Jerk Based Motion Planning

Master of Science Thesis

Nino Ayrton Vroom

Design of a jerk based motion planning algorithm capable of producing comfortable motion using non-linear optimization methods



Jerk Based Motion Planning

Master of Science Thesis

by



to obtain the degree of Master of Science at the Delft University of Technology, to be defended publicly on Tuesday July 8, 2021 at 9:00 AM.

Student number:4359577Thesis committee:Dr. Barys Shyrokau,
Dr. Riender Happee,
TU Delft
Dr. Laura FerrantiTU Delft
TU Delft

An electronic version of this thesis is available at http://repository.tudelft.nl/.





The work in this thesis was supported by Siemens Digital Industries AG. Their cooperation is hereby gratefully acknowledged.





Copyright $\ensuremath{\mathbb{O}}$ Department of Cognitive Robotics (CoR) All rights reserved.

Abstract

The concept of motion planning for a path following task is highly relevant. Having a comfortable reference motion is an advantage for automated vehicles, since this leads to more higher comfort for the passengers, in turn leading to higher adoption of ADAS. Defining a speed profile can be done in three different ways. First is a heuristics based methods, such as seen in IPG CarMaker's driver model. Secondly, line-of-sight based methods use the knowledge that blind corners can contain potential moving obstacles, leading to slower speeds to compensate for the braking distance. Finally, optimization based methods, such as laptime optimization, can be used by defining dynamics, constraints and a cost function to find the optimal motion for the given scenario. Comfort is primarily a function of acceleration and jerk. The RMS of the acceleration should be as low as possible for more comfort, while jerk lower than a determined value can be ignored from a comfort point of view. Finally, the scope and the contributions to the state of the art have been brought forward.

A curvilinear approach in combination with a distance domain approach is utilized, since this allows for only the curvature of the road to be used as a reference. To generate a speed profile longitudinal point mass dynamics are utilized with the jerk as an input to the optimization. The maximum speed is determined beforehand using the curvature of the road and assuming steady-state cornering. For each point along the road a speed is now determined that should not be exceeded. Generating a speed profile using a linear optimization can not be done accurately. However, an assumption can be made that the vehicle takes the same time for each distance step, leading to an approximate solution. The non-linear optimization is able to overcome the shortcomings of the linear optimization, by including the time as a state in the solution. By incorporating the lateral dynamics in the EOM the predetermined maximum speed is no longer needed. The lateral dynamics are simplified by assuming the vehicle is always aligned with the reference, therefore ignoring yaw dynamics. Leading to a complete description of the longitudinal and lateral motion that can be solved using a non-linear optimization method. ACADO is used for the non-linear optimization.

A benchmark is performed against a state of the art literature example based on a non-linear optimization method that is able to minimize motion sickness based on vibrations experienced by the passengers. The speed profile generation based on linear optimization is not able to reproduce the resulting speed profile as seen in the example. For the non-linear optimization method the speed profile obtained for the time optimal cost function is very close, however for the comfortable motion, the speed through the corners was still too high to be considered comfortable. By minimizing the acceleration and jerk the motion planning system including the lateral motion is able to accurately reproduce similar speed profiles for the three different cost functions of the example.

In Simcenter Prescan a complex 10 degrees of freedom vehicle model containing both sprung and unsprung mass has been presented. A driver model, based on two PD controllers for longitudinal control and the Prescan built-in path follower for lateral control, has been introduced. The motion planning containing both lateral and longitudinal motion has been used to define a comfortable reference speed and trajectory. The simulation results show that the reference motion is comparable to the resulting motion, both in the longitudinal and the lateral direction. Indicating that the motion planning system produces accurate results. On the other hand it is also observed that the longitudinal control of the vehicle needs a more complex control scheme. For the lateral control a predictive control scheme is needed to be able to follow the road with low lateral offset.

Contents

1	Intro	oduction 1
	1.1	Motivation
	1.2	An Overview of Speed Planning
		1.2.1 Speed during Steady-State Cornering
		1.2.2 Cubic Splines
		1.2.3 Third Order Jerk and Acceleration Limited
		1.2.4 Speedprofile Generation in IPG CarMaker's Driver Model
		1.2.5 Line-Of-Sight Based Speed
		1.2.6 Optimal Control Approach
		1.2.7 Laptime Optimization
		1.2.8 Comfort from Driveability
		1.2.9 Acceleration and Jerk Limits
	1.3	Scope
	1.4	Contributions
	1.5	Summary
2	Pro	nosed Motion Planning 11
-	2 1	Cartesian and Curvilinear Coordinates 11
	2.1	Distance Domain
	2.3	ACADO 13
	2.0	231 SQP 14
		2.3.2 Active Set Method
	2.4	Linear Optimization
	2.5	Non-Linear Optimization
	2.6	Motion Planning using Non-Linear Optimization
	2.7	Summary
2	Bon	25
J	3 1	Benchmark Description 25
	3.2	Kev Performance Index 27
	3.3	Speedprofile using Linear Optimization 29
	3.4	Speedprofile using Non-Linear Optimization 33
	3.5	Speedprofile using Non-Linear Optimization with Lateral Motion
	3.6	Effects of Settings
		3.6.1 Cost Function Weights
		3.6.2 Lateral Jerk Inclusion in Motion Planning
		3.6.3 Distance Step Size
		3.6.4 Timestep Variance for Linear Method
		3.6.5 Initial Conditions for Non-Linear Method and Motion Planning
	3.7	Summary
4	Con	nparison of Motion Planning Against a Complex Vehicle Model 45
•	4.1	Vehicle Model and Experiment Settings
	4.2	Kev Performance Index
	4.3	Driver Model
		4.3.1 Controller Tuning
	4.4	Scenario Motion Profile
	4.5	Simulation Results
		4.5.1 Longitudinal Results
		-

	4.6	Summary	54
5	Con	clusions and Future Work	55
	5.1	Conclusions	55
	5.2	Recommendations and Future Work	56
Α	Арр	endix	57
	A.1	Linear Speedprofile Generation Final EOM	57
	A.2	Non-Linear Speedprofile Generation Final EOM	57
	A.3	Motion Planning Final EOM	58
	A.4	Benchmark Format Results	59
	A.5	Vehicle Parameters.	60
	A.6	Path Follower Parameters	62

List of Figures

1.1 1.2 1.3 1.4	Six possible speed profiles using the cubic splines method from [4]	3 3 4
1.5	improves the vehicle accelerates [37]	4 5
2.1 2.2 2.3	A path defined as a series of points	11 12
	aligned with the reference (black line)	21
3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 3.10 3.12 3.12 3.13 3.14 3.15	The three trajectories used in [16]	26 27 27 29 30 31 32 33 34 34 34 36 37 37
3.16	The resulting speedprofiles using the proposed motion planning method in the bench- mark format	39
3.17	The speedprofile including lateral jerk using the motion planning method	40
3.19	Acceleration and jerk for the linear method as determined by the algorithm (Input) and calculated afterwards (Real). Assuming a speed of $40m/s$ (left) and $10m/s$ (right)	42
4.1	calculated afterwards (Real). Assuming a speed of $40m/s$ (left) and $10m/s$ (right) The 10 degrees of freedom of the vehicle dynamics model. Source: Prescan help quide	42 45
4.2	The error defined by the path follower from Prescan. Source: Prescan help guide	49
4.3	The road in Prescan	50
4.4 45	The speedprofile as a function of distance used as the reference for the simulation	51
4.6	The lateral offset as a function of color over the resulting trajectory	51
4.7	The resulting speed over the trajectory	52
4.8	The resulting longitudinal acceleration	52
4.9	The resulting longitudinal jerk	52
4.10	The longitudinal control inputs as a function of distance	52
4.11	The resulting lateral offset compared to the centerline of the road	53
4.12	The resulting lateral acceleration	53
4.13	The resulting lateral jerk	53
4.14	The steering wheel angle as a function of distance	53

= 0
58
59
59
59
60
60
61
61
62

Introduction

In the automotive world self-driving vehicles and driving assistance systems are becoming more prevalent. Many new cars are equipped with advanced driver assistance systems (ADAS), such as adaptive cruise control (ACC) and lane keep assist (LKA). These systems aim to improve safety for the occupants by helping the driver with the driving task. Widespread adoption of ADAS can therefore improve safety on the road [26], not only for the vehicles that are equipped with these systems, but also the other traffic participants, such as pedestrians and motorcycles. Acceptance of systems such as ACC is higher than standard cruise control [28]

However, these systems will only help improve safety if they are actually used. The adoption of safety systems will only happen if the users are willing to use it. Therefore comfort has to be an important consideration in the design phase of the ADAS [12].

In the current interpretation of ADAS the control is usually divided into two separate systems that aim to control the vehicle in parallel [30]. The main example for this is the combination of ACC and LKA, which allows the driver to give control of the throttle and brakes to the ACC and the steering wheel to the LKA. This is especially the case for longitudinal control for a vehicle. Where the control action is not a result of a single integrated control system, but it is a combination of many different systems either working together or overwriting the control input in specific situations. ADAS such as ACC have a focus on comfort while maintaining safety. On the other hand, safety systems such as active emergency braking (AEB), ABS and electronic stability control are focused on safety, while comfort has no priority in system design. Many studies that focus on path following do so by decoupling the longitudinal control from the lateral control. According to a survey on control schemes for automated highway driving the control task is often partitioned into longitudinal and lateral control [30]. For lateral path following many different methods are viable methods to accomplish decent results, even the simplest methods can perform well in common scenarios. However a single type of control strategy used can not obtain perfect results for each situation [33]. In low lateral acceleration scenarios with constant longitudinal speeds such as during vehicle following on highway-like tracks a linear feedback controller with feedforward action shows good results [32].

This research will focus on the longitudinal part of path following and more specifically, building reference speed profiles that can easily be followed and are able to adhere to specific requirements, such as time optimality and comfort.

This introduction chapter goes over a literature review on speed planning methods. Then the scope of the research and the contribution to the state of the art are discussed. The second chapter proposes three ways to build speedprofiles for a known path. The third chapter benchmarks the proposed motion planning method against a state of the art speed planning method based on minimization of motion sickness. The fourth chapter verifies the proposed motion planning using the complex vehicle model from Simcenter Prescan. Finally, the conclusions and recommendations for future work are presented.

1.1. Motivation

Using a linear controller such as PID means that with regards to comfort no constraints can be added to the controller. And knowing that speed following is difficult due to the highly non-linear nature of the systems involved. The search for good ways to find a reference speed that can be used during path following is an important step to good path following. When the reference speed is comfortable and easy to follow the resulting motion when a path follower is subsequently used will be better.

Speed profile generation and motion planning are interesting from a comfort point of view in that certain objectives, such as jerk minimization can be determined over a longer section than is typically possible or convenient in predictive control methods. Speed profile generation can be easily done by making use of smoothing functions based on set speed limits. However this completely ignores the effects of the lateral motion. In this research a method is developed that will result in a speed profile that can be used as a reference speed for a driver model. This method will take into account the longitudinal motion and the lateral motion ensuring that a comfortable reference is created regardless of the determined path.

The question then becomes how can this reference speed be determined and it what ways can it be described as comfortable.

1.2. An Overview of Speed Planning

This section will go over the basic methods of generating a speedprofile. First, a definition for lateral acceleration is defined that can be used to define a maximum velocity. Then, five different methods of defining a speedprofile are discussed.

1.2.1. Speed during Steady-State Cornering

Assuming steady-state cornering an upper bound for the speed can be chosen. By choosing a value for the lateral acceleration and using the curvature of the road the speed can be defined.

$$u_{max} = \sqrt{a_y \cdot R} \tag{1.1}$$

with a_y the maximum allowed lateral acceleration and *R* the radius of curvature. When the desired path is known, the curvature can be determined for every point, a comfortable lateral acceleration can be chosen and the maximum longitudinal velocity is then defined [7]. However, if the curvature is not smooth, the resulting speedprofile will not be smooth. The non-smooth speedprofile can still be used as a reference.

Extending this idea, instead of using just the lateral acceleration, the combined longitudinal and lateral acceleration, a comfort parameter, can be used. This is done as follows:

$$a_{w} = \sqrt{n_{x} \cdot a_{x}^{2} + n_{y} \cdot a_{y}^{2} + n_{z} \cdot a_{z}^{2}}$$
(1.2)

Where a_i are the accelerations in three directions with n_i as their respective weights. The lateral acceleration a_y is replaced by the comfort parameter a_w in 1.1. This results in a speedprofile based on this comfort parameter instead of just the lateral acceleration.

1.2.2. Cubic Splines

Creating a speedprofile with continuous velocities and accelerations and a jerk minimization can be achieved by interpolating cubic splines [4]. Possible speed profiles that are generated are seen in Figure 1.1. The generated speed profiles are able to adhere to boundary conditions such as trajectory length and starting velocity. Since the profiles are generated using a third order polynomial, the jerk can be taken into account. This approach is used in combination with an optimization algorithm that iteratively tries to find suitable profiles with the lowest absolute jerk that adhere to the boundary conditions. Resulting in minimal absolute jerk trajectories that are computationally simple due to their algebraic nature. The main drawback lies in the fact that there is no limit on the amount of acceleration that is allowed. This means that as long as the absolute jerk is low the acceleration is uncapped which can be less than ideal for comfort.



Figure 1.1: Six possible speed profiles using the cubic splines method from [4]

1.2.3. Third Order Jerk and Acceleration Limited

To combat the problem of uncapped accelerations a different strategy can be employed [13]. The speedprofile is divided into different sections: The acceleration phase and the cruising phase. The acceleration phase is comprised by using the maximum jerk until the maximum acceleration is achieved. Then the maximum jerk (but in the opposite direction) is used to reduce the acceleration. The vehicle is now moving with a constant speed, called the cruising phase. To slow down the vehicle the acceleration phase is used again, but in reverse order. This can be seen graphically in Figure 1.2. Because the algorithm uses maximum jerk and acceleration as much as possible it is time-optimal while adhering to maximum velocity constraints. The profile can be computed within 0.5ms meaning that it is also a good candidate for online generation.



Figure 1.2: A third order trajectory with jerk and acceleration limit [13]

1.2.4. Speedprofile Generation in IPG CarMaker's Driver Model

IPG CarMaker is an example of vehicle simulation software which contains a driver model with an integrated speedprofile. The first step in determining the speedprofile is with the help of the cruising speed (V_{cruising}), the maximum lateral acceleration and the curvature (k). The cornering speed is determined from the curvature and the lateral acceleration as seen earlier. The desired speed is the minimum of the cruising speed and the cornering speed. the second step is to reduce the slopes of the speedprofile to make sure the maximum longitudinal acceleration/deceleration is not exceeded. The final step is to smooth out the speedprofile to make sure the desired jerk does not go to infinity. This step also includes a plausibility test to smooth out or entirely remove unnecessary peaks. These three steps can be seen in Figure 1.3.



Figure 1.3: Generating a speedprofile in 3 steps as per IPG

1.2.5. Line-Of-Sight Based Speed

A complete different way to generate a reference speed can be done on the basis of line-of-sight and potential hazards. When a driver comes up to an intersection that they are unable to overlook they will naturally slow down. To model this the controller needs to have information not only about the field of view, but also about the possible moving obstacles. The desired speed is determined by first assuming an obstacle is right outside of field of view. This is illustrated in Figure 1.4. The stopping distance of the vehicle is determined by the speed. The maximum speed is then chosen to be the speed at which the vehicle is able to stop if an obstacle came around the corner. This leads to a speedprofile where the vehicle will start to slow down coming to an intersection and then when getting closer to the blind corner it will slowly start to speed up as the field of view increases. This behaviour can be modeled and used to create a more comfortable driver model [31]. This method can also be used to create a controller that is able to closely resemble human driving in urban environments with bad visibility. [37]. On top of being able to have a more comfortable ride it also safer since possible collisions are avoided by slowing down to appropriate speeds.



Figure 1.4: Ego-vehicle facing a blind intersection. The vehicle anticipates potential moving obstacles coming from the blind area and slows down. When the visibility of the intersection improves the vehicle accelerates [37].

1.2.6. Optimal Control Approach

Creating a speedprofile can be done by making use of optimal control procedures [34]. The motion states x, the control u, the initial time t_0 and the final time t_f need to be found that optimize the performance index function J that measures the quality of the path.

$$J = \Phi(x, t_0, u, t_f) + \int_{t_f}^{t_0} \Lambda([x, u, t]) dt$$
 (1.3)

subject to state constraints (representing system dynamics) x that contain the distance, velocity, acceleration and jerk.

$$\dot{x} = f[x, u, t] \tag{1.4}$$

The state constraints *C* which contain the velocity, acceleration and jerk limits. These are the maximum speed defined by the lateral acceleration and curvature and the maximum longitudinal acceleration and jerk as set by comfort desires.

$$C_{min} \le C[x, u, t] \le C_{max} \tag{1.5}$$

And finally the boundary conditions

$$\phi_{min} \le \phi[x, t_0, x, t_f] \le \phi_{max} \tag{1.6}$$

An example of a method to solve such an optimal control problem can be seen in Figure 1.5.

Algorithm 1: Solving optimal control problems with		
direct methods		
Input: Initial guess with zero for all unknown nodes		
Output: State and control profiles $w.r.t$ time: x, u		
1 Initialize the solver;		
2 while Error is larger than the accepted tolerance do		
Read the current solution of x and u;		
4 Convert the problem to a constrained nonlinear		
optimization problem;		
5 Calcuate the numerical solution in a differential		
form (2) and in an integral form (1);		
6 Optimize and update the solution with the		
gradient-based method;		
7 Compute the error and update the solution;		
s end		

Figure 1.5: An example algorithm from [34]

Using this framework different performance measures can be defined: Minimum time or minimum jerk. The minimum time method results in the following performance function

$$J = t_f - t_0 \tag{1.7}$$

The minimum jerk can be defined in different ways. One way is to set the performance function as the absolute value of the jerk *j*.

$$J = \int_{t_0}^{t_f} |j| \mathrm{d}t \tag{1.8}$$

Another way is to use the minimum square of the jerk.

$$J = \int_{t_0}^{t_f} j^2 \mathrm{d}t \tag{1.9}$$

The biggest advantage lies in the fact that it is easily and clearly defined due to the performance function. It is can also be extended to use methods like non-linear MPC or neural networks to improve runtime performance. Another nice advantage is that due to the fact that there is a performance index a value can be assigned to the speedprofile and the quality of the speedprofile can be determined independent of the vehicle dynamics and control.

1.2.7. Laptime Optimization

An extension of an optimal control approach is by taking a look at laptime optimizations. Laptime optimization is used to determine the fastest way around a race track. This is usually done using very complex vehicle dynamics models to get as accurate results as possible [21]. The main drawback lies in the fact that the accuracy of the results are much more important than the computation time, therefore resulting in very long computation times that are not feasible to compute in realtime. Since a laptime optimization is based on general optimization the cost function (and/or underlying dynamics) can be altered. An altered version can be used to generate a speedprofile that is able to minimize motion sickness [16].

However, the laptime optimization systems are not confined to traditional vehicle models. A go-kart is a specific type of racing car that does not utilize a differential to drive the rear wheels. A simulation has been used to study the peculiar dynamics of go-karts and focused on the tyre slippage dynamics, which are influenced by the lack of differential [22].

Another approach is presented in [6], where the optimization is focused on the longitudinal aspect with an electric race car. Two different transmission strategies are compared to find which is the better option.

A laptime optimization can also be used to find the effects of various vehicle parameters. The impact on driven line, vehicle stability and manoeuvre time is evaluated for the different settings for the vehicle model [17]. This allows the use of the optimization to find the best set up for a vehicle without having to do real world tests.

1.2.8. Comfort from Driveability

The driveability of a vehicle is the terminology that describes vehicle responsiveness, operating smoothness and driving comfort. it evaluates the overall driver feeling under various driving conditions. The following issues are considered as driveability problems [35]:

- Hesitation or delay
- Sluggish
- Hard start
- Surge
- · Idle roughness and instability
- · Noise and oscillations

for motion planning and control systems the interesting points are surge, and noise and oscillations. Surge means that the engine power will vary under steady throttle. This is relevant in the sense that for autonomous driving variation of speeds when they are not necessary, such as moving in a straight line, should be avoided. Noise and oscillations are also obvious candidates. Interior noise level is mostly a function of the sound dampening of the chassis of a vehicle, which can not be influenced by a vehicle dynamics control system, but the engine rpm and load in an ICE vehicle do contribute. Meaning that for comfort it is desirable to keep the load and RPM as low as possible. This can also have an effect on safety since larger noise generation can decrease detectability of outside warnings and siren signals. On the other hand oscillations can create discomfort in passengers. Vibrations between 0.5 and 80 Hz are significant in exciting human body response. Horizontal vibrations between 1 and 2 Hz especially should be avoided, while for vertical vibrations between 4 and 8 Hz should be avoided. This can be interesting for roads that contain speedbumps or other regular vertical uneveness since the frequency of these vertical excitations are directly proportional to the speed of the vehicle. The speed should therefore be set such that these vertical excitations do not occur with undesired frequency.

1.2.9. Acceleration and Jerk Limits

Finding acceleration and jerk limits has been a topic of research even before automated vehicle systems were feasible [14]. This study came back inconclusive to actual limits, but it did conclude that lower acceleration and jerk lead to more a comfortable ride.

For lateral and longitudinal acceleration and jerk limits ISO 22179:2009 [1] can be referenced. According to this the maximum lateral acceleration is between $a_y = 2.0m/s^2$ and $a_y = 2.3m/s^2$ depending on the curve radius. This is derived from average driver behaviour in curves (95% of drivers). The average deceleration (over 2 seconds) shall not exceed $3.5m/s^2$ and the negative jerk shall not exceed $2.5m/s^3$ while travelling above 20m/s. While travelling below 5m/s they should not exceed $5m/s^2$ and $5m/s^3$ respectively. The average acceleration (over 2 seconds) shall not exceed $2m/s^2$ when the vehicle is travelling above 20m/s. The average acceleration should not exceed $4m/s^2$ when the vehicle is travelling below 5m/s. This can be seen in table 1.1. The transition from the low speed to the high speed values is linear, i.e., the acceleration at 12.5m/s shall not exceed $3m/s^2$.

Longitudinal Direction	Low Speed ($\leq 5m/s$)	High Speed ($\geq 20m/s$)
Acceleration (m/s^2)	4	2
Deceleration (m/s^2)	5	3.5
Negative Jerk (m/s^3)	5	2.5

Table 1.1: Longitudinal limits according to [1]

Another way to measure comfort is not by checking the acceleration and jerk limits, but by looking at the Root Mean Square (RMS) of the acceleration [19]. The acceleration RMS value calculates the average acceleration during a certain period of time. The variable \tilde{a} is the weighted vehicle acceleration in m/s^2 filtered by a band-pass filter with the bandwidth of 1 to 32 *Hz*. t_f and t_0 are the final and starting time respectively.

$$RMS = \sqrt{\int_{t_f}^{t_0} \tilde{a}^2 dt / (t_f - t_0)}$$
(1.10)

According to ISO 2631-1 1997 [2] the following acceleration RMS values based on an 8-hour exposure can be used as a comfort reference as seen in table 1.2.

Acceleration RMS Value (m/s ²)	Comfort Reaction
≤ 0.315	not uncomfortable
0.315 ~ 0.63	a little uncomfortable
0.5 ~ 1	fairly uncomfortable
0.8 ~ 1.6	uncomfortable
1.25 ~ 2.5	very uncomfortable
≥2	extremely uncomfortable

Table 1.2: Comfort reaction to vibration environments [2]

The RMS has been used in research to evaluate passenger comfort in automated vehicles [8]. The RMS gives insight into comfort, since a road with many curves may lead to a higher RMS acceleration than a road with less curves, even though the control strategy in both scenarios adheres to similar acceleration and/or jerk limits.

During the design of transportation structures the lateral jerk has to be taken into account. During a study the minimum horizontal curve radius was derived using the limit value of lateral jerk. The limit value of the lateral jerk is determined to lies somewhere between $0.3 - 0.9m/s^3$ [18]. Any jerk value lower than this limit should not influence comfort negatively.

1.3. Scope

The use of lap time optimization frameworks to obtain a comfortable reference trajectory has not been widely researched. The aim of this thesis was to gain an understanding of how motion planning can be used to improve comfort for autonomous vehicles. This research focuses on optimization methods to determine comfortable speedprofiles for known paths and motion planning combining both speed and displacement with regards to the path to further improve comfort levels. To achieve this the following research objectives are defined:

- 1. Apply a laptime optimization framework to propose comfortable speed and motion planning algorithms.
- 2. Benchmark the proposed motion planning against published results from literature.
- 3. Verify the resulting motion is feasible and accurate using the high-fidelity simulation environment of Simcenter Prescan.

This thesis does not cover motion planning based on extensive vehicle dynamics models. This has been excluded since the accuracy of the results are heavily influenced by the quality of the dynamics that are fed into the algorithm and such accurate vehicle dynamics models are not readily available. A general architecture for complete motion planning is done in four distinct steps [27]. The first step is the route planning, this is usually done by a navigation system. After, the behavioral layer is introduced. When determined what the vehicle is supposed to do the motion planning comes in to complete a description for the states necessary to accomplish this goal. Finally, the local feedback control is introduced. This thesis is confined to the motion planning step. Therefore, it is assumed that the route planning and behavioural layer have already happened. Meaning that all relevant information, such as road curvature and width is known with perfect accuracy. The goal for the proposed algorithm then becomes to complete a description that is able to be used as a reference for a driver model.

1.4. Contributions

This thesis contributes to the state of the art by bringing forward a threefold of assets:

- Jerk based motion planning: The motion and speed planning is done using the jerk as the input. This allows the algorithm to take into account jerk limits, but also jerk minimization by including the jerk in the cost function for the optimization.
- Simplified lateral dynamics: Instead of using complex lateral vehicle dynamic models that are computationally expensive, simplified lateral dynamics are proposed. The longitudinal dynamics are unaltered and can be extended. Allowing for less computationally expensive motion planning.
- 3. Optimization methods comparison: Three distinct speed profile generation methods are compared against each other. A linear and a non-linear speed planning method are compared. On top of that, these are compared to full motion planning. Therefore, additional knowledge has been gained to decide which level of complexity is most suitable for a certain application.

1.5. Summary

This chapter has introduced the concept of motion planning and shown its relevance. Having a comfortable reference motion is an advantage for automated vehicles, since this leads to more higher comfort for the passengers, in turn leading to higher adoption of ADAS. Defining a speed profile can be done in three different ways. First is a heuristics based methods, such as seen in IPG CarMaker's driver model. Secondly, line-of-sight based methods use the knowledge that blind corners can contain potential moving obstacles, leading to slower speeds to compensate for the braking distance. Finally, optimization based methods, such as laptime optimization, can be used by defining dynamics, constraints and a cost function to find the optimal motion for the given scenario. Comfort is primarily a function of acceleration and jerk. The RMS of the acceleration should be as low as possible for more comfort, while jerk lower than a determined value can be ignored from a comfort point of view. Finally, the scope and the contributions to the state of the art have been brought forward.

 \sum

Proposed Motion Planning

In this chapter three different ways of building a speedprofile are discussed. To start, two important design options are discussed, the coordinate system and the time domain against the distance domain. To solve non-linear optimization problems the software package ACADO is used. A short section is given on the workings of the SQP and the active set method that ACADO utilises. The first method to define a speed profile is based on a linear optimization using only the longitudinal dynamics, while the second one is based on a non-linear optimization. The third method takes into account the lateral motion and is therefore able to not only generate a speed profile, but also a lateral offset from the reference path.

2.1. Cartesian and Curvilinear Coordinates

Tracking the position of a vehicle with respect to the path the two most common ways to achieve this are the following. The first way is to define the path as a series of points characterised by their *X*, *Y* coordinate and θ . Where θ is the heading of the path.



Figure 2.1: A path defined as a series of points

The heading is needed to be able to define the heading error of the vehicle compared to the path. To track the position and the orientation of the vehicle using this definition of the path is usually done by taking the location of the vehicle in the global Cartesian frame of reference and every point of the path. The distance between the vehicle location and each of these points are computed. The point which has the shortest distance to the vehicle is used as the reference trajectory. The distance between two points is usually defined by taking the euclidean distance.

$$d(p,q)^2 = (p_1 - q_1)^2 + (p_2 - q_2)^2$$
(2.1)

Where *p* and *q* denote the two points, in this case the vehicle location and a single point of the path, and $d(p,q)^2$ the distance between these points squared. Taking the square instead of the actual distance reduces the computational complexity by avoiding a square root calculation for each point. This is valid because the following holds.

$$d_1^2 < d_2^2$$
 then $d_1 < d_2$ $d \in \mathbb{R}$

Therefore no square root calculation is needed to find the closest point. Another method to reduce computational complexity is by indexing the path based on distance from the start. Combining this with the fact that the vehicle is moving at a finite speed, only a small portion of the trajectory needs to be considered, significantly reducing the search space for long trajectories.

Another method of tracking the position and orientation of the vehicle is by employing the triple curvilinear coordinates (s, n, α) [23]. Where *s* is the distance along the path (longitudinal direction), *n* the distance perpendicular to the path (lateral direction) and θ the heading relative to the road. This results in a curved frame of reference that follows the curvature of the path.



Figure 2.2: A visual representation of curvilinear coordinates

To track the position and the orientation of the vehicle using the curvilinear method the location of the vehicle in the global frame of reference is no longer sufficient to determine the relative position. The curvilinear coordinates only define the trajectory with regards to the start of the path. This means that the progression of the coordinates needs to be taken into account. The vehicle speed is related to the curvilinear coordinates according to the following differential relations.

$$\frac{d}{dt}s = \frac{u\cos\alpha - v\sin\alpha}{1 - n\kappa(s)}$$
(2.2)

$$\frac{d}{dt}n = u\sin\alpha + v\cos\alpha \tag{2.3}$$

$$\frac{d}{dt}\alpha = r - \frac{u\cos\alpha - v\sin\alpha}{1 - n\kappa(s)}\kappa(s)$$
(2.4)

Where $\kappa(s)$ is the curvature of the road as a function of the distance. Assuming the path is flat and contained in the *xy*-plane. The curvature of the road can be determined from the Cartesian coordinates as a function of the travelled distance as follows:

$$\kappa(s) = \sqrt{\left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2}$$
(2.5)

The main difference between the two discussed methods lies in the fact that for the curvilinear approach an integration of velocities is needed. For the cartesian vehicle tracking no prior location or orientation information is needed to find the current location and orientation of the vehicle with respect to the path. Another advantage of using curvilinear coordinates is that it is easier to identify longitudinal and lateral kinematics. This is due to the fact that when the heading error is zero, the lateral and longitudinal motions decouple. This can be seen when taking equations 2.2, 2.3, 2.4 and applying $\alpha = 0$. The differential relation of the longitudinal position *s* is no longer a function of the lateral speed *v* and the lateral position *n* is no longer a function of the longitudinal speed *u*.

$$\frac{d}{dt}s = \frac{u\cos\alpha}{1 - n\kappa(s)} \tag{2.6}$$

$$\frac{d}{dt}n = v\cos\alpha \tag{2.7}$$

Motion planning is done with a fixed starting position. Therefore the ability of cartesian coordinates being able to be used regardless of starting position is no longer relevant. Combining this with that fact that for curvilinear coordinates only a curvature reference is needed means that the curvilinear approach is the preferred method of position and orientation tracking for motion planning of a vehicle.

2.2. Distance Domain

Motion planning is a method to find a sequence of valid configurations that moves the object from the initial point to the endpoint. In this context it is used to find the locations and velocities needed to obtain a suitable trajectory to get from the starting position to the final position. As the final value of the time variable *t* is clearly undefined, while the curvilinear distance *s* varies from between fixed initial point s = 0 and endpoint s = S, it is convenient to mathematically formulate the problem in terms of the independent variable *s* instead of *t*. Taking the time variable *t* as the free variable is referred to as the time domain, while taking the distance variable *s* is referred to as the distance domain. The combination of using the distance domain and the curvilinear approach is one often found in laptime optimization methods [6], [17], [21], [22].

Vehicle dynamics are generally defined as a set of differential equations in the time domain. To be able to use the vehicle dynamics in the distance domain the equations of motion need to be altered. The definition of the time derivative can be combined with the curvilinear approach to obtain a suitable transformation. This allows the (vehicle) dynamics to be defined in the time domain and still be able to be used in a distance domain framework.

$$\dot{x} = \frac{dx}{dt} \tag{2.8}$$

$$\dot{x} \cdot \frac{dt}{ds} = \frac{dx}{dt} \cdot \frac{dt}{ds} = \frac{dx}{ds}$$
(2.9)

The definition for the change of states per distance step $\frac{dx}{ds}$ is defined using eq. 2.2. This approach immediately introduces a problem. Due to a division two distinct boundary conditions arise.

$$u\cos\alpha - v\sin\alpha \neq 0 \tag{2.10}$$

$$1 - n\kappa(s) \neq 0 \tag{2.11}$$

The first condition implies that the combined speed of the vehicle can not go to zero, meaning that the vehicle has to keep moving (not necessarily only in the longitudinal direction). Moving only perpendicular to the direction of the path is also not permitted. These boundary conditions make intuïtive sense in the way that either not moving or moving perpendicular to the path means that the time between the current distance step and the next goes to infinity and therefore is not computable. The second condition implies that the resulting motion can not pass the instant centre of rotation. At that exact point the previous, current and next distance steps can no longer be disambiguated and the dynamics in the distance domain fall apart.

Another advantage lies in the results that are obtained. The lateral offset and the speed are a function of the distance, not the time, therefore it can easily be mapped onto the original path. If a part of the resulting motion is no longer correct, a time based approach no longer works, while for a distance domain approach the bad section can be easily removed and the rest of the motion could potentially still be used. A bad section can result from changes in the road network, such as road works or data corruption.

2.3. ACADO

ACADO is used for the nonlinear optimization. [15] ACADO for Matlab is a Matlab interface for ACADO Toolkit. It brings the ACADO Integrators and algorithms for direct optimal control, model predictive control and parameter estimation to Matlab. ACADO for Matlab uses the ACADO Toolkit C++ code base and implements methods to communicate with this code base. The freely availabe toolkit ACADO has been show to solve multi-objective optimal control problems appearing in chemical engineering [20]. ACADO allows the user to explicitly define the equations of motion and then convert the system to an optimal control problem. Meaning that the equations of motion (discrete or continuous), bounds and cost function are directly implemented using ACADO for Matlab.

2.3.1. SQP

ACADO works using Sequential Quadratic programming (SQP). SQP type algorithms have been in use for a long time, as early as 1963 [36]. SQP methods aim at solving nonlinear optimization problems (NLP) by using a linear-quadratic approximation of the original problem in each iteration [5]. This is done by using newtons method on a series of problems with quadratic cost functions with linear constraints, a so-called quadratic program (QP). A QP can be solved efficiently using dedicated solvers. The QP solver used by ACADO is the Active Set Method via the software qpOASES [10].

Consider a general NLP problem of the form as discussed before.

$$\min_{x} f(x)$$

s.t. $h(x) = 0$
 $g(x) \le 0$ (2.12)

where $f(x) \in \mathbb{R}$ is the objective function to be minimized, $h(x) \in \mathbb{R}^m$ the equality constraints and $g(x) \in \mathbb{R}^p$ the inequality constraints.

To convert the NLP into a QP problem, the first step is to define a Lagrange function L as the objective function of the OCP. Using Lagrange multipliers Lambda and mu, the Lagrange function is defined as follows:

$$L = f(x) + \lambda h(x) + \mu g(x)$$
(2.13)

The function L couples the constraints and objective function in one formulation. Taking the Jacobian of the Lagrange function gives directly the first KKT condition as follows:

$$\nabla L = \nabla f(x) + \lambda \nabla h(x) + \mu \nabla g(x) = 0$$
(2.14)

The NLP can be converted into a QP in which the cost function is quadratic and the constraints are linear functions. To achieve such a form, Taylor expansion is utilized. This step also shows the need for a twice continuously differentiable cost function. The taylor expansion results in the following equations:

$$L \approx L + \nabla L(x - x^{k}) + \frac{1}{2}(x - x^{k})HL(x - x^{k})$$
(2.15)

$$h(x) \approx h(x^k) + \nabla h(x^k)(x - x^k)$$
(2.16)

$$g(x) \approx g(x^k) + \nabla g(x^k)(x - x^x)$$
(2.17)

Where H is the hessian matrix of the cost function and one of the main computationally expensive steps of the algorithm. The standard ACADO method is used, which is a method called the Gaus-Newton algorithm. To simplify the equations $d(x) = x - x^k$ is defined. Resulting in a QP problem to find the next iterate for the SQP problem.

$$\min_{d(x)} \quad \nabla Ld(x) + \frac{1}{2}d(x)^{T}HLd(x)$$

s.t.
$$h(x^{k}) + \nabla h(x^{k})d(x) = 0$$
$$g(x^{k}) + \nabla g(x^{k})d(x) \le 0$$
 (2.18)

However, a QP problem containing inequality constraints is a nontrivial problem and is solved using the Active Set Method (ASM) via the software qpOASES which is embedded in the ACADO software package.

2.3.2. Active Set Method

The strategy for ASM is to start from an arbitrary point and then find the next iterate by setting

$$x^{k+1} = x^k + \alpha_k d^k \tag{2.19}$$

Where α_k is the step length and d^k is the search direction. The problem then becomes to determine the search direction and the step length.

First the search direction is determined. At the current iterate x^k the index set of active inequality constraints is determined.

$$\mathcal{A}^{k} = \{ j \mid g_{j}(x^{k}) + \nabla g_{j}^{T}(x)x^{k} = 0, j = 1, \dots, p \}$$
(2.20)

The goal is to rewrite the system to an equality constrained QP. Reason being that the solution process to solve equality constrained QP is linear. First step is to combine equation 2.18 and 2.19. Leading to the following optimization problem.

$$\min_{d^{k}} \quad \nabla L(x^{k} + d^{k}) + \frac{1}{2}(x^{k} + d^{k})^{T} H L(x^{k} + d^{k})$$

s.t. $h(x^{k}) + \nabla h(x^{k})(x^{k} + d^{k}) = 0$
 $g(x^{k}) + \nabla g(x^{k})(x^{k} + d^{k}) = 0, j \in \mathcal{A}^{k}$ (2.21)

Expanding this problem in accordance with [11] the final form of the QP problem is obtained.

г

п

$$\min_{d^{k}} \quad (g^{k})^{T} d^{k} + \frac{1}{2} (d^{k})^{T} H L d^{k}$$

$$s.t. \quad \nabla h(x^{k}) d^{k} = 0$$

$$\nabla \tilde{q}(x^{k}) d^{k} = 0$$
(2.22)

Where g^k and $\tilde{g}(x)$ are defined as follows:

$$g^{k} = \left[HLx^{k} + \nabla L\right] \tag{2.23}$$

$$\tilde{g}(x) = \begin{bmatrix} \ddots \\ \nabla g_j^T(x) \\ \vdots \end{bmatrix}, \ j \in \mathcal{A}^k$$
(2.24)

(2.25)

(0, 0, 7)

Leading to the KKT optimality conditions:

$$HLd^{k} + g^{k} + \nabla h(x)^{T}\lambda^{k} + \nabla \tilde{g}(x)^{T}\mu^{k} = 0$$
(2.26)

$$\nabla h(x)d^k = 0 \tag{2.27}$$

$$7\tilde{g}(x)d^{\kappa} = 0 \tag{2.28}$$

Where λ and μ are Lagrange multipliers corresponding to equality and active inequality constraints respectively. The search direction is obtained by solving the QP such that the KKT optimality conditions are satisfied. Algorithms for equality constrained QP can now be applied to obtain d^k . If d^k is a solution of QP then there are λ and μ such that the KKT optimality condition holds. Solving this QP for d^k leads to two different cases. First the case for when $d^k \neq 0$. Then a step length α_k is needed that guarantees $x^{k} + \alpha_{k}d^{k}$ is feasible. A common α_{k} that guarantees the satisfaction of all constraints as elaborated in [11] is the following:

$$\alpha_k = \min\left\{1, \frac{b_j - b_j^T d^k}{b_j^T d^k} | j \notin \mathcal{A}^k \text{ and } b_j^T d^k > 0\right\}$$
(2.29)

Finally, if $\alpha = 1$ the active index set is not updated.

$$\mathcal{A}^{k+1} = \mathcal{A}^k \tag{2.30}$$

And if $\alpha_k < 1$ the active index set is updated to remove the active inequality constraint as follows:

$$\mathcal{A}^{k+1} = \mathcal{A}^k \cup \{J_0\} \tag{2.31}$$

The ASM then starts a new iteration using the new active index set.

The other case arises when $d^k = 0$. When $d^k = 0$ the system reduces to the following:

$$g^k + A^T \lambda^k + B^T \mu^k = 0 \tag{2.32}$$

In this case, two subcases can be identified. If $\mu^k \ge 0$ the problem simplifies to the original problem. Therefore, if $d^k = 0$ and $\mu_k \ge 0$, $x^{k+1} = x^k$ is a KKT point and the problem has converged and the optimal solution is found.

In the case that some components of μ^k are negative, then x^k is not the optimal solution. An assumption is done $\mu_{j0} = \min\{\mu_j | \mu_j < 0, j \in \mathcal{A}^k\}$. Then the index J_0 is removed from \mathcal{A}^k and the previous steps are repeated to determine d^k and μ_k , until the optimal solution has been found.

In short, an active set based on inequality constraints is defined. Then the inequality constraints are converted to equality constraints and this modified QP is solved to find the search direction d^k .

2.4. Linear Optimization

In this section a linear optimal control approach to determine a speedprofile will be proposed. By utilising a linear approach based on a quadratic cost function, the optimization can be quickly performed. The curvature of the path is assumed to be known for equidistant (*h*) points along the path. Using the curvature and an allowed lateral acceleration, the maximum speed can be determined using equation 1.1. Additionally a speed limit ($u_{speedlimit}$) is chosen for the low curvature (straight) sections. An upper limit for the longitudinal speed at each point of the path is now defined.

$$u_{max,s} = min(u_{speedlimit}, u_{curvature,s})$$
(2.33)

The optimal control procedure can now be defined, starting with the states that are to be optimized. The state vector \mathbf{x} consists of the speed u, acceleration a and jerk J for each distance step. For an optimization of length i the state vector is of length 3i.

$$\mathbf{x} = [u_1, a_1, J_1, \dots, u_i, a_i, J_i]^T$$
(2.34)

The cost function is defined as the sum of the differences between the states and a set reference. The reference for the speed is as seen in eq 2.33, while the reference for the acceleration and jerk are set to zero. By utilising a zero reference minimization for the acceleration and jerk can be achieved. This leads to the following cost function:

$$\mathcal{L} = \sum_{s=1}^{l} v_c \cdot (u_{max,s} - u_s)^2 + a_c \cdot a_s^2 + J_c \cdot J_s^2$$
(2.35)

Where v_c , a_c , J_c are the cost variables for the different states.

Point mass dynamics are incorporated to ensure proper cohesion between the different states. The acceleration is defined as a first order discrete derivative of the speed, while the jerk is defined as a second order discrete derivative of the speed. The dynamics are employed as follows:

$$dt \cdot a_s = v_s - v_{s-1} \tag{2.36}$$

$$dt^2 \cdot J_s = v_{s+1} + v_{s-1} - 2 \cdot v_s \tag{2.37}$$

Since the timestep dt varies for each distance index due to the speed not being constant

Up to this point the system was linear and could be described using a linear set of equations. However, the timestep dt can vary for each distance index since the speed is not constant. This leads to nonlinearity. To simplify and make the problem linear, dt can be assumed to be constant. Using the speedlimit $u_{speedlimit}$ for the timestep leads to the following definition.

$$dt = \frac{h}{u_{speedlimit}}$$
(2.38)

When dt is defined using the speed limit, the real timestep is always longer than the assumed one. This is due to the boundary condition that the speed should never exceed the speedlimit. Since $u_{speedlimit} \ge u_s$, and thus $dt_{real} \le dt$, therefore the calculated acceleration and jerk are always larger than the real acceleration and jerk. Defining bounds on the calculated acceleration and jerk means that the real acceleration and jerk will always be within the defined bounds. The speed constraint is defined based on a set minimum speed, larger than zero, and the previously defined maximum speed. The acceleration and jerk constraints can be freely defined.

$$u_{min} \le u_s \le u_{max,s} \tag{2.39}$$

$$a_{min} \le a_s \le a_{max} \tag{2.40}$$

$$J_{min} \le J_s \le J_{max} \tag{2.41}$$

To solve an optimal control problem of this form the built-in Matlab command quadprog will be considered. This solver takes input in the form of eq 2.42 and returns the optimized set of states \mathbf{x} . Due to the inherent limitations only the speed at each point will be used to evaluate the performance.

The acceleration and jerk will be regarded as intermediate states that can give insight into limitations that arise.

$$\min_{\mathbf{x}} \quad \frac{1}{2} \mathbf{x}^T H \mathbf{x} - f^T \mathbf{x} \quad \text{such that} \begin{cases} A \cdot \mathbf{x} \le b \\ A_{eq} \cdot \mathbf{x} = b_{eq} \\ lb \le \mathbf{x} \le ub \end{cases} \tag{2.42}$$

Converting eq 2.35 to the needed form the following matrices are found. The matrix *H* is defined as a square symmetric matrix of size 3i containing the cost parameters. The matrix f^T is defined using the maximum speed that is determined in equation 2.33 and is of size 3i.

$$H = \begin{bmatrix} v_c & 0 & 0 & & & \\ 0 & a_c & 0 & & 0 & \\ 0 & 0 & J_c & & & \\ & & \ddots & & & \\ & & v_c & 0 & 0 & \\ 0 & 0 & a_c & 0 & \\ 0 & 0 & 0 & J_c \end{bmatrix}$$

$$f^T = \begin{bmatrix} u_{max,1}, 0, 0, \dots, u_{max,i}, 0, 0 \end{bmatrix}$$
(2.44)

To take into account the relation between the speed, acceleration and jerk the equality constraints are used. Having the equality constraints as a linear function significantly reduces the computational complexity and is a prerequisite to using this specific solver. The equality matrix A_{eq} is of size $2i \times 3i$ and corresponding equality vector b_{eq} of size 2i are defined as follows:

$$\mathbf{T} = \begin{pmatrix} -1 & -dt & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 2 & 0 & dt^2 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -dt & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & 0 & dt^2 \end{pmatrix}$$
(2.45)

$$A_{eq} = \begin{bmatrix} \mathbf{T} & 0 \\ & \ddots \\ 0 & \mathbf{T} \end{bmatrix}$$
(2.46)

$$b_{eq} = [0, \dots, 0]^T \tag{2.47}$$

т

The inequality constraint matrices A and b are not utilised int his implementation. Finally, the speed, acceleration and jerk bounds are included as the following lower and upper bound matrices lb, ub:

$$lb = [u_{min}, a_{min}, J_{min}, \dots, u_{min}, a_{min}, J_{min}]^{T}$$
(2.48)

$$ub = \left[u_{max,1}, a_{max}, J_{max}, \dots, u_{max,i}, a_{max}, J_{max}\right]^{\prime}$$
(2.49)

Note that the lower bound the values are constant across the entire distance, while for the upper bound the speed limit is a function of the distance index.

2.5. Non-Linear Optimization

To combat the shortcomings of the linear method a non-linear method is proposed. ACADO for Matlab has been used for the implementation. The non-linear method can be seen as an extension of the linear variant, where the states are extended to include the time t. Leading to the following state vector **x** and control input **u**:

$$\mathbf{x} = \left[u, a, t\right]^T \tag{2.50}$$

$$\mathbf{u} = J \tag{2.51}$$

Note that the state vector and control input are determined for each distance step.

This implementation allows for more complex cost functions. Common cost functions in literature are time optimality, minimization of accelerations or a combination of both. This results in the following cost functions:

$$\mathcal{L} = t \tag{2.52}$$

$$\mathcal{L} = w \cdot t + \int_0^S a_s^2 ds \tag{2.53}$$

$$\mathcal{L} = \int_0^S a_s^2 ds \tag{2.54}$$

Where *S* is the total length of the path and *w* is a weight factor that determines the trade-off between the time and acceleration part. Since increasing the speed as much as possible leads to a minimization of both the cost function in eq 2.52 and eq 2.35, with a_c , J_c set to zero, it can be concluded that they are equivalent.

The next step is to define the dynamics of the system. Discrete point mass dynamics are employed that lead to the following definitions for speed u, acceleration a and time passed t:

$$u_{s+1} = u_s + a_s \cdot dt_s \tag{2.55}$$

$$a_{s+1} = a_s + J_s \cdot dt_s \tag{2.56}$$

$$t_{s+1} = t_s + dt_s (2.57)$$

All previous presented dynamics are linear as long as the timestep dt is constant. However, in this implementation the timestep is variable for each distance step, leading to the definition of the timestep:

$$dt_s = \frac{h}{u_s} \tag{2.58}$$

Due to the fact that the timestep is a function of the speed, the dynamics can no longer be described as a linear set of equations. Finally, state constraints are introduced as bounds on the speed, acceleration and jerk. The maximum speed $u_{max,s}$ is determined according to eq 2.33

$$u_{min} \le u_s \le u_{max,s} \tag{2.59}$$

$$a_{min} \le a_s \le a_{max} \tag{2.60}$$

$$J_{min} \le J_s \le J_{max} \tag{2.61}$$

2.6. Motion Planning using Non-Linear Optimization

By including the ability to deviate from the predetermined path the configuration space is extended, allowing for more flexibility for the optimization to find a suitable motion. The equations of motion are defined in multiple steps. Starting with the necessary assumptions and the states and control inputs. Once both are defined, the equations of motion are derived.

The following three assumptions are made.

- 1. The equations of motion are defined in distance domain, not time domain with a curvilinear coordinates system using the curvature of the path as the only reference.
- 2. All dynamics are discrete.
- 3. The actor is always aligned with the reference, therefore the yaw dynamics are ignored.

The states for the EOM are comprised of the position, velocity, acceleration and global information. The position information is defined using the lateral offset and the distance along the path, which is the free variable in the distance domain. Therefore the position information is defined using only the lateral offset (as a function of the distance).

• Lateral Offset (n)

The velocity information is defined using the longitudinal and lateral velocity.

- Longitudinal Velocity (u)
- Lateral Velocity (v)

The acceleration information is similarly defined using the longitudinal and lateral acceleration.

- Longitudinal Acceleration (a_x)
- Lateral Acceleration (a_v)

The global information is comprised of the time and the global heading. The time is needed to be able to define a minimum time cost function, while the global heading is used to check the integration of the reference curvature.

- Time (*t*)
- Global Heading (θ)

The control input is in the form of the longitudinal and lateral jerk.

- Longitudinal Jerk (J_x)
- Lateral Jerk (J_{ν})

The state vector \mathbf{x} and the control input \mathbf{u} for each distance step become the following:

 $\mathbf{x} = \left[n, u, v, a_x, a_y, t, \theta\right]^T$ (2.62)

$$\mathbf{u} = \left[J_{x}, J_{y}\right]^{T} \tag{2.63}$$

The length of the distance step is *h*. The angle between the previous and the current step is represented by ϕ . The lateral offset at the current step is n_s and the lateral offset at the next step is n_{s+1} . A small angle approximation is used, this means that it is assumed that $cos(\phi) = 1$ and $sin(\phi) = \phi$.



Figure 2.3: A visual representation of the simplified movement of the vehicle showing it is always aligned with the reference (black line)

The goal of the equations of motion is to find a description for the evolution of the states only depending on the control inputs and the states at a previous step.

$$\mathbf{x}_{s+1} = f\left(\mathbf{x}, \mathbf{u}\right) \tag{2.64}$$

In the following derivation, when there is no subscript attached to a state or control input the value at arbitrary distance step *s* is used. In other words $\mathbf{x} = \mathbf{x}_s$.

The first step is to determine how the angle ϕ is related to the reference curvature κ_{ref} .

$$\phi = \kappa_{ref} \cdot h \tag{2.65}$$

The distance travelled in direction (dx) of reference and perpendicular(dy) are defined as follows.

$$dx = h \cdot \cos(\phi) + n_{s+1} \cdot \sin(\phi) \tag{2.66}$$

$$dx = h + n_{s+1} \cdot \phi \tag{2.67}$$

$$dy = n - n_{s+1} \cdot \cos(\phi) + h \cdot \sin(\phi) \tag{2.68}$$

$$dy = n - n_{s+1} + h \cdot \phi \tag{2.69}$$

The time it takes between distance steps (dt) is defined by the longitudinal travelled path divided by the longitudinal speed.

$$dt = \frac{h + n_{s+1} \cdot \phi}{u} \tag{2.70}$$

Defining the evolution of the lateral offset starts with the relationship between the lateral speed, lateral distance travelled and the time.

$$v \cdot dt = dy \tag{2.71}$$

substituting dt and dy we get the following:

$$v \cdot \frac{[h + n_{s+1} \cdot \phi]}{u} = n - n_{s+1} + h \cdot \phi$$
 (2.72)

Reordering the equation to include all n_{s+1} terms at the left side we get the following:

$$n_{s+1} + n_{s+1} \cdot \frac{v \cdot \phi}{u} = n + h \cdot \phi - \frac{v}{u} \cdot h$$
(2.73)

This allows us to define the lateral offset at the next distance step n_{s+1} as a function of current states.

$$n_{s+1} = [1 + \frac{v \cdot \phi}{u}]^{-1} \cdot [n + h \cdot \phi - \frac{v}{u} \cdot h]$$
(2.74)

The next step is to complete a description of the lateral speed. The lateral speed is defined as follows:

$$v_{s+1} = v + a_v \cdot dt \tag{2.75}$$

The assumption that the vehicle is always facing in the same direction as the reference means that the lateral speed needs to be defined more carefully. Therefore an additional factor is needed in the calculation. This extra term makes sure that zero lateral speed and acceleration will go straight ahead instead of following the reference curvature.

$$v_{s+1} = v + (a_y - a_{y,corr}) \cdot \frac{h}{u}$$
 (2.76)

Where the correction factor ay_{corr} is the expected lateral acceleration based on speed, lateral offset and reference curvature.

$$a_{y,corr} = u^2 \cdot \frac{\kappa_{ref}}{1 + n_{s+1} * \kappa_{ref}}$$
(2.77)

The global heading and the total time are defined as follows

$$t_{s+1} = t + \frac{h + n_{s+1} \cdot \phi}{\nu}$$
(2.78)

$$\theta_{s+1} = \theta + \phi \tag{2.79}$$

The longitudinal speed is defined as follows

$$u_{s+1} = u + a_x \cdot \frac{h}{u} \tag{2.80}$$

Finally, the lateral and longitudinal accelerations are defined as functions of the jerk input as follows

$$a_{x,s+1} = a_x + J_x \cdot \frac{h}{u} \tag{2.81}$$

$$a_{y,s+1} = a_y + J_y \cdot \frac{h}{u} \tag{2.82}$$

Now, all the states at the next distance step are defined only as a function of states at the current step and/or the input states and the equations of motion can be written in the form as seen in eq 2.64.

2.7. Summary

In this chapter the differences between a cartesian and curvilinear coordinate system have been discussed. For this application a curvilinear approach is the more optimal solution. In combination with a curvilinear approach, the distance domain is a better solution, allowing for only the curvature of the road to be used as a reference. SQP and ASM have been presented, combined ACADO uses these methods to solve non-linear optimal control problems. To generate a speed profile longitudinal point mass dynamics are utilized with the jerk as an input to the optimization. The maximum speed is determined beforehand using the curvature of the road and assuming steady-state cornering. For each point along the road a speed is now determined that should not be exceeded. Generating a speed profile using a linear optimization can not be done accurately. However, an assumption can be made that the vehicle takes the same time for each distance step, leading to an approximate solution. The non-linear optimization is able to overcome the shortcomings of the linear optimization, by including the time as a state in the solution. By incorporating the lateral dynamics in the EOM the predetermined maximum speed is no longer needed. The lateral dynamics are simplified by assuming the vehicle is always aligned with the reference, therefore ignoring yaw dynamics. Leading to a complete description of the longitudinal and lateral motion that can be solved using a non-linear optimization method.

3

Benchmark

This chapter benchmarks the proposed speed planning methods against a state of the art literature example. It starts by describing the scenario and the method that was used as a benchmark result. Then it explains what differentiates speedprofiles from each other in terms of performance by introducing KPI that can be used as performance evaluation. At the end the effects of the different settings that were used are presented.

3.1. Benchmark Description

Benchmarking the proposed speedprofile generation methods is done against the method used in [16]. Minimizing motion sickness is done by minimizing the changes in speed and thus accelerations. A more in depth look will be taken in the following section to understand the approach used.

The vehicle is simplified as a point mass model that can accelerate within bounds, the friction circle. The dynamics equations are defined as the first and second derivative of the position, therefore containing the position S, velocity v and acceleration a in both the longitudinal and lateral direction.

$$\dot{S}_x = u \tag{3.1}$$

$$\dot{S}_{y} = v \tag{3.2}$$

$$\ddot{S}_x = a_x \tag{3.3}$$

$$\ddot{S}_y = a_y \tag{3.4}$$

The friction circle is chosen as a constraint and is defined as the euclidean distance between the two acceleration components. Since the value for the friction constraint is not mentioned, the value is assumed to be based on a friction coefficient of $\mu = 1$ and gravitational acceleration $g = 9.81m/s^2$.

$$\sqrt{a_x^2 + a_y^2} = \mu g \tag{3.5}$$

The curvature of the centreline of the road as a function of distance has been used as the only reference. this means that a curvilinear approach in the distance domain has been utilised.

Motion sickness is determined based on whole body exposure to mechanical vibrations and repeated shock. The total motion sickness dose value (MSDV) resulting from the the lateral and longitudinal motion is give as the following:

$$MSDV = \left(\int_0^T (a_{x,w}(t))^2 dt)\right)^{\frac{1}{2}} + \left(\int_0^T (a_{y,w}(t))^2 dt)\right)^{\frac{1}{2}}$$
(3.6)

where $a_{x,w}$ and $a_{y,w}$ are frequency weighted accelerations with W_F the weighting factor.

$$a_{x,w}(t) = a_x(t) \times W_f \tag{3.7}$$

$$a_{y,w}(t) = a_y(t) \times W_f \tag{3.8}$$

A simple linear approximation between MSDV and the illness rating (IR) is given as follows:

$$IR = K \times MSDV \tag{3.9}$$

Where *K* is an empirically derived constant set to $K = \frac{1}{50}$ [16]. In this implementation the illness rating was used to represent the motion sickness.

Now that the objectives and dynamics are defined the next step is define the optimal control problem. The cost function is defined for three distinct cases. The first case is the minimization of journey time. The third case the minimization of motion sickness. For the second case a combination of the first and third case is taken. This leads to the following cost functions.

Case 1 :
$$J = \int_0^s \frac{1}{s} ds$$
 (3.10)

Case 2 :
$$J = w_1 \cdot \int_0^s \frac{1}{s} ds + w_2 \cdot IR$$
 (3.11)

$$Case \ 3: J = IR \tag{3.12}$$

Where w_1 , w_2 are the weighting factors and s0, sf are the starting and final distance respectively. A minimum speed of $u_{min} = 5m/s$ has been utilised to overcome a problem that is introduced with minimization of acceleration. A minimization of accelerations can lead to the speed going to zero and this is to be avoided due to the limitations in using the distance domain.

The results that are obtained using the presented motion sickness minimization algorithm are based on a simple road scenario. There are three road curvature profiles presented. The trajectories can be seen in Figure 3.1. Only the most aggressive curvature road profile will be considered, which corresponds to road K1. Which means the road consists of two hairpin turns joined together by three straight sections. The radius of the corners is estimated to be R = 8m corresponding to a curvature of $\kappa = 0.125m^{-1}$.



Figure 3.1: The three trajectories used in [16]

Taking the scenario as described above, the resulting speedprofiles are found in Figure 3.2. This Figure contains the speedprofiles for the three different cases for a single trajectory. For case 3 it can be seen that the speedprofile does not immediately go to the minimum speed, therefore it can be concluded that time is still included in the cost function. This concludes the description for the benchmark.


Figure 3.2: The resulting speedprofiles from [16]

The reference curvature and the other boundary conditions that are to be used in the proposed methods are now discussed. The length of integration has been set at S = 250m with distance steps of h = 1m. Influence of the change of the settings will be discussed at the end of this chapter. The curvature profile that has been used as the reference can be seen in Figure 3.3, leading to the trajectory in space as seen in Figure 3.4. The first corner starts at 78m and ends at 102m, the second corner starts at 178m and ends at 202m. Acceleration limits are set to $a_{max} = 9.81m/s^2$, while jerk limits are not enforced.



Figure 3.3: The curvature as a function of distance



Figure 3.4: The trajectory in space

3.2. Key Performance Index

To be able to accurately asses the objective metrics for a comfortable ride it is necessary to divide the resulting motion into individual maneuvers [3]. Therefore the KPI are defined with the benchmark scenario in mind.

Travel time is the most important value in the context of lap time optimization, since this is the primary goal. Also, it is an easy to intuitively understand parameter. Therefore travel time is a valuable performance indicator in the context of time optimality and it can help differentiate strategies that perform similarly from a comfort standpoint. Travel time is defined as the travelled distance divided by the speed.

$$t = \sum_{s=0}^{S} \frac{h}{u(s)}$$
(3.13)

According to [2] the RMS of the acceleration can be used as a comfort parameter. To approximate the comfort parameters the RMS of the longitudinal acceleration is taken. The RMS is calculated as

follows:

$$a_{RMS} = \sqrt{\frac{1}{S} \sum_{s=0}^{S} a_x(s)^2}$$
(3.14)

The RMS value shows how much change of speed occurs over the length of the trajectory. For the motion planning the lateral acceleration is determined and thus an additional KPI is also included which is based on RMS of the combined acceleration.

$$a_{RMS,combined} = \sqrt{\frac{1}{S} \sum_{s=0}^{S} \left[a_x(s)^2 + a_y(s)^2 \right]}$$
(3.15)

The final KPI is the peak jerks in the longitudinal direction. Jerk limits are used to determine comfort during curves and influences design of transportation structures [18]. The highest and lowest value can be used to determine if the trajectory is deemed acceptable according to [1]. According to [18] the lateral jerk of transportation infrastructure should not exceed $0.9m/s^3$. Therefore it can be concluded that in a real world setting the lateral jerk is of this magnitude, therefore a longitudinal jerk that is lower than this limit can be regarded as not having an influence on comfort. The values are presented as the minimum value followed by the maximum value.

$$J_{x,peak} = [min(J_x(s)) , max(J_x(s))]$$
(3.16)

In short, up to four KPI values for performance evaluation are determined and are listed below, with their accompanying preferred value. For the travel time, RMS acceleration and RMS acceleration combined a lower value is always better than a higher value. For the peak jerk the value needs to only be within bounds.

- The objective of the travel time is to achieve a value as low as possible.
- · The objective of the RMS of the accelerations is to be as low as possible.
- The peak jerk should be lower than predetermined bounds, based on comfort criteria.

3.3. Speedprofile using Linear Optimization

The cost functions of the linear method are slightly different to the functions used in the benchmark. Time minimization is defined as the difference between the realised speed and the reference speed, while the motion sickness is defined as the minimization of longitudinal acceleration. Since the lateral acceleration is not taken into account during the optimization, the friction circle constraint can not be guaranteed, only longitudinal acceleration limits. The initial speed, acceleration and jerk have not been specified and thus are determined by the optimization.

Case 1, time minimization is realised by only including the speed cost in the cost function, which ensures that the only goal for the optimization is to maximise the speed as much as possible. Representing this, the following cost function values have been chosen.

$$v_c = 1 \quad a_c = 0 \quad J_c = 0$$

This leads to the speedprofile as seen in Figure 3.5 The linear method does not calculate the acceler-



Figure 3.5: The speedprofile with a linear method for case 1

ation and jerk during the trajectory correctly. To determine the RMS for the acceleration and jerk only the speedprofile is taken and the acceleration and jerk are determined afterwards. This can be done by assuming a constant acceleration between the different speeds. Here, the timestep at each distance index is also needed.

$$dt_s = \frac{h}{u_s} \tag{3.17}$$

$$a_{x,real,s} = \frac{u_{s+1} - u_s}{dt_s}$$
(3.18)

$$J_{x,real,s} = \frac{a_{x,real,s+1} - a_{x,real,s}}{dt_s}$$
(3.19)

Travel time	a _{RMS}	a _{RMS,combined}	J _x ,peak
[s]	$[m/s^2]$	$[m/s^2]$	$[m/s^3]$
18.7	4.59	-	[-224, 24.2]

Table 3.1: Case 1 KPI for the linear method benchmark

It can be seen that the RMS for the acceleration is fairly high, which is to be expected, since the system is trying to optimize the travel time. However, that acceleration is not close to the acceleration limit that is imposed. This is due to the fact that the timestep is not varying during the optimization and is the result of eq 2.38. Peak jerk shows that the guidelines found in [1] are not adhered to, which is to be expected since the optimization does not take into account comfort. The lower peak

jerk indicates that the change from acceleration to deceleration is more sudden and to be expected, since this value corresponds to the point in between the curves where the speedprofile goes from increasing to decreasing, resulting in a change of acceleration twice the acceleration constraint in a single distance step. this can be seen in Figure 3.6 where there is a single negative jerk data point where the speedprofile goes from acceleration to deceleration.

The speedprofile shows a linearly changing speed with respect to distance, while the benchmark result shows the speedprofile tapering off at higher speeds. The benchmark result shows expected behaviour, since the higher the speed, the shorter the distance step takes, the lower the change in speed over a given distance with the same acceleration. Linearly changing speed is an expected shortcoming of the linear method, because the timestep is assumed constant.



Figure 3.6: The jerk as a function of distance

Case 2, motion sickness minimization is realised by a combination of minimisation of acceleration and time minimization. Jerk has not been included in the cost function, leading to the following cost functions. The specific values have been determined through trial and error to match the visual pattern of the speedprofile to the result from the benchmark.

$$v_c = 1$$
 $a_c = 5$ $J_c = 0$

This leads to the speedprofile as seen in Figure 3.7



Figure 3.7: The speedprofile with a linear method for case 2

Travel time [s]	a_{RMS} $[m/s^2]$	$\begin{vmatrix} a_{RMS,combined} \\ [m/s^2] \end{vmatrix}$	$\begin{bmatrix} J_{x,peak} \\ [m/s^3] \end{bmatrix}$
21.2	1.59	-	[-2.84, 22.8]

Table 3.2: Case 2 KPI for the linear method benchmark

Travel time has increased by roughly 13% compared to Case 1. The RMS of the acceleration has decreased by 65%. This means that there is a meaningful trade off between travel time and minimization of acceleration. The fact that the jerk was not included in the cost function can be observed by the peak jerk values. The peak negative jerk was significantly lower than the peak positive jerk, indicating that the deceleration was done in a smoother fashion than the acceleration. The acceleration plot can be seen in Figure 3.8 which supports this claim. The speed during the corners is constant, but at the maximum allowed speed, leading to high lateral accelerations. Since the lateral acceleration is not taken into account for the minimization of accelerations the speed is not reduced during the curves, which explains why the travel time is only slightly longer than the time minimization case.



Figure 3.8: The acceleration as a function of distance

Up until this point the jerk term in the cost function was not used. However in the benchmark a minimization was done of certain frequencies of acceleration. A vibration means quick changes in speed and acceleration, meaning relatively large jerk values. Therefore, the jerk term is introduced in the cost function for Case 3 leading to the following values.

$$v_c = 1 \quad a_c = 5 \quad J_c = 1$$

This leads to the speedprofile as seen in Figure 3.9

Travel time	a _{RMS}	a _{RMS,combined}	J _{x,peak}
[s]	$[m/s^2]$	$[m/s^2]$	$[m/s^3]$
23.4	1.02	-	[-1.05, 0.85]

Table 3.3: Case 3 KPI for the linear method benchmark

The travel time has increased by 10% compared to case 2, which is explained by a twofold of reasons. Firstly, the introduction of the jerk term means that the relative weight of the time minimization term in the cost function is lower. Secondly, jerk minimization leads to less aggressive application of accelerations, leading to more time between peak acceleration and deceleration, in turn leading to longer travel times. RMS of the acceleration is lower compared to case 2. However the difference is not deemed significant. The reason for this is that the lateral acceleration as a result of the speed during the corners that is expected will have a much larger influence on total accelerations experienced. In combination with peak jerks being lower, this leads to a more comfortable trajectory than found in case 2. Finally, it can be observed that the speed during the curves is no longer constant. This is consistent



Figure 3.9: The speedprofile with a linear method for case 3

with results found in the benchmark, however this introduces a problem. Now there is longitudinal acceleration during the curves, meaning that the friction circle constraint can no longer be guaranteed.

An advantage to using a linear optimization means that the computation time is very short. For this scenario, regardless of the used cost function, the computation time was roughly 0.5 seconds. This makes the linear approach feasible for online speedprofile generation from a computational complexity point of view.

3.4. Speedprofile using Non-Linear Optimization

In this section the non-linear speedprofile generation method will be compared to the benchmark results. The non-linear method aims to combat some of the shortcomings observed in the previous section. The initial conditions are set to free variables and are to be determined by the optimization algorithm.

For Case 1 time minimization the cost function is purely comprised of the time.

50

$$\mathcal{L} = t$$
(3.20)

200

250

Figure 3.10: The speedprofile for case 1 using the non-linear method

0

Travel time	a_{RMS}	a _{RMS,combined}	$J_{x,peak}$ $[m/s^3]$
[s]	$[m/s^2]$	[m/s ²]	
15.0	8.80	-	[-324,96.7]

100

00 150 Distance [m]

Table 3.4: Case 1 KPI for the non-linear method benchmark

Travel time is lower than in the linear variant. This is related to a good description of the acceleration and jerk which can accurately predict the acceleration. this is also seen in the RMS of the acceleration, which is closer to the friction circle constraint than the value found in the linear method, meaning that more of the available acceleration is used. The peak jerk indicates that this is not a comfortable trajectory. Comparing the resulting speedprofile to the benchmark result a similar speed at the start, at the peak on the second straight, and during cornering is observed. It can also be seen that the speed tapers off at higher speeds, which is expected behaviour and shows that the non-linear method improves upon the linear variant in this regard.

For case 2 the cost function is made up of the time and the integration of the longitudinal acceleration. The value for the weighting variable has been determined as $w_1 = 0.001$.

$$\mathcal{L} = t + w_1 \cdot \int_0^S a_s^2 ds \tag{3.21}$$

ravel time a_{RMS} [s] $[m/s^2]$		$a_{RMS,combined}$ [m/s ²]	$J_{x,peak}$ $[m/s^3]$	
16.0	2.84	-	[-6.33,72.7]	

Table 3.5: Case 2 KPI for the non-linear method benchmark

Travel time has increased by 6.7% with a reduction of the RMS for the acceleration of 68%. The only slightly longer travel time can be attributed to having the same speed during the curves. At the same



Figure 3.11: The speedprofile for case 2 using the non-linear method

time the peak jerk has been lowered compared to case 1, but is still considered too high according to [1].

For case 3 the following cost function is used. The weighting variables are determined to be $w_1 = 0.001$ and $w_2 = 0.005$.

$$\mathcal{L} = t + w_1 \cdot \int_0^S a_s^2 ds + w_2 \cdot \int_0^S J_s^2 ds$$
 (3.22)



Figure 3.12: The speedprofile for case 3 using the non-linear method

Travel time [s]	a_{RMS} $[m/s^2]$	a _{RMS,combined} [m/s ²]	$\begin{bmatrix} J_{x,peak} \\ [m/s^3] \end{bmatrix}$
21.2	1.85	-	[-1.04, 2.26]

Table 3.6: Case 3 KPI for the non-linear method benchmark

Travel time is now noticeably longer than previous cases. Mainly two reasons can be thought of for this, the first being the lower speeds on the middle straight, the other reason is the lower speed during the curves. However as was the case with the linear method, the friction circle constraint can not be guaranteed, since there is longitudinal acceleration during the curves. With peak jerk values that are considered comfortable case 3 shows the most comfortable trajectory of the three different strategies. Non-linear optimization is more computationally complex than a linear optimization. The non-linear

optimization took 88s for case 3 using the settings as specified, which is substantially longer than the travel time. However, the computation time is heavily influenced by the specific settings used and the effects of settings section goes more in depth.

These results show that the non-linear method is able to more accurately capture the dynamics that are necessary to produce not only a comfortable trajectory, but especially a more time optimal one, compared to the linear method. This comes at the trade-off in computation time which is significantly shorter for the linear variant. However another shortcoming has emerged here in the fact that the lateral motion is not taken into account and even with a seemingly comfortable speedprofile the comfort during lateral motion is determined only by a lateral acceleration limit. In the next section motion planning is utilized that will aim to combat this shortcoming and produce suitable speedprofiles taking into account both longitudinal and lateral motion.

3.5. Speedprofile using Non-Linear Optimization with Lateral Motion

Main two differences between the motion planning and the linear/non-linear method are firstly that the lateral motion is now taken into account and the cost functions will be changed to reflect this. Resulting in no longer needing a reference speed and therefore this reference speed is no longer included in the results plots. Secondly, the amount of distance steps has been reduced to 125 resulting in a stepsize of h = 2m. This is done to reduce the computational load and make the calculation of the result feasible.

Case 1 is done using the following cost function:

$$\mathcal{L} = t \tag{3.23}$$

The following speedprofile is found:



Figure 3.13: The speedprofile for case 1 using motion planning

Travel time [s]	a_{RMS} $[m/s^2]$	a _{RMS,combined} [m/s ²]	$J_{x,peak}$ $[m/s^3]$
15.0	8.80	9.78	[-189, 52.5]

Table 3.7: Case 1 KPI for the motion planning benchmark

For the motion planning, exactly the same travel time and RMS acceleration as the non-linear method are found. Where this does not hold is when looking at the jerk. The jerk is applied at a single step with the required magnitude to apply the highest possible acceleration at the following distance step. Since the step size is taken as h = 2m compared to h = 1m for the non-linear method the jerk is applied for a longer distance and thus longer period of time, resulting in a lower peak jerk. RMS for the combined acceleration is close to the friction circle constraint suggesting that the used method is time optimal.

For case 2, minimization of accelerations, the combined longitudinal and lateral accelerations term has been added to the cost function. The main difference between this implementation is that the lateral acceleration is considered during the optimization and thus is included. Leading to the cost function below with the weight factor $w_1 = 0.001$.

$$\mathcal{L} = t + w_1 \cdot \int_0^S \left[a_{x,s}^2 + a_{y,s}^2 \right] ds$$
(3.24)

Travel time [s]	a_{RMS} $[m/s^2]$	a _{RMS,combined} [m/s ²]	$\begin{bmatrix} J_{x,peak} \\ [m/s^3] \end{bmatrix}$
17.3	3.95	5.40	[-8.23, 46.5]

Table 3.8: Case 2 KPI for the motion planning benchmark



Figure 3.14: The speedprofile for case 2 using motion planning

The travel time has increased by 15%, which is more than the non-linear method, which is explained by the lower speed during the curves.By including the lateral acceleration in the cost function, the speed in curves is lowered slightly compared to the non-linear optimization, where this approach was not posssible. RMS acceleration compared to case 1 has been lowered, but is too high to be considered comfortable.

For case 3 the longitudinal jerk has been included in the cost function. Only the longitudinal jerk is used in the cost function, not the lateral jerk. The lateral jerk is mostly defined by the curvature of the road and since the curvature in this scenario is not smooth, the result has high lateral jerk. Leading to speedprofiles that will minimize speed on corner entry and exit, which introduces more changes in speed than is desired from a driveability point of view. Case 3 is done using the following cost function with the weighting values $w_1 = 0.0001$ and $w_2 = 0.00005$:

$$\mathcal{L} = t + w_1 \cdot \int_0^S \left[a_{x,s}^2 + a_{y,s}^2 \right] ds + w_2 \cdot \int_0^S J_{x,s}^2 ds$$
(3.25)



Figure 3.15: The speedprofile for case 3 using motion planning

Travel time	a_{RMS}	a _{RMS,combined}	J _{x,peak}
[<i>s</i>]	$[m/s^2]$	[[m/s ²]	$[m/s^3]$
26.7	1.44	2.22	[-0.84, 1.99]

Table 3.9: Case 3 KPI for the motion planning benchmark

Travel time increased has increased by 78% compared to case 1, however a reduction in RMS acceleration can also be noted. More interestingly the lateral acceleration has also been reduced as can be seen by the RMS acceleration combined, which is only slightly higher than the RMS acceleration, as a result of the slower speed during the curves. The jerk minimization means that jerk has been reduced to below the limits, therefore this speed profile can be considered comfortable. By including the longitudinal jerk in the cost function for the motion planning algorithm a speedprofile can be found that possesses similar properties to one found that is able to minimise motion sickness. Therefore comfortable speedprofiles. Due to the more complex nature of the equations of motion, the computation time of the motion planning algorithm was similar to that of the non-linear method even though the motion planning used half the amount of distance steps (125, against 250 for the non-linear method).

3.6. Effects of Settings

Different settings for the dynamics, cost function or ACADO result in different outcomes. In this section four different settings are further examined. First, the decision to omit the lateral jerk in the cost function for the motion planning is elaborated and the effect is demonstrated with an example. Second, the distance step size is varied to investigate the trade off between computational complexity and accuracy. Third, the linear method timestep is varied to find the effects a larger assumed timestep has on the resulting motion. Finally, the initial conditions for the ACADO optimization are varied to show that for similar scenarios there can be a significant difference in computation time and convergence.

3.6.1. Cost Function Weights

The goal of the benchmark is to check the results of the proposed motion planning against the described state of the art example. The resulting speed profile is dependent on the weights that are chosen for the acceleration and jerk. Since the KPI of the benchmark reference results are not known, the chosen weights of the cost function were not done based on KPI. By increasing the weights the optimization will put less focus on the travel time and more focus on the acceleration and jerk. To show the potential of the proposed motion planning algorithms, the weights are chosen such that the resulting speedprofile has a similar visual profile. On top of that, there was extra attention given to make sure the speedprofiles have a somewhat similar starting speed and top speed on the straight between the two corners. By presenting the results in a similar visual format as the benchmark reference result, it can be observed that the proposed motion planning algorithm is able to produce similar speed profiles.



Figure 3.16: The resulting speedprofiles using the proposed motion planning method in the benchmark format

3.6.2. Lateral Jerk Inclusion in Motion Planning

The lateral jerk is not included in the cost function for the motion planning optimization in the benchmark results. The reason for this is that the results better resemble the benchmark results when the lateral jerk is omitted from the cost function. The lateral jerk is a function of both the speed and the change of curvature of the road. Therefore when the road curvature changes, such as at the start and end of a constant radius corner, high lateral jerk is required to be able to follow the road. Minimization of the lateral jerk leads to a minimization of speed only at the start and end of a corner. To illustrate this the following cost function is set up with the weights equal to those used in the benchmark scenario, $w_1 = 0.005$ and $w_2 = 0.002$.

$$\mathcal{L} = t + w_1 \cdot \int_0^s \left[a_{x,s}^2 + a_{y,s}^2 \right] ds + w_2 \cdot \int_0^s \left[J_{x,s}^2 + J_{y,s}^2 \right] ds$$
(3.26)

The resulting speedprofile can be seen in figure 3.17. During cornering the speed is not constant, but lower at the corner entry, peaking at the middle of the corner and then slow again at the corner exit. This behaviour is not ideal from a comfort point of view, because this introduces more changes of acceleration, jerk, which impacts comfort negatively. Looking at the KPI for this speed profile it can be seen that the travel time has increased with a lower $a_{RMS,combined}$ indicating the opposite, a more comfort oriented trajectory. However, where the lateral jerk minimization does not improve on comfort is in the peak longitudinal jerk, which is double that of the values seen in the benchmark and they are too high to be considered comfortable. Showing that when no control can be had over the lateral motion, including the lateral jerk in the cost function does not necessarily improve comfort and needs to be considered on a case to case basis.



Figure 3.17: The speedprofile including lateral jerk using the motion planning method

Travel time [s]	a_{RMS} $[m/s^2]$	$a_{RMS,combined}$ $[m/s^2]$	$J_{x,peak}$ $[m/s^3]$
29.0	1.48	1.96	[-0.95, 4.37]

Table 3.10: KPI for the motion planning with lateral jerk inclusion

3.6.3. Distance Step Size

Distance step size is the amount of length between each data point for which a speed is determined. Since the benchmark trajectory is a fixed length of 250m, varying the distance step is done by adjusting the amount of steps the trajectory is divided into. The amount of steps influences the computation time by needing to perform more calculations. On the other hand when the step size increases too much the accuracy of the results are compromised and no longer useable. Here To see the effects the amount of steps for the speedprofile generation the KPI for four different step sizes are shown in table 3.11.

Distance steps	Step size	Travel time	a_{RMS} $[m/s^2]$	$J_{x,peak}$ $[m/s^3]$	Computation time
[]	[]	[9]			[3]
250	1	15.0	8.80	[-324, 96.7]	236
125	2	15.0	8.78	[—159, 52.5]	30
50	5	14.4	8.86	[-64.5, 24.9]	2.0
25	10	14.6	8.60	[-32.0, 15.0]	0.40

Table 3.11: KPI and computation time for different stepsizes for the non-linear method benchmark

First thing to notice is the peak of the longitudinal jerk, which is proportional to the amount of distance steps. Since the amount of distance steps is inversely proportional to the step size, this is to be expected. The jerk is assumed to be applied over the entirety of a distance step. Therefore a larger step size leads to a longer application of the jerk, which in turn leads to a lower peak value. Therefore, without jerk limitation or minimization the step size is the defining parameter that defines peak jerk values. Next is the fact that the travel time is shorter with less steps, suggesting that the obtained results are not accurate. To investigate this further the speedprofile for 25 distance steps is shown in figure 3.18. This figure clearly shows that the step size is too large to gain accurate results. In this symmetrical scenario the optimal speed profile should also be symmetrical, however this is not the case, in combination with a different travel time compared to the short step sizes the result is deemed inaccurate and not usable.



Figure 3.18: The speedprofile for case 1 using the non-linear method with only 25 distance steps

3.6.4. Timestep Variance for Linear Method

The linear method simplifies the dynamics by assuming a constant speed across the whole trajectory for the timestep calculation.

$$dt = \frac{h}{U} \tag{3.27}$$

where *U* denotes the assumed speed in m/s and *h* the distance step in *m*. Therefore the selected assumed speed has a significant influence on the results. For the benchmark the most conservative approach was taken in the form of taking the highest speed possible, the speedlimit 40m/s, as the assumed speed. When the assumed speed is higher than realised speeds the acceleration and jerk will always be overestimated. On the other hand, when the assumed speed is lower than the realised speed the acceleration and jerk are underestimated and can become infeasible due to exceeding friction circle constraints. Taking the benchmark scenario with case 3 jerk minimization the effects of the assumed speed is investigated. Table 3.12 shows the KPI for various different assumed speed.

Assumed Speed [<i>m</i> / <i>s</i>]	Travel time [s]	a_{RMS} $[m/s^2]$	$J_{x,peak}$ $[m/s^3]$	Acceleration Peak $[m/s^2]$	Speed Peak [m/s]
40	24.6	0.66	[-0.33, 0.53]	1.27	13.5
30	22.5	1.38	[-1.05, 1.18]	2.49	17.1
20	19.4	3.43	[-4.12, 3.39]	6.05	24.3
10	15.8	9.72	[-19.6, 16.9]	17.3	36.1

Table 3.12: KPI for different timesteps for the linear method benchmark

It can be seen that all the KPI are dependent on the assumed speed. To get the most accurate results the assumed speed should therefore be chosen such that they are as close as possible to the realised speed. The last entry shows the peak speed as determined by the linear optimization. The difference between the peak speed and the assumed speed is a good indication for the accuracy of the results. When the assumed speed is larger than the peak speed the acceleration and jerk are overestimated and therefore a longer travel time and more comfortable KPI are observed. Conversely, a lower assumed speed than peak speed indicates that the acceleration and jerk are underestimated and the real values are higher. Leading to a lower travel time and less comfortable KPI with higher accelerations and jerk. In the case of an assumed speed of 10m/s the peak acceleration exceeds the friction circle constraint, meaning that not only is it not an accurate result, it is also not a feasible trajectory. To highlight the different cases figure 3.19 shows the acceleration and jerk as estimated by the algorithm and as determined afterwards based on the speedprofile. The real acceleration and jerk are always lower than the real values when the assumed speed is overestimated. When the assumed speed is underestimated the real acceleration is consistently higher than expected values. The real difference is when looking at the jerk, not only is the magnitude too high, but the sign is often

in the opposite direction. Therefore, an overestimation of the assumed speed leads to a conservative estimate of the optimal solution, while an underestimation of the assumed speed leads to inaccurate and/or infeasible results.



Figure 3.19: Acceleration and jerk for the linear method as determined by the algorithm (Input) and calculated afterwards (Real). Assuming a speed of 40m/s (left) and 10m/s (right)

3.6.5. Initial Conditions for Non-Linear Method and Motion Planning

For the motion planning algorithm using ACADO the initial guess that is used for initialization of the SQP problem is based on the bounds that are set. Setting the speed at the first step of the trajectory influences the computation time and even the overall convergence. This effect is shown in table 3.13.

Initial Speed	Computation time	SQP iterations	Convergence
[m/s]	[<i>s</i>]	[-]	[-]
-	26.3	35	YES
5	37.6	48	YES
6	45	55	YES
8	89.5	79	YES
10	16.4	-	NO
15	17.1	-	NO

Table 3.13: Computation time and convergence for case 3 motion planning benchmark

For an initial speed of more than 10m/s it can be seen that ACADO is no longer able to find a solution. This comes from the fact that ACADO uses the bounds of the system for a starting state. Setting the speed at every point along the trajectory to 10m/s will lead to an infeasible state, since that speed would lead to lateral acceleration during the corner that would exceed the friction circle constraint. By looking at the final solution of the optimization we can see that the initial speed actually exceeds this value. Now looking at the computation time it is observed that there is high variance between the different initial speeds. This stems from the fact that more SQP iterations were needed to get to the optimal solution. This example shows that even for seemingly similar scenarios there can be a significant difference in computation time or even convergence as a whole. For this reason the computation time is not included as a KPI, even though it is mentioned.

3.7. Summary

This chapter introduced a state of the art literature example based on a non-linear optimization method that is able to minimize motion sickness based on vibrations experienced by the passengers. The speed profile generation based on linear optimization is not able to reproduce the resulting speed profile as seen in the example. For the non-linear optimization method the speed profile obtained for the time optimal cost function is very close, however for the comfortable motion, the speed through the corners was still too high to be considered comfortable. The motion planning system including the lateral motion is able to accurately reproduce similar speed profiles for the three different cost functions. The effects of the different settings for the cost function have been presented. It has been shown that the initial conditions chosen for ACADO can have an impact on computation time and convergence.



Comparison of Motion Planning Against a Complex Vehicle Model

This chapter goes over the simulation results to verify the effectiveness and accuracy of the motion planning algorithm. The vehicle model and the simulation are first presented. Then KPI are defined to evaluate the simulation results. A driver model is defined by two PID based controllers for the longitudinal inputs and the built-in Prescan path follower for the lateral control. Finally, the simulation results are presented and discussed.

4.1. Vehicle Model and Experiment Settings

In the Prescan GUI there are many different vehicle types available. For this experiment a Mazda RX-8 has been chosen with a vehicle model called 3D-simple. For this research the most important things to note are the width of the vehicle, the width of the road and the dynamics model of the vehicle. The width of the vehicle is a function of the vehicle it is based on. A Mazda RX-8 has a width of 1.77m. The width of the road is set to 3.5m, which is the standard width of a dutch road. To make sure the simulation has good enough accuracy, the simulation is set to 200hz, the recommended simulation frequency for the dynamics model used.

The dynamics model consists of two different masses that are connected, the sprung and the unsprung mass. The sprung mass (the mass that is supported above the suspensions) has 6 degrees of freedom. Three displacements (x, y and z) and three rotations (roll, pitch and yaw). And the unsprung mass (the mass below the suspensions; the 4 wheels) has 4 degrees of freedom namely the 4 vertical displacements. Between the sprung mass and unsprung mass are placed the suspensions. The suspension transmits forces between the unsprung mass of the wheels and the sprung mass of the body. For each individual wheel a stiffness and a damping is defined. The steering system directly connects the angle of the front wheels with the angle of the steering wheel. It also dictates the maximum amount of steering angle that is permitted.



Figure 4.1: The 10 degrees of freedom of the vehicle dynamics model. Source: Prescan help guide

The engine model consists of a torque map that can be modified. The torque map indicates the

engine torque as a function of the throttle position and the engine rpm. This allows for engine braking when no throttle is applied. The driveline and the transmission indicate the resulting wheel torque as a function of engine torque and the chosen gear. In this research the vehicle is operating in a single gear, therefore reducing the driveline to a simple reduction gear set powering the rear wheels.

All the parameter values that are used for the vehicle can be found in the Appendix.

4.2. Key Performance Index

In the works from [9] the controller was tuned according to the RMS of the lateral error. In this experiment the lateral error RMS does not give insight into the following of the path through the corner, which is the most interesting section from a controller performance point of view. On top of that, if the vehicle has a steady state error on the straight, leading to a high RMS value, does not necessarily mean that the performance is poor. The vehicle is allowed to operate freely within the lane boundaries, therefore the lateral controller is evaluated as a satisficing controller, instead of an optimizing controller. Instead of searching for a lateral offset as small as possible, anywhere in the lane is fine. The main KPI that indicates the lateral performance is therefore chosen to be the maximum lateral offset compared to the centerline of the road, not the reference trajectory.

$$n_{max} = \max\left(e_y + n_{ref}\right) \tag{4.1}$$

The centerline of the road directly correlates to the border of the road, which should not be exceeded. Maximum allowed offset is determined by the width of the vehicle and the width of the road. The width of the used vehicle is 1.77m, combined with a standard dutch road of width 3.5m, leads to the following value for the maximum lateral offset to make sure the vehicle does not leave the road:

$$n_{max} = 0.865m$$
 (4.2)

Comfort is determined mainly by the jerk experienced by the passengers. According to different sources from the literature the peak jerk should not exceed certain values. In this research any jerk lower than $0.9m/s^3$ is considered to be trivial and can be ignored, therefore an important KPI for comfort is in the form of the peak jerk, both in the longitudinal and the lateral direction.

RMS of the acceleration is an indicator of comfort according to [2], therefore it is a good KPI to evaluate the comfort of the resulting motion. The acceleration RMS should not be considered without taking into account the other KPI, since a lower RMS value can be obtained by bad tracking performance. To compute the RMS of the acceleration the following definition is used:

$$a_{RMS,combined} = \sqrt{\frac{1}{S} \sum_{s=0}^{S} \left[a_x(s)^2 + a_y(s)^2 \right]}$$
(4.3)

To compare the acceleration from the simulation and the reference the ratio of the peak acceleration is compared. The peak acceleration ratio is defined by taking the peak of the acceleration from the simulation and the reference as follows.

$$a_{x,ratio} = \frac{a_{x,sim,peak}}{a_{x,ref,peak}}$$
(4.4)

The peak acceleration ratio indicates how much the vehicle has to 'catch up' to the changing reference speed. It has to be noted that this ratio is a function of the performance of the longitudinal controller, which is not the scope of this research. Only the longitudinal direction is considered, since the lateral acceleration showed very good results and adding this value does not help the evaluation of the resulting motion.

To evaluate the performance of the simulation four different KPI have been established with an explanation of their preferred value below.

- The border of the road should not be exceeded, thus the lateral offset has to be within bounds.
- The objective of the peak jerk is to keep within comfortable bounds.
- The RMS of the acceleration should as low as possible.
- The objective of the peak acceleration ratio is get a value as close as possible to 1.

4.3. Driver Model

For the control scheme a PID controller is utilised. Ease of implementation and lack of constraint handling features make it a good controller to evaluate the reference quality. A PID controller is a linear feedback controller that takes an error signal and produces a control input u. For the error the difference between the current value and a reference is taken in addition to the integral and the derivative.

$$u = K_p \cdot e + K_i \cdot \int e + K_d \cdot \frac{d}{dt}e$$
(4.5)

Longitudinal control is done based on the current speed and the reference speed. The derivative part of the controller corresponds to direct acceleration control.

Therefore the error is defined as follows:

$$e_{long} = u_{current} - u_{ref} \tag{4.6}$$

The output of the longitudinal controller is split into two different controls for the throttle and the brakes. The throttle input is in the form of a percentage of travel and thus operates between 0 - 100, while the input for the brakes is brake pressure in bar, which ranges from 0 - 150. To account for the different dynamics and control input ranges two separate controllers are used for the throttle and the brakes.

The vehicle has three control inputs: Throttle, brakes and steering.

A different controller for each of these was utilised, resulting in three different controllers that together are able to control the vehicle. For the throttle control $u_{throttle}$ a PD control scheme with feedforward action based on reference acceleration is implemented. Leading to the following throttle control:

$$u_{throttle} = K_{p,throttle} \cdot e_{long} + K_{d,throttle} \cdot \frac{d}{dt} e_{long} + K_{ff,throttle} \cdot a_{x,ref}$$
(4.7)

The obtained controller gains are the following:

$$K_{p,throttle} = 20$$
$$K_{d,throttle} = 40$$
$$K_{ff,throttle} = 20$$

The next control input is the brakes u_{brakes} . The controller for the brakes is a PD controller without feedforward action. Leading to the following brakes control:

$$u_{brakes} = K_{p,brakes} \cdot e_{long} + K_{d,brakes} \cdot \frac{d}{dt} e_{long}$$
(4.8)

where the found controller gains are the following:

$$K_{p,brakes} = 80$$

 $K_{d,brakes} = 40$

The lateral control is achieved by the steering input. Simcenter Prescan comes included with a sophisticated path follower. The Path Follower is based on the Optimal Preview Control theory [24] [25]. It calculates the optimal front wheel steer angle to minimize the lateral error of the actor position (vehicle center of gravity) and a reference path. It approximates average human driver behavior.



Figure 4.2: The error defined by the path follower from Prescan. Source: Prescan help guide

The algorithm uses 10 preview points to calculate the steering wheel angle. The preview distance is calculated by multiplying the actor velocity in m/s by the 'Preview time' (default 1 second). This is actually the distance the vehicle moves in the preview time of 1 second. The 10 lateral errors are calculated by first finding the closest point on the trajectory and then the distance from this point to the actor longitudinal axis. The path follower settings are the default ones and can be found in the Appendix.

4.3.1. Controller Tuning

PID tuning is done by trial and error [9]. The goal of the controller is to find gains for both the longitudinal and the lateral controller that are able to achieve satisfactory results. Tuning is done based on a set of priorities. First priority is the longitudinal control, then the lateral control. The speed of the vehicle influences how the steering behaviour should change, therefore making sure the vehicle is able to somewhat follow the speedprofile is the first step. Longitudinal control tuning is done by evaluating the steady state error that is obtained and the oscillations in the control input that are needed to obtain this error. It was observed that high frequency oscillations, chattering, in the control input lead to good speed following behaviour. Therefore the trade off in the longitudinal control input was to find the tuning values without chattering, that still lead to decent speed following. For the brake control gains, the proportional gain is increased until oscillations occurred. Then the derivative gain was increased to try and dampen these. The derivative gain translates to direct acceleration control.

$$\frac{de_{long}}{dt} = \frac{du_{current}}{dt} + \frac{du_{ref}}{dt} = a_{x,current} - a_{x,ref}$$
(4.9)

The integral part of the controller was not utilised, since it introduced more oscillations and worsened performance. The longitudinal gains were chosen such that the peak jerk is a value of around $0.9m/s^3$.

For the throttle control the first step was to find a gain for the feedforward action that is able to somewhat follow the acceleration reference. This extra control gain is needed to overcome the bias in longitudinal control where no control action equates to a slightly deceleration due to external factors such as rolling and aerodynamic resistance. When a suitable gain is established a similar tuning pattern was used for the proportional and the derivative gain as the brake control.

Tuning of the lateral controller turned out to not be necessary, since the default settings of the path follower resulted in suitable lateral results. Suitable results in this scenario were marked by a lateral offset lower than the width of the road minus the width of the vehicle as defined in the KPI's and lateral jerk that was lower than a value of $0.9m/s^3$ with minimal oscillations and/or chattering.

4.4. Scenario Motion Profile

The scenario is inspired by the benchmark scenario. The road consists of three straight sections connected together by two curved roads with a radius of 8m. The total length of the road is 235m due to shortening of the final straight compared to the benchmark. A picture of the road as built in the prescan GUI can be seen in figure 4.3.



Figure 4.3: The road in Prescan

From the road the curvature is extracted. Resulting in the curvature plot as seen in figure 4.4.



Figure 4.4: The curvature of the road in Prescan

Most notable from this plot is the fact that the curvature does not seem to be consistent during the corners. This is due to the way Prescan handles building the road. In the background of prescan the road is built from a series of points that are connected to each other, where the any point between is determined using interpolation. A combination of the way the interpolation works and the amount of points used to create the road lead to the curvature shown.

For the reference that is used in the simulation the motion planning algorithm is used. The maximum allowed lateral offset is set to 0.5m to allow a small lateral offset to still be within the lane borders. The resulting motion profile is defined using two main parameters. The first one being the speed profile that makes sure the vehicle slows down before the corners. The second parameter is the lateral offset, which is defined as a function of the centerline of the road. To allow the vehicle to be able to follow the reference a comfortable motion profile is defined. The cost function that is used is similar to the one used in case 3 of the benchmark.

$$\mathcal{L} = t + w_1 \cdot \int_0^s \left[a x_s^2 + a y_s^2 \right] ds + w_2 \cdot \int_0^s j x_s^2 ds$$
(4.10)

With the following weights:

$$w_1 = 0.05 \qquad w_2 = 0.2 \tag{4.11}$$

A stepsize of h = 1m is used since this showed good results compared to larger distance steps and is a good trade off between accuracy and computation time. The obtained speed profile can be seen in figure 4.5. The speed profile is charaterised by slow speeds during the corners and smooth accelerations on the straights, which is ideal for comfort.





Figure 4.5: The speedprofile as a function of distance used as the reference for the simulation



Lateral motion is visualised in figure 4.9. The color indicates the lateral offset compared to the centerline of the road, where red means the reference is to the right and blue to the left of the road centerline. The lateral motion is characterised by a 'racing line' through the corners making use of the entire width of the road to lengthen the radius of the corner. For the second corner similar behaviour is observed as in the first corner.

Resulting in the following KPI values for the motion profile as described.

Fravel time	a_{RMS}	a _{RMS,combined}	$J_{peak} \\ [m/s^3]$
[s]	$[m/s^2]$	[m/s ²]	
40.4	0.52	0.86	[-0.16, 0.3]

Table 4.1: KPI for the motion planning reference used in the simulation

Peak jerk indicate that the resulting motion are considered comfortable, while the acceleration combined RMS indicate that the value is lower than what is defined as uncomfortable. Therefore it can be concluded that the resulting motion is considered comfortable and a good fit as a reference for a path follower.

4.5. Simulation Results

The results of the simulation are split into two sections, longitudinal and lateral. Since the vehicle is operating under very low accelerations and jerk, the vehicle model can be approximated by a linear bicycle model. Meaning that the longitudinal and the lateral dynamics are no longer coupled. The resulting motion as predicted by the motion profile should be a good fit for a vehicle that experiences low accelerations and jerk. Since a vehicle that is operating at low accelerations can be modeled accurately with a linear bicycle model [29]. The resulting KPI are as follows:

n _{max} [m]	$\begin{bmatrix} J_{x,peak} \\ [m/s^2] \end{bmatrix}$	$\begin{bmatrix} J_{y,peak} \\ [m/s^2] \end{bmatrix}$	a _{rms,combined} [m/s ³]	$a_{x,ratio}$ $[m/s^2]$
0.68	0.62	0.86	0.85	1.05

Table 4.2: KPI for the motion planning simulation results

From the KPI it can be seen that the lateral offset is lower than the road width. Both the longitudinal and lateral jerk are lower than the $0.9m/s^3$ threshold, meaning that from a comfort point of view the maximum jerk values are acceptable. RMS acceleration is not considered uncomfortable and shows similar values to the motion planning. Peak acceleration ratio comes in at 1.05, meaning that there is only a 5% overshoot, indicating good acceleration following.

4.5.1. Longitudinal Results

The longitudinal results are represented by the speed, acceleration, jerk and control inputs.



Figure 4.7: The resulting speed over the trajectory

Figure 4.8: The resulting longitudinal acceleration

Speed following shows good performance, especially during deceleration. During acceleration there is a slight undershoot in speed leading to the vehicle not reaching the maximum speed on the straight. The speed difference on the straight is around 0.4m/s, which for speed in a straight line is not significant and will only influence travel time. The acceleration reference shows the difference in acceleration and deceleration following. Looking at peak acceleration shows the 5% overshoot during acceleration. When switching between acceleration and deceleration there is a noticeable change in rate of acceleration. This effect can be seen right around 150m. Combining these results shows that the speed profile with accompanying acceleration are able to be followed fairly accurately.



Figure 4.9: The resulting longitudinal jerk

Figure 4.10: The longitudinal control inputs as a function of distance

Peak longitudinal jerk is higher than predicted from the motion planning, but it is within reasonable bounds from a comfort point of view. When the controller switches between throttle and brake inputs, namely around 90m, 150m and 190m, there is a noticeable spike in jerk. Therefore the higher than predicted jerk is a direct result of the control action. On the other hand, the steady state jerk from the simulation is very close to the values predicted by the motion planning, as seen around the 110m to 140m mark. Combining these two points it can be concluded that the motion planning algorithm is able to accurately predict the longitudinal acceleration and jerk.

4.5.2. Lateral Results

The lateral results are represented by the lateral offset, acceleration, jerk and control input.



Figure 4.11: The resulting lateral offset compared to the centerline of the road



The lateral offset shows that the vehicle is able to stay within the lane boundary at 0.86*m* at every point on the track. The vehicle is on the outside of path, which can be seen by a positive lateral offset in the first curve and a negative lateral offset in the second curve. Similar profiles are observed in both corners, just mirrored. Indicating that there is no direction dependency. The changing lateral offset during the middle straight is followed close, without noticeable oscillations or lateral error. The lateral acceleration that is expected based on steady-state cornering matches the realised lateral acceleration from the simulation. This means that the determined lateral acceleration from the motion planning is close to the values that are to be expected in the real world. The simulation shows slightly lower lateral acceleration than the reference from the motion planning. This is due to the fact that the vehicle has a lateral offset compared to the reference, leading to the vehicle following a lower curvature path than the reference. A lower curvature with a similar speed amounts to lower lateral accelerations during steady-state cornering.



Figure 4.13: The resulting lateral jerk

Figure 4.14: The steering wheel angle as a function of distance

The controller shows a smooth input without any noticeable oscillations. The result of this smooth control input results in a smooth lateral jerk without large peaks. Due to the good control from the Prescan path follower the resulting lateral jerk is very similar to the expected jerk as predicted by the motion planning algorithm. The fact that the lateral jerk from the simulation is very similar to the jerk as predicted by the motion planning, both in magnitude and profile, indicates that the assumptions made in the motion planning algorithm are valid.

250

4.6. Summary

This chapter has presented a scenario where the motion planning method has been used in Simcenter Prescan. A complex 10 degrees of freedom vehicle model containing both sprung and unsprung mass has been presented. KPI to evaluate the performance of the simulation have been defined. As a driver model two PID controllers have been established for the throttle and the brake control. For the steering, the built in path follower from Prescan has been used. To ensure representative results the method for tuning the controller has also been presented. The motion planning containing both lateral and longitudinal motion has been used to define a comfortable reference speed and trajectory. The simulation results show that the reference motion is comparable to the resulting motion, both in the longitudinal and the lateral direction. Indicating that the motion planning system produces accurate results. On the other hand it is also observed that the longitudinal control of the vehicle needs a more complex control scheme for optimal speed tracking, mostly to improve during the switching between throttle and brakes. For the lateral control a predictive control scheme is needed to be able to follow the road with very low lateral offset.

5

Conclusions and Future Work

5.1. Conclusions

In the literature review it was shown that motion planning as a reference for a subsequent path follower has advantages for automated driving tasks. The proposed motion planning algorithms that are designed are able to construct comfortable reference motion based on the curvature of the road. To validate the proposed methods, they are benchmarked against a state of the art literature example. The resulting motion is used as a reference trajectory and speed in a simulation using a complex vehicle model. This allows the comparison of the resulting motion with the predicted motion as determined by the motion planning algorithm. Accurate results are obtained showing that the assumptions are valid under these conditions. Therefore, below mentioned are the main conclusions that are obtained.

Speed profile generation using a linear optimization is a fast and relatively accurate method that can be used if computation speed is of utmost importance.

Jerk based motion planning using a non-linear optimization gives a more flexible solution than speedprofile generation with similar accuracy. In a situation where computation time is important linear speedprofile generation is preferred over non-linear speedprofile generation, because it is not quick enough. At the same time, for accuracy dependent situations motion planning is preferred for the extra flexibility.

The results for the proposed motion planning are feasible and have been shown to be good reference signals for a path follower.

Finally, the largest shortcomings from the motion planning are from two different sources. The first is the fact that all dynamics are discrete. The second is the fact that due to the nature of the simplifications that are made the lateral dynamics can not (easily) be extended to include for instance a bicycle model. Therefore this method is not suited for situations where the lateral dynamics are important, such as laptime optimization. This method however is very suited to extension of longitudinal dynamics, such as powertrain or aerodynamics models.

5.2. Recommendations and Future Work

During this research, a jerk based motion planning algorithm was designed, that has been shown to be able to tackle different goals, from time optimality to motion sickness minimization. However, the results are not perfect. Future work can be divided into two directions. First is general improvements in the optimization framework and further implementation. The second is in the equations of motion used for the optimization. Computation times for the non-linear optimization were very long compared to the length of the planned motion. In combination with bad convergence, there is a lot of improvements to be had to be able to deploy such a system in a realtime algorithm. On the other hand the EOM are currently based on point mass dynamics assuming steady-state cornering, which is a simplification that is not valid with larger accelerations.

Below mentioned are different points in which the current designed method can be improved and extended.

- Improvements in the optimization framework to reduce computation time and improve convergence for more general situations. One of the ways this can be achieved is by using a different solver that is more suited to this specific optimization problem.
- More complex longitudinal dynamics. By incorporating the likes of drivetrain dynamics and aerodynamic drag. This can be done to not only result in a speedprofile, but also reference throttle and brake inputs.
- Use of the motion planning algorithm in more complex road situations. In this research two corners
 put together with straight roads have been used. The optimization approach allows the motion to
 be much more comfortable when both longitudinal and lateral acceleration are to be combined.
- For motion planning algorithms for automated driving the goal is to be able to compute the resulting motion in realtime. Receding horizon implementation is a good candidate to keep the computation time feasible. This also allows the algorithm to be able to adapt the path and speed to changing environments.
- The reference curvature that was used in the creation of the reference motion assumes perfect knowledge. As the saying goes garbage in, garbage out. The effect of imperfect knowledge of the reference curvature has to be researched and if necessary estimators have to be used to compensate.



Appendix

A.1. Linear Speedprofile Generation Final EOM

States **x** and Inputs **u**:

$$\mathbf{x} = [u, a]$$
$$\mathbf{u} = [J]$$

The EOM:

$$u_{s+1} = u_s + a_s \cdot dt$$
$$a_{s+1} = a_s + J_s \cdot dt$$
$$dt = \frac{h}{u_{speedlimit}}$$

Constraints:

$$u_{min} \le u_s \le u_{max,s}$$
$$a_{min} \le a_s \le a_{max}$$
$$J_{min} \le J_s \le J_{max}$$

A.2. Non-Linear Speedprofile Generation Final EOM

States **x** and Inputs **u**:

$$\mathbf{x} = [u, a, t, dt]$$
$$\mathbf{u} = [J]$$

The EOM:

$$u_{s+1} = u_s + a_s \cdot dt_s$$
$$a_{s+1} = a_s + j_s \cdot dt_s$$
$$t_{s+1} = t_s + dt_s$$
$$dt_s = \frac{h}{u_s}$$

Constraints:

 $u_{min} \le u_s \le u_{max,s}$ $a_{min} \le a_s \le a_{max}$ $J_{min} \le J_s \le J_{max}$

A.3. Motion Planning Final EOM

States \mathbf{x} and Inputs \mathbf{u} :

$$\mathbf{x} = [n, u, v, a_x, a_y, t, \theta]^T$$
$$\mathbf{u} = [J_x, J_y]^T$$

A visual representation of the directions of the motion and reference:



Figure A.1: A visual representation of the simplified motion as assumed by the motion planning algorithm

In figure A.1 the direction of the longitudinal speed u is always aligned with the reference, while the lateral speed v is always perpendicular. For the angle ϕ a small angle approximation is utilised. The EOM:

$$\begin{split} n_{s+1} &= [1 + \frac{v_s \cdot \phi_s}{u_s}]^{-1} \cdot [n_s + h \cdot \phi_s - \frac{v_s}{u_s} \cdot h] \\ u_{s+1} &= u_s + a_{x,s} \cdot \frac{h}{u_s} \\ v_{s+1} &= v_s + (a_{y,s} - u_s^2 \cdot \frac{\kappa_{ref,s}}{1 + n_{s+1} * \kappa_{ref,s}}) \cdot \frac{h}{u_s} \\ a_{x,s+1} &= a_{x,s} + J_{x,s} \cdot \frac{h}{u_s} \\ a_{y,s+1} &= a_{y,s} + J_{y,s} \cdot \frac{h}{u_s} \\ t_{s+1} &= t_s + \frac{h + n_{s+1} \cdot \phi_s}{u_s} \\ \theta_{s+1} &= \theta_s + \kappa_{ref,s} \cdot h \end{split}$$

Constraints:

$$n_{min} \le n_s \le n_{max}$$
$$u_{min} \le u_s \le u_{max}$$
$$a_x^2 + a_y^2 \le a_{max}^2$$

A.4. Benchmark Format Results



Figure A.2: The resulting speedprofiles from [16]



Figure A.3: The resulting speedprofiles using the linear method in the benchmark format



Figure A.4: The resulting speedprofiles using the non-linear method in the benchmark format



Figure A.5: The resulting speedprofiles using the motion planning method in the benchmark format

A.5. Vehicle Parameters

The vehicle used can be described by the physical dimensions, the suspension, and the engine.



Figure A.6: The parameters for the vehicle used in the Prescan simulation



Figure A.7: The suspension stiffness and damping for the vehicle used in the Prescan simulation



Figure A.8: The engine torque map for the vehicle used in the Prescan simulation

A.6. Path Follower Parameters

Block Parameters: PathFollower_Preview_1	×
PreScan Block: Path Follower with Preview (mask)	
The Path Follower Model is selected in the GUI to simulate dr behavior.	iver
For more information, please click Help.	
Controller Parameters System Parameters	
Parameters	
Path Unique ID	
978	:
Mass vehicle total [kg]	
1480	:
Inertia vehicle around yaw axis [kg m2]	
2562	:
Distance between COG and front axle [m]	
1.08	:
Length of wheelbase [m]	
2.7	:
Maximum steering wheel angle [deg]	
	E
Delay time [s]	
0.1	:
Low-speed limit [km/h]	_
20	
Maximum steering wheel angle rate [deg/s]	
1200	:
Lateral controller frequency [Hz]	
200	:
OK Cancel Help	Apply

Figure A.9: Path follower settings used in the Prescan simulation
Bibliography

- ISO 22179:2009. "Intelligent transport systems Full speed range adaptive cruise control (FSRA) systems — Performance requirements and test procedures". In: (2009).
- [2] ISO 2631:1997. "Mechanical vibration and shock Evaluation of human exposure to whole-body vibration Part 1: General requirements". In: (1997).
- [3] Hanna Bellem et al. "Objective metrics of comfort: developing a driving style for highly automated vehicles". In: *Transportation research part F: traffic psychology and behaviour* 41 (2016), pp. 45– 54.
- [4] CG Lo Bianco, Aurelio Piazzi, and Massimo Romano. "Velocity planning for autonomous vehicles". In: *IEEE Intelligent Vehicles Symposium, 2004*. IEEE. 2004, pp. 413–418.
- [5] Paul T Boggs and Jon W Tolle. "Sequential quadratic programming". In: *Acta numerica* 4 (1995), pp. 1–51.
- [6] Olaf Borsboom et al. "A convex optimization framework for minimum lap time design and control of electric race cars". In: *IEEE Transactions on Control Systems Technology* (2020).
- [7] Haotian Cao et al. "An optimal model-based trajectory following architecture synthesising the lateral adaptive preview strategy and longitudinal velocity planning for highly automated vehicle".
 In: Vehicle system dynamics 55.8 (2017), pp. 1143–1188.
- [8] Yuchuan Du, Chenglong Liu, and Yishun Li. "Velocity control strategies to improve automated vehicle driving comfort". In: *IEEE Intelligent transportation systems magazine* 10.1 (2018), pp. 8– 18.
- [9] Wael Farag and Zakaria Saleh. "Tuning of PID track followers for autonomous driving". In: 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT). IEEE. 2018, pp. 1–7.
- [10] Hans Joachim Ferreau et al. "qpOASES: A parametric active-set algorithm for quadratic programming". In: *Mathematical Programming Computation* 6.4 (2014), pp. 327–363.
- [11] Abebe Geletu. Quadratic programming problems-a review on algorithms and applications (Activeset and interior point methods).
- [12] Franziska Hartwich, Matthias Beggiato, and Josef F Krems. "Driving comfort, enjoyment and acceptance of automated driving–effects of drivers' age and driving style familiarity". In: *Ergonomics* 61.8 (2018), pp. 1017–1032.
- [13] Robert Haschke, Erik Weitnauer, and Helge Ritter. "On-line planning of time-optimal, jerk-limited trajectories". In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE. 2008, pp. 3248–3253.
- [14] Lawrence L Hoberock. "A survey of longitudinal acceleration comfort studies in ground transportation vehicles". In: (1977).
- [15] Boris Houska, Hans Joachim Ferreau, and Moritz Diehl. "ACADO toolkit—An open-source framework for automatic control and dynamic optimization". In: *Optimal Control Applications and Meth*ods 32.3 (2011), pp. 298–312.
- [16] Zaw Htike et al. "Motion Sickness Minimisation in Autonomous Vehicles Using Optimal Control". In: International Conference on Robotics in Alpe-Adria Danube Region. Springer. 2020, pp. 275– 282.
- [17] Daniel Patrick Kelly. "Lap time simulation with transient vehicle and tyre dynamics". In: (2008).
- [18] Ahmet Sami Kilinc and Tamer Baybura. "Determination of minimum horizontal curve radius used in the design of transportation structures, depending on the limit value of comfort criterion lateral jerk". In: *TS06G-Engineering Surveying, Machine Control and Guidance*. 2012.

- [19] K Senthil Kumar, S Pal, and RC Sethi. Objective evaluation of ride quality of road vehicles. Tech. rep. SAE Technical Paper, 1999.
- [20] Filip Logist et al. "Multi-objective optimal control of chemical processes using ACADO toolkit". In: Computers & Chemical Engineering 37 (2012), pp. 191–199.
- [21] Roberto Lot and Francesco Biral. "A curvilinear abscissa approach for the lap time optimization of racing vehicles". In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 7559–7565.
- [22] Roberto Lot and Nicola Dal Bianco. "Lap time optimisation of a racing go-kart". In: *Vehicle System Dynamics* 54.2 (2016), pp. 210–230.
- [23] Roberto Lot and S Evangelou. "Lap time optimization of a sports series hybrid electric vehicle". In: 2013 World congress on engineering. 2013, pp. 1–6.
- [24] Charles C MacAdam. "An optimal preview control for linear systems". In: (1980).
- [25] Charles C MacAdam. "Application of an optimal preview control for simulation of closed-loop automobile driving". In: *IEEE Transactions on systems, man, and cybernetics* 11.6 (1981), pp. 393– 399.
- [26] Mark Mario Morando, Long T Truong, and Hai L Vu. "Investigating safety impacts of autonomous vehicles using traffic micro-simulation". In: *Australasian Transport Research Forum*. 2017, pp. 1–6.
- [27] Brian Paden et al. "A survey of motion planning and control techniques for self-driving urban vehicles". In: *IEEE Transactions on intelligent vehicles* 1.1 (2016), pp. 33–55.
- [28] Merja Penttinen and Juha Luoma. "Acceptance and use of ADAS". In: 8th Transport Research Arena, TRA 2020-Conference cancelled. Liikenne-ja viestintävirasto Traficom. 2020, p. 66.
- [29] Hongbin Ren et al. *Development of effective bicycle model for wide ranges of vehicle operations*. Tech. rep. SAE Technical Paper, 2014.
- [30] Astrid Rupp and Michael Stolz. "Survey on control schemes for automated driving on highways". In: *Automated driving*. Springer, 2017, pp. 43–69.
- [31] Taishi Sawabe et al. "Comfortable autonomous navigation based on collision prediction in blind occluded regions". In: 2015 IEEE International Conference on Vehicular Electronics and Safety (ICVES). IEEE. 2015, pp. 75–80.
- [32] Antoine Schmeitz et al. "Towards a generic lateral control concept for cooperative automated driving theoretical and experimental evaluation". In: 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS). IEEE. 2017, pp. 134– 139.
- [33] Jarrod M Snider et al. "Automatic steering methods for autonomous automobile path tracking". In: Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08 (2009).
- [34] Yuyang Wang, Jean-Rémy Chardonnet, and Frédéric Merienne. "Speed Profile Optimization for Enhanced Passenger Comfort: An Optimal Control Approach". In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE. 2018, pp. 723–728.
- [35] Xi Wei and Giorgio Rizzoni. *Objective metrics of fuel economy, performance and driveability-A review*. Tech. rep. SAE Technical Paper, 2004.
- [36] Robert B Wilson. "A simplicial algorithm for concave programming". In: *Ph. D. Dissertation, Grad*uate School of Bussiness Administration (1963).
- [37] Yuki Yoshihara et al. "Autonomous predictive driving for blind intersections". In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE. 2017, pp. 3452–3459.