed using MATLAB (R2019a) with the previously discussed software tools, ICLOCS2 and OpenCRG, coupled to it. In Section 5.1, the optimization algorithm is verified using a trivial example of driving a straight road section with both the objective functions, i.e., MTM and MEVM. For the trivial example, a change in the objective functions does not lead to different results. Therefore, Section 5.2 studies the influence of changing the objective function on the generated optimal trajectory for other road geometries. Section 5.3 will elaborate on the sensitivity analysis conducted, where the influence of varying vehicle and tire parameters is studied. Finally, in Section 5.4, a conclusion is formulated for the results found in this chapter.

5.1 Optimization validation

In this section, the optimization algorithm is validated using a trivial example to see if the coupling with ICLOCS2 and OpenCRG is working, and the results are as expected. A straight road section of 30 $[m]$ is simulated with a RWD topology and the dry asphalt tire employed for the validation. The boundary conditions for this simulation are set according to Table 4.2. As already mentioned in the introduction of this chapter, both objective functions from Section 4.2.2 are tested. The optimized trajectory for the two simulations is shown in Figure 5.1, including the relevant model states and control inputs. In this case, there is no use in showing the other four model states (β, r, Δy, and $\Delta\psi$) since the vehicle is driving in a straight line without steering. From this figure, it can be concluded that the change of the objective function does not influence the results since all states and inputs coincide.

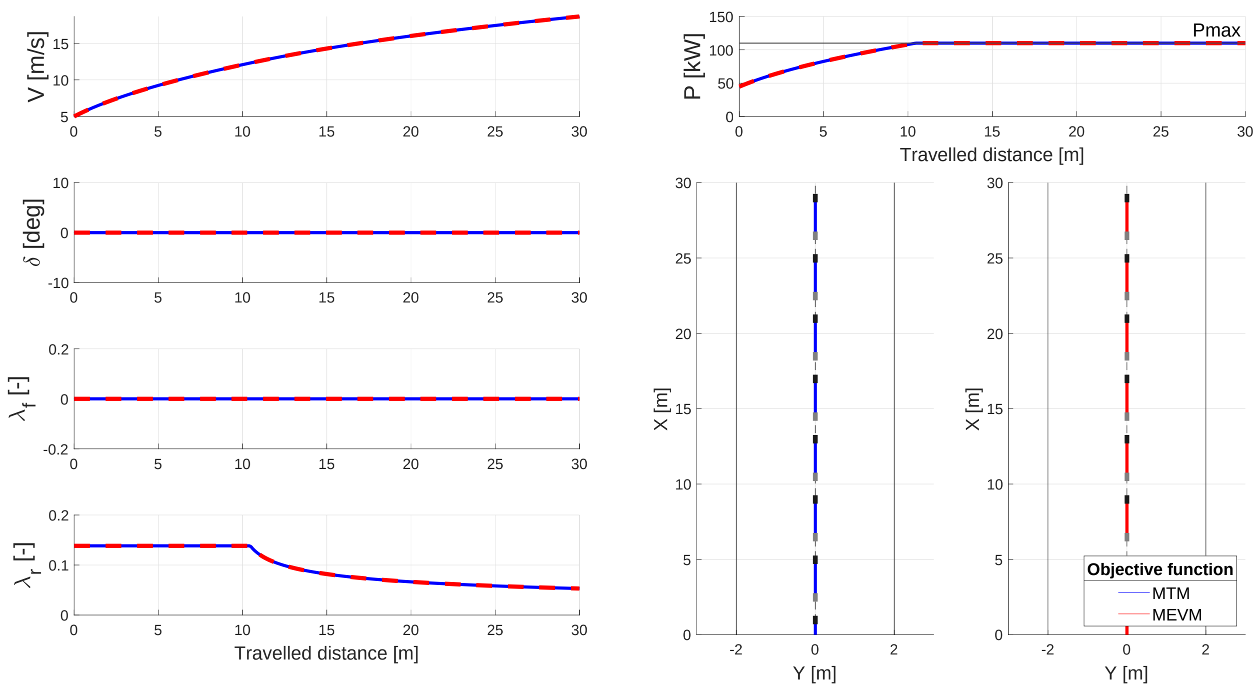


Figure 5.1: Validation of the optimization algorithm for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

The results for the trivial example are as expected for both of the objective functions. Since the vehicle is driving on a straight road section, the optimal trajectory uses the (optimal) rear longitudinal slip value λ_r that results in the maximum longitudinal tire force F_x. This statement is valid for the result from both objective functions, MTM and MEVM. From Figure 3.4, it can be seen that this optimum slip value is approximately 0.13. This value is also the utilized slip in the optimal trajectory until the maximum engine power available is reached. From that point on, the applied slip value decreases for an increasing velocity while the vehicle remains to operate at the maximum available engine power.

The ICLOCS2 software is also capable of estimating the absolute local error using Romberg quadrature. The error estimates for each segment $(t_{k+1} - t_k)$ of the state variables are presented in Figure 5.2. This method repeatedly applies trapezoidal quadrature at 2^n points in each segment and then uses Richardson extrapolation to get an estimation for the discretization error (n is set to 10 in ICLOCS2). According to Betts [38], Romberg quadrature has a tolerance close to machine precision. This property makes it a useful method for estimating the discretization error e in each segment.

The absolute local errors for the four states that are equal to zero in this simulation (β, r, Δy, and $\Delta\psi$) are most likely present due to rounding errors in the computation. These four states are zero because the vehicle is driving in a straight line. From 5.2, it can be concluded that the absolute local error for the states that are changing (V) is still small. The small absolute error can be explained by inspecting Figure 4.2b. Since the velocity is slowly increasing when the vehicle progresses, it can accurately be represented by a quadratic spline. Therefore, the absolute local error is small for this particular state and simulation.

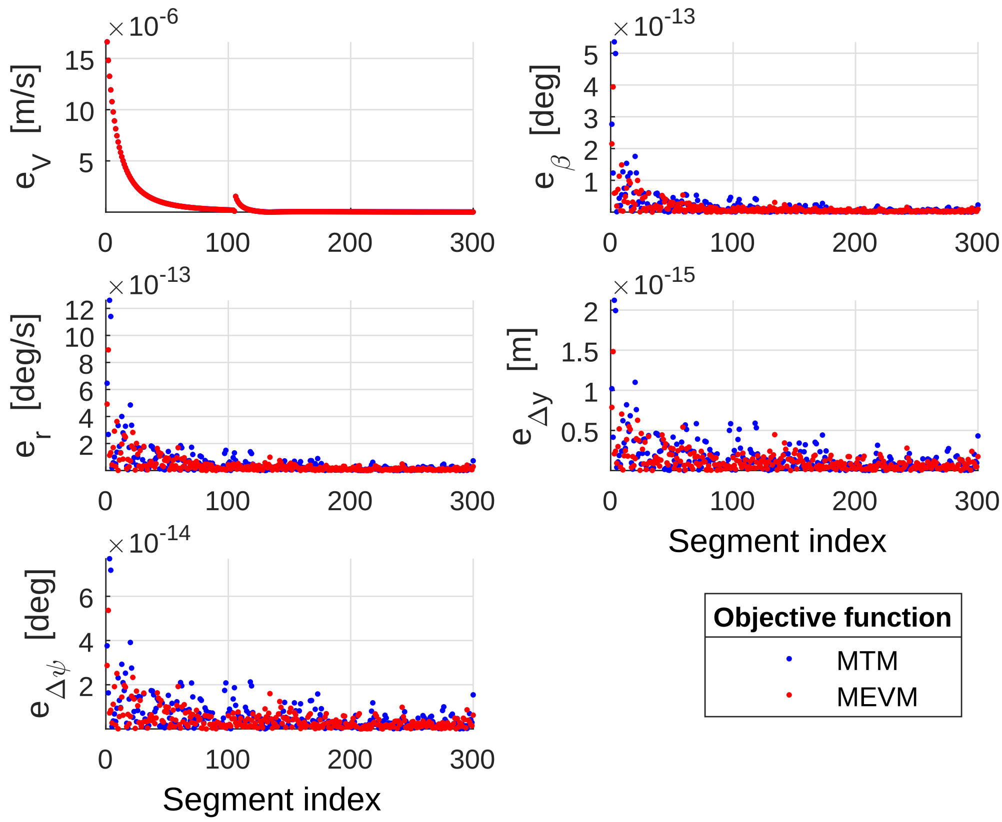


Figure 5.2: Absolute local error for the state variables, based on the integral of the error over each segment (due to discretization using the trapezoidal method).

In Appendix H, the absolute error plots are added for the optimal maneuvers of negotiating a 90° corner. The same conclusion can be drawn as for the absolute error plot in this section. The errors are a couple of orders of magnitude smaller than the values of the model states. Therefore, no further attention will be paid to the discretization error e.

5.2 Influence of the objective function

For the trivial example of driving on a straight road, it has been concluded that changing the objective function from MTM to MEVM did not influence the optimal solution. In this section, the two objective functions will be applied to find the optimal trajectory for negotiating a 180° corner, with a corner radius of 10 $[m]$ for the road centerline and two straight road sections of 20 $[m]$. Additionally, all optimizations carried out in this section use an RWD vehicle topology.

The boundary conditions from Section 4.2.3 are applied to the optimizations shown in this section. The initial velocity V_0 is free, i.e., it follows from the solution of the minimized/maximized objective function. The initial conditions for the other four model states (β_0, r_0, Δy_0, and $\Delta\psi_0$) are fixed to be zero to ensure that the vehicle starts driving straight ahead in the middle of the road at the start of the simulation. The final velocity V_f and lateral offset with respect to the road centerline Δy_f are free. The other three remaining vehicle states (β_f, r_f, and $\Delta\psi_f$) are fixed to be zero to ensure that the vehicle is driving straight ahead again at the end of the maneuver. A summary of the boundary conditions can be found in Table 5.1. The constraints discussed in Section 4.2.4 are active as well.

Table 5.1: Boundary conditions for the 180° corner simulations

Variable	**Initial BC**	**Final BC**
V	Free	Free
β	0	0
r	0	0
Δy	0	Free
$\Delta\psi$	0	0

The optimization is conducted for two types of tires, the dry asphalt tire, and the gravel tire, and the results are shown in Figure 5.3 and Figure 5.4 respectively. In addition, the longitudinal and lateral tire force characteristics for these tires have been shown in Figure 3.4.

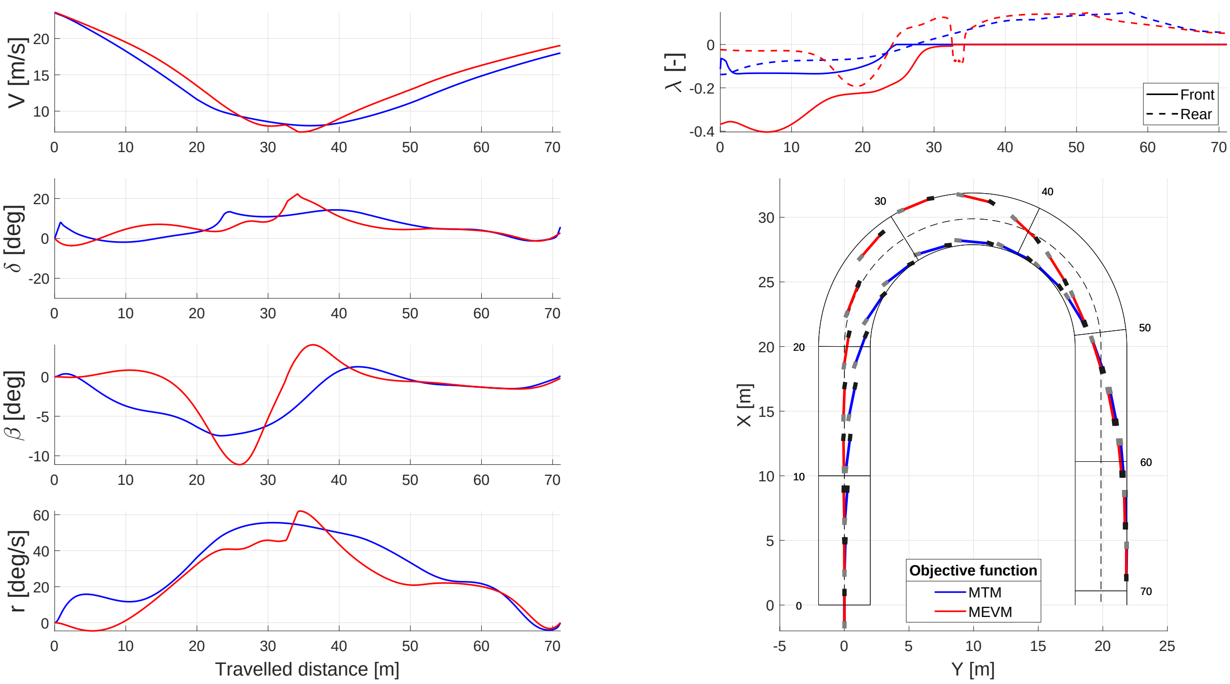


Figure 5.3: Simulation of a 180° corner with tire 1 (dry asphalt) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

The blue lines represent the results corresponding to the optimization with the MTM objective function in both figures, and the red lines represent the MEVM maneuver. For clarity, the usage of these colors is consistent throughout this section and Section 6.1. The dark grey box in the trajectory plot represents the front wheel of the single-track vehicle model, and the light grey box depicts the rear wheel. The traveled distance $[m]$ along the road centerline enables the reader to understand the vehicle behavior along the trajectory better. All model states and control inputs, except ΔF_z, are included to compare the optimal vehicle behavior for both objective functions. As discussed in Section 4.2.1, the algebraic variable ΔF_z is used as a control input to solve an algebraic loop present in the OP, and therefore this variable is not visualized in the figures.

The optimal maneuvers with the dry asphalt tire employed for both objective functions, MTM and MEVM, are shown in Figure 5.3. What stands out in the figure is that both maneuvers are characterized by relatively low vehicle sideslip angles β and steering angles δ. In contrast to the optimal maneuver using the dry asphalt tire, the optimal maneuvers where the gravel tire is employed are characterized by relatively large vehicle sideslip angles β and counter steering. Both objective functions trigger the vehicle to drift through the corner, as presented in Figure 5.4. The driving behavior for both employed tires can be explained by looking at their longitudinal and lateral tire force characteristics. For the dry asphalt tire, the maximum forces in both directions are achieved at small slips. Unlike the tire characteristics for the dry asphalt tire, the maximum tire forces for the gravel tire are achieved at large slips.

The concept of the corner apex should be explained first to describe the trajectories for both objective functions. In motorsports, the concept of the corner apex is used to describe the point where the vehicle is closest to the inner road boundary of the corner. The corner apex is also referred to as the clipping point of the corner. Depending on the road section that comes after the corner, a professional racing driver would apply one of the two optimized trajectories, MTM or MEVM, to maximize its driving performance independent of the employed tire. A professional driver will almost always take the corner using a late apex when a long straight follows after the corner since maximizing the exit velocity will gain more time than cornering with a slightly higher velocity. As can be seen in Figures 5.3 and 5.4, this is true independent of the tire choice. If the exit velocity is less important because multiple corners are chained together, then maximizing the corner velocity results in faster lap times. Especially for the gravel tire, faster times can be achieved since the vehicle can approach the corner at a significantly higher velocity.

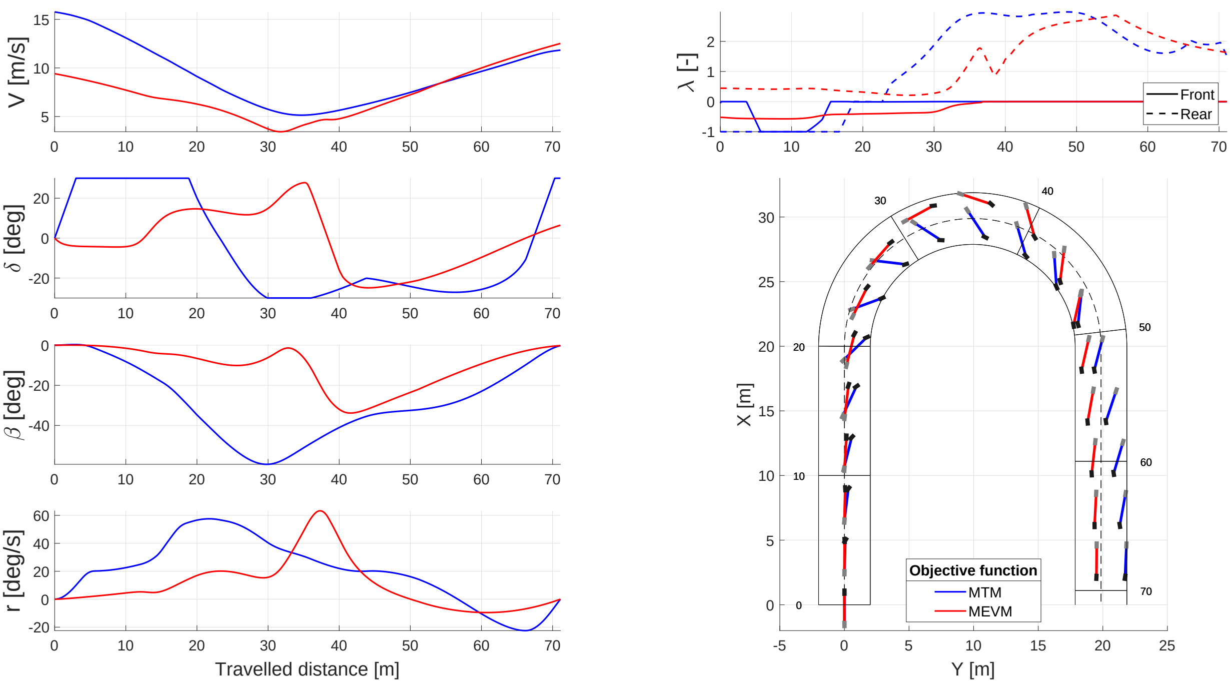
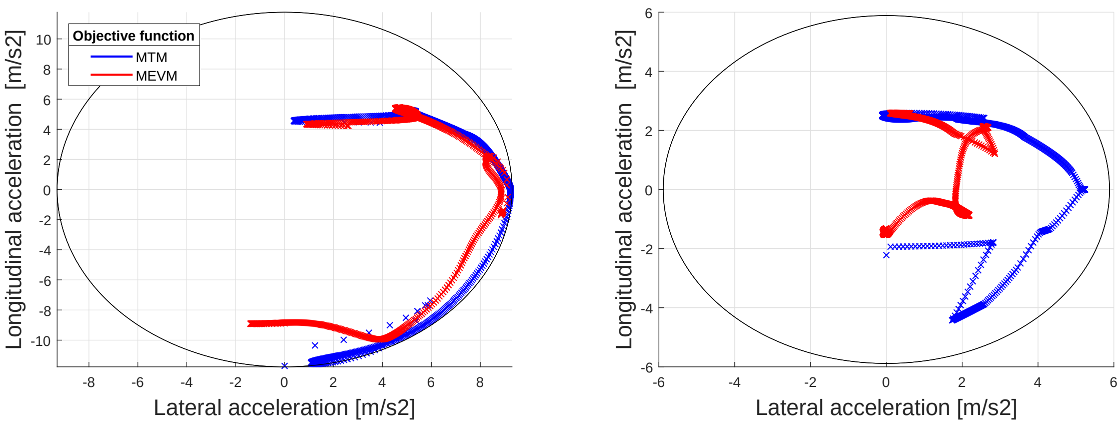


Figure 5.4: Simulation of a 180° corner with tire 2 (gravel) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

In order to visualize how closely the vehicle is operating to its limits, a G-G diagram is created for the optimal trajectories of the two objective functions, with a dry asphalt or gravel tire employed. A G-G diagram is a graph where the longitudinal and lateral acceleration of a vehicle are plotted against each other. The G-G diagrams for the two tires are displayed in Figure 5.5. An ellipse equation is implemented for this research to plot the 'limit'-circle, based on the friction coefficients in the longitudinal and lateral direction. The longitudinal weight transfer implemented in the vehicle model is neglected in this case. According to Kritayakirana [6], the addition of the longitudinal weight transfer results in the GG-diagram no longer being an ellipse, but it is slightly distorted. The 'limit'-circle is only used to indicate how close the vehicle is operating to its limits. Therefore, the deviation in shape is taken for granted.

The G-G diagram of the dry asphalt tire, provided in Figure 5.5a, illustrates that the optimal trajectory for an MTM is operating close to the vehicle's limits. I.e., the maximum capabilities of the vehicle in the longitudinal and lateral direction, using the combined slip tire model, are exploited in the optimal solution. However, this conclusion cannot be drawn for the MEVM. Instead, the vehicle takes a late apex in order to apply a positive longitudinal slip (throttle) as early as possible. Since this results in a faster utilization of the longitudinal tire force, the final velocity is higher than the final velocity of an MTM maneuver.

Where the maximum tire forces in the longitudinal and lateral direction are achieved at small slips for the dry asphalt tire, the maximum tire forces for the gravel tire are attained at large slips. From the G-G diagram of the gravel tire, see Figure 5.5b, it is apparent that the optimal solution is not exploiting the maximum tire forces that are available at large slips. Thus, a trade-off may be present between developing extremely large slips to reach the maximum tire force in either direction and the cost following from the objective function. In the optimal solution, this trade-off is most likely balanced to minimize the maneuver time or maximize the exit velocity.



(a) Tire 1: dry asphalt

(b) Tire 2: gravel

Figure 5.5: G-G diagram for the simulations of a 180° corner.

In Appendix H, the results of the optimization are presented for a 90° corner. The conclusions drawn in this section are also applicable to the optimal trajectories for this road geometry.

5.3 Sensitivity analysis

All optimized trajectories shown in the previous sections utilize an RWD topology. In the first subsection, the change in topology will be combined with varying vehicle and road parameters. This subsection will also exploit the effect of changing the vehicle topology, FWD or RWD, on the optimal vehicle sideslip angle β. The last subsection will vary the gravel tire curves to see what effect a lower/higher friction coefficient has on the vehicle sideslip angle β of the found optimal trajectory. The original vehicle and tire parameters used in the previous sections are summarized in Appendix B. The primary purpose of this section is to increase understanding of the relationship between the input parameters of the OP and the optimal solution found by the algorithm. Furthermore, to improve the understanding of this relationship, the influence of parameter variations on the optimal solution is analyzed.

5.3.1 Varying vehicle and road parameters

This section aims to provide an overview of the results that followed from varying the vehicle and road parameters in the optimization algorithm. The results will be used to study the effect of varying these parameters on the vehicle sideslip angle β. All optimizations utilize a road geometry with two straight sections of 20 $[m]$, a 90° or 180° corner with a radius of 10 $[m]$, except for the set where the corner radius is varied. In addition, the lane width is 2 $[m]$ for all sets, except for the set where this is the varied parameter. All parameter sets are summarized in Table 5.2, including the tires employed for each set.

The boundary conditions presented in Table 5.1 are also applied for the optimizations shown in this section. The initial velocity V_0 is free. The initial conditions for the other four model states (β_0, r_0, Δy_0, and $\Delta\psi_0$) are fixed to be zero to ensure that the vehicle starts driving straight ahead in the middle of the road at the start of the simulation. The final velocity V_f and lateral offset with respect to the road centerline Δy_f are free as well. The other three remaining vehicle states (β_f, r_f, and $\Delta\psi_f$) are fixed to be zero to ensure that the vehicle is driving straight ahead at the end of the maneuver.

Table 5.2: Sensitivity analysis - varying vehicle parameters

Set	Varied parameter	Tire used	Range	Step size
1	Mass	1	900 - 1700	25 [kg]
2	Weight distribution	1	65/35 - 35/65	1 [%]
3	Mass	2	900 - 1700	25 [kg]
4	Weight distribution	2	65/35 - 35/65	1 [%]
5	Corner radius	2	5 - 30	1 [m]
6	Lane width	2	0.5 - 3	0.1 [m]

In this subsection, only the results of the first, second, and fifth set will be shown. The figures for the three remaining sets are added in Appendix H. Nevertheless, the results of these three sets will be discussed. In the first two sets, the mass, Figure 5.6, and weight distribution, Figure 5.7, are varied to see how it influences the vehicle sideslip angle β, while the dry asphalt tire is employed.

The yaw moment of inertia I_z should be made mass-dependent and thus varied to get a realistic comparison for the results where the mass is varied. The yaw moment of inertia is approximated, assuming that the car is a rectangular block of uniform density. It is defined as follows:

$$I_z = m\frac{L^2 + W^2}{12}, \tag{5.1}$$

where m is the vehicle mass, and L and W are the length and width of the vehicle (rectangular block). Here it is assumed that the length and width of the vehicle are constant. Using this assumption, the equation for the yaw moment of inertia I_z can be rewritten into:

$$I_{z,new} = I_{z,original}\frac{m_{new}}{m_{original}}, \tag{5.2}$$

where the values for $I_{z,original}$ and $m_{original}$ are equal to the yaw moment of inertia and vehicle mass in the vehicle parameter table in Appendix B. The new vehicle mass m_{new} is a parameter varied in the sensitivity analysis, as indicated in Table 5.2.

From Figure 5.6, the conclusion can be drawn that the result of varying the vehicle mass only has a minor effect on the vehicle sideslip angle β. The combination of the road geometry, 90° or 180°, and vehicle topology, RWD or FWD, have a more significant influence on the minimum vehicle sideslip angle β found in the optimal maneuver. Additionally, the mass affects the initial and final velocity, and thus the cost increases for an increasing vehicle mass. The static weight distribution is 61/39 (front to rear). Therefore, a traction advantage is present for the FWD topology over the RWD topology. That also explains the slightly lower maneuver times for optimal solutions using the FWD topology.

From the results with the gravel tire employed, added to the appendix, the following conclusions can be drawn. The vehicle mass does have very little influence on the developed vehicle sideslip angle β. The results are mainly dictated by the combination of the vehicle topology and the type of corner implemented in the road geometry. These findings are somewhat surprising given the fact that the mass affects the results when the dry asphalt tire is employed, as shown in Figure 5.6. The maximum available engine power is most likely not utilized on this relatively short maneuver when the gravel tire is used. Therefore, the vehicle mass probably has very little influence on the vehicle sideslip angle β.

On the contrary, varying the weight distribution affects the vehicle sideslip angle β significantly. This effect is displayed in Figure 5.7. Relatively low slip values are expected for the dry asphalt tire due to the shape of the tire curve. More information regarding this can be found in Section 3.2.3. Therefore, a variation of a couple of degrees for the vehicle sideslip angle β is called a significant effect. Especially for the optimization using an RWD topology and 180° corner in the road geometry, the weight distribution plays a significant role in the vehicle sideslip angle β that is found in the optimal trajectory. The difference can be found in the location where this vehicle sideslip angle β emerges in the maneuver. In the situation where most of the weight is on the front wheel, trail-braking is applied, resulting in the development of a vehicle sideslip angle β early in the corner. When most of the weight is on the rear wheel, the vehicle starts to oversteer mid-corner, translating to the relatively high vehicle sideslip angle β.

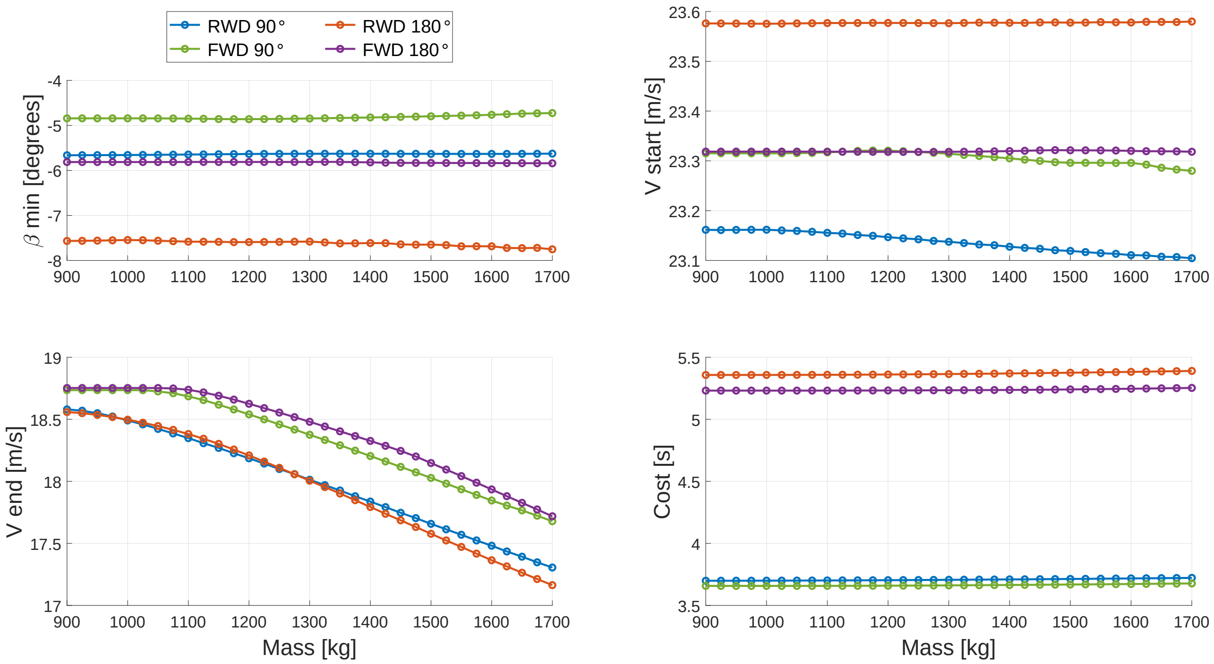


Figure 5.6: Sensitivity Analysis: Tire 1, dry asphalt – Mass [kg] (Set 1).

The effect of the weight distribution on the maneuver time applies to this set as well. For the FWD topologies, the maneuver time is increasing when more weight is placed at the rear wheel. The opposite is true for RWD topologies. More weight at the back results in a traction advantage and thus a decrease in the maneuver time. Conclusions formulated for the influence of the weight distribution on the optimal vehicle sideslip angle β are applicable as well for the optimization with the gravel tire employed. Thus, it is found that the optimal vehicle sideslip angle β is determined by the combination of the vehicle topology, road geometry, and the weight distribution.

The two optimizations using set 5 and 6 of the parameters, varying the corner radius and lane width, are not conducted for the dry asphalt tire in this research. Therefore no comparison can be made between the results. However, for both optimizations, the results are somewhat trivial because, with an increasing corner radius or lane width, the initial and final velocity for the optimal solution are increasing as well. What is somewhat surprising is the increasing cost for an increasing corner radius, but due to the increased corner radius, the total traveled distance increases. This effect is displayed in Figure 5.8. Thus, a longer distance is the reason for the increasing cost for an increasing corner radius.

For both parameters, a similar change in behavior is found. An increasing corner radius or lane width results in an increase of the found optimal vehicle sideslip angle β. Again, this trend is not influenced by the vehicle's topology or the corner implemented in the road geometry.

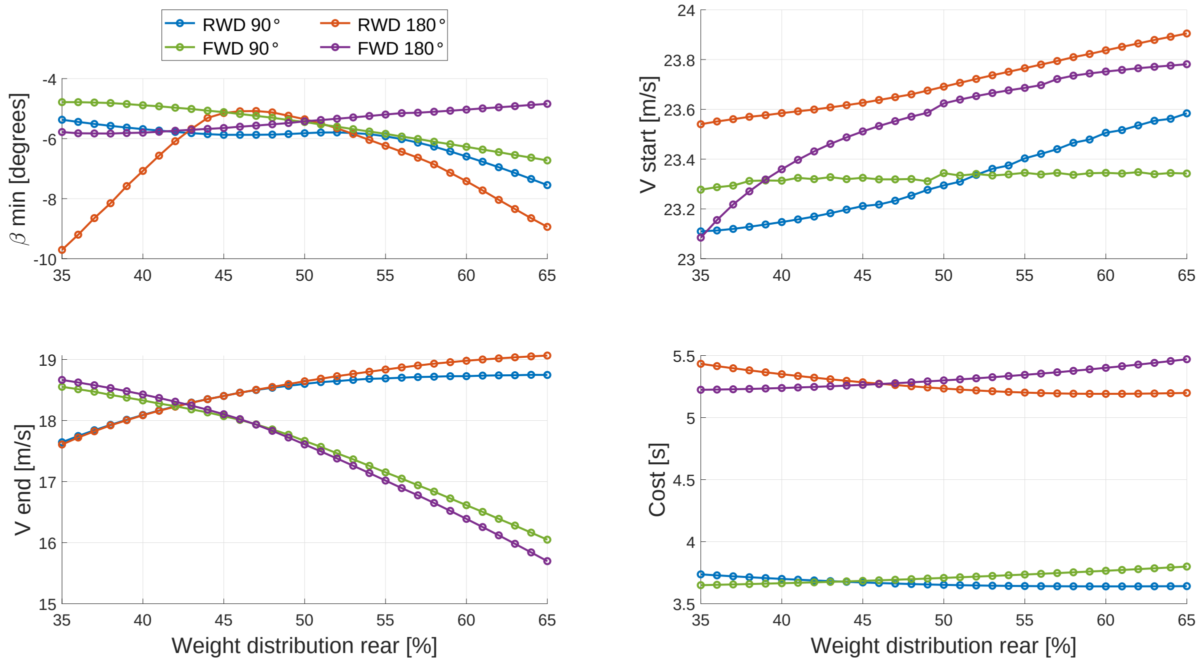


Figure 5.7: Sensitivity Analysis: Tire 1, dry asphalt – Weight distribution rear [%] (Set 2).

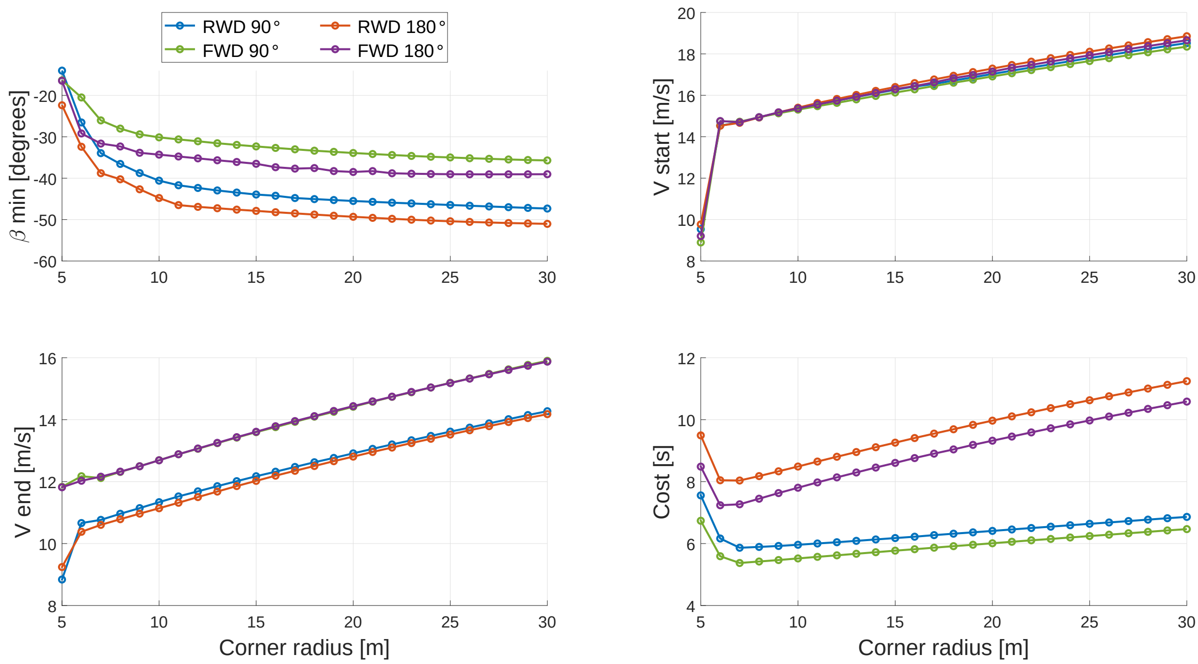


Figure 5.8: Sensitivity Analysis: Tire 2, gravel – Corner radius [m] (Set 5).

5.3.2 Varying tire curves

In this subsection, the tire curve of the gravel tire will be varied by varying the friction coefficients in the longitudinal ($\mu_{x,f}$ and $\mu_{x,r}$) and lateral ($\mu_{y,f}$ and $\mu_{y,r}$) direction. All other coefficients required by the combined slip Magic Formula model are kept original and can be found in Table B.2 in Appendix B. Four new tire curves are created, two with lower friction coefficients and two with higher friction coefficients. The curves with the corresponding friction coefficients are displayed in Figure 5.9. Here the magenta line represents the tire curve with the original coefficients and is thus equal to the gravel curve presented in Figure 3.4.

The first optimization performed is utilizing the five tire curves and has the MTM objective function implemented. Both the initial and final velocities are left free to be determined by the optimization algorithm. From the results displayed in Figure 5.10, it can be concluded that the model states and control inputs are very similar. The tire curve with the highest friction coefficient has the lowest optimal maneuver time because a higher velocity is achieved over the entire trajectory. All five optimal trajectories are plotted over each other in the road geometry plot. Only the trajectory for the black tire curve is visible because the optimal trajectories for all five tire curves are almost identical. A difference is visible in the yaw rate r plot since the trajectories with higher velocities are rotating faster. If this is not the case, then the driven trajectories would not have been identical. In Appendix H, the result for the same optimization with the MEVM objective function is added. The trajectories are again almost identical but differ significantly from the results with the MTM objective function. The effect of the objective function on the optimal trajectory has already been covered in Section 5.2.

A similar optimization is again performed, but this time the initial velocity is fixed to see how this influences the optimal trajectories. The optimal trajectories for this optimization, shown in Figure 5.11, are very similar to the optimization where the initial velocity is left free. Except for the first couple of meters of the trajectories, because the vehicle is accelerating till the velocity matches the velocities of the optimal trajectories shown in Figure 5.10.

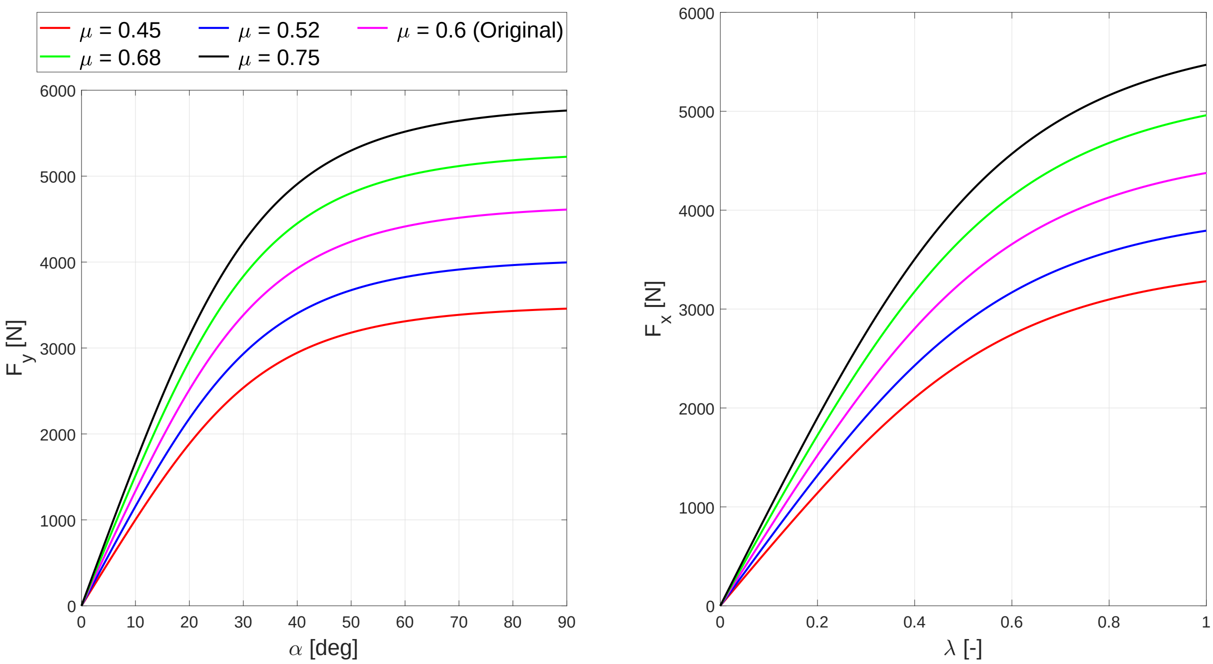


Figure 5.9: Sensitivity analysis - varying tire curves: Lateral tire force F_y and longitudinal tire force F_x characteristics as function of the lateral slip angle α and longitudinal slip λ

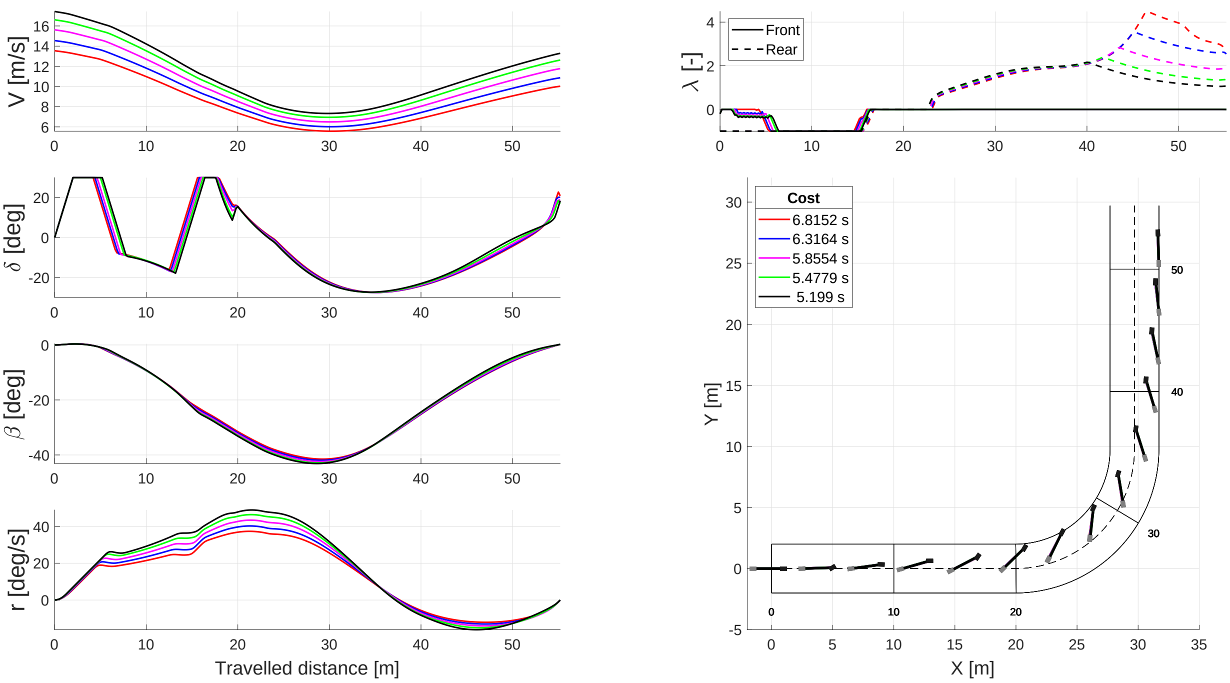


Figure 5.10: Trajectories for the five tire curves shown in Figure 5.9, simulated with a MTM objective function, and a free initial and final velocity.

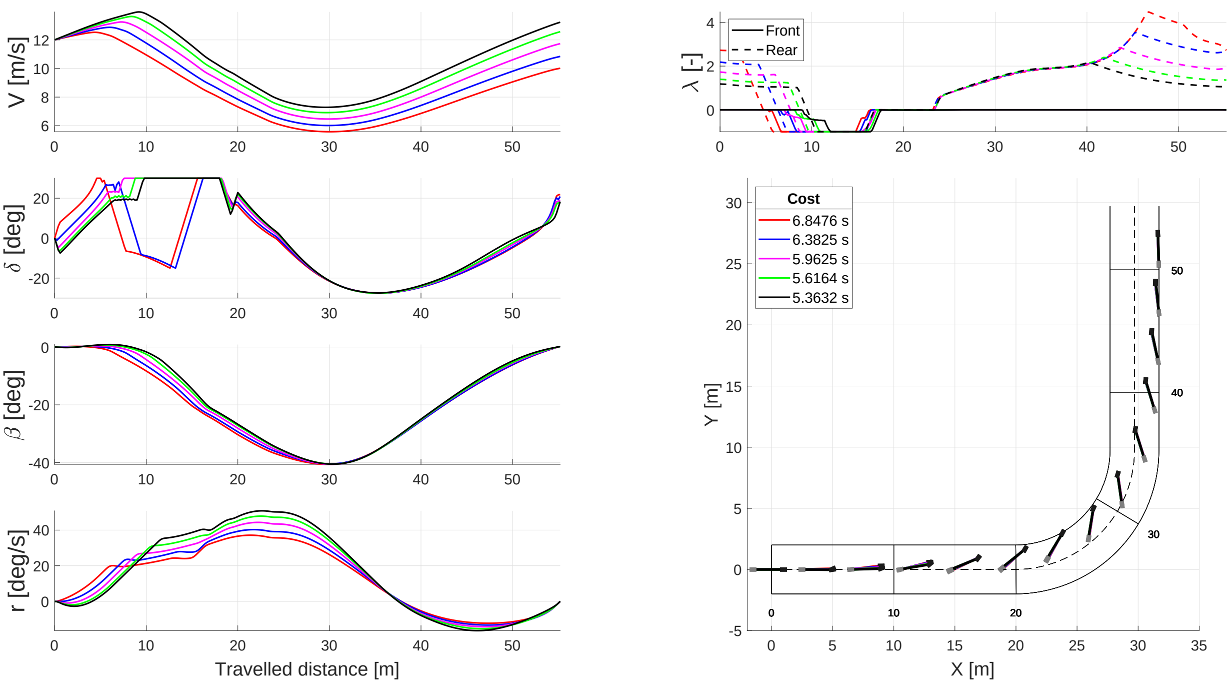


Figure 5.11: Trajectories for the five tire curves shown in Figure 5.9, simulated with a MTM objective function, and a fixed initial velocity and free final velocity.

5.4 Conclusion

This chapter validated the developed optimization algorithm using a trivial example problem of driving on a straight road section with the MTM and MEVM objective function. As expected, no difference is found in the optimal trajectories for both the objective functions. Therefore, the same objective functions have been used to see how it influences the optimal trajectory for a road section with a 90° or 180° corner, with and without obstacles in the road geometry. It is found that objective function affects the corner approach of the vehicle. In situations where the exit velocity is of importance, the corner is negotiated using a late apex. The corner is negotiated using a mid apex for maneuvers where a higher cornering velocity is crucial. This is similar to how professional drivers approach both situations. In addition, the maneuvers employing the dry asphalt tire are characterized by relatively low vehicle sideslip angles β. Where the gravel tire is employed, the maneuvers are characterized by relatively large vehicle sideslip angles β.

This chapter also discussed the influence of changing the vehicle topology, the road geometry, vehicle parameters, and tire curves on the optimal vehicle sideslip angle β. The vehicle parameters varied in the sensitivity analysis: mass, weight distribution, corner radius, and lane width. For varying the tire curves, the original gravel curve, shown in Figure 3.4 has been used as a basis to construct four new tire curves. It is proven that predicting the optimal vehicle sideslip angle β upfront will be complex because the value depends on many variables, such as the vehicle's topology, the employed tire, the road geometry, and the model parameters of the vehicle. However, using the optimization algorithm developed in this research, it is possible to predict the optimal trajectory for the vehicle sideslip angle β for a given set of variables.

By varying vehicle parameters in the sensitivity analysis conducted in this research, an estimate can be developed of how sensitive the found optimal trajectory is to changes in one of the varied vehicle parameters. From the vehicle parameters varied in this chapter, it can be concluded that the vehicle mass has a minor effect on the value of the vehicle sideslip angle β. Thus, this value is mainly dictated by the vehicle's topology, the road geometry, the tire employed, and the other three parameters varied in the sensitivity analysis. This statement also supports the claim that it is difficult to accurately predict the optimal vehicle sideslip angle β because of the many influencing variables.

Changing the tire curves by varying the friction coefficients in the longitudinal and lateral direction affects some of the model states and control inputs, but it hardly impacts the found optimal trajectory. It shows that the optimization algorithm developed in this research is reasonably robust against changes in the friction coefficients of the road surface. From a practical point of view, this is also the expected result because more friction results in faster maneuver times. I.e., in Formula 1, a softer tire mounted to the car results in faster lap times, but it does hardly change the ideal racing line. Thus, the shape of the tire curve, combined with the vehicle topology, road geometry, and vehicle design parameters such as the weight distribution, mainly dictate the optimal trajectory.

Chapter 6

Simulation results

In the previous chapter, it has been proved that the optimization algorithm developed for this research is capable of generating race maneuvers such as trail-braking using the vehicle's maximum capabilities. Depending on the choice of the objective function, other racing techniques also automatically emerge as a solution, e.g., a Scandinavian flick. The optimal solution will corner with a mid-apex for an MTM or with a late-apex for a MEVM. This chapter will discuss the utilization of the algorithm for road geometries with obstacles included.

Section 6.1 covers obstacles in the road geometry. In Section 6.2, the model constraints are adapted for collision avoidance. Three models are created: last-point-to-brake, last-point-to-steer where the vehicle is initially driving straight ahead, and last-point-to-steer where the vehicle is initially steady-state cornering. Section 6.3 will benchmark the results of the optimization models with the kinematic last-point-to-brake and last-point-to-steer models. From the results of the kinematic models, it is assumed that either braking or steering is the preferred control input for an evasive maneuver. Braking is preferred for low velocities and steering for higher velocities because the kinematic models are inadequate to utilize combined inputs. However, a combination of braking and steering might prove to be a better solution for collision avoidance. Therefore, Section 6.4 will combine these control inputs to validate this assumption. Finally, in Section 6.5, a conclusion is formulated for the results found in this chapter.

6.1 Obstacles within the road boundaries

In this section, the optimization algorithm is extended to include obstacles in the road geometry to see how this influences the optimal solution. The optimal solutions will be provided in a similar way as in Section 5.2, but this time with obstacles included in the road geometry. Previously, it has been allowed to position the vehicle at any point in the road geometry, provided that the model states still comply with the constraints and bounds enforced on the trajectory.

The obstacle is implemented approximately halfway through a 90° corner for the optimizations conducted in this section. Since the optimization utilizes a bicycle model, the object within the road boundaries increases in width with half the vehicle track width (0.75 $[m]$). Additionally, a safety distance of 0.25 $[m]$ is added to the three sides that do not coincide with the road boundary. Together both margins form the obstacle around the object shown in Figure 6.3. The dry asphalt and gravel tires are employed in combination with the same objective functions as discussed before. The optimal trajectories are displayed in Figure 6.1 and Figure 6.2, for the dry asphalt and gravel tire respectively. Figure 6.3 shows a zoomed section of the previously mentioned figures. It is clearly visible that the vehicle avoids the obstacle and object. Although the wheel crosses the obstacle, the reason for this has already been explained in Section 4.2.4. For simplicity, only the midpoint of the front and rear wheel is constrained to prevent it from exceeding the road boundaries and obstacles.

Conclusions that have been drawn in the previous section are again applicable here. Both maneuvers employing the dry asphalt tire are characterized by relatively low vehicle sideslip angles β and steering angles δ. With the gravel tire employed, the maneuvers are characterized by relatively large vehicle sideslip angles β and counter-steering when the vehicle is drifting. When the results are compared with the optimal trajectories for a 90° corner without the obstacle, see Appendix H, it becomes apparent that the mid-corner velocity is decreased. Due to the implemented obstacle, the driving radius is significantly smaller, and therefore the mid-corner velocity is lower. This smaller driving radius results from the apex being positioned at the road centerline instead of the left border of the road.

A G-G diagram is created for the optimized trajectories with an obstacle included in the road geometry. The longitudinal and lateral accelerations for the dry asphalt tire are depicted in Figure 6.4a, and for the gravel tire in Figure 6.4b. As it can be concluded that the optimized trajectory with the MTM objective function and the dry asphalt tire employed is operating close to the limits during the entire maneuver. The MEVM maneuver sacrifices the initial velocity to be able to reach the highest possible end velocity. Similar to the previous chapter, the same conclusion is drawn for the G-G diagram where the gravel tire is employed. A trade-off is likely balanced between developing extremely large slips to reach the maximum tire force in either direction and the cost following from the objective function.

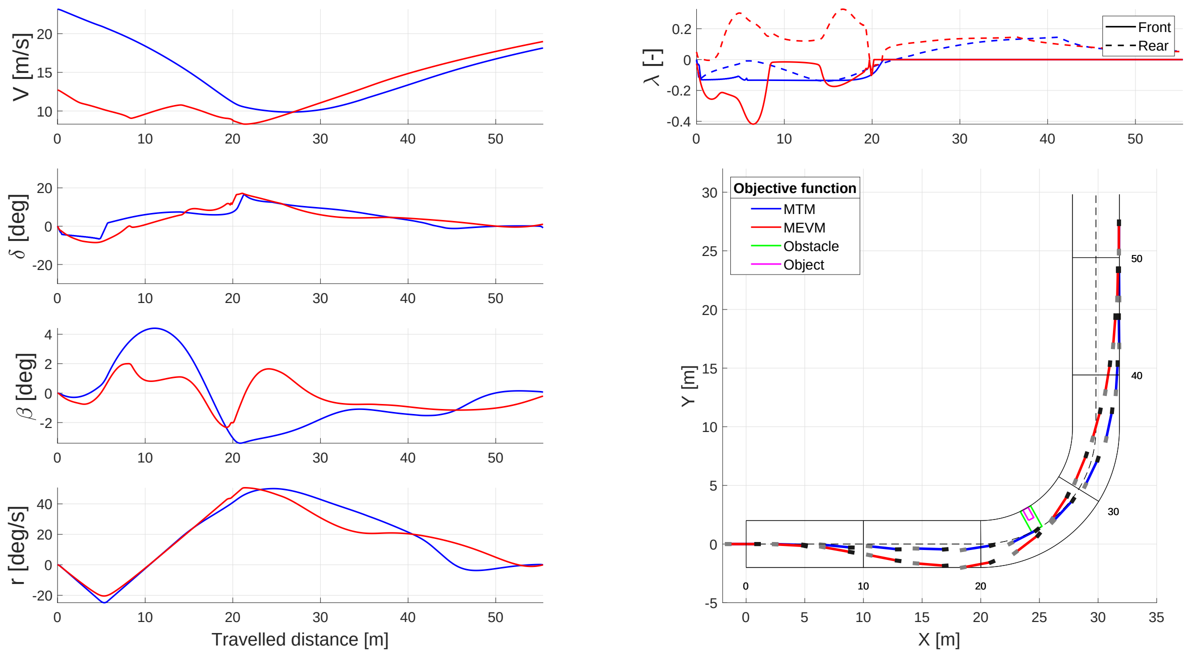


Figure 6.1: Simulation of a 90° corner and an obstacle on the left side of the road, with tire 1 (dry asphalt) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

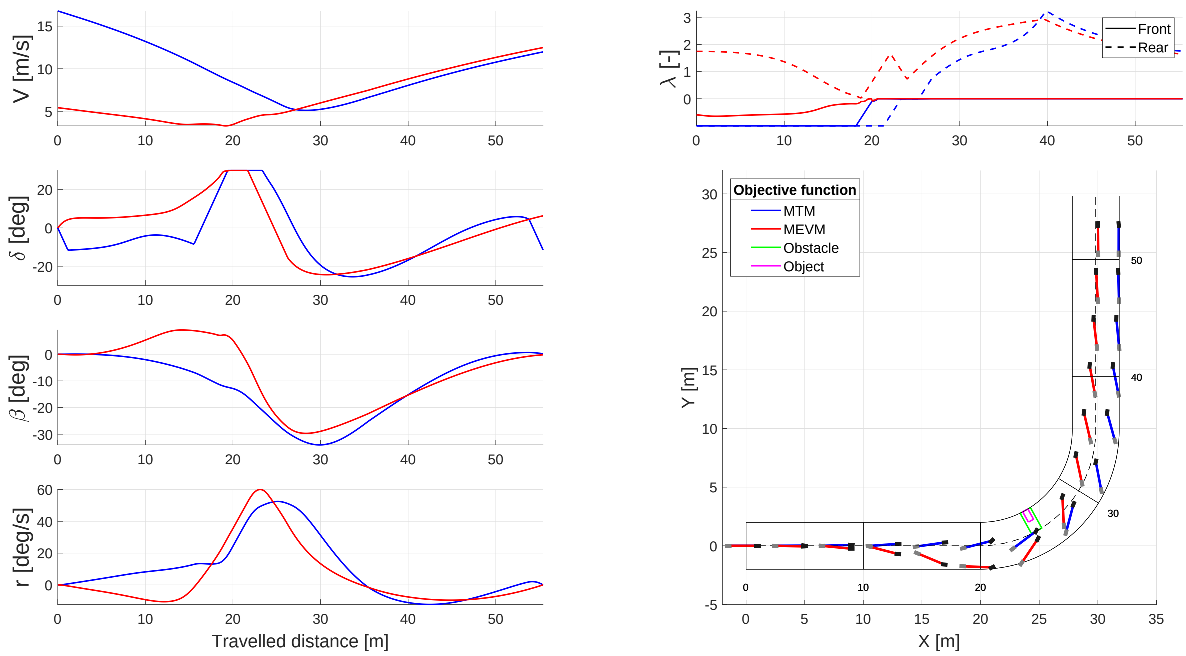
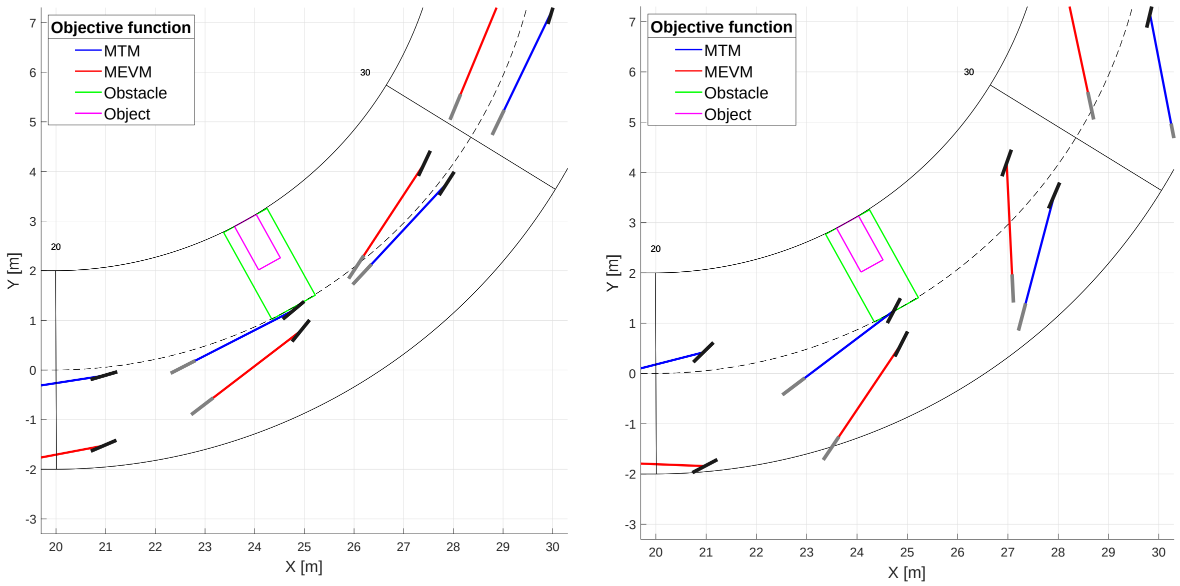


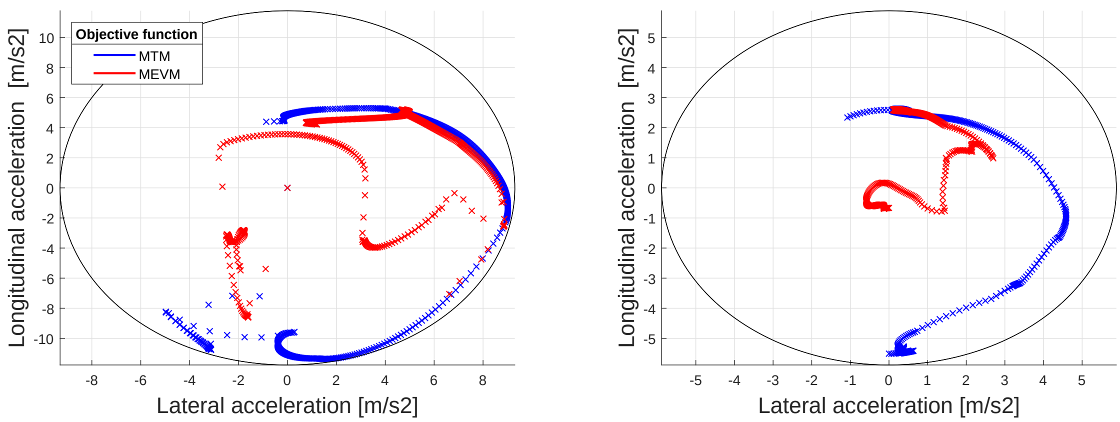
Figure 6.2: Simulation of a 90° corner and an obstacle on the left side of the road, with tire 2 (gravel) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).



(a) Tire 1: dry asphalt

(b) Tire 2: gravel

Figure 6.3: Simulation of a 90° corner and an obstacle on the left side of the road: zoomed



(a) Tire 1: dry asphalt

(b) Tire 2: gravel

Figure 6.4: G-G diagram for the simulations of a 90° corner and an obstacle on the left side of the road geometry.

In Appendix I the results of the optimization with the obstacle positioned on the right side of the road in combination with the gravel tire are presented. A considerable difference is found for the minimum vehicle sideslip angle β for both objective conditions when comparing the two road geometry scenarios. The optimization results using the dry asphalt tire are not added because the optimal solution is equal to the results of optimization without the obstacle. Both maneuvers already avoid the obstacle when it is implemented mid-corner on the right side of the road, see Figure H.1.

6.2 Model constraint adaptation

In this section, optimization of trajectories will be used to benchmark the results found using a nonlinear single-track vehicle model and a nonlinear combined slip tire model with the results of the last-point-to-brake and last-point-to-steer kinematic models. The kinematic equations for both models are defined in Appendix A.

6.2.1 Last-point-to-brake

The last-point-to-brake optimization problem utilizes a slightly different objective function than used before in this research. Previously, the objective has been to minimize the maneuver time or maximize the exit velocity of the maneuver. Both objective functions are of less importance for evasive maneuvers. In collision avoidance, the largest velocity for which the vehicle can stop within the given critical distance is of interest. In order to find the largest velocity, maximizing the initial velocity of the maneuver is implemented as the objective function. For the Last-point-to-brake optimization, the following assumption is made:

- The tire and brake system dynamics are neglected because the wheel slips are used as inputs. I.e., the vehicle is allowed to immediately apply a longitudinal slip for the front and rear wheel, which instantly results in a longitudinal tire force (brake force).

This results in optimizing the largest initial velocity for a given critical distance in which the vehicle needs to stop. Since the kinematic model assumes a constant longitudinal acceleration, the decision has been made to neglect the response time and tire relaxation length. Therefore, these items are not built into the optimization algorithm. However, the two neglected items in this model can be added later, resulting in a longer critical distance for the same initial velocity.

In Figure 6.5, the optimal solution is shown for a critical distance of 30 $[m]$. The initial velocity V_0 is left free for the optimization since this is the cost to be maximized. The final velocity V_f is set to zero to ensure that the vehicle does not hit the object. To be completely clear, the CG of the vehicle has to travel the critical distance in the longitudinal direction. This CG of the vehicle is positioned at [0,0] at the start of the simulation. The results correspond to a vehicle braking using ABS, i.e., operating at the longitudinal slip value that results in the largest braking force. Therefore, the vehicle is braking with a constant longitudinal acceleration, which is also assumed in the kinematic model. From this figure, it is apparent that the results found by the optimization algorithm are very similar to the results of the last-point-to-brake kinematic model. For the interested reader, the result for a critical distance of 15 $[m]$ is added in Appendix I.

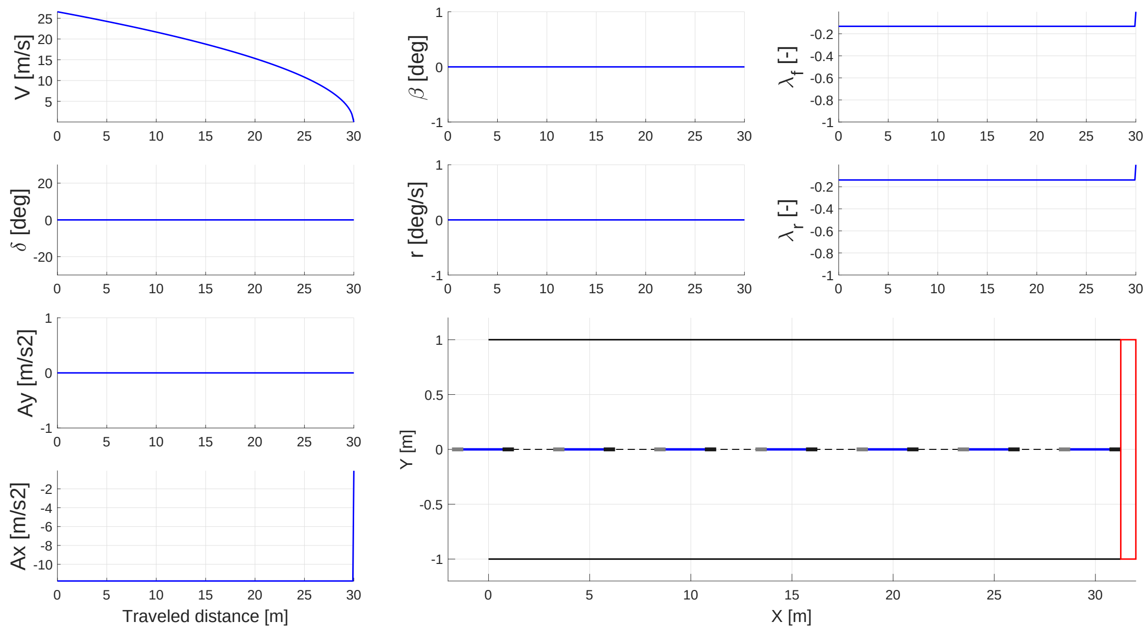


Figure 6.5: Last-point-to-brake optimization: critical distance 30 $[m]$.

6.2.2 Last-point-to-steer: initially straight ahead driving

The last-point-to-steer optimization problem utilizes the same objective function as the last-point-to-brake model. In addition, the initial velocity is implemented as the objective function to find the largest velocity for which the vehicle can avoid the obstacle within the given critical distance while steering. For the last-point-to-steer optimization, where the vehicle is initially driving straight ahead, the following assumptions are made:

- Only the CG of the vehicle has to cover a specified lateral distance (in this case, 2.5 $[m]$) to keep it comparable to the kinematic model.
- The steering dynamics are neglected. I.e., the vehicle is allowed to immediately apply a steering angle at the start of the simulation (for instant lateral tire forces and lateral acceleration). However, the vehicle sideslip angle (β) and yaw rate (r) are equal to zero because the vehicle is driving straight ahead before the obstacle is detected. Therefore, from that moment, the vehicle can generate a vehicle sideslip angle (β) and yaw angle ($\Delta\psi$). This scenario probably represents the most realistic situation for collision avoidance since it is similar to what would happen in an actual driving situation.

The optimal solution for a critical distance of 30 $[m]$ is displayed in Figure 6.6. For the optimization, the initial velocity V_0 and the final velocity V_f are left free. To ensure that the CG covers the specified lateral distance, the initial condition for the lateral position of the vehicle with respect to the road centerline $\Delta y_0 = 2.5$ $[m]$ and the final condition for this model state is $\Delta y_f = 0$ $[m]$.

A negative steering angle δ is immediately applied in the generated trajectory to steer away from the obstacle. This also initiates the drifting maneuver, which is required to achieve the lateral accelerations shown in Figure 6.6. For this maneuver, a G-G diagram is created to show that the vehicle is partly utilizing its maximum capabilities see Figure 6.9a. During the drift initiation, the vehicle only utilizes a part of the vehicle's capability in the lateral direction. In Section 6.4, it will be shown that the drifting maneuver is not initiated when the vehicle is not utilizing the maximum capabilities in the lateral and longitudinal direction. Thus, the lateral accelerations are significantly lower as well.

The magenta line in Figure 6.6 represents the location of the CG of the vehicle. A red rectangle is used to indicate the obstacle in the road geometry. From this figure, it can be concluded that the found optimal solution meets the assumption that only the CG has to cover a specified lateral distance. For the interested reader, the result for a critical distance of 15 $[m]$ is added in Appendix I.

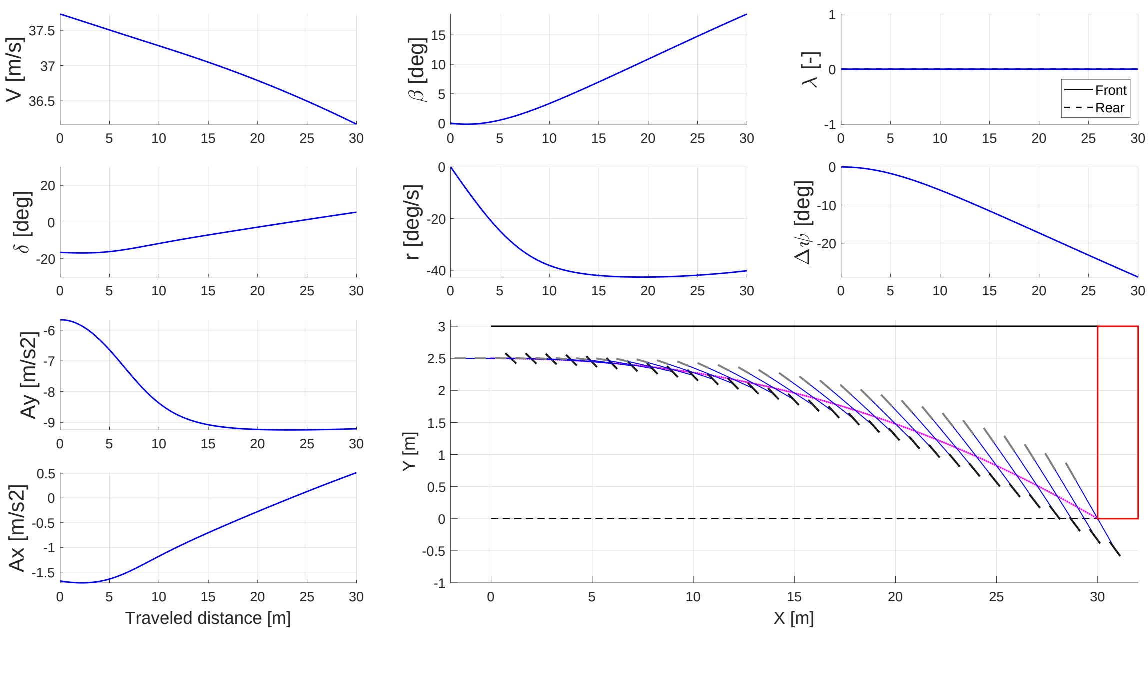


Figure 6.6: Last-point-to-steer optimization straight road: only steering – critical distance 30 $[m]$.

6.2.3 Last-point-to-steer: initially steady-state cornering

In the previous subsection, it has been shown that the lateral acceleration needs some time to build up to steady-state maximum lateral acceleration due to the yaw dynamics of the vehicle. To confirm this theory, the initial conditions of a vehicle during steady-state cornering will be applied in this section. I.e., driving on a circle with a fixed radius R_0 and constant steering angle δ. However, only the initial conditions of the vehicle are similar to a vehicle that is driving on a circle because the road geometry is still straight. The kinematic last-point-to-steer model considers a constant lateral acceleration. Therefore, the initial conditions in the new problem are determined using the kinematic model and the equations to determine steady-state cornering drifting equilibria from Section 2.1. To generate a scenario close to the kinematic model, i.e., where the vehicle starts the maneuver while utilizing, or close to utilizing, the maximum lateral acceleration.

The new last-point-to-steer optimization problem implements the same objective function as discussed before, i.e., maximizing the initial velocity. For the last-point-to-steer optimization model with steady-state cornering initial conditions, the following assumption is made: only the CG of the vehicle has to cover a specified lateral distance of 2.5 $[m]$ (from point A to point B) to keep it comparable to the kinematic model. This is visualized in Figure 6.7. Since the initial conditions of a vehicle driving on a circle are considered, it is not reasonable to assume that the initial vehicle sideslip angle (β_0), yaw angle ($\Delta\psi_0$), and yaw rate (r_0) are equal to zero.

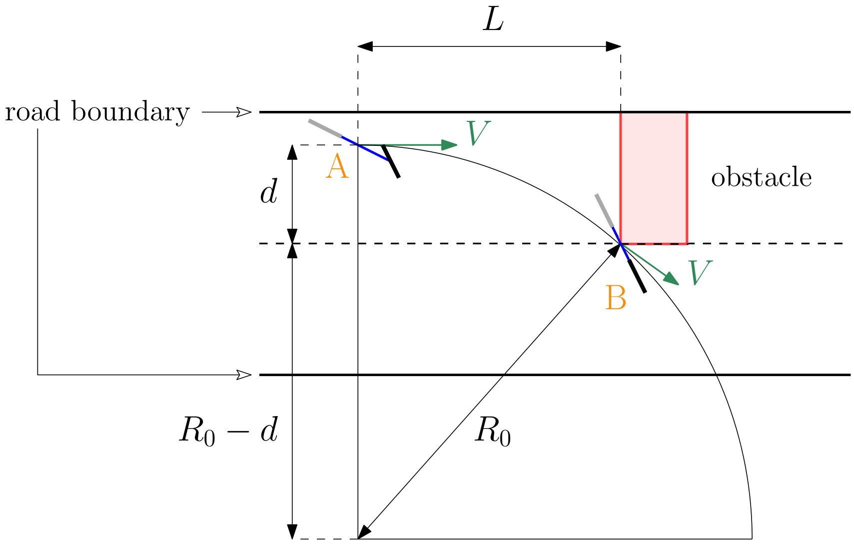


Figure 6.7: Road geometry for a circular road section.

In order to determine the initial yaw rate r_0, the constant corner radius R_0 is required. The equations for the last-point-to-steer kinematic model are based on the longitudinal distance L between the obstacle and the vehicle and the lateral distance d the vehicle has to travel to avoid the obstacle. The values for L and d are known, and thus the corner radius can easily be computed using trigonometry:

$$(R_0 - d)^2 + L^2 = R_0^2. \tag{6.1}$$

Rearranging (6.1) results in the equation for the corner radius R_0:

$$R_0 = \frac{d^2 + L^2}{2d} \tag{6.2}$$

The initial guess for the velocity V_0 is determined using a rewritten equation for the kinematic last-point-to-steer model. For the kinematic model, the velocity is used as an input to determine the critical distance $L_{steering}$, see Appendix A. In the optimization problem, the equation is used the other way around, i.e., a known critical distance L is used to determine the initial guess for the velocity. The lateral acceleration a_y required to determine V_0 follows from the lateral friction coefficient μ_y of the employed tire. Thus, with the corner radius R_0 and initial velocity V_0 known, the initial yaw rate r_0 can be determined:

$$r_0 = \frac{V_0}{R_0}. \tag{6.3}$$

Next, the initial steering angle δ_0 is required to find the initial vehicle sideslip angle β_0. In order to find the initial steering angle, one more vehicle parameter needs to be known that is not used yet, the understeer coefficient η. In previous scenarios, the vehicle always started while driving straight ahead. Therefore, the value of the understeer coefficient η has not been required to be known before. Now, it is required to find the value of the initial steering angle δ_0. The value of this parameter can also be used to determine whether the vehicle has oversteer, understeer, or neutral steer. The understeer coefficient η is defined as follows:

$$\eta = \frac{F_{z,f}}{C_{F\alpha,f}} - \frac{F_{z,r}}{C_{F\alpha,r}} \quad \rightarrow \quad \eta = \frac{1}{B_{y,f} \cdot C_{y,f} \cdot \mu_{y,f}} - \frac{1}{B_{y,r} \cdot C_{y,r} \cdot \mu_{y,r}}, \tag{6.4}$$

where $F_{z,f}$ and $F_{z,r}$ are the static tire loads, and $C_{F\alpha}$ the cornering stiffness of the tire. This cornering stiffness can be computed using the following equation $C_{F\alpha} = B_y \cdot C_y \cdot \mu_y \cdot F_z$. The values for the Magic Formula coefficients in this equation, for both the front and rear tire, can be found in Appendix B. The initial steering angle δ_0 can now be calculated using:

$$\delta_0 = \frac{a+b}{R_0} + \frac{\eta V^2}{g R_0}. \tag{6.5}$$

The determined initial conditions can now be used in combination with the equations to determine steady-state cornering drifting equilibria (Section 2.1) to find a value for the initial vehicle sideslip angle β_0.

Now that the steady-state cornering initial conditions for the new optimization problem are known, the model can be used to find an optimal trajectory. In Figure 6.8, the results are shown for the optimization of straight road section with a critical distance L of 30 $[m]$. Since this optimization ensures that the vehicle is driving on the limit, the same drifting behavior is found as for the optimization problem discussed in Subsection 6.2.2. The only difference is that the vehicle is already cornering at the beginning of the maneuver. Therefore, a significantly higher initial velocity is found by the optimization algorithm compared to the problem where the vehicle is initially driving straight ahead, see Figure 6.6.

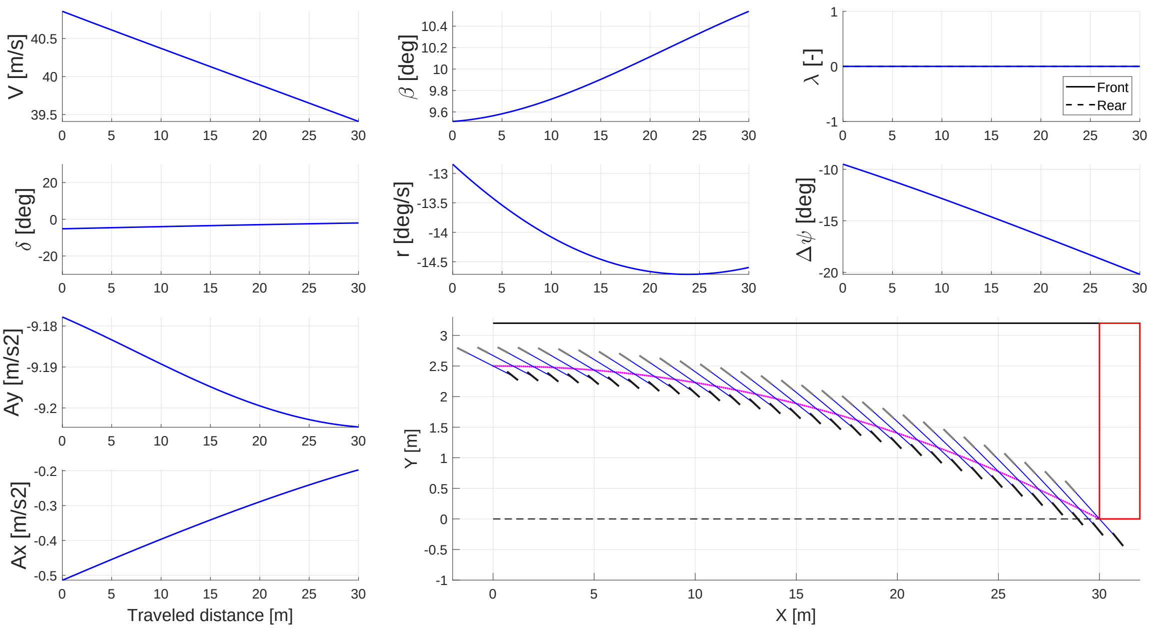
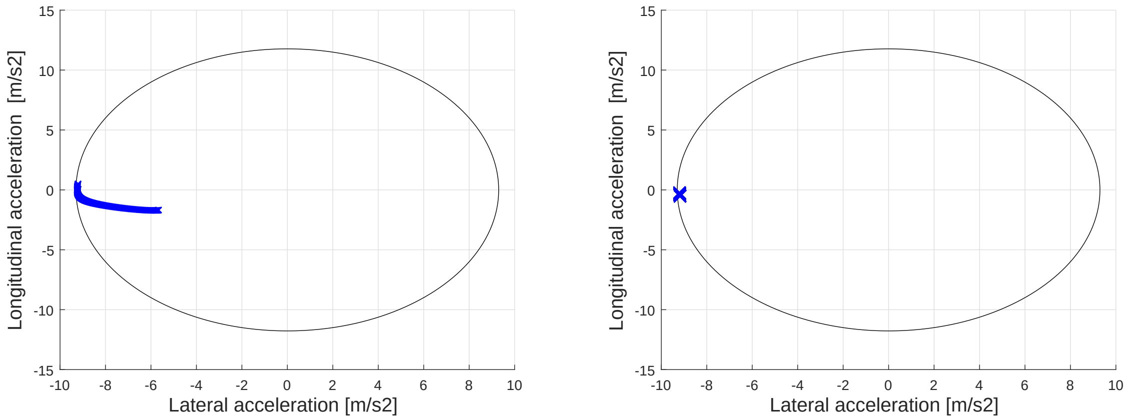


Figure 6.8: Last-point-to-steer: initially steady-state cornering – critical distance 30 [m].

Subsection 6.2.2 already shortly discussed the results displayed in the G-G diagram for an evasive maneuver of a vehicle that is initially driving straight ahead. Figure 6.9 presents the G-G diagrams for both optimization problems discussed in this section. From Figure 6.9b it becomes apparent that the vehicle is utilizing the maximum capability of the vehicle in the lateral direction from the start of the maneuver for the problem where the vehicle is initially steady-state cornering. This statement is not true for the other maneuver because the lateral acceleration starts to build up when the drift is initiated. Since the maneuver with steady-state cornering initial conditions utilizes the vehicle's maximum capability from the start, it can cover the same lateral (critical) distance with a slightly larger initial velocity V_0. Again, this is under the assumption that the vehicle is cornering/drifting at the start of the maneuver. In the end, the evasive maneuvers resulting from the optimization problems discussed in the previous and this subsection show a vehicle driving beyond the stable limits of handling to avoid an obstacle.

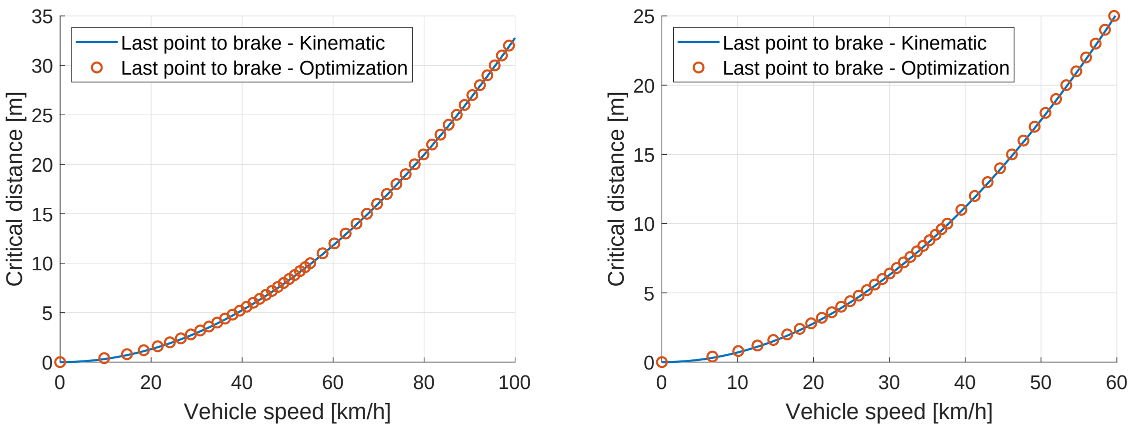


(a) Initial condition: Straight ahead driving (b) Initial condition: Steady-state cornering

Figure 6.9: G-G diagram for the simulations of an evasive maneuver on a straight road segment with different initial conditions (Figure 6.6) and circular road segment (Figure 6.8).

6.3 Comparing the kinematic and optimization based models

The previous section discussed the setups of the optimization problem for both the last-point-to-brake and the last-point-to-steer where the vehicle initially is driving straight ahead or steady-state cornering. In this section, the three models and the kinematic models are used to find the optimal initial velocity for a series of specified critical distances. Thereafter, the optimization results will be benchmarked with the results of the two kinematic models. These optimizations are conducted for two tires, the dry asphalt tire, and the gravel tire.



(a) Tire 1: dry asphalt (b) Tire 2: gravel

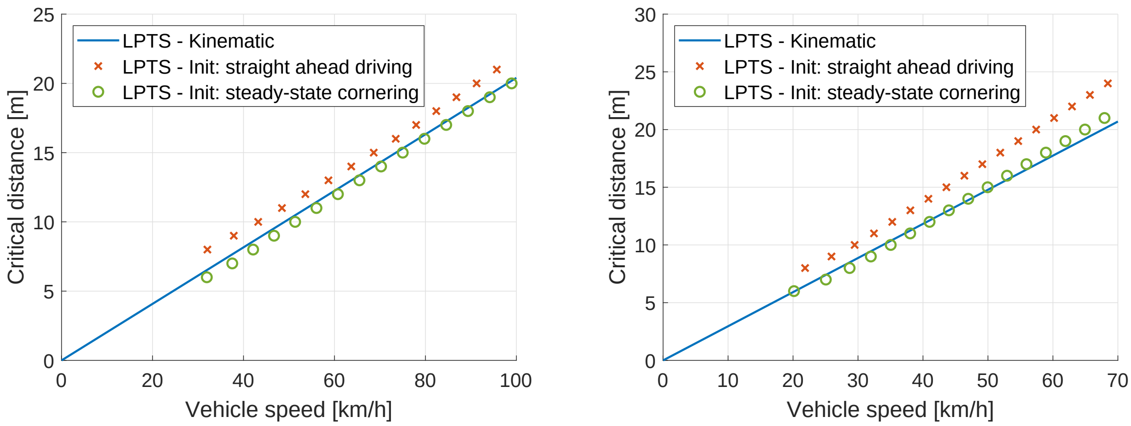
Figure 6.10: Last-point-to-brake: kinematic model vs optimization

From Figure 6.10 it can be concluded that the results of the optimization and kinematic model for last-point-to-brake are very similar. Thus, the relatively simple kinematic model for the last-point-to-brake is thus able to find a critical distance, similar to the critical distance found by the more complex optimization model. This finding suggests that the kinematic last-point-to-brake model is adequate for predicting a stopping distance based on the available friction in the longitudinal direction.

A similar conclusion cannot be drawn for the last-point-to-steer kinematic model because the assumptions made in this model are not entirely comparable to real-world driving situations. For example, in the kinematic model, the vehicle is considered to have constant lateral acceleration, but this is not realistic for collision avoidance, where the vehicle is assumed to be driving straight ahead when the obstacle is detected. As denoted in the assumptions for the straight-ahead driving optimization problem, this means that the vehicle sideslip angle (β_0) and yaw rate (r_0) are equal to zero at the start of the maneuver. Therefore, the decision has been made to plot the results of a vehicle with steady-state cornering initial conditions. For this scenario, the initial vehicle sideslip angle (β_0), yaw angle ($\Delta\psi_0$), and yaw rate (r_0) are not equal to zero and are estimated using the equations for steady-state cornering.

In Figure 6.11, the results of the last-point-to-steer (LPTS) kinematic model and the two optimization problems are shown. From the initially steady-state cornering problem results, it can be concluded that these are fairly similar to the results of the kinematic model. However, as concluded before, this is not exactly true for the initially straight-ahead driving problem.

It can be seen from the data in Figure 6.10 and Figure 6.11 that the kinematic models can be used to get a rough estimation of the critical distance. A future study could look into options to combine the steering and braking optimization models to find the safest maneuver for static and non-static obstacles. In this research, only static obstacles have been considered due to limitations in the method used for generating the road geometry. It is also advised to look for an objective function that can weigh the decision to either brake or steer based on a risk assessment. The next step is to validate the assumption that the preferred control inputs for an evasive maneuver are a combination of braking and steering. Therefore, the subsequent section will combine these two control inputs using the developed optimization algorithm.



(a) Tire 1: dry asphalt

(b) Tire 2: gravel

Figure 6.11: Last-point-to-steer: kinematic model vs optimization

6.4 Combined braking and steering

Throughout this research, it has already been shown that the developed optimization approach is able to deal with different tire-road interaction curves, road configurations (e.g., straight or curve), initial conditions, objective functions, and varying vehicle and road parameters. However, the algorithm neglects the brake, steer, and tire dynamics. The developed approach provides a framework to integrate those dynamics in the optimization problem and, therefore, could provide a more accurate estimate of the best combination of steering and braking for evasive or typical driving situations. In this section, the utilization of combined control inputs will be compared to using only one control input for evasive maneuvering scenarios.

The optimization method results in Section 6.2 and 6.3 only used one control input, i.e., either braking or steering. From the results of the kinematic models, the conclusion emerges that this is the best strategy to avoid an obstacle. However, it is assumed in general that it is better to use a combination of braking and steering for collision avoidance. In this section, that assumption will be validated by comparing the results of using only one input to the results of using simultaneous steering and braking.

Two situations are compared, corresponding to the usage of the control inputs:

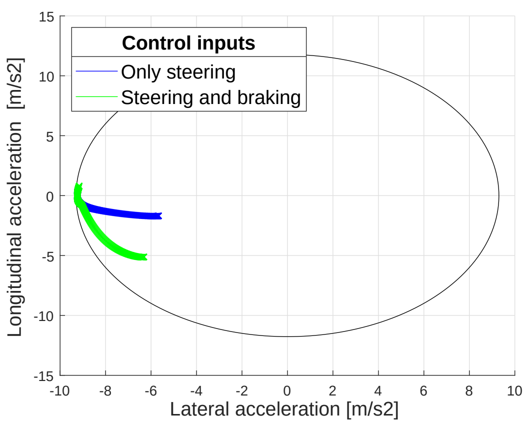
1. Only steering
2. Steering and braking ($\lambda \leq 0$)

Both situations stated above are compared to see how the choice of control inputs influences the outcome of the optimization algorithm.

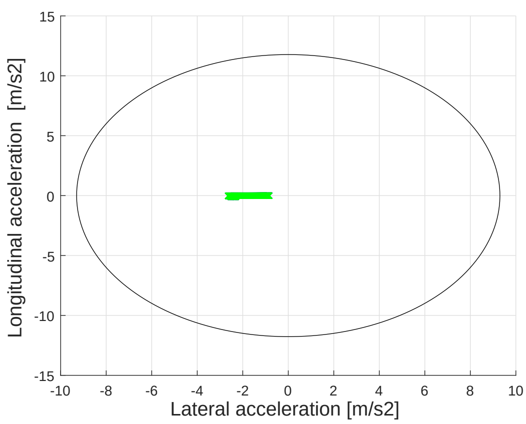
The results of optimizing the two situations are visualized in Figure 6.13. To be clear, the results of the only steering situation are equal to the results presented in Figure 6.6. With steering as the only control input, the initial velocity is noticeably lower than when braking can also be applied. The optimal solution for the combined control inputs is using the brakes to build up the vehicle sideslip angle β and yaw rate r faster. Braking early in the maneuver results in the ability to avoid the obstacle with a higher initial velocity. The G-G diagram for both situations is presented in Figure 6.12. The results visualized in this figure show that the vehicle is operating closer to the vehicle limits when using the two control inputs simultaneously to avoid the obstacle. Therefore, the general assumption that utilizing both control inputs is better for collision avoidance is partly validated.

Nevertheless, for a relatively short critical distance, it might still be better to use the brakes only because of the low vehicle speed. These situations are not tested in this research, and therefore, no conclusions can be drawn about this.

Where the optimal solution found for maximizing the initial velocity of the maneuver tends to develop a drift to avoid the obstacle, this is not the case when the initial velocity is fixed to a significantly lower velocity. This situation is optimized using the MTM objective function because it is useless to maximize a fixed initial velocity. The optimization is conducted with a fixed initial velocity of 20 $[m/s]$, and again a critical distance of 30 $[m]$ is used to be able to compare the outcome with the results in Figure 6.13. Results for the optimization with a fixed initial velocity are presented in Figure 6.14. It is apparent from this figure that the vehicle is not driving on the limit with the fixed initial velocity because the accelerations are significantly smaller than in Figure 6.13. This observation is supported by the results shown in the G-G diagram in Figure 6.12b. This figure shows that the vehicle only uses a part of the vehicle's capability in the lateral direction. Additionally, there is hardly any difference between the results for the two situations tested. In this situation, it might even be possible to apply throttle while still avoiding the obstacle because the vehicle does not utilize its maximum capabilities in either of two directions.



(a) Combined inputs



(b) Combined inputs: limited V_0

Figure 6.12: G-G diagram for the simulations of an evasive maneuver with a combination of the braking and steering control inputs.

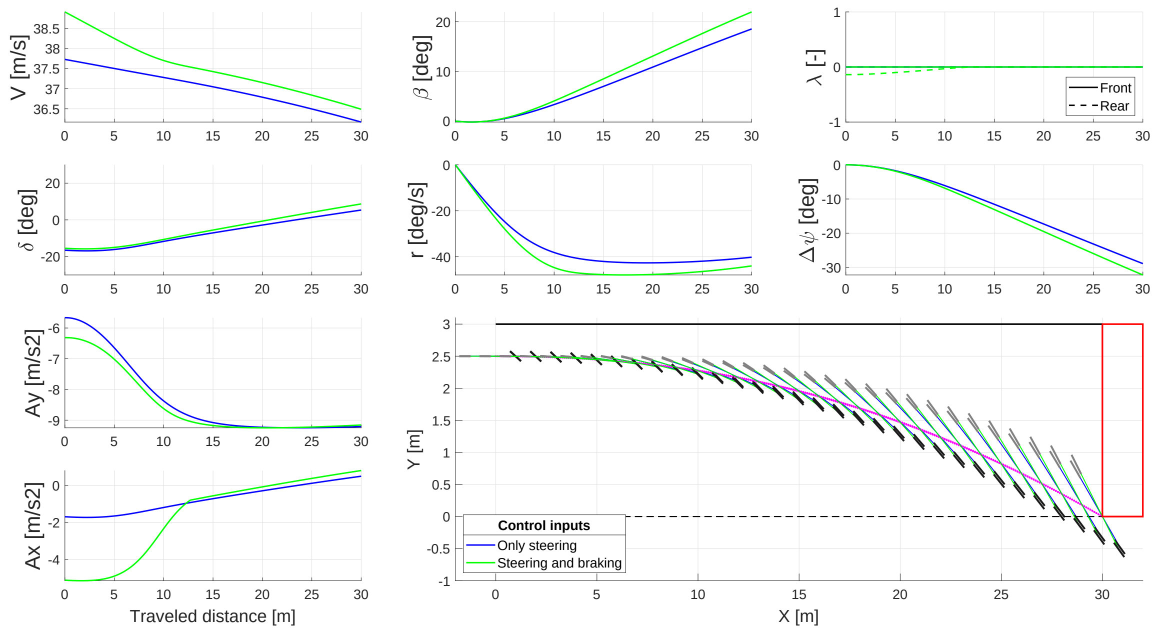


Figure 6.13: Last-point-to-steer optimization straight road: blue (only steering) and green (steering and braking) – critical distance 30 [m].

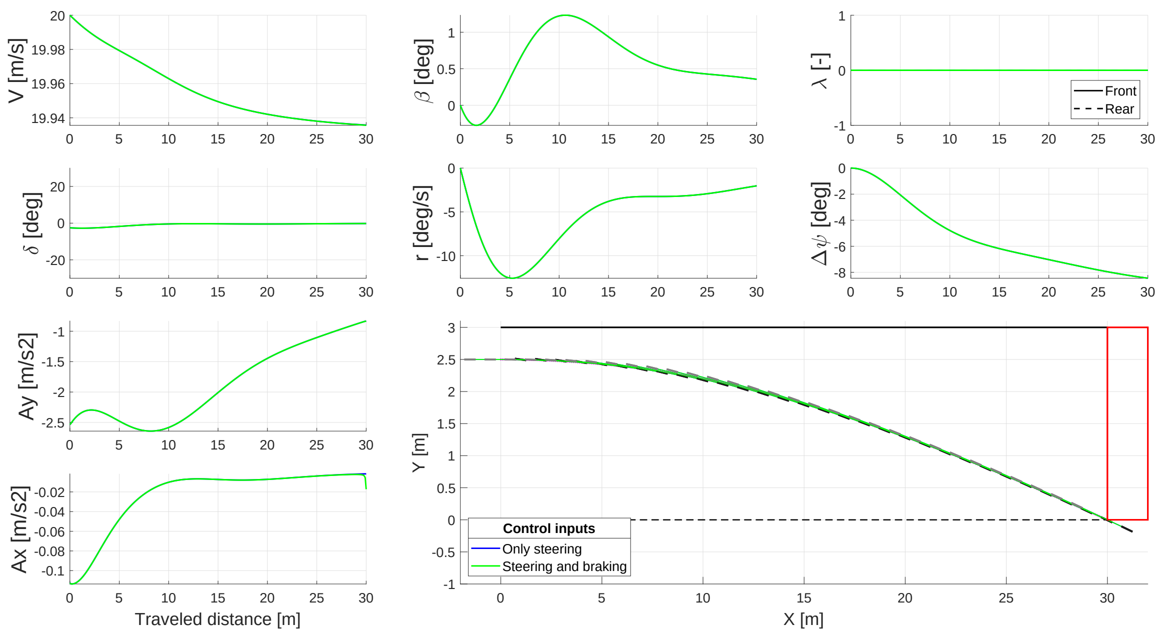
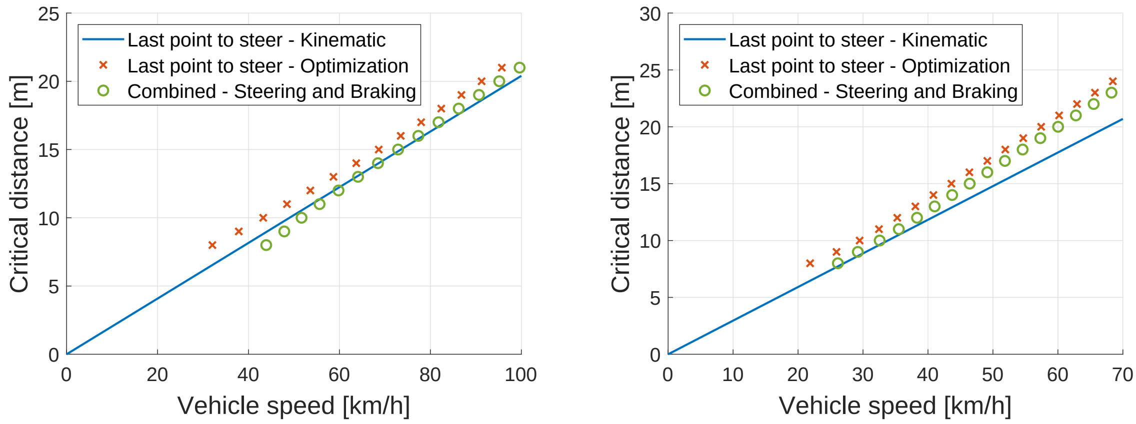


Figure 6.14: Last-point-to-steer optimization straight road: blue (only steering) and green (steering and braking) – limited initial velocity.

Using a combination of braking and steering is compared to the results of the last-point-to-steer kinematic model and optimization (straight road segment) approach for a series of specified critical distances. Both the dry asphalt and gravel tire are employed again, and the results for all three models are presented in Figure 6.15. From the figure, it can be concluded that utilizing the two control inputs simultaneously is at least beneficial for a critical distance > 8 $[m]$. For the tested critical distances, the model using both braking and steering is able to avoid the obstacle with a larger initial velocity, without exceptions. This statement is true independent of the tire employed for the optimization. Although, the results are not entirely comparable to the kinematic last-point-to-steer model as already concluded in Section 6.3. The reason for this is that constant acceleration is considered for the kinematic model. As shown in 6.12a this is not a reasonable assumption for the optimization model where the vehicle is initially driving straight ahead.



(a) Tire 1: dry asphalt

(b) Tire 2: gravel

Figure 6.15: Kinematic models vs optimization

6.5 Conclusion

This chapter discussed the gathered results using the developed optimization algorithm for road geometries with obstacles included. Similar conclusions have been drawn for the optimal trajectories of negotiating a road with obstacles included, as for the road geometries without obstacles. The maneuvers employing the dry asphalt tire are characterized by relatively low vehicle sideslip angles β and steering angles δ. On the other hand, with the gravel tire employed, the maneuvers are characterized by relatively large vehicle sideslip angles β and counter-steering when the vehicle is drifting.

Three separate optimization approaches have been developed for this research: the last-point-to-brake model, the last-point-to-steer model where the vehicle is initially driving straight ahead, and the last-point-to-steer model where the vehicle is initially steady-state cornering. The three optimization approaches are benchmarked with the results of the kinematic last-point-to-brake and last-point-to-steer models. The results of the kinematic model and optimization approach are very similar for the last-point-to-brake simulations. Furthermore, depending on the initial conditions used for the optimization model, driving straight ahead or steady-state cornering, the results are fairly comparable to the last-point-steer kinematic model. The kinematic model assumes a constant lateral acceleration, but this is not a reasonable assumption for a vehicle driving straight ahead initially. Due to the yaw dynamics of the vehicle, it takes some time before the vehicle reaches the steady-state maximum lateral acceleration.

Finally, the influence of using braking and steering simultaneously on the optimal solution for an evasive maneuver has been tested. Without exceptions, using combined control inputs results in larger initial velocities for a specific critical distance to avoid the obstacle. The results presented suggest that a combination of braking and steering is better for evasive maneuvers. At least for the situations tested in this research (critical distance > 8 $[m]$). G-G diagrams showed that the vehicle utilizes more of its maximum capabilities in the lateral and longitudinal direction using simultaneous braking and steering. The kinematic collision avoidance models can be used to get a rough estimate for the critical distance required to avoid an obstacle for a given initial velocity. However, for a more detailed description of the evasive maneuver, the optimization algorithm developed in this research is required.

Chapter 7

Conclusion and recommendations

7.1 Conclusion

This research aimed to generate optimal trajectories for evasive maneuvers based on a vehicle's capabilities in the lateral and longitudinal direction. Racing-inspired objective functions, minimum time or maximum exit velocity, are employed in the optimization problem to utilize the vehicle's maximum capabilities. The system dynamics in the optimization problem are composed of a combined vehicle and road model. A nonlinear single-track vehicle model combined with a nonlinear combined slip tire model has been used for this research. The implemented tire model provides possibilities to describe the differences in tire behavior for driving on different road surfaces, e.g., asphalt or gravel. The road geometry is modeled using a curvilinear coordinate system. In addition, the road geometry data required by the optimization algorithm is generated using functions from the OpenCRG tool. The trapezoidal collocation direct method and the IPOPT NLP solver are used to solve the optimization problem.

Various research projects on the topic of collision avoidance and collision mitigation have been examined. Some projects used discrete optimization for collision avoidance with a finite set of predetermined feasible trajectories. However, none of the projects discussed on collision avoidance focused on utilizing the maximum capability of the vehicle in the lateral and longitudinal direction. In addition, almost all of these researches employed a kinematic vehicle model, which cannot capture the evasive maneuver characteristics. Utilizing these characteristics is thus still a topic open for investigation. Hardly any information is available on the comparison with the kinematic last-point-to-brake and last-point-to-steer model. The kinematic models assume that either braking or steering is the preferred control input for an evasive maneuver because these models are inadequate in utilizing combined inputs. However, it is assumed in general that it is better to use a combination of braking and steering for collision avoidance. Therefore, validating this assumption has also been a topic open for research.

The developed algorithm is first validated for a trivial example, and thereafter the influence of the previously mentioned objective functions on the optimal solution is tested. From these tests, it could be concluded that the outcome follows the approach professional drivers use to negotiate corners. When a long straight section follows the corner, a driver will almost always take a late apex. The optimization algorithm finds a similar approach for the MEVM objective function. For sections with multiple corners chained together, the exit velocity is of less importance. In these situations, maximizing the corner velocity will result in faster lap times. Similar cornering behavior is found for the MTM objective function. These conclusions are supported by G-G diagrams and proved to be true for both tires tested in this research.

One difference in the optimized trajectories that emerges for the different tires used, the dry asphalt and gravel, is the cornering behavior of the vehicle. The maneuvers using the dry asphalt tire are characterized by relatively low vehicle sideslip angles β and steering angles δ, while the optimal maneuvers using the gravel tire are characterized by relatively large vehicle sideslip angles β and counter steering. This behavior is expected due to the shape of the tire-road interaction curves and confirmed by the solution found by the optimization algorithm. The found behavior is independent of the selected objective function. The optimal trajectory for a road geometry with an obstacle implemented mid-corner also showed the same characteristic driving behavior, low vehicle sideslip angles β for the dry asphalt tire, and relatively large vehicle sideslip angles β for the gravel tire.

A sensitivity analysis showed the influence of changing the vehicle topology, the road geometry, and varying the vehicle parameters and tire curves, which can significantly impact the found optimal vehicle sideslip angle β. From the results, it could be concluded that the vehicle mass has a minor effect on the value of the optimal vehicle sideslip angle β. Furthermore, changing the tire curves by varying the friction coefficients in the longitudinal and lateral direction affects the model states and control inputs found by the optimization algorithm, but it hardly impacts the found optimal trajectory. Thus the value of the optimal vehicle sideslip angle β is mainly dictated by the vehicle's topology, the weight distribution between the front and rear tire, the road geometry, and the shape of the tire curve of the employed tire.

After comparing the results of the last-point-to-brake kinematic model with the last-point-to-brake optimization method, it could be concluded that the results are fairly similar, independent of the tire choice. Thus, the relatively simple kinematic model is able to find the same critical distance as the more complex optimization model. This insight suggests that the kinematic last-point-to-brake model is adequate in predicting the stopping distance for collision avoidance situations. However, the optimization approach can consider different aspects of the vehicle topology and road geometry. For instance, if the vehicle brakes while driving on a circular road, a part of the forces used in the lateral direction are then compromised by the longitudinal braking force. These phenomena cannot be addressed using the relatively simple last-point-to-brake kinematic model.

Depending on the assumptions applied for the last-point-to-steer optimization model, the results are comparable to the kinematic last-point-to-steer model. Two different initial conditions have been considered in this research, one where the vehicle is initially driving straight ahead and one where the vehicle is initially steady-state cornering. It is assumed that the initial vehicle sideslip angle β_0 and yaw rate r_0 are equal to zero for model where the vehicle is initially driving straight ahead. On the contrary to this model, the initial vehicle sideslip angle (β_0), yaw angle ($\Delta\psi_0$), and yaw rate (r_0) are estimated using the equations for steady-state cornering for the model where the vehicle is initially steady-state cornering. It could be concluded that the results for this optimization model are fairly similar to the results of the kinematic model by comparing both models. However, this conclusion is not valid for the model where the vehicle is initially driving straight ahead. The kinematic model assumes constant lateral acceleration, and this is not true because the lateral acceleration first has to be build up from the start of the simulation. Building up the lateral acceleration happens by initiating a drift to utilize the vehicle's maximum capabilities. In addition, it is concluded that shorter critical distances could be achieved using both braking and steering simultaneously in comparison to using only steering for a critical distance $>$ 8 $[m]$. This finding party proved the general assumption that a combination of braking and steering is preferred for evasive maneuvers. However, it might still be better to utilize the brakes only below this critical distance of 8 $[m]$, but these situations have not been tested in this research.

To conclude, the main objective of this research has been accomplished. The developed optimal trajectory generator is capable of finding evasive maneuvers based on a vehicle's maximum capabilities in the lateral and longitudinal direction. Thus, it can find the optimal trajectory for an evasive maneuver to avoid a potential collision within and beyond the stable limits of handling.

7.2 Recommendations

The results found in this research provided insight into cornering behavior within and beyond the stable limits of handling on different terrains. This section identifies the areas of interest for future research. The recommendations elaborate on potential improvements that emerged while carrying out this research and yield valuable insights to make future vehicles safer.

1. **Motor and tire dynamics**
   The motor and tire dynamics are neglected in this research, meaning the longitudinal slip, used as a control input in this research, immediately results in a tire force. For future research, it might be interesting to study the effect of motor dynamics and tire relaxation lengths on the optimal solution found in this research. Furthermore, these effects need to be considered for collision avoidance since the critical distance found by the optimization algorithm will increase due to a delay introduced by these effects.

2. **Model fidelity**
   The single-track vehicle model implemented in this research only incorporated longitudinal weight transfer. For a more accurate result, it may be considered to implement lateral load transfer in the single-track vehicle model or use a double-track vehicle model. The load transfer can then be combined with a differential between the wheels, depending on the required results. Aerodynamic forces acting on the vehicle are also neglected in this research. Especially for racing vehicles, the addition of these forces in the vehicle model could improve the accuracy of the results since these vehicles are often equipped with aerodynamic devices to reduce the drag and improve the downforce acting on it.

3. **Variable friction coefficients in the road geometry**
In this research, it is assumed that the tire-road interaction curves remained constant during the entire maneuver. Varying friction coefficients in the road geometry, e.g., a modeled μ-drop, to see how this influences the optimal trajectory for both cornering and collision avoidance could provide valuable insight for handling such a situation in a future level five autonomous vehicle. The μ-drop situation can be compared to a vehicle driving on an asphalt road (high μ) and suddenly encounters a road section covered in, e.g., snow or gravel (low μ). It would be interesting to study how this situation, or the inverse situation (low μ to high μ), influences the vehicle's driving behavior.

4. **Dynamic obstacles**
The implemented obstacles in this research are considered to be static obstacles. However, this simplifies the situations encountered in real-life driving, where follow road users are also moving. Therefore, future research should consider dynamic obstacles to find the safest maneuver for static and non-static obstacles in the road geometry. Additionally, an objective function that can weigh the decision to either brake or steer should be included because both options are optimized separately in this research. For example, a potential high collision risk could be found at first for a dynamic obstacle, but it might be better to slightly brake to match velocities instead of immediately steering away from it if the vehicle is driving away.

5. **Closed loop simulations**
An interesting option for future research would be to combine the optimization with MPC/NMPC for closed-loop simulations. Later it can then be implemented on an experimental vehicle platform. E.g., a Radio Controlled (RC) vehicle as used in the research of Baars [10].

6. **Adaptive grid meshing**
Consider options for adaptive meshing of the trajectory. In this research, a relatively dense grid is used over the entire trajectory. Adaptive meshing only applies a dense grid to the segments where it is necessary and a coarse grid for the segments where the local error estimation is considerably smaller.

7. **PSOPT - an open source optimal control package**
It is recommended to look into an implementation of the proposed trajectory generator in, e.g., C or Java for future research. This implementation could help to improve the calculation speed since these programming languages are generally considered more efficient than MATLAB. Consider using PSOPT instead of ICLOCS2 since this software package is also an open-source application written in C++. The reason for choosing ICLOCS2 eventually in this research is that it proved to be a working solution to solve OPs for motion planning. However, it has not been the goal to find the most accurate or efficient software available. Therefore, it is recommended to reconsider the options available, and the PSOPT software package is one of these options.

Bibliography

[1] World Health Organization, "Global status report on road safety 2018," 2018. [Online]. Available: https://www.who.int/violence_injury_prevention/road_safety_status/2018/en/

[2] S. Singh, "Critical reasons for crashes investigated in the National Motor Vehicle Crash Causation Survey," *National Highway Traffic Safety Administration Report No. DOT HS 812 115*, 2015.

[3] R. Y. Hindiyeh, "Dynamics and Control of Drifting in Automobiles," Ph.D. dissertation, Stanford University, 2013.

[4] J. N. Dang, "Statistical Analysis of the Effectiveness of Electronic Stability Control (ESC) Systems Final Report," *National Highway Traffic Safety Administration Report No. DOT HS 809 790*, 2007.

[5] On-Road Automated Driving (ORAD) committee, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles J3016," SAE International, Tech. Rep., 2018.

[6] K. Kritayakirana, "Autonomous Vehicle Control at the Limits of Handling," Ph.D. dissertation, Stanford University, 2012.

[7] E. Velenis, D. Katzourakis, E. Frazzoli, P. Tsiotras, and R. Happee, "Steady-state drifting stabilization of rwd vehicles," *Control Engineering Practice*, vol. 19, no. 11, pp. 1363–1376, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066111001614

[8] J. Y. Goh and J. C. Gerdes, "Simultaneous stabilization and tracking of basic automobile drifting trajectories," in *IEEE Intelligent Vehicles Symposium, Proceedings*, vol. 2016-Augus. Institute of Electrical and Electronics Engineers Inc., 2016, pp. 597–602.

[9] E. Wachter, "Lateral Path Tracking in Limit Handling Condition using SDRE Control," Master's thesis, Chalmers University of Technology, 2016.

[10] M. Baars, "Modelling and Control of a Scaled Car In and Beyond Stable Limit Handling," Master's thesis, Delft University of Technology, 2018.

[11] J. Y. M. Goh, "Automated Vehicle Control Beyond the Stability Limits," Ph.D. dissertation, Stanford University, 2019.

[12] E. Joa, H. Cha, Y. Hyun, Y. Koh, K. Yi, and J. Park, "A new control approach for automated drifting in consideration of the driving characteristics of an expert human driver," *Control Engineering Practice*, vol. 96, p. 104293, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0967066119302345

[13] d. Bárdos, d. Domina, Z. Szalay, V. Tihanyi, and L. Palkovics, "MIMO Controller Design for Stabilizing Vehicle Drifting," in *2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo)*, 2019, pp. 187–192.

[14] A. Eckert, B. Hartmann, M. Sevenich, and P. Rieth, "Emergency Steer & Brake Assist - A Systematic Approach for System Integration of Two Complementary Driver Assistance Systems," *Proc. of 22nd International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2011.

[15] K. Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1599–1616, 2012.

[16] X. Hu, L. Chen, B. Tang, D. Cao, and H. He, "Dynamic path planning for autonomous driving on various roads with avoidance of static and moving obstacles," *Mechanical Systems and Signal Processing*, vol. 100, pp. 482–500, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888327017303825

[17] N. Kapania, "Trajectory Planning and Control for an Autonomous Race Vehicle," Ph.D. dissertation, Stanford University, 2016.

[18] E. Ono, S. Hosoe, H. D. Tuan, and S. Doi, "Bifurcation in vehicle dynamics and robust front wheel steering control," *IEEE Transactions on Control Systems Technology*, vol. 6, no. 3, pp. 412–420, 1998.

[19] C. G. Bobier-Tiu, C. E. Beal, J. C. Kegelman, R. Y. Hindiyeh, and J. C. Gerdes, "Vehicle control synthesis using phase portraits of planar dynamics," *Vehicle System Dynamics*, vol. 57, no. 9, pp. 1318–1337, 2019. [Online]. Available: https://doi.org/10.1080/00423114.2018.1502456

[20] A. Gasparetto, P. Boscariol, A. Lanzutti, and R. Vidoni, *Path Planning and Trajectory Planning Algorithms: A General Overview.* Cham: Springer International Publishing, 2015, pp. 3–27. [Online]. Available: https://doi.org/10.1007/978-3-319-14705-5_1

[21] J. H. Jeon, S. Karaman, and E. Frazzoli, "Anytime computation of time-optimal off-road vehicle maneuvers using the RRT," in *Proceedings of the IEEE Conference on Decision and Control*, 2011.

[22] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011. [Online]. Available: https://doi.org/10.1177/0278364911406761

[23] D. Casanova, R. S. Sharp, and P. Symonds, "Minimum time manoeuvring: The significance of yaw inertia," *Vehicle System Dynamics*, vol. 34, no. 2, pp. 77–115, 2000. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1076/0042-3114%28200008%2934%3A2%3B1-G%3BFT077

[24] A. Heilmeier, A. Wischnewski, L. Hermansdorfer, J. Betz, M. Lienkamp, and B. Lohmann, "Minimum curvature trajectory planning and control for an autonomous race car," *Vehicle System Dynamics*, vol. 58, no. 10, pp. 1497–1527, 2020. [Online]. Available: https://doi.org/10.1080/00423114.2019.1631455

[25] E. Velenis and P. Tsiotras, "Minimum time vs maximum exit velocity path optimization during cornering," in *IEEE International Symposium on Industrial Electronics*, 2005.

[26] E. Velenis, P. Tsiotras, and J. Lu, "Aggressive maneuvers on loose surfaces: Data analysis and input parametrization," in *2007 Mediterranean Conference on Control and Automation, MED*, 2007.

[27] ——, "Modeling aggressive maneuvers on loose surfaces: The cases of Trail-Braking and Pendulum-Turn," in *2007 European Control Conference, ECC 2007*, 2007.

[28] ——, "Optimality properties and driver input parameterization for trail-braking cornering," *European Journal of Control*, vol. 14, no. 4, pp. 308–320, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0947358008707751

[29] B. Olofsson, K. Lundahl, K. Berntorp, and L. Nielsen, "An investigation of optimal vehicle maneuvers for different road conditions," *IFAC Proceedings Volumes*, vol. 46, no. 21, pp. 66–71, 2013, 7th IFAC Symposium on Advances in Automotive Control. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016383446

[30] D. Tavernini, M. Massaro, E. Velenis, D. I. Katzourakis, and R. Lot, "Minimum time cornering: the effect of road surface and car transmission layout," *Vehicle System Dynamics*, vol. 51, no. 10, pp. 1533–1547, 2013. [Online]. Available: https://doi.org/10.1080/00423114.2013.813557

[31] D. Tavernini, E. Velenis, R. Lot, and M. Massaro, "The Optimality of the Handbrake Cornering Technique," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 4, 2014, 041019. [Online]. Available: https://doi.org/10.1115/1.4026836

[32] K. Berntorp, B. Olofsson, K. Lundahl, and L. Nielsen, "Models and methodology for optimal trajectory generation in safety-critical road–vehicle manoeuvres," *Vehicle System Dynamics*, vol. 52, no. 10, pp. 1304–1332, 2014. [Online]. Available: https://doi.org/10.1080/00423114.2014.939094

[33] R. Verschueren, S. De Bruyne, M. Zanon, J. V. Frasch, and M. Diehl, "Towards time-optimal race car driving using nonlinear mpc in real-time," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 2505–2510.

[34] J. V. Frasch, A. Gray, M. Zanon, H. J. Ferreau, S. Sager, F. Borrelli, and M. Diehl, "An auto-generated nonlinear mpc algorithm for real-time obstacle avoidance of ground vehicles," in *2013 European Control Conference (ECC)*, 2013, pp. 4136–4141.

[35] C. Choi and Y. Kang, "Simultaneous braking and steering control method based on nonlinear model predictive control for emergency driving support," *International Journal of Control, Automation and Systems*, vol. 15, no. 1, pp. 345–353, 2017. [Online]. Available: http://dx.http//www.springer.com/12555

[36] A. S. P. Singh and O. Nishihara, “Nondimensionalized univariate equation characterizing optimal state feedback control for collision avoidance,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 10, pp. 3344–3359, 2018.

[37] D. E. Kirk, *Optimal Control Theory: An Introduction.* London, England: Prentice-Hall, 1971.

[38] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming.* Society for Industrial and Applied Mathematics, 2010.

[39] G. Perantoni, “Optimal Control of Vehicle Systems,” Ph.D. dissertation, University of Oxford, 2013.

[40] S. van Koutrik, “Optimal Control for Race Car Minimum Time Maneuvering,” Master’s thesis, Delft University of Technology, 2015.

[41] D. Tavernini, E. Velenis, R. Lot, and M. Massaro, “On the optimality of handbrake cornering,” in *52nd IEEE Conference on Decision and Control*, 2013, pp. 2233–2238.

[42] G. Perantoni and D. J. Limebeer, “Optimal control for a Formula One car with variable parameters,” *Vehicle System Dynamics*, vol. 52, no. 5, pp. 653–678, 2014. [Online]. Available: https://www.tandfonline.com/action/journalInformation?journalCode=nvsd20

[43] M. Kelly, “Transcription methods for trajectory optimization: a beginners tutorial,” *arXiv: Optimization and Control*, 2017.

[44] “Canon Tutorial,” Accessed: 2021-05-15. [Online]. Available: http://www.matthewpeterkelly.com/tutorials/trajectoryOptimization/canon.html

[45] B. Jackson, “Exploring Quadrature Rules in Direct Collocation Methods for Trajectory Optimization,” Tech. Rep., 2018.

[46] H. S. Rodrigues, M. T. T. Monteiro, and D. F. M. Torres, “Optimal control and numerical software: an overview,” *Systems Theory: Perspectives, Applications and Developments*, pp. 93–110, 2014. [Online]. Available: http://arxiv.org/abs/1401.7279

[47] S. Leyffer and A. Mahajan, *Software For Nonlinearly Constrained Optimization.* American Cancer Society, 2011. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470400531.eorms0570

[48] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.

[49] M. E. Vázquez-Méndez and G. Casal, “The clothoid computation: A simple and efficient numerical algorithm,” *Journal of Surveying Engineering*, vol. 142, no. 3, p. 04016005, 2016. [Online]. Available: https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29SU.1943-5428.0000177

[50] P. Lugner and J. Edelmann, *Basics of Vehicle Dynamics, Vehicle Models.* Cham: Springer International Publishing, 2019, pp. 1–45. [Online]. Available: https://doi.org/10.1007/978-3-319-79008-4_1

[51] M. Acosta, S. Kanarachos, and M. E. Fitzpatrick, “A hybrid hierarchical rally driver model for autonomous vehicle agile maneuvering on loose surfaces,” in *ICINCO 2017 - Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics*, 2017.

[52] H. B. Pacejka, *Tire and Vehicle Dynamics*, Thirdal ed. Butterworth-Heinemann, 2012.

[53] M. Kelly, “An introduction to trajectory optimization: How to do your own direct collocation,” *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017. [Online]. Available: https://doi.org/10.1137/16M1062569

[54] Y. Nie, O. Faqir, and E. C. Kerrigan, “ICLOCS2: Try this Optimal Control Problem Solver Before you Try the Rest,” in *2018 UKACC 12th International Conference on Control (CONTROL)*, 2018, pp. 336–336.

[55] M. J. Weinstein and A. V. Rao, “Algorithm 984: Adigator, a toolbox for the algorithmic differentiation of mathematical functions in matlab using source transformation via operator overloading,” *ACM Trans. Math. Softw.*, vol. 44, no. 2, 2017. [Online]. Available: https://doi.org/10.1145/3104990

[56] N. D. Bianco, E. Bertolazzi, F. Biral, and M. Massaro, “Comparison of direct and indirect methods for minimum lap time optimal control problems,” *Vehicle System Dynamics*, vol. 57, no. 5, pp. 665–696, 2019. [Online]. Available: https://doi.org/10.1080/00423114.2018.1480048

[57] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006. [Online]. Available: https://link.springer.com/article/10.1007/s10107-004-0559-y

[58] “ASAM OpenCRG®.” [Online]. Available: https://www.asam.net/standards/detail/opencrg/

[59] I. J. M. Besselink, “Vehicle Dynamics - 4AT000, lecture notes 2018,” 2018, Eindhoven University of Technology.

[60] M. Bian, L. Chen, Y. Luo, and K. Li, “A Dynamic Model for Tire/Road Friction Estimation under Combined Longitudinal/Lateral Slip Situation,” in *SAE Technical Papers 2014-01-0123*, 2014. [Online]. Available: https://doi.org/10.4271/2014-01-0123

Appendix A

Kinematic models

Last-point-to-brake kinematic model equations:

Motion of the vehicle:

$$\begin{aligned} x &= v_{x0}t + \tfrac{1}{2}a_x t^2, \quad \text{with } v_{x0} = \text{initial velocity} \\ v_x &= v_{x0} + a_x t \end{aligned} \tag{A.1}$$

Time to stop, i.e. $v_x = 0$:

$$t = -v_{x0}/a_x. \tag{A.2}$$

Substituting the time to stop in the distance to stop equation results in the last-point-to-brake equation:

$$L_{brake} = \frac{v_{x0}^2}{2a_x}, \tag{A.3}$$

where L_{brake} is the critical last-point-to-brake distance for a given initial velocity, and a_x is the maximum longitudinal acceleration based in the friction coefficient in this direction.

Last-point-to-steer kinematic model equations:

Motion of the vehicle:

$$\begin{aligned} x &= v_{x0}t, \quad \text{with } v_{x0} = \text{constant} \\ y &= \tfrac{1}{2}a_y t^2 \end{aligned} \tag{A.4}$$

Eliminating time results in the last-point-to-steer equation:

$$L_{steer} = v_{x0}\sqrt{\frac{2y}{a_y}}, \tag{A.5}$$

where L_{steer} is the critical last-point-to-steer distance for a given velocity, y is the lateral offset the vehicle has to travel in order to avoid the obstacle, and a_y is the maximum lateral acceleration based in the friction coefficient in this direction.

Appendix B

Vehicle and tire parameters

B.1 Vehicle parameters

In Table B.1 an overview of all parameters of the single-track vehicle model is provided.

Table B.1: Parameters of the single-track vehicle model

Parameter	**Value**	**Unit**	**Description**
m	1300	kg	Vehicle mass
g	9.81	m/s^2	Gravity
I_z	2000	$kg\ m^2$	Yaw moment of inertia
l	2.5	m	Wheelbase
a	0.975	m	Distance CG to front wheel
b	1.525	m	Distance CG to rear wheel
WD	61/39	%	Weight distribution
h_{cg}	0.5	m	Height CG
W	1.5	m	Track width
R_w	0.28	m	Wheel radius
I_w	1.8	$kg\ m^2$	Axle inertia
P_{max}	110	kW	Engine power
δ_{max}	30	°	Absolute maximum steering angle
k_t	0/1	−	Propulsive torque distribution (FWD/RWD)

B.2 Tire parameters

Different coefficients are used for the front and rear wheel of the single-track vehicle model because various parameters are dependent on the normal load F_z. A complete overview of all the coefficients used in this research is provided in Table B.2.

Table B.2: Combined slip Magic Formula coefficients

Parameter	**Dry asphalt**	**Gravel**
$\mu_{x,f}$	1.20	0.6
$\mu_{x,r}$	1.20	0.6
$B_{x,f}$	11.7	1.529
$B_{x,r}$	11.1	1.529
$C_{x,f},\ C_{x,r}$	1.69	1.09
$E_{x,f}$	0.377	-0.951
$E_{x,r}$	0.362	-0.951
$\mu_{y,f}$	0.935	0.6
$\mu_{y,r}$	0.961	0.6
$B_{y,f}$	8.86	1.529
$B_{y,r}$	9.30	1.529
$C_{y,f},\ C_{y,r}$	1.19	1.09
$E_{y,f}$	-1.21	-0.951
$E_{y,r}$	-1.11	-0.951
$C_{x\alpha,f},\ C_{x\alpha,r}$	1.09	1.02
$B_{x1,f},\ B_{x1,r}$	12.4	75.4
$B_{x2,f},\ B_{x2,r}$	-10.8	-43.1
$C_{y\lambda,f},\ C_{y\lambda,r}$	1.08	0.984
$B_{y1,f},\ B_{y1,r}$	6.46	33.8
$B_{y2,f},\ B_{y2,r}$	4.20	42.0

Appendix C

Track model derivation

In this appendix, the track model will be derived from Figure 3.1 provided in Section 3.1. From this figure two equations can be derived, see (C.1) and (C.2):

$$v_s = (R - \Delta y)\dot{\psi}_s, \tag{C.1}$$

$$\begin{aligned} v_s &= \dot{u}\cos(\Delta\psi) - \dot{v}\sin(\Delta\psi), \\ v_s &= V\cos(\Delta\psi + \beta), \end{aligned} \tag{C.2}$$

where v_s is the projected vehicle speed along the road centerline, R is the radius of curvature, ψ_s the heading angle of the road centerline. Substituting (C.2) in (C.1) and rewriting results in:

$$\dot{\psi}_s = \frac{V\cos(\Delta\psi + \beta)}{R - \Delta y} \tag{C.3}$$

The vehicle's velocity along the path $\dot{s} = \frac{ds}{dt}$ is then defined by:

$$\dot{s} = R \cdot \dot{\psi}_s \tag{C.4}$$

Substituting (C.3) and rewriting results in:

$$\dot{s} = \frac{V\cos(\Delta\psi + \beta)}{1 - \Delta y \cdot \kappa} \tag{C.5}$$

From Figure 3.1 another equation can be derived, see (C.6).

$$\Delta\psi = \psi - \psi_s \tag{C.6}$$

$$\dot{\Delta\psi} = r - \dot{\psi}_s \tag{C.7}$$

Substituting (C.3) and rewriting results in:

$$\dot{\Delta\psi} = r - \frac{V\cos(\Delta\psi + \beta)}{R - \Delta y} \tag{C.8}$$

Since the second term in the equation is equal to $\dot{s}$, (C.8) can be rewritten:

$$\dot{\Delta\psi} = r - \kappa \cdot \dot{s} \tag{C.9}$$

Appendix D

OpenCRG

For the curvilinear coordinate system, a road geometry is defined by the road centerline length s, the road width Δy, and the radius of curvature R or its reciprocal, the road curvature κ. In order to construct the road geometry and extract the information required by the optimization algorithm, the open-source ASAM OpenCRG [58] (from now on OpenCRG) tool is used. This tool can be used to create, modify, and evaluate road surfaces. Road segments are described by a heading angle and start position, and this information is stored in a specific data layout, called "curved regular grid"(CRG). In this research, the tool is used to construct road geometries from nine predefined variables:

- Length of straight section [m]
- Corner radius [m]
- Corner angle [deg]
- Road width [m]
- Corner direction: -1 (CW) and 1 (CCW)
- Clothoid included: 0 (NO) and 1 (YES)
- Start heading angle clothoid [deg]
- Final heading angle clothoid [deg]
- Discretization step [m]

These nine variables are used to construct the different road segments present in the road geometries tested in this research: straight road sections, curved road sections, and segments that describe a transition between a straight and curved segment (clothoid). By appending all road segments using the OpenCRG tool, a road geometry is constructed, and from this geometry, the information required by the optimization algorithm is extracted. The following data is extracted from the constructed road geometry:

- Road centerline x/y-positions
- Road left boundary x/y-positions
- Road right boundary x/y-positions
- Road length array s
- Road heading angle array ψ_s
- Road curvature array κ

A flowchart is created, see Figure D.1, to explain how the OpenCRG tool is used to create the road geometries shown in this research. The steps shown in green are part of the initialization of the optimization algorithm and thus are not part of OpenCRG. Only the steps indicated by the blue border are part of the OpenCRG tool. The most important functions utilized from the OpenCRG tool are *crg_write*, *crg_append* to append the straight and curved road sections, and *crg_road_borders* to determine the coordinates of the road boundaries. The other data is automatically generated when the road centerline data is written to the OpenCRG data layout. After the three function calls, the required data stated above can be extracted. This data can then be used in the optimization algorithm or for plotting the road geometry.

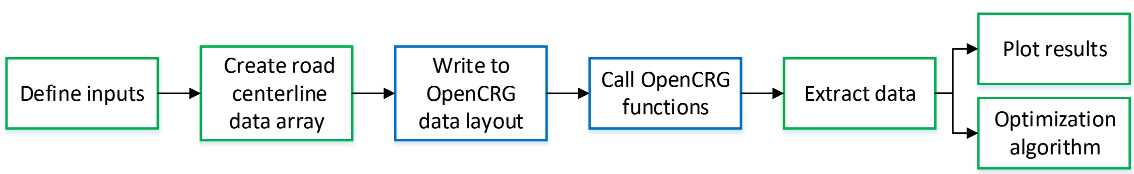


Figure D.1: Flowchart - usage of the OpenCRG tool.

Appendix E

Vehicle model

In this appendix the single-track vehicle model used in Section 3.2.1 is explained. In addition, the model states, that are not related to the curvilinear coordinates, and control inputs as defined in Section 4.2 are further elaborated here. The model states and control inputs are defined as:

$$x = (V \quad \beta \quad r)^\top, \tag{E.1}$$

$$u = (\delta \quad \lambda_f \quad \lambda_r \quad \Delta F_z)^\top, \tag{E.2}$$

where V is the absolute velocity, β the vehicle sideslip angle, r the yaw rate, δ the steering angle, λ_f and λ_r the longitudinal slip values of the front and rear wheel respectively, and ΔF_z represents the load transfer between the front and rear wheel.

The state derivatives are as follows:

$$\dot{V} = \frac{F_{x,f}\cos(\delta-\beta) - F_{y,f}\sin(\delta-\beta) + F_{x,r}\cos\beta + F_{y,r}\sin\beta}{m}, \tag{E.3}$$

$$\dot{\beta} = \frac{F_{x,f}\sin(\delta-\beta) + F_{y,f}\cos(\delta-\beta) - F_{x,r}\sin\beta + F_{y,r}\cos\beta}{mV} - r, \tag{E.4}$$

$$\dot{r} = \frac{a(F_{y,f}\cos\delta + F_{x,f}\sin\delta) - bF_{y,r}}{I_z}, \tag{E.5}$$

where V is the absolute velocity, β the vehicle sideslip angle, r the yaw rate, F_{xf} and F_{xr} the longitudinal forces, F_{yf} and F_{yr} the lateral forces, δ the steering angle, m the vehicle mass, and I_z the yaw moment of inertia.

In order to compute the tire forces in the lateral direction it is necessary to determine the lateral tire slip angles of the front and rear wheel. These two slip angles are calculated using:

$$\alpha_f = \delta - \text{atan}\left(\frac{V\sin\beta + a\cdot r}{V\cos\beta}\right), \tag{E.6}$$

$$\alpha_r = -\text{atan}\left(\frac{V\sin\beta - b\cdot r}{V\cos\beta}\right), \tag{E.7}$$

To compute the engine power used by the vehicle, the rotational velocities of the front and rear wheel and torques applied to these wheels are required. These variables are defined as:

$$\omega_f = \frac{(V\cos(\beta-\delta) + a\cdot r\cdot\sin\delta)(\lambda_f+1)}{R_w}, \tag{E.8}$$

$$\omega_f = \frac{V\cos\beta\ (\lambda_r+1)}{R_w}, \tag{E.9}$$

$$T_f = F_{x,f}\cdot R_w, \tag{E.10}$$

$$T_f = F_{x,r}\cdot R_w, \tag{E.11}$$

where R_w is the radius of the tire.

Using these variables the engine power can be computed as follows:

$$P = \frac{T_f \cdot \omega_f + T_r \cdot \omega_r}{1000}, \tag{E.12}$$

where P is the used engine power in $[kW]$.

Appendix F

Tire models

This appendix will provide an overview of three tire model commonly used in literature for vehicle dynamics simulations and more related to the research in this thesis, automated drifting.

Linear tire model
For small values of longitudinal and lateral slip, the tire behavior is approximately linear and can be expressed using the following equations [59]:

$$F_x = C_{F\lambda}\lambda \quad \text{and} \quad F_y = C_{F\alpha}\alpha, \tag{F.1}$$

where $C_{F\lambda}$ is the longitudinal slip stiffness, $C_{F\alpha}$ the cornering stiffness, λ the longitudinal slip and α the tire slip angle. At high values of slip the linear model will be inaccurate since it does not model tire force limits and combined slip behavior. For accurate modeling of the tire characteristics of tire forces near saturation a nonlinear tire model is required. Since this research is about drifting with large slip angles and tires near saturation, this model will not be used but it is added for completeness.

Brush tire model
The brush tire model is a simple physical model, which represents the tire in the form of three components: a ring, a carcass and bristles that touch the road surface. The evolution of the lateral force distribution in the contact patch for an increasing slip angle is shown in Figure F.1. The complete derivation of the brush model will not be will not be discussed in this section, for more information the reader may consult the PhD report of Rami Yusef Hindiyeh [3]. Eventually, the lateral tire force can be found by integrating the lateral force per unit length $q_y(x)$ for a specific slip angle over the contact patch length. This results in the following function for the lateral tire force F_y:

$$F_y = \begin{cases} -C_\alpha \tan\alpha + \frac{C_\alpha^2}{3\mu F_z}|\tan\alpha|\tan\alpha - \frac{C_\alpha^3}{27\mu^2 F_z^2}\tan^3\alpha, & |\alpha| \leq \alpha_{sl} \\ -\mu F_z \text{sgn}\alpha, & |\alpha| > \alpha_{sl} \end{cases} \tag{F.2}$$

It is also possible to extend this model to the combined slip brush tire model, for modeling both lateral and longitudinal forces, but this adds significant complexity to the equations. The combined slip model depends on the addition of the wheel speed ω since the longitudinal slip λ is a function of the wheel speed. A simpler approach to model the combined slip characteristics of a tire is using the friction circle in combination with a derating factor ε. The derating factor ε is based on the assumption that the longitudinal force is treated as a direct input to the vehicle. Here the friction circle formula in (2.1), can be used to compute ε which describes the lateral force as a fraction of the total available force. According to Hindiyeh [3], this technique make it possible to account for the coupling of the longitudinal and lateral force during drifting without the need to include the longitudinal dynamics in the model.

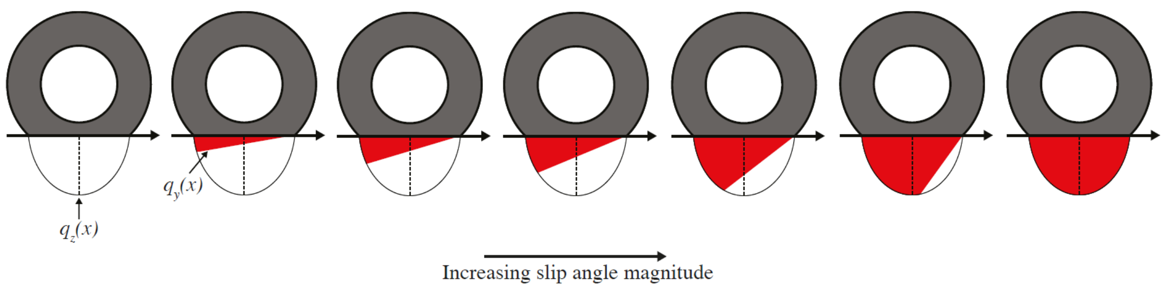


Figure F.1: Evolution of lateral force per unit length $q_y(x)$ (shown in red) with increasing slip angle. [3]

Modified Dugoff tire model
A different semi-empirical tire model is the Modified Dugoff tire model, which is extension of the original Dugoff model. The original model is not able to model the decrease of the friction force when it enters the nonlinear region of the tire. According to Bian et al. [60], the modified model is able to accurately capture the transient behavior for pure and combined slip conditions. Although the accuracy of the Modified Dugoff model is slightly lower than that of the Magic Formula it is used in literature as a replacement because it has simpler equations and a short convergence time which makes it suitable for control purposes. This is the exact reasoning why Mart Baars used this model in his research to control the Delft Scaled Vehicle in a drifting maneuver [10]. The modified Dugoff tire model is defined (F.3).

$$F_x = C_\lambda \frac{\lambda}{\lambda + 1} f(\sigma) G_\lambda \tag{F.3a}$$

$$F_y = C_\alpha \frac{\tan(\alpha)}{\lambda + 1} f(\sigma) G_\alpha, \tag{F.3b}$$

with

$$f(\sigma) = \begin{cases} \sigma(2-\sigma), & \text{if} \quad \sigma < 1 \\ 1, & \text{if} \quad \sigma \geq 1 \end{cases} \tag{F.4}$$

where σ is described by:

$$\sigma = \frac{\mu F_z (1 + \lambda)}{2\sqrt{(C_\lambda \lambda)^2 + (C_\alpha \tan(\alpha))^2}}. \tag{F.5}$$

The following magnifying factors G_σ and G_α are used to model decreasing tire forces after the maximum force peak for increasing tire slips:

$$G_\lambda = (1.15 - 0.75\mu)\lambda^2 - (1.63 - 0.75\mu)\lambda + 1.27, \tag{F.6}$$

$$G_\alpha = (\mu - 1.6)\tan\alpha + 1.155. \tag{F.7}$$

This is not possible with the original Dugoff tire model, hence why the modified Dugoff tire model is presented here.

Isotropic versus non-isotropic
Two models are compared in Figure F.2, the Magic Formula combined slip model and the Magic Formula similarity method. The similarity method used in [30] is considered to be an isotropic model since it has the same characteristic in the longitudinal and lateral direction. The Magic Formula coefficients for this method are adopted from that reference as well. The combined slip model used in this research is considered to be a non-isotropic model because of the different characteristics in the two directions. In order to visualize and quantify the difference between the two models, an illustrative 3D surface defined as the resulting force F_{res} can be plotted [32].

The resulting tire force is defined as:

$$F_{res} = \sqrt{F_x^2 + F_y^2}. \tag{F.8}$$

According to Berntop et al. [32], the absence of the characteristic peaks in F_{res} for an isotropic model has a significant influence on the behavior of the tire model. Therefore, it also has a significant affect on the vehicle behavior.

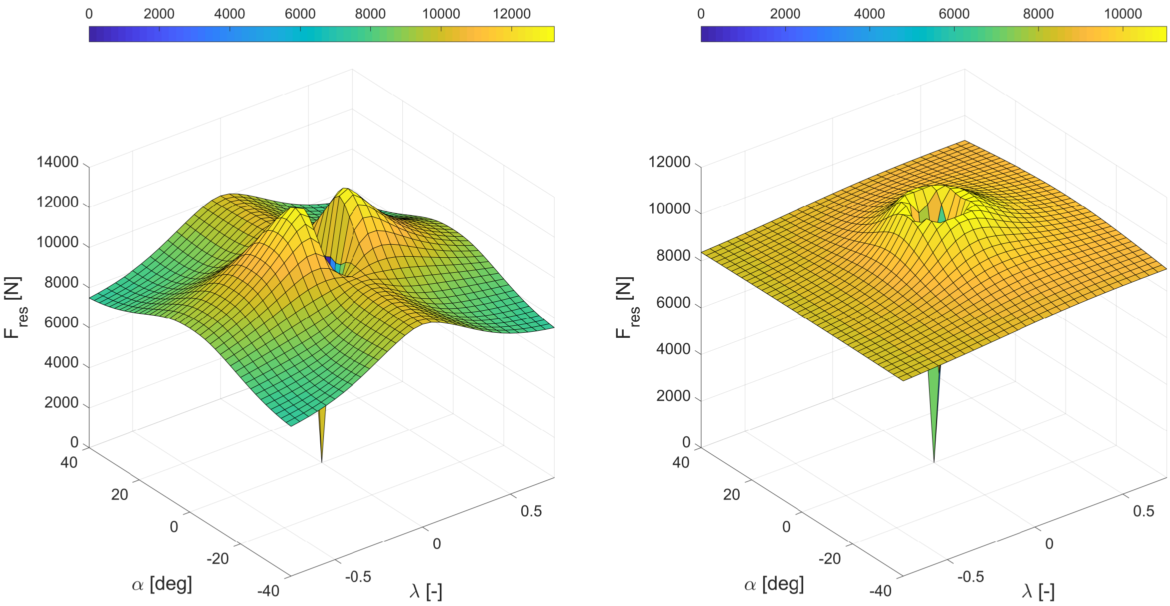


Figure F.2: Resulting tire forces F_{res}, for the Magic Formula combined slip model (left) and the Magic Formula similarity method (right).

Appendix G

Trapezoidal collocation

According to Jackson [45], the transcription process can be summarized in six steps:

1. Discretize the problem
2. Define an initial guess
3. Create the augmented state
4. Calculate the constraints
5. Solve the NLP
6. Interpolate the results

The first step for discretization is to approximate the continuous integral, Lagrange term, of the objective function using a summation. This step is where the trapezoid rule for integration becomes important, which is applied between each collocation point to approximate the value of the integrand $w(t_k) = w_k$ at the collocation points s_k along the trajectory. This yields the following approximation, where $h_k = s_{k+1} - s_k$:

$$\int_{s_0}^{s_f} L(\boldsymbol{x}(s), \boldsymbol{u}(s), s) \approx \sum_{k=0}^{N-1} \tfrac{1}{2} h_k \cdot (w_k + w_{k+1}). \tag{G.1}$$

The trapezoid rule for integration is used as well to construct the collocation constraints. Direct collocation methods represent the system dynamics as a collection of constraints, the collocation constraints. In order to achieve this, the system dynamics are written in integral form, as displayed in (G.2), and then the trapezoidal method is used to approximate the integral. The system dynamics between each of the collocation points are approximated using:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}, \\ \int_{s_0}^{s_f} \dot{\boldsymbol{x}} ds &= \int_{s_0}^{s_f} \boldsymbol{f} ds, \\ \boldsymbol{x}_{k+1} - \boldsymbol{x}_k &\approx \tfrac{1}{2} h_k \cdot (\boldsymbol{f}_{k+1} + \boldsymbol{f}_k). \end{aligned} \tag{G.2}$$

The complete collection of collocation constraints follows from the application of (G.2) between every pair of collocation points in the trajectory,

$$\boldsymbol{x}_{k+1} - \boldsymbol{x}_k = \tfrac{1}{2} h_k \cdot (\boldsymbol{f}_{k+1} + \boldsymbol{f}_k), \qquad k \in 0, ..., (N-1), \tag{G.3}$$

where $\boldsymbol{x}_k$ is a decision variable and $\boldsymbol{f}_k$ follows from the evaluation of the system dynamics at each of the collocation points. A decision variable represents an unknown in the NLP, and it is the parameter an optimization solver adjusts to arrive at the optimal solution. There are two types of decision variables, the model states and control inputs. For more information, the reader is referred back to Section 4.2.

The system dynamics are not the only constraints that are enforced on the trajectory. As discussed in Subsection 4.2.4 and 4.2.5, there are also bounds, path constraints, and boundary constraints present in the OP for motion planning. First, the bounds on the model states and control inputs will be discussed. These constraints are enforced by approximating them at every collocation point. In a mathematical form, this equates to:

$$\begin{aligned} \boldsymbol{x} < \boldsymbol{0} \qquad &\rightarrow \boldsymbol{x}_k < \boldsymbol{0} \quad \forall k, \\ \boldsymbol{u} < \boldsymbol{0} \qquad &\rightarrow \boldsymbol{u}_k < \boldsymbol{0} \quad \forall k. \end{aligned} \tag{G.4}$$

Path constraints, the equality and inequality constraints, are similarly enforced on the trajectory by applying them to every collocation point:

$$\boldsymbol{h}(s,\boldsymbol{x},\boldsymbol{u}) \leq \boldsymbol{0} \qquad \rightarrow \boldsymbol{h}(s_k,\boldsymbol{x}_k,\boldsymbol{u}_k) \leq \boldsymbol{0} \quad \forall k. \tag{G.5}$$

The boundary conditions are constraints that enforce a restriction on the initial and final states of a trajectory, as explained in Subsection 4.2.3. In contrast to the two types of constraints previously discussed here, the boundary constraints are only enforced at the first and last collocation points:

$$\begin{aligned} \boldsymbol{g}(s_0,\boldsymbol{x}(s_0),\boldsymbol{u}(s_0)) \leq \boldsymbol{0} \qquad &\rightarrow \boldsymbol{g}(s_0,\boldsymbol{x}_0,\boldsymbol{u}_0) \leq \boldsymbol{0}, \\ \boldsymbol{g}(s_f,\boldsymbol{x}(s_f),\boldsymbol{u}(s_f)) \leq \boldsymbol{0} \qquad &\rightarrow \boldsymbol{g}(s_f,\boldsymbol{x}_f,\boldsymbol{u}_f) \leq \boldsymbol{0}. \end{aligned} \tag{G.6}$$

The next step would be solving the resulting NLP using the augmented state, and the constraints formulated previously. The augmented state z is a variable that is passed to the NLP solver, and it includes the initial and final value of the independent variable and the discretized model states and control inputs at the collocation points as shown in (2.3). The structure of the augmented state z is as follows:

$$z = [s_0, s_f, x_0, u_0, x_1, u_1, ..., , x_k, u_k, ..., x_N, u_N]. \tag{G.7}$$

Various solvers for the NLP have been discussed in the literature review, Section 2.4.3. The NLP solver selected for this research, which has shortly been discussed in Section 4.5, solves the model state and control input values at each of the collocation points in order to minimize/maximize the objective function.

After solving the NLP, the next step would be to construct a continuous trajectory by interpolating the discrete solution between the collocation points. The applied quadrature scheme determines the order of the piecewise polynomials used for interpolating. In the trapezoidal method, the trajectories of the control inputs and system dynamics are approximated using linear splines. Since the trapezoidal method is a Lobatto method, the knot points are coinciding with the collocation points. A knot point denotes a point that joins two segments of the piecewise polynomial.

Since the trapezoidal method is used, the trajectory of the control inputs is constructed using a piecewise linear function. The value of independent variable s and the control at each collocation point u_k is known, so from this point it is relatively simple to derive an expression for $u(s)$ using linear interpolation on the interval $s \in [s_k, s_{k+1}]$:

$$\boldsymbol{u}(s) \approx \boldsymbol{u}_k + \frac{\tau}{h_k}(\boldsymbol{u}_{k+1} - \boldsymbol{u}_k), \tag{G.8}$$

where τ is defined as $\tau = s - s_k$, and as formulated before $h_k = s_{k+1} - s_k$.

The same method is applied to approximate the system dynamics over one interval. Although, the model state $\boldsymbol{x}$ is of interest and not $\dot{\boldsymbol{x}}$, so both sides of (G.9) are integrated, to obtain the quadratic expression of the model states shown in (G.10):

$$\boldsymbol{f}(s) = \dot{\boldsymbol{x}}(s) \approx \boldsymbol{f}_k + \frac{\tau}{h_k}(\boldsymbol{f}_{k+1} - \boldsymbol{f}_k), \tag{G.9}$$

$$\boldsymbol{x}(s) = \int \dot{\boldsymbol{x}}(s)d\tau \approx \boldsymbol{c} + \boldsymbol{f}_k\tau + \frac{\tau^2}{2h_k}(\boldsymbol{f}_{k+1} - \boldsymbol{f}_k). \tag{G.10}$$

The constant of integration c in (G.10), can be found by solving the equation at a collocation point $\tau = s - s_k = 0$ (state at the boundary). Substituting this in (G.10) results in the final expression for the model state:

$$\boldsymbol{x}(t) \approx \boldsymbol{x}_k + \boldsymbol{f}_k\tau + \frac{\tau^2}{2h_k}(\boldsymbol{f}_{k+1} - \boldsymbol{f}_k). \tag{G.11}$$

Appendix H

Algorithm assessment

H.1 Influence of the objective function

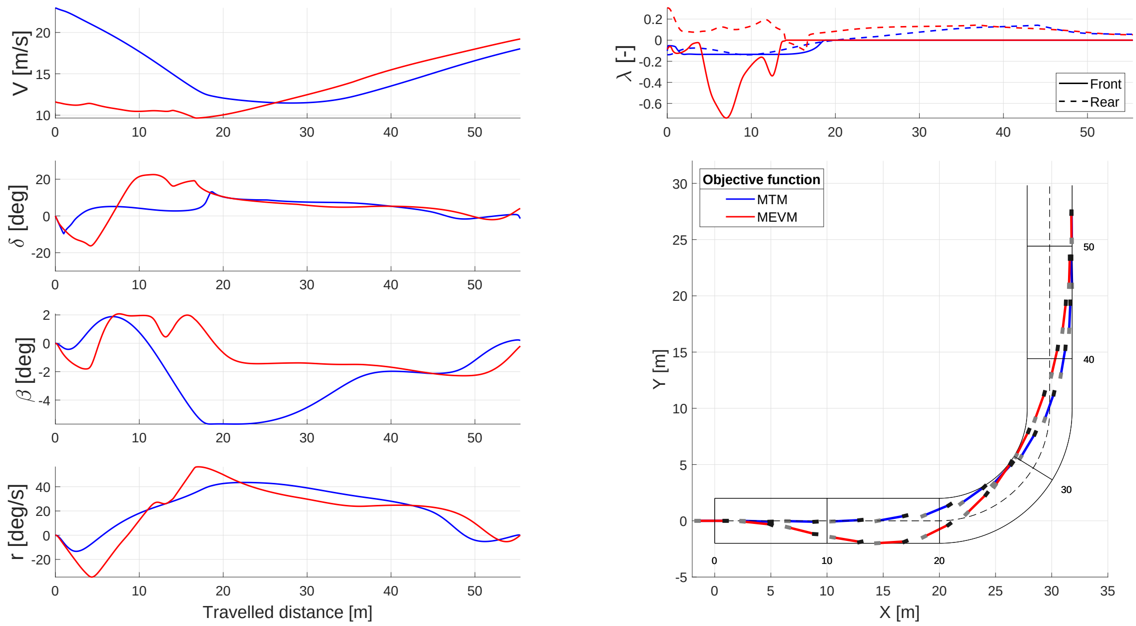


Figure H.1: Simulation of 90° corner with tire 1 (dry asphalt) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

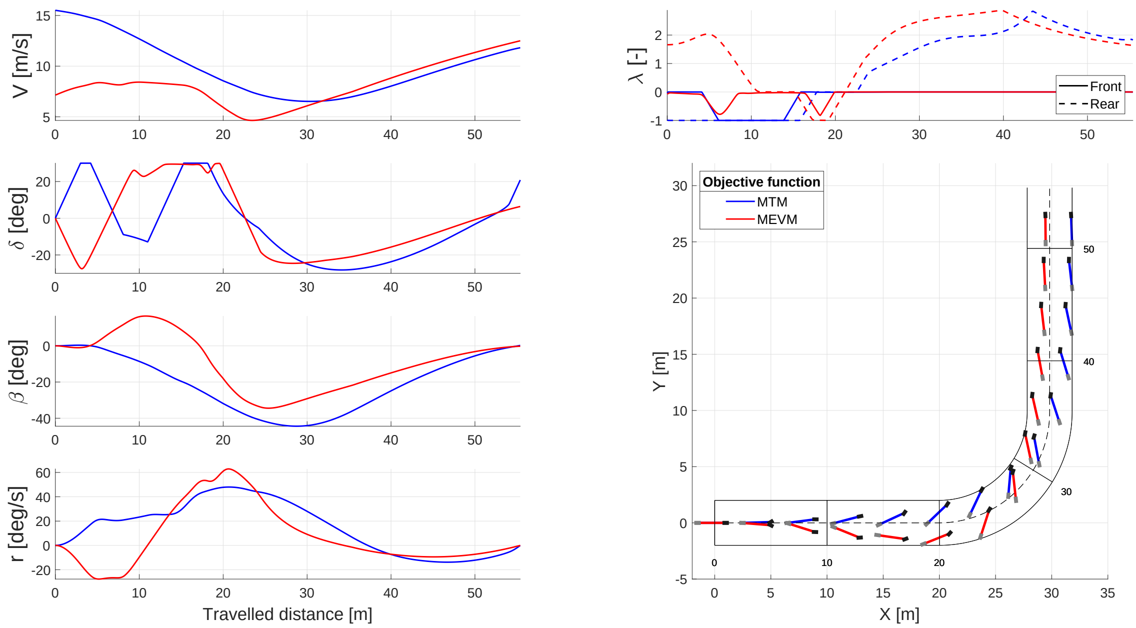


Figure H.2: Simulation of 90° corner with tire 2 (gravel) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

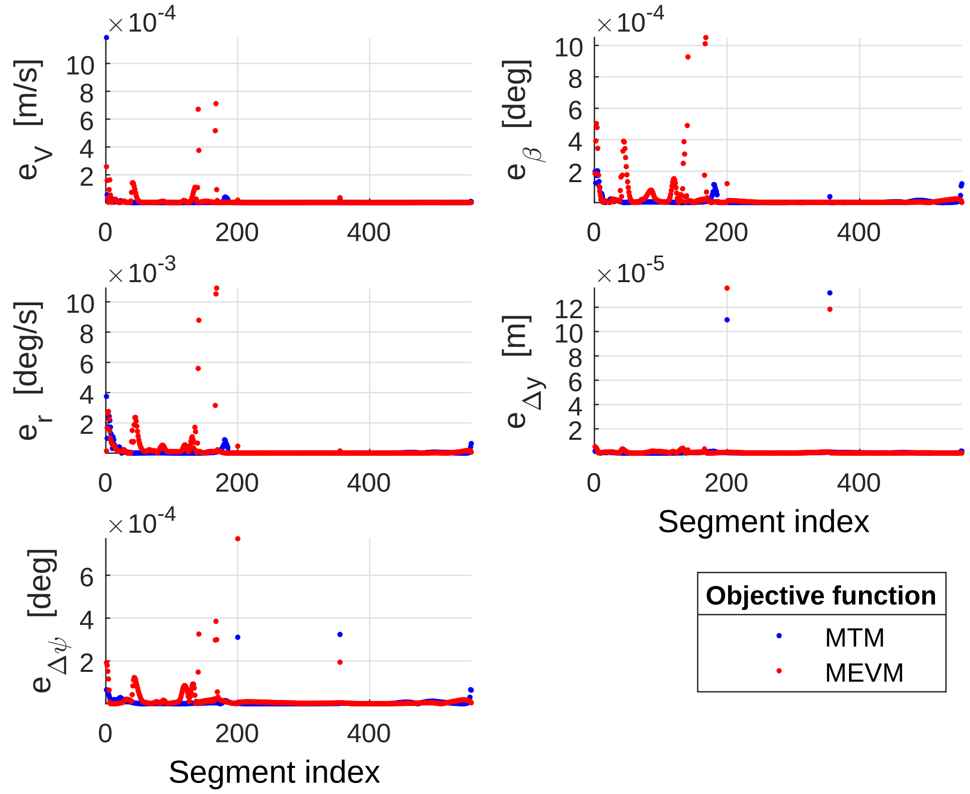


Figure H.3: Absolute local error for optimization shown in Figure H.1.

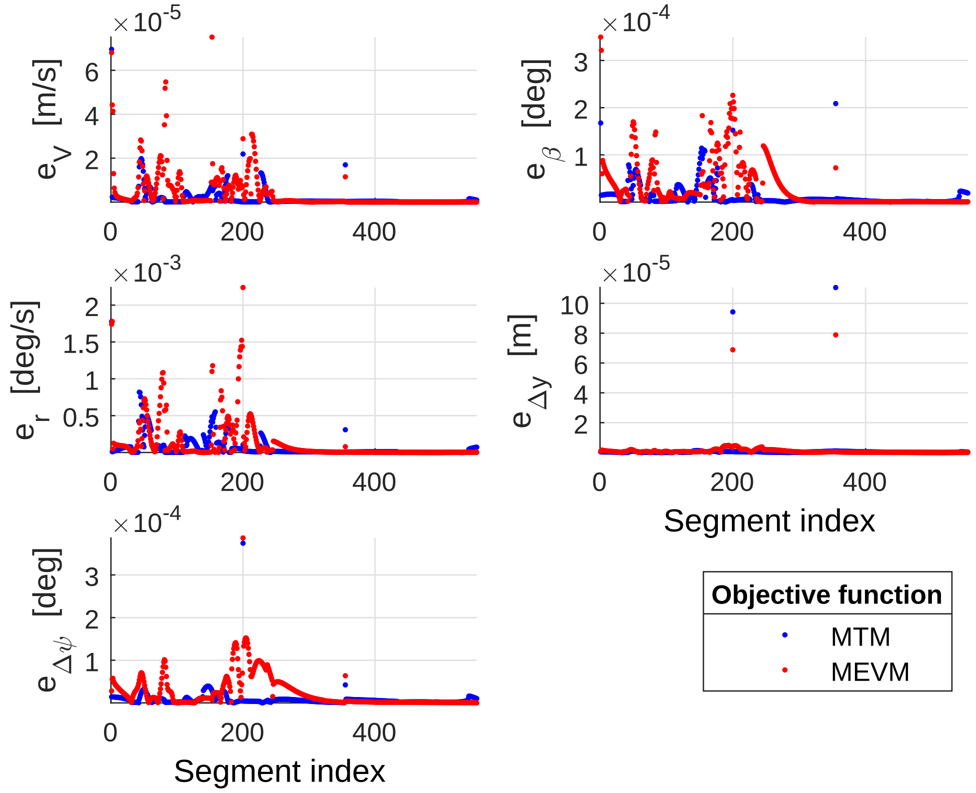
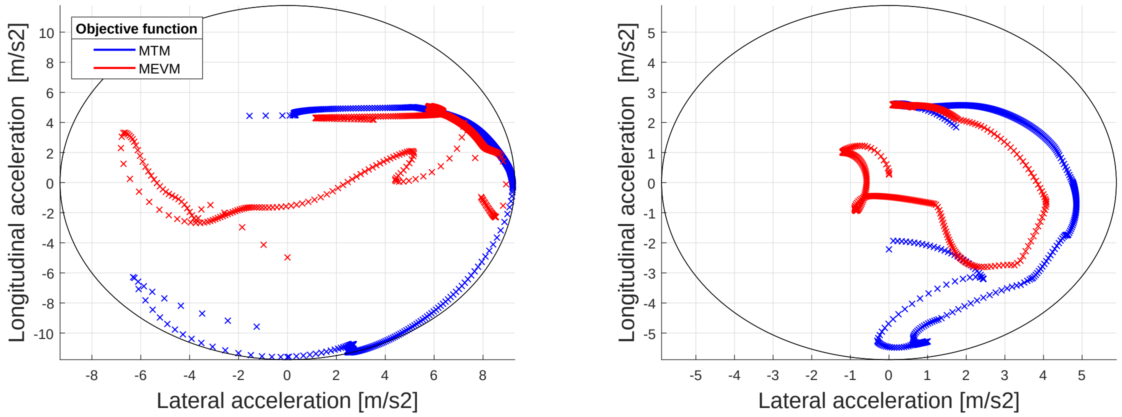


Figure H.4: Absolute local error for optimization shown in Figure H.2.



(a) Tire 1: dry asphalt (b) Tire 2: gravel

Figure H.5: G-G diagram for the simulations of a 90° corner.

H.2 Varying vehicle parameters

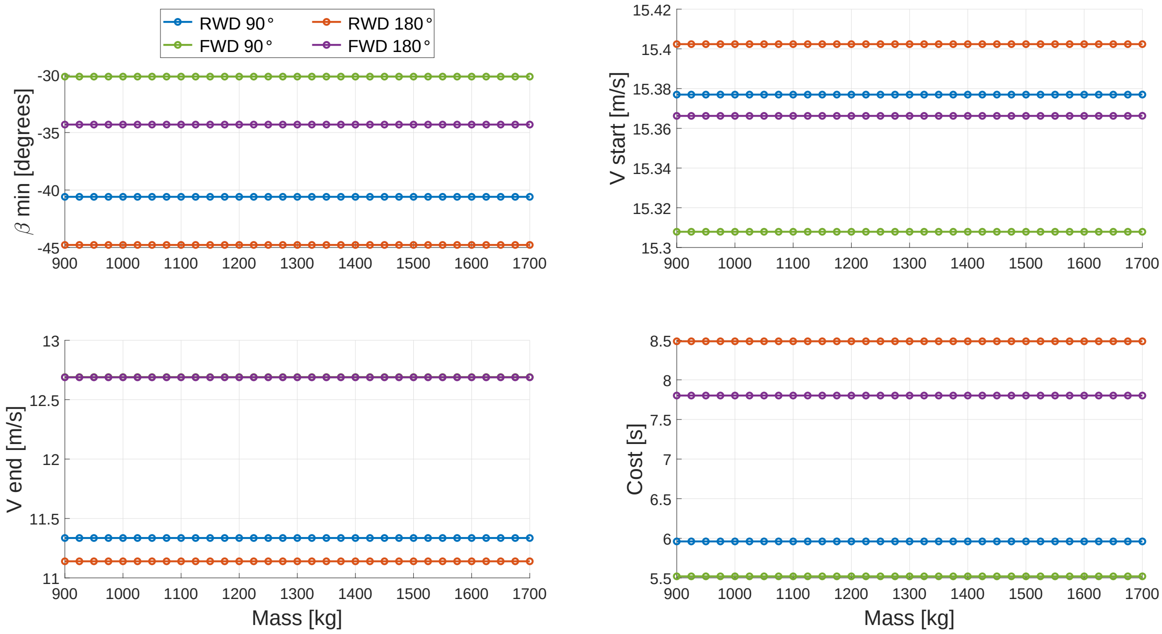


Figure H.6: Sensitivity Analysis: Tire 2, gravel – Mass [kg] (Set 2).

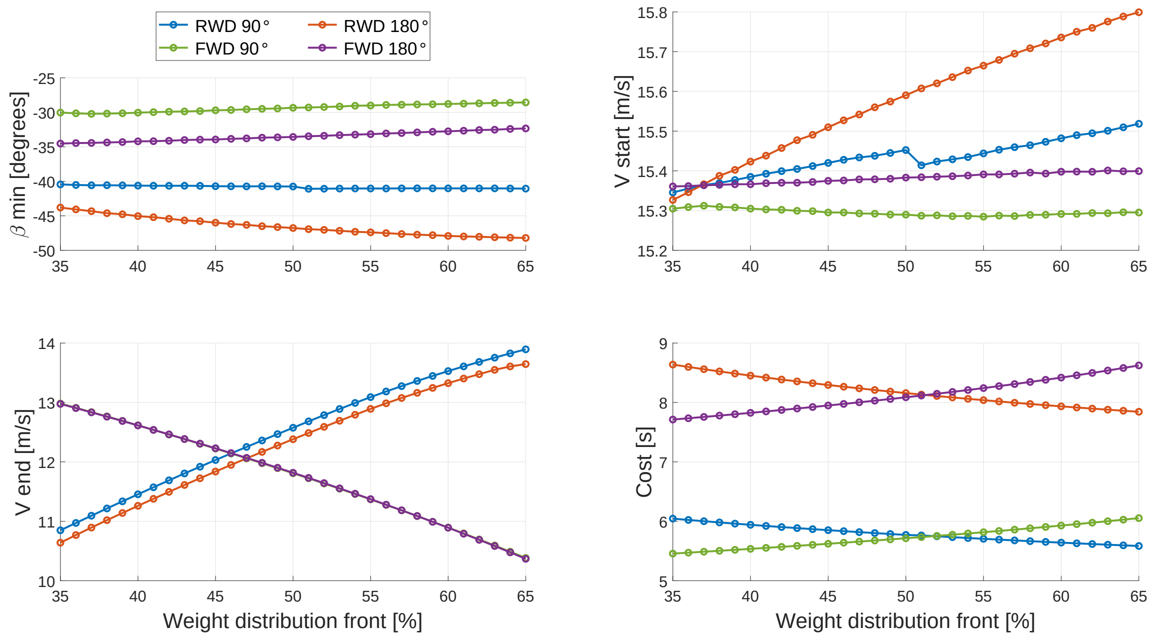


Figure H.7: Sensitivity Analysis: Tire 2, gravel – Weight distribution rear [%] (Set 4).

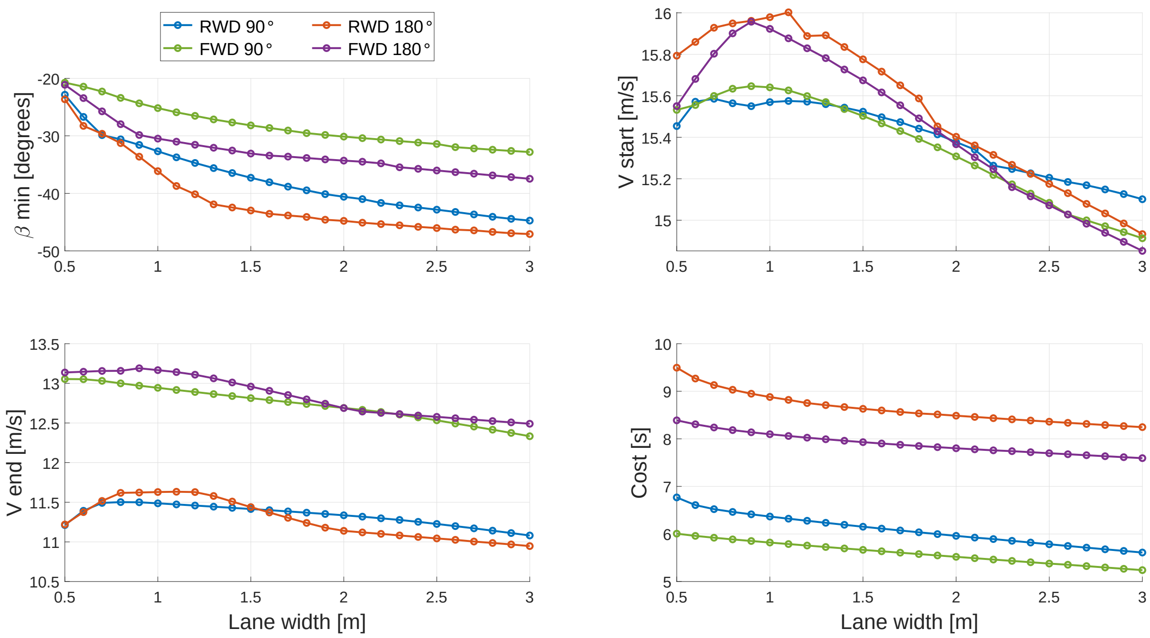


Figure H.8: Sensitivity Analysis: Tire 2, gravel – Lane width [m] (Set 6).

H.3 Varying tire curves

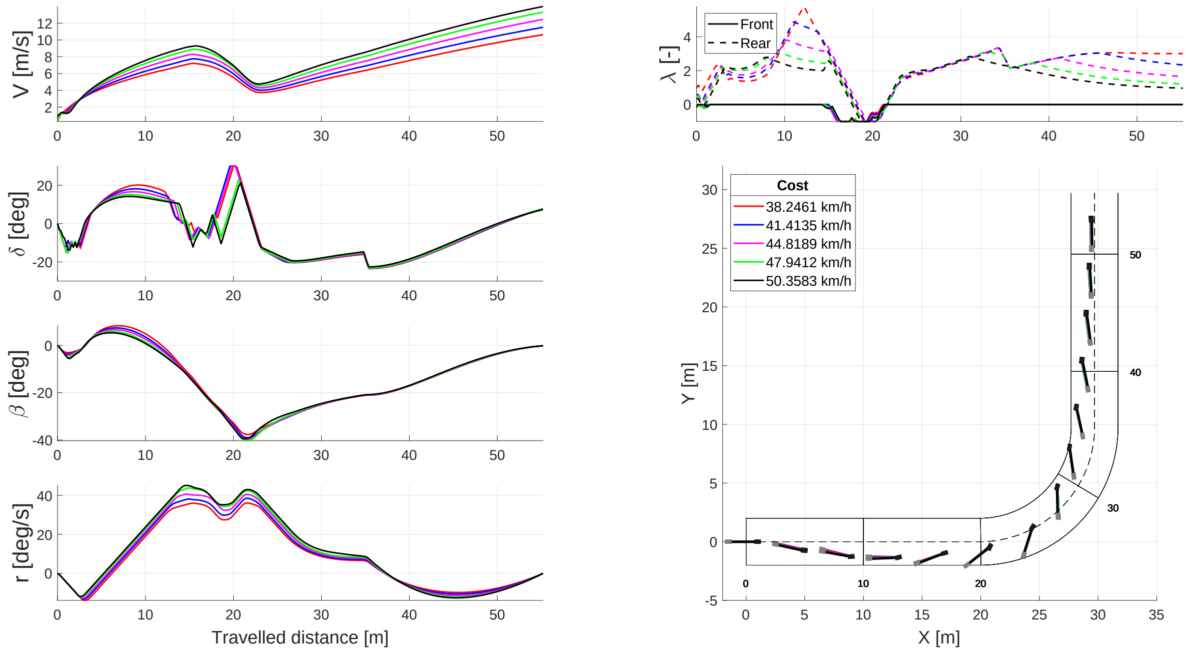


Figure H.9: Trajectories for the five tire curves shown in Figure 5.9, simulated with a MEVM objective function, and a free initial and final velocity.

Appendix I

Simulation results

I.1 Collision avoidance

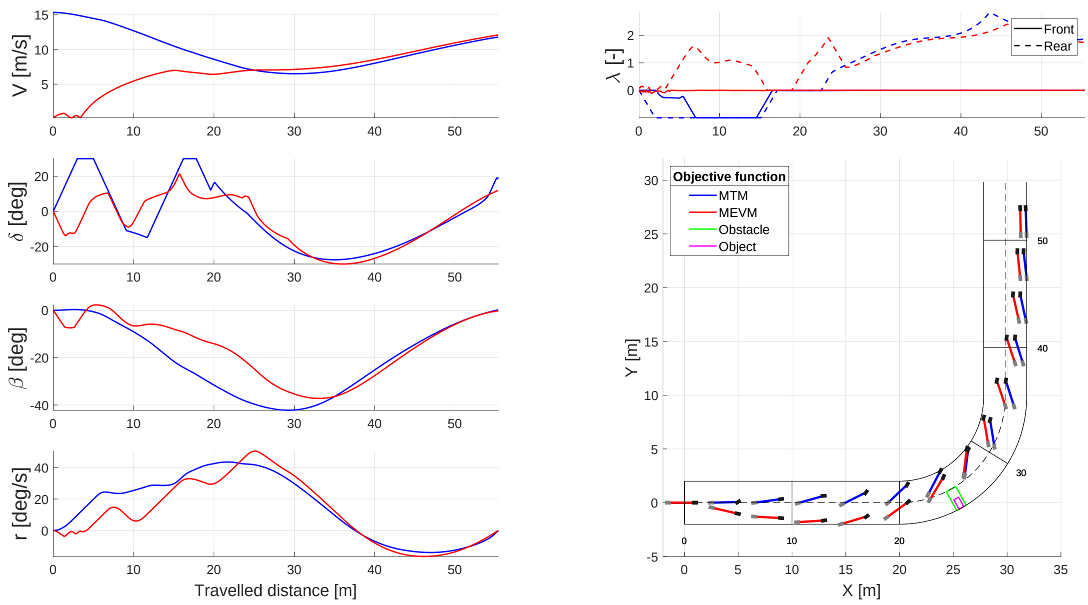


Figure I.1: Simulation of a 90° corner and an obstacle on the right side of the road, with tire 2 (gravel) for both objective functions: Minimum Time of travel Maneuver (MTM) and Maximum Exit Velocity Maneuver (MEVM).

I.2 Model constraint adaptation

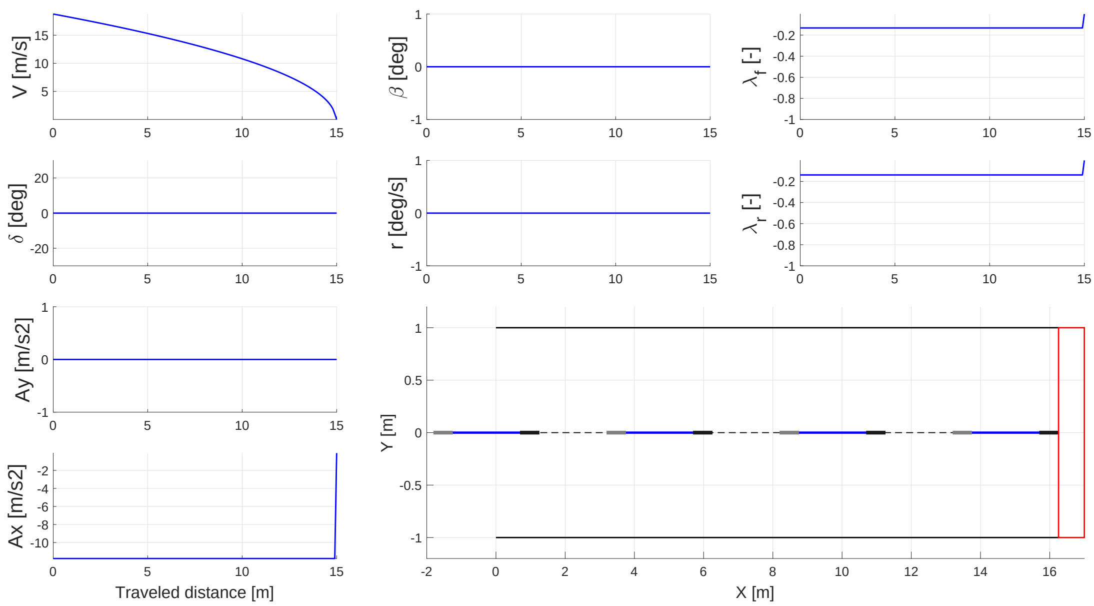


Figure I.2: Last-point-to-brake optimization: critical distance 15 $[m]$.

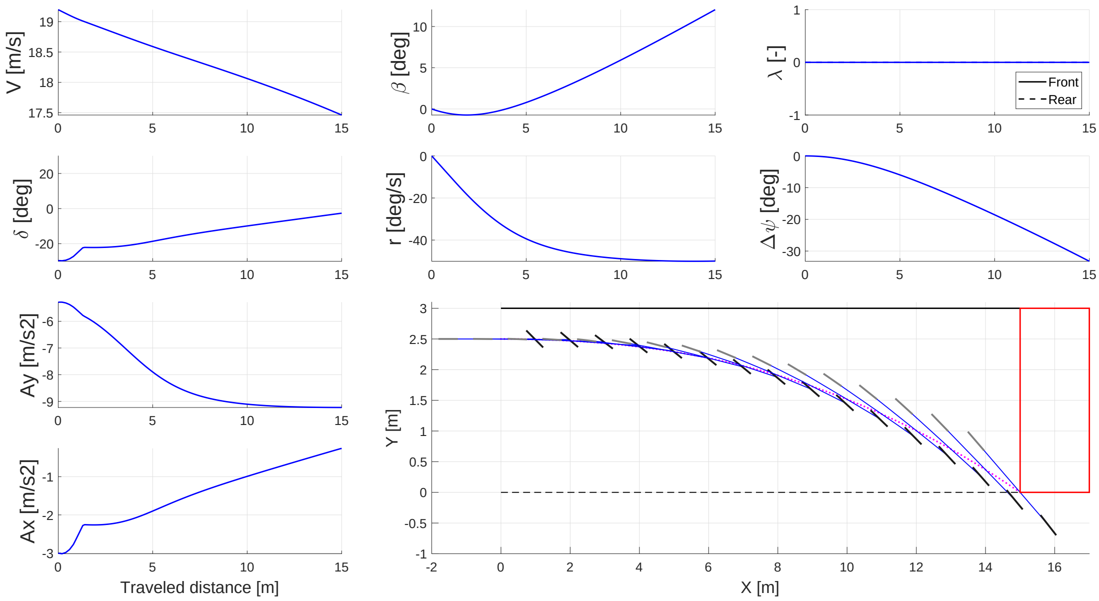


Figure I.3: Last-point-to-steer: initially steady-state cornering – critical distance 15 [m].

PO Box 513
5600 MB Eindhoven
The Netherlands
tue.nl

DEPARTMENT OF MECHANICAL ENGINEERING