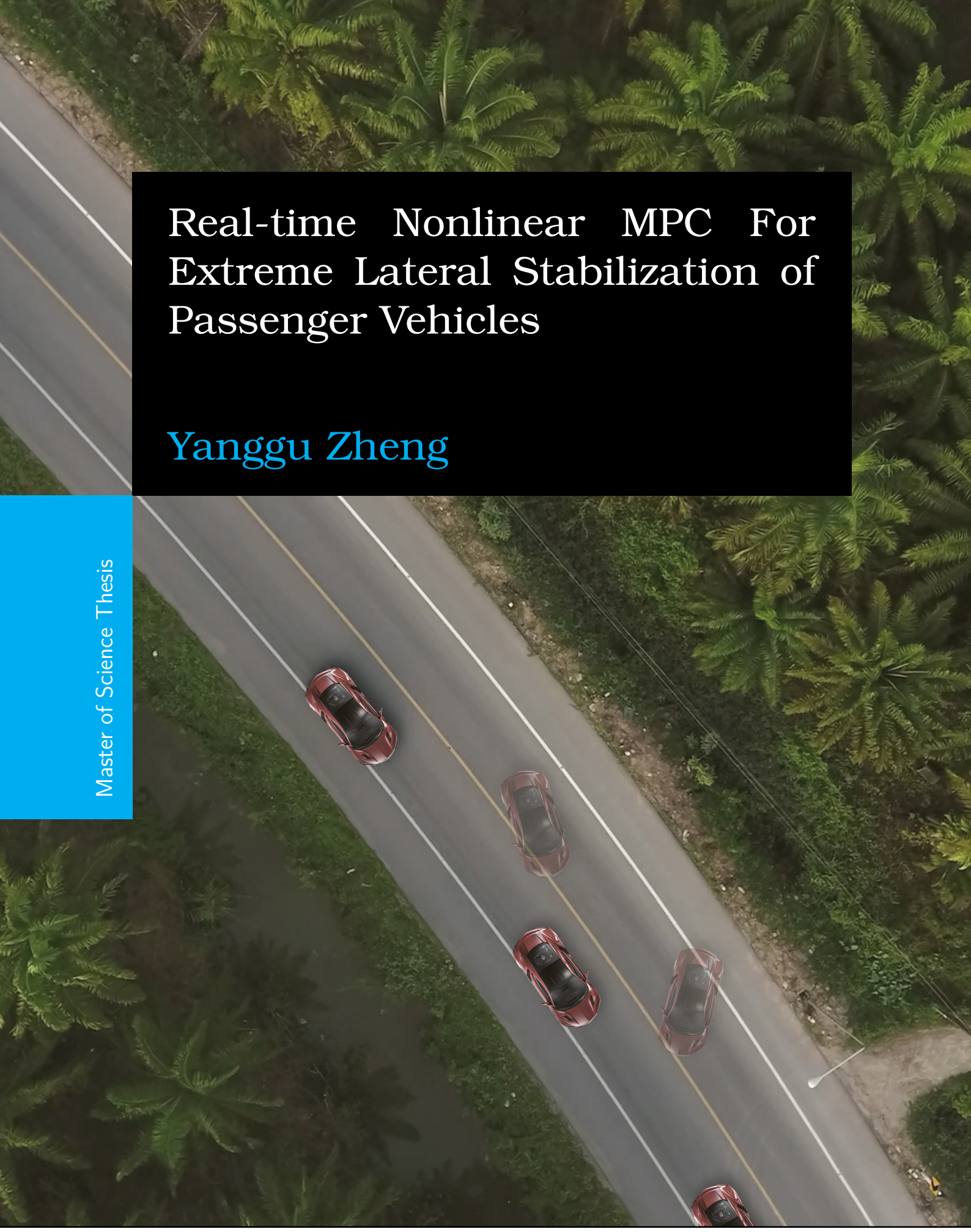

Real-time Nonlinear MPC For Extreme Lateral Stabilization of Passenger Vehicles

Yanggu Zheng

Master of Science Thesis

TU Delft Delft University of Technology

Faculty of Mechanical, Maritime and Materials Engineering

Real-time Nonlinear MPC For Extreme Lateral Stabilization of Passenger Vehicles

Master of Science Thesis

For the degree of Master of Science in Vehicle Engineering at Delft University of Technology

Yanggu Zheng

July 3, 2018

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

Abstract

Loss of lateral stability is still a major cause of road accidents in recent years. Nonlinear model predictive control (NMPC) is regarded as a powerful tool to improve vehicle safety by fully utilizing the tire-road friction. However, the computational load is excessive for on-board hardware, thus NMPC is yet distant from real-time implementation in vehicle control. To tackle the problem, this study proposes a method to improve the computational efficiency in NMPC. First, a lookup table of initial guess of the NMPC solution is calculated based on a piecewise-affine (PWA) approximation of tire's nonlinear behavior. Then, a local optimization process starting from the initial guess searches for the optimal control input, using perpendicular search plus line search method. The proposed method was examined through a set of numerical tests and in real-world scenarios using experimentally calibrated multibody model. The code was also tested on dSPACE for proof of real-time implementation. Results show a significant reduction in computational time, thus real-time implementation has been achieved with a huge margin. The loss in the accuracy of the optimal solution is negligible. The performance in improving vehicle safety is promising too, as the vehicle can be recovered from unstable motion with body slip angle up to 30 degrees.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisor Dr. Barys Shyrokau for his brilliant assistance through the research and the writing of the thesis. I would like to also thank my girlfriend for editing the cover photo for me. Last but not least, thanks to my family who supported my study in the last two years.

Delft University of Technology
July 3, 2018

Yanggu Zheng

”If in doubt, flat out.”
— *Colin McRae*

Chapter 1

Introduction

1-1 Background

According to multiple resources of statistics [1, 2, 3], fatalities caused by road accidents each year has dropped by 66.2 % in the past 25 years in the EU. Even greater improvement can be discovered when taking into account the influence of increasing use of cars, as the road fatalities per billion vehicle kilometer or per 10,000 registered vehicles has decreased by more than 80 % in several countries including Austria, Belgium and Denmark. Meanwhile, the number of injury accidents did not decrease as much as fatal accidents do, with over 1 million cases happened in the year 2014 in EU countries and 1.7 million cases reported in the US. Adverse road conditions is a major cause of crashes, contributing to 24 % of total crashes in the US from 1995 to 2005 [4]. The number did not drop greatly through the years, as until 2014 there are still 14 % of fatal accidents and 20 % of total accidents caused by low-friction road surfaces [5]. It is also surprising to notice that among all fatal crashes, around 20 % happened when the driver's negotiating curves and around 30 % are collisions with fixed objects like poles, trees and guardrails. It appears that a loss of yaw stability is a common factor that contributed to these types of accidents and they could possibly be avoided if the vehicle's steering was under control.

The widespread application of electronic systems in active chassis systems has been enabling various active safety functions that aim to prevent accidents from happening, among which the anti-lock braking system (ABS) and electronic stability program (ESP) are two well-known examples. It has been a great success since their introduction as they mitigate the shortcomings of pure mechanical systems thus help keep the vehicle under the driver's control. These systems can effectively avoid accidents in most cases. According to statistics from NHTSA[6], electronic stability control reduced the fatal single-vehicle crashes by 36 %, fatal run-off-road by 36 % and fatal rollovers by 70 % for passenger cars. However, there are still exceptional situations where the effort of traditional vehicle dynamics control systems turns out to be vain, although human error could also be blamed. List of possible causes includes aggressive steering maneuvers (for collision avoidance), angled rear-end collisions, abrupt changes in friction level, etc. It is commonly observed from these cause that during

a short period of time the rear axle is unable to generate as much lateral force as demanded in order to counteract the yaw moment exerted by the front axle. As a result, the vehicle's side-slip angle increased quickly, leading the rear tires to saturation. Saturated rear tires significantly change the yaw dynamics of the vehicle as the lateral force no longer increase monotonically along with the side-slip angle. As stated in [7], the ESP system was developed to limit the side slip angle within the range where average drivers can handle. But when a large side slip angle already occurs, the system is unable to bring it down.

1-2 Model Predictive Control of Vehicle Dynamics

Model predictive control has received a good amount of interest from automotive industry, who desires to improve road safety. Despite the wide application of active safety systems like anti-lock braking system (ABS) and electronic stability control (ESC), road accident is still a major cause of unnatural deaths [8]. Loss of lateral stability contributed to 20-30% of total road fatalities in US[9] in 2015. Manufacturers have been trying to achieve better utilization of tire-road friction by means of integrated vehicle dynamics control (IVDC), i.e. coordinated control of multiple actuators including steering, braking, (semi-)active suspension, etc. Control allocation is a major problem in coordinating the actuators and the nonlinear dynamics at the limit of friction further increased the difficulty of integrated control. It has been a challenging task in vehicle motion control to handle the tire's highly coupled and nonlinear behavior in generating longitudinal and lateral forces which is mainly a function of longitudinal slip ratio κ and side-slip angle α and is also subject to variations in friction level μ, vertical load F_z, camber angle γ, etc. Quick development of computers and advances in numerical optimization have enabled the application of nonlinear model predictive control in many systems. NMPC is a powerful method to incorporate nonlinearities in the plant and coordinate multiple control inputs. Thus it can be a good solution to IVDC. The modelling of vehicle dynamics has been thoroughly studied and documented through years. The characteristics of tire-road friction can also be well described with empirical models like Pacejka's magic formula. However, the accuracy of modelling comes at the cost of complexity in computation. The complexity of numerical optimization problem hinders the application of NMPC in quick dynamics. Borrelli et al[10] first used NMPC to control the front steering angle alone and pointed out the problem in computational complexity. In the follow-up study[11], the researchers also included the braking torque on individual wheels as the control input. The authors claimed that simulation was time-consuming and tuning was difficult. A recent study[12] claimed to have achieved real-time NMPC control of torque-vectoring using a commercial toolbox. The numerical quality is not satisfactory, though, as the final cost could be 28% higher than the actual optimum in some cases. The problem could be related with the choice of optimization method and tuning of parameters. Global optimization of non-convex object function still remains an unsolved problem. Existing algorithms can obtain good object values but require very intensive computation and can hardly converge within the sampling time. The problem is even more severe when considering real-time on-board implementation, where the hardware for computation is less powerful. Therefore, how to reduce the computational time is regarded as the top question of all.

By manipulating the model of dynamics, the optimization problem can be simplified. Beal and Gerdes[13] developed an affine-force-input model for the control of active front steering,

which led to a linear MPC. Lateral force on the front tire was adopted as the control input of MPC. The force was then interpreted into steering angle for the low-level controller to follow. The rear tires were treated as linear and constraints on vehicle state were introduced in the optimization to guarantee the validity of linear models. Yet not all of the situations can be properly described with linear models. For limit-handling control, the nonlinear dynamics must be accurately described. In addition, external disturbances like collisions and hydroplaning may directly force the vehicle into the range of motion where the model deviates drastically from the actual behavior of the vehicle. Falcone et al[14] looked into the linear time-varying (LTV) modelling. The LTV model avoids the highly-demanding nonlinear optimization but still incorporates the nonlinear dynamics by changing the parameters on the linear model with respect to system states. As a result, the controller simply needs to solve a quadratic programming problem repetitively. LTV-MPC is not valid for long-term prediction, though. The control input given by LTV-MPC is only optimal if the system follows the linearized dynamics through the entire prediction horizon, which is usually not the case. Hybrid MPC has also been investigated by Bernardini et al[15] and Di Cairano et al[16], in order to achieve a balance between accuracy and simplicity. PWA model describes nonlinear dynamics with an approximation using linear sections. Hybrid MPC features moderate complexity of computation and sub-optimal solutions. However, it might be difficult to further improve model accuracy when implementing hybrid MPC. The efficiency of solving mixed-integer quadratic programming (MIQP) problem drops drastically when the number of modes in the system increases.

Apart from changing the modelling techniques, the optimization process itself can also be improved. Tondel[17] attempted with explicit MPC, which aims to move intensive computation offline using the multi-parametric programming technique. The technique was also suggested by Bernardini et al[15] and later on adopted by Di Cairano et al[16] to enable real-time experiments. Explicit MPC performs only tree-search operation and evaluation of affine functions online. But the storage of offline solution demands plenty of memory. Bernardini claimed that around 5,000 space partitions is needed for approximating the hybrid MPC solution and the number grows to 11,277 in Tondel's study where a nonlinear model was used. Nevertheless, these studies limited the state variables in a rather narrow range and the prediction horizons were short ($N_p = N_c = 3$ for both).

Improvements are also possible within the optimization process itself. It is generally believed that the initial guess (i.e. starting point) plays an important role in numerical optimization[18]. The initial guess is usually required as input or randomly determined. For non-convex problems, multiple random initial guesses might be necessary to increase the chance of finding the global optimum. The computational complexity increases as a consequence, though. On the other hand, artificially introduced randomness, as implemented in simulated annealing and genetic algorithm, could to some extent avoid being trapped at a local minimum. Nevertheless, initial guess still has a role to play in these algorithms. It has been studied that good initial guess can accelerate the optimization process[19]. It is also reasonable to roughly store the solution to the optimization problem to explicitly serve as the initial guess. Zeilinger et al[20] proposed a method for real-time implementation of MPC in large-scale linear systems. The initial guess was calculated with PWA function that approximates the actual solution to the QP problem. Consequently, only limited iteration is needed in online optimization to obtain a good cost value.

1-3 Focus of the Study

The study develops a nonlinear MPC controller of vehicle dynamics. By combining the steering and braking action, the utilization of tire-road friction can be maximized. Active front steering and individual generation of braking torque is enabled by actuators that are commercially available with acceptable cost, hence experimental tests may be carried out in the future. The incorporation of nonlinear dynamics also enlarges the range of motion where stabilization of the vehicle can be achieved.
As stated above, intensive computation has hindered the application of nonlinear MPC in the control of vehicle dynamics. By simplifying the vehicle model in nonlinear MPC, the computation time can be reduced but the performance was also degraded. Hence, this study focuses on finding a method to enable real-time implementation of nonlinear MPC in the stabilization of a passenger vehicle with a fully nonlinear vehicle model, by means of improving the computational efficiency.
Regarding the scenario of application, past studies have favored standard tests including double-lane-change, sine-with-dwell, etc.. These scenarios mainly tests the capability of stabilizing the vehicle against aggressive maneuvers, where the magnitude of the destabilizing impact is limited by the tire-road friction. In real-world driving, however, there may be external disturbances that exceed the friction limit of the vehicle. A lateral collision on the rear end can force the vehicle into unstable motion (featuring large body slip angle β) so quickly that the active safety systems are unable to counteract. Recovery of such motion has received less interest, despite the catastrophic aftermath. Yang et al [21] worked on the post-collision trajectory control using braking forces alone, with front steering angle assumed to be zero. The controller developed in this thesis will try to handle the similar situation but will only concentrate on achieving the quick stabilization of the vehicle motion. It still makes sense to temporarily exclude vehicle trajectory from the scope because once stabilized, the vehicle can be easily controlled to follow the desired trajectory. Additionally, if the duration of the unstable motion is short enough, the vehicle may not deviate far from the original trajectory.

1-4 Proposed Solution

In order to achieve real-time implementation without any trade-off regarding control performance, the computational efficiency in solving nonlinear MPC problem must be solved directly. This study proposes to accelerate the numerical optimization process in nonlinear MPC with the help of properly determined initial guesses. The initial guesses must be close enough to the actual solution so that only local optimization should be performed online. To find such initial guesses, hybrid model predictive control shall be exploited. If the vehicle dynamics can be properly approximated with a hybrid model, the control input given by the corresponding hybrid MPC controller should yield a sub-optimal control input and hence a good initial guess to the nonlinear MPC problem. By solving the hybrid MPC problem offline and store the solutions explicitly, the online computation can be significantly reduced. It may seem even more helpful if the initial guess is given directly by the offline solution to the nonlinear MPC problem. However, there is no algorithm that guarantees the global optimality, even when given sufficient time. In contrast, the mixed-integer quadratic programming

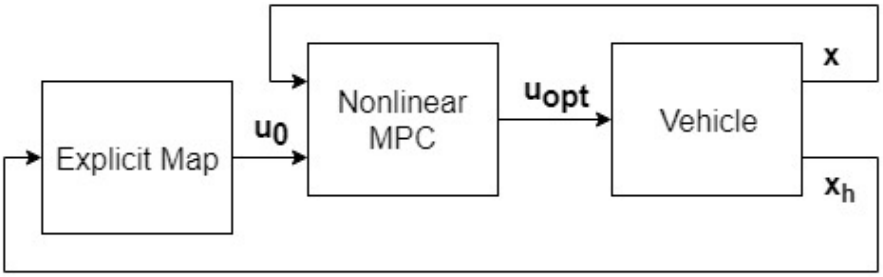

Figure 1-1: Schematic drawing of proposed system

problem corresponding to hybrid MPC can be solved efficiently. Since a hybrid system can be described with only binary variables and continuous variables, the solution to the hybrid MPC problem is indeed globally optimal [22]. Nevertheless, the efficiency of local optimization algorithm should be improved, due to the lack of gradient information of the objective function in nonlinear MPC. To avoid extra computational burden introduced by numerical evaluation of gradient, a gradient-free optimization algorithm should be implemented. The overall description of the proposed method is given by Figure 1-1. In this study, the control input to the vehicle is $\mathbf{u} = [F_{bFL}, F_{bFR}, F_{bRL}, F_{bRR}, \delta]^T$, i.e. independently controlled braking force on each wheel plus the steering angle of front wheels. Vehicle states $\mathbf{x_h} = [r, \beta]^T$ are fed to the explicit map and states $\mathbf{x} = [r, \beta, v_x]^T$ are fed to the NMPC controller.

1-5 Layout of the Thesis

The thesis report is structured as follows. In Chapter 2, a hybrid model of vehicle's lateral and yaw dynamics is derived based on a PWA description of the tire's combined-slip behavior. Next, the corresponding hybrid MPC controller is generated and the solution is calculated offline. Chapter 4 includes the nonlinear MPC problem with a nonlinear 2-track model, as well as a local optimization algorithm for the controller to solve the nonlinear MPC problem. The controller is then tested under a scenario that reflects a certain type of real-world accident in Chapter 5. The benchmarking test of the numerical performance of the proposed controller as well as the proof of real-time implementation with dSPACE is documented in Chapter 6. Chapter 7 summarizes the contribution and shortage of this study and recommendations for future works. Some technical details and less important results are placed in the appendices and a draft journal paper is attached at the end of the thesis.

Chapter 2

Hybrid Modelling of Vehicle Dynamics

In this chapter, a piecewise affine model of the tire is developed first, which extracts the combined-slip behavior in the generation of friction force. Using this tire model, the vehicle's lateral and yaw dynamics can be modelled as a hybrid system that contains multiple linear local dynamics. This hybrid model of vehicle dynamics will be exploited in the next chapter as the key component in the formulation of the hybrid MPC problem for vehicle stabilization.

2-1 Piecewise Affine Tire Model

In the modelling of vehicle's lateral dynamics, the nonlinear characteristics are mainly introduced by the frictional force between the tire and the road. There are several empirical models that describe the characteristics of the generation of the frictional force, including Burckhardt model, Dugoff model, Pacejka's magic formula, etc.. Among these empirical models, Pacejka's magic formula is widely adopted because of its good accuracy and incorporation of additional factors. In the magic formula, the longitudinal force and the lateral force (F_x and F_y) are described as functions of longitudinal slip ratio κ, side slip angle α and other parameters. In this study, the influence of vertical load F_z is incorporated while the rest are omitted. In order to enable hybrid modelling of vehicle dynamics, the vertical load is assumed constant in the PWA tire model.
PWA model approximates the nonlinear behavior with multiple linear sections. As mentioned above, Pacejka's magic formula describes the longitudinal and lateral force as separate functions of κ and α (see Figure 2-1). The straightforward approach of PWA tire modelling is to approximate both of the functions. This approach has two drawbacks. First of all, the complexity of the consequent hybrid vehicle model will increase. Moreover, an additional low-level controller is required to track the reference value of κ.

An alternative solution has been adopted instead, where the Pacejka's model is modified into a function of α and braking force F_b. The braking force F_b is directly adopted in the control

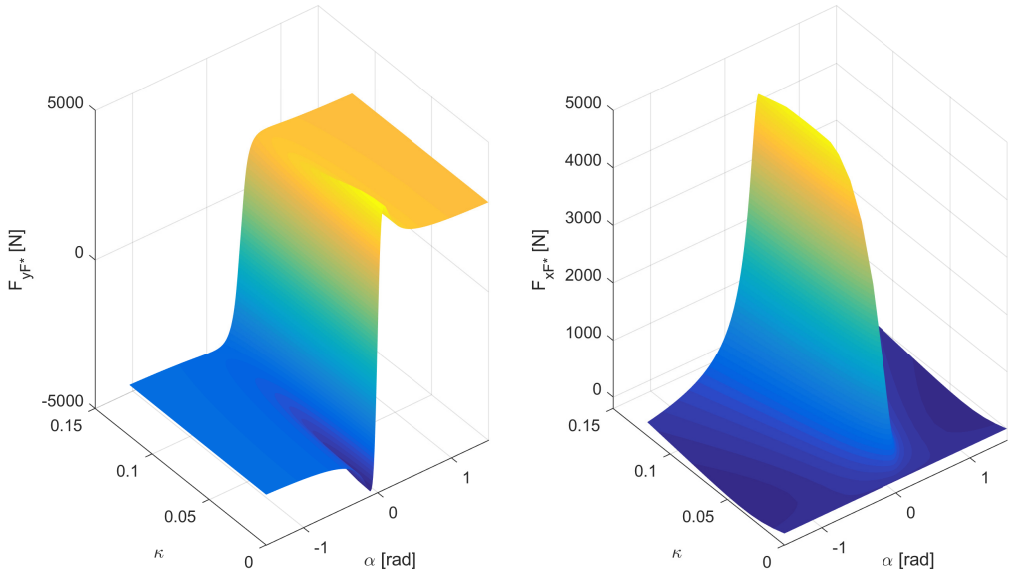

Figure 2-1: Tire forces given by Pacejka's magic formula

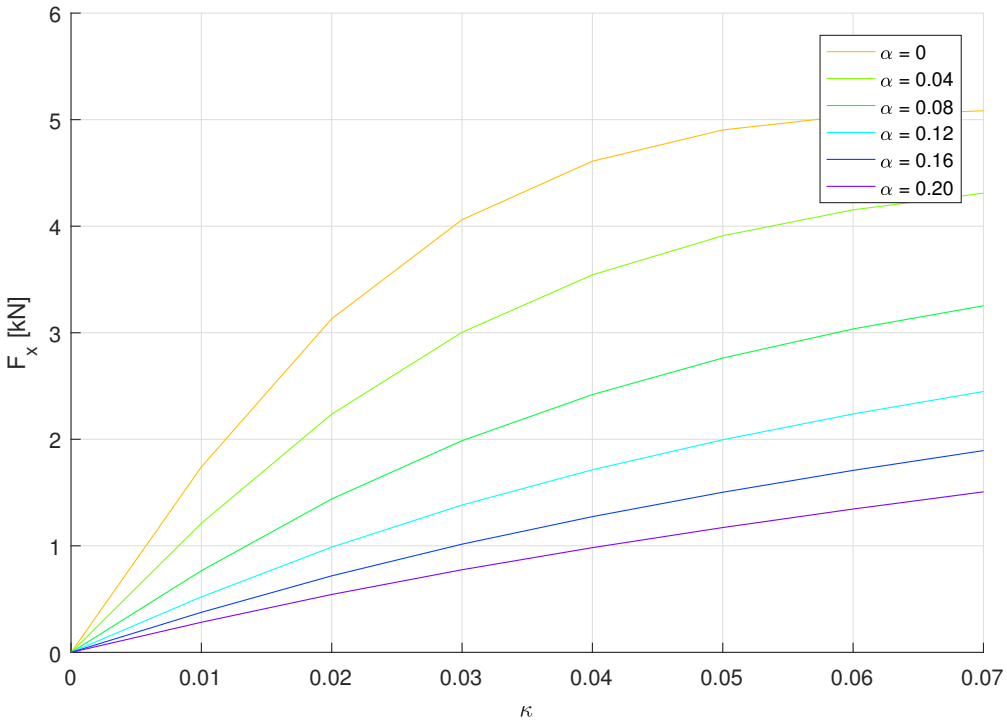

Figure 2-2: Longitudinal force as function of longitudinal slip ratio at different side-slip angles

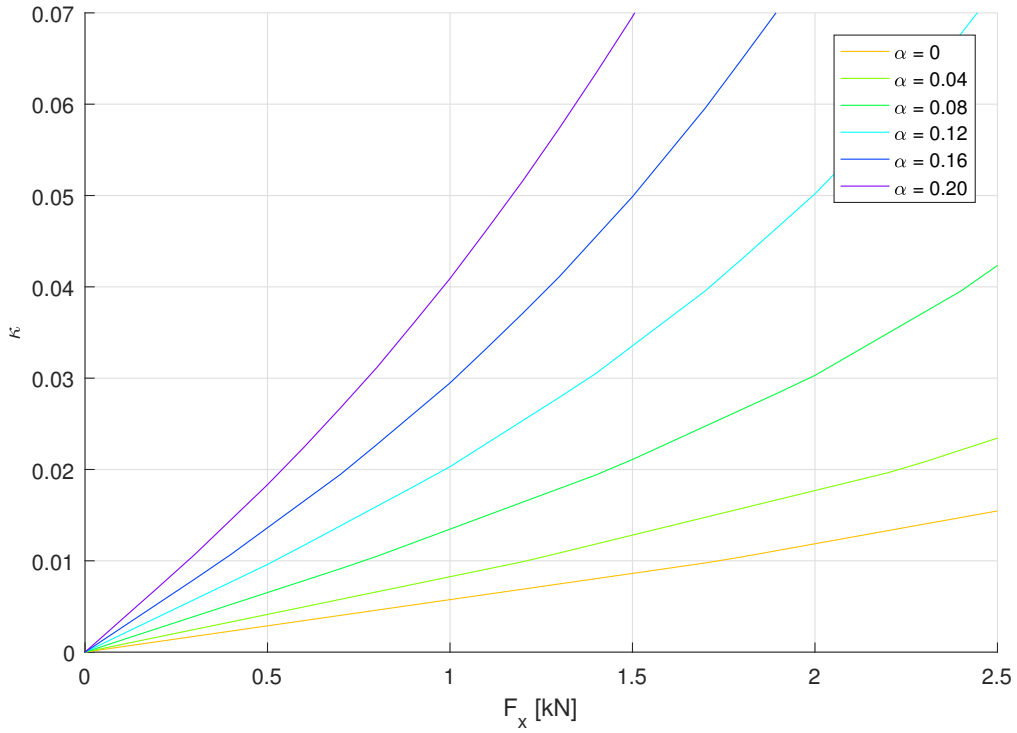

Figure 2-3: Slip ratio as function of longitudinal force at different side-slip angles

input. The modification is done follows. By first neglecting the wheel dynamics, $F_b = -F_x$ can be assumed. The vertical load on a single tire is calculated according to vehicle parameters. Next, given a certain value for α, F_b is a function of κ, as described by the magic formula. By evaluating F_b with different values for κ, the inverse function, i.e. κ as a function of F_b, can be generated using 1-D interpolation (see Figure 2-3). It should be emphasized of course, that the inverse function only exists when F_b grows monotonically with κ. Hence, the upper limit of braking force should be chosen with extra care. By repeating the previous steps with different values for α, a surface of F_y versus F_b and α can be generated, as given in Figure 2-4 and Figure 2-5. The surface shows following features:

1. The peak of F_y is obtained at a narrow range of α. This range is hardly influenced by F_b;

2. Increasing F_b has a negligible effect on the value of F_y;

3. When α is large, it may be impossible to find a corresponding κ for the desired F_b. Hence the available F_b is dependent on α;

The linear sections for approximating the modified nonlinear tire model should capture these features in the simplest possible way (see Figure 2-4 and Figure 2-5). First of all, a rectangular

section is placed in the neighborhood of $\alpha = 0$, spanning 0.06 rad on each side. The peak of F_y is found at $\alpha = \pm 0.09 rad$ and thus one rectangular section on each side is needed to connect the edge of the first section to the peak line. The previous sections are solely defined with the upper and lower bounds on α and F_b. Due to the difference in vertical loads, the upper bound of braking force is 2.5 kN for the front tires and 1.5 kN for the rear tires. After that, a trapezoid area on each side is added, which represent the descent in both lateral force and braking force allowed by the friction limit. The descent is modelled by an additional linear constraint on F_b. The linear constraint intersects the previous upper bound at $\alpha = 0.12 rad$ and $\alpha = 0.15 rad$ for the front and rear tires, respectively, and crosses zero at $\alpha = 0.5 rad$ for all tires. The last sections beyond that describe the lowered rate of reduction in F_y with α ranging from 0.5 rad to 1.5 rad. Taking into the fact that braking action is not available anymore, the last two sections are to pieces of straight line. To avoid numerical issues, the upper bound of F_b in the last sections is chosen as 1.0×10^{-5} and hence they are actually very narrow rectangles. In total 7 sections is needed to model a single tire. Each section represents a mode of tire behavior. The mathematical expression of the PWA tire model is given by (2-1). F_{yij} is the lateral force of tire i in mode j and the mode is active when the corresponding k constraints are satisfied. The parameters of the PWA tire model can be found in Appendix A.

$$\begin{aligned} & F_{yij} = a_{ij}\alpha_i + b_{ij}F_{bi} + c_{ij} \\ & d_{ijk}\alpha_i + e_{ijk}F_{bi} + g_{ijk} \leq 0 \\ & i \in \{FL, FR, RL, RR\} \\ & j \in \{1, 2, \cdots N_{modes}\} \\ & k \in \{1, 2, \cdots N_{constraints}\} \end{aligned} \tag{2-1}$$

2-2 Hybrid 2-track Model

In order to incorporate the yaw moment generated by braking actions, the adoption of a 2-track vehicle model is necessary. The original 2-track vehicle model can be expressed with (2-2). The force components are explained by (2-3). Approximations are needed to formulate the original model into a hybrid (PWA) model. First of all, the longitudinal dynamics are ignored and v_x is fixed to 20 m/s. Next, the effect of δ on the transformation of tire forces is neglected, i.e. $sin\delta \approx 0, cos\delta \approx 1$. As a result, the equations of motion are now linear functions of state variables and control inputs. Again, using small angle approximation, α_F and α_R can be expressed as linear functions of $\mathbf{x}$ and $\mathbf{u}$, as given in (2-4). The equality between the left and right tire on each axle is assumed with the exclusion of the influence of toe-in angle and Ackermann steering geometry. Using the tire model developed in the previous section, F_{yi} and the corresponding constraints are PWA functions of α_i and F_{bi}. Since linear transformation preserves linearity, F_{yi} is a PWA function of $\mathbf{x}, \mathbf{u}$. Eventually, the equations of motion are PWA functions of state variables and control inputs and hence the vehicle model is indeed a PWA model.

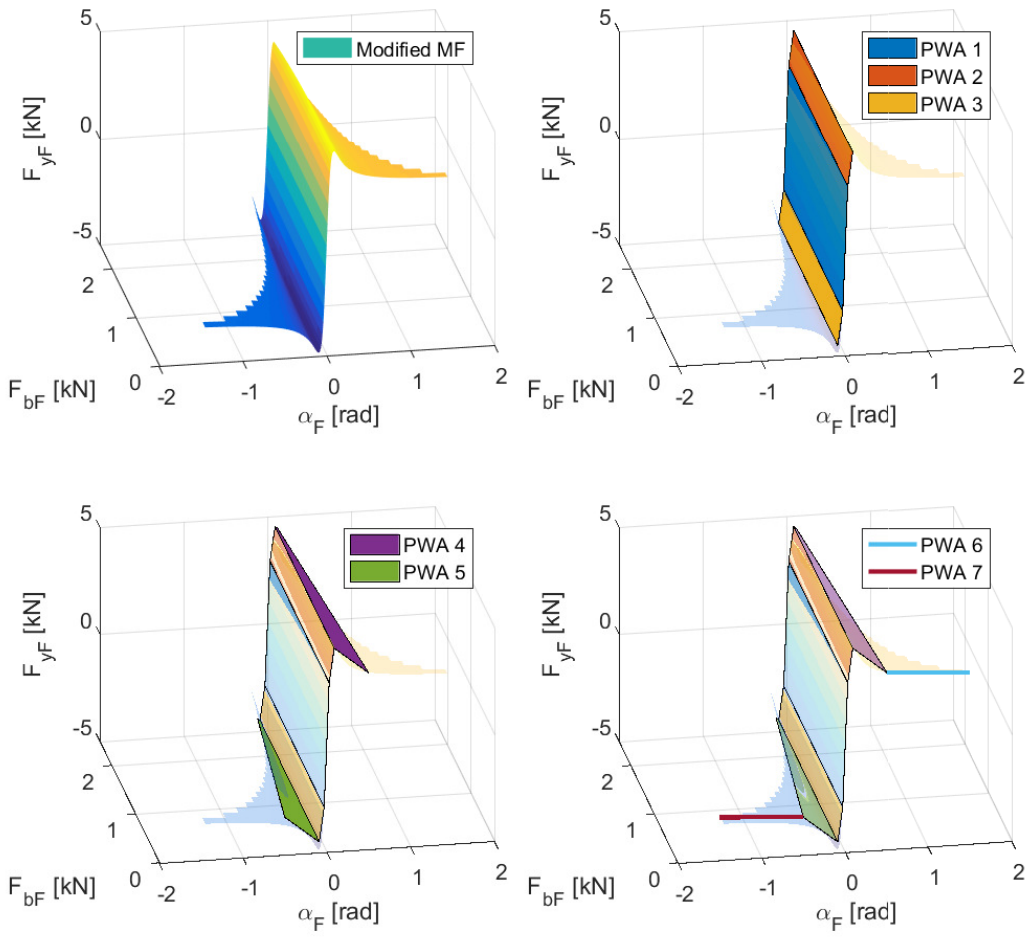

Figure 2-4: PWA modelling of front tires, compared to Pacejka's magic formula

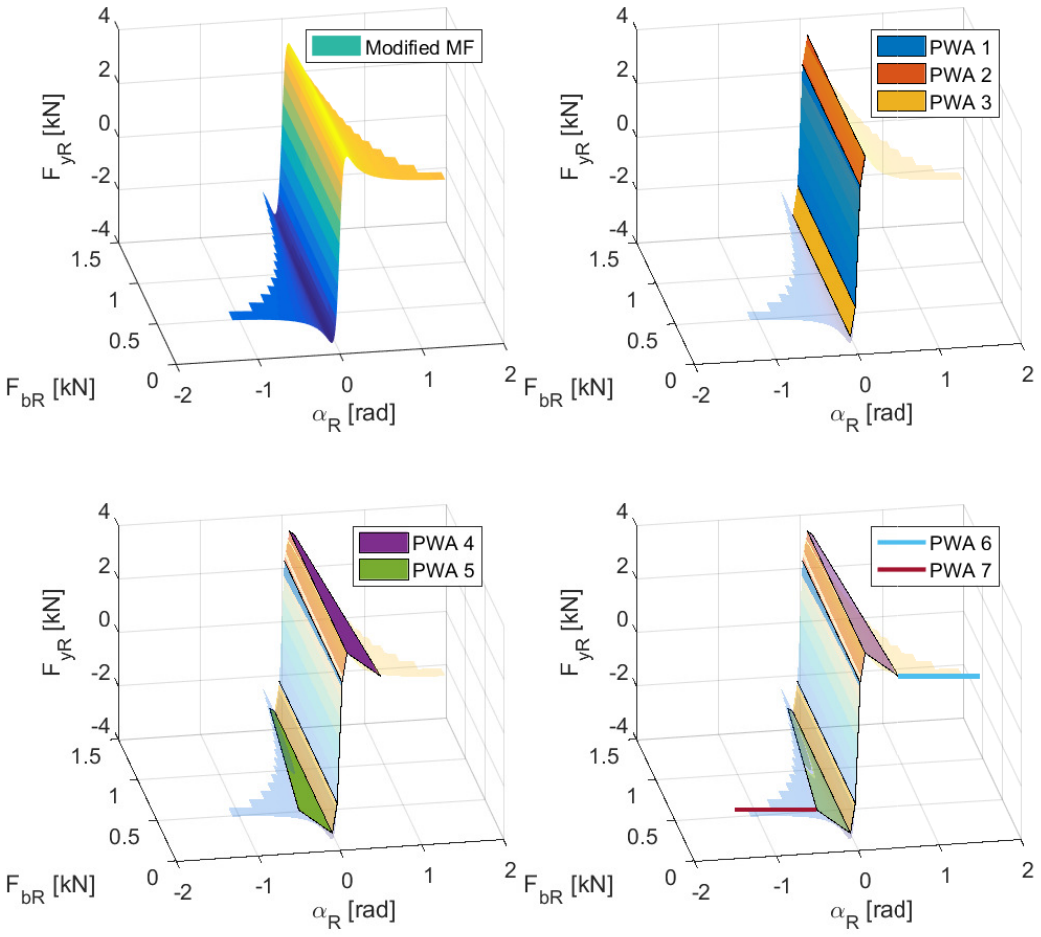

Figure 2-5: PWA modelling of rear tires, compared to Pacejka's magic formula

$$\begin{aligned}
\dot{r} &= \frac{1}{I_z}\left(F_{yF}l_F - F_{yR}l_R + (F_{xR} - F_{xL})\frac{B}{2}\right) \\
\dot{\beta} &= \frac{1}{m}(F_{yF} + F_{yR}) - r \\
\dot{v}_x &= \frac{1}{m}(F_{xL} + F_{xR})
\end{aligned} \tag{2-2}$$

$$\begin{aligned}
F_{yF} &= (F_{yFL} + F_{yFR})\cos\delta - (F_{bFL} + F_{bFR})\sin\delta \\
F_{yR} &= F_{yRL} + F_{yRR} \\
F_{xL} &= -F_{bFL}\cos\delta - F_{yFL}\sin\delta - F_{bRL} \\
F_{xR} &= -F_{bFR}\cos\delta - F_{yFR}\sin\delta - F_{bRR}
\end{aligned} \tag{2-3}$$

$$\begin{aligned}
\alpha_{FL} = \alpha_{FL} = \alpha_F &= \delta - \beta - r * l_F/v_x \\
\alpha_{RL} = \alpha_{RR} = \alpha_R &= r * l_R/v_x - \beta
\end{aligned} \tag{2-4}$$

As previously mentioned, each tire has 7 modes. The individual control of braking force means the mode of each tire is independent of each other. Therefore, the number of combinations of tire modes is $7^4 = 2,401$. Each combination will result in a different mode of vehicle dynamics, i.e. the PWA 2-track model has 7^4 modes. Such amount of local dynamics is not at all desired by the hybrid MPC (which will be discussed in the next chapter). The number can be significantly reduced to $7^2 = 49$ by taking into account the following facts. First, the equality of side slip angle on each axle has been assumed in (2-4). In addition, the tire modes are distinguished by the value of α alone. Hence, the left and right tires on each axle have to be in the identical mode, meaning that the independence of tire modes only exists between the front and rear axle. Because each axle has 7 modes, there are 7^2 modes in the vehicle dynamics. The equations of motion must be formulated into a discrete-time state-space model for the implementation of hybrid MPC controller. The continuous-time state-space model can be determined first, as given by (2-5). The discretization of state-space model is achieved by using (2-6). The choice of sampling time $T_s = 0.05s$ will be discussed in the next chapter and determined by comparing the closed-loop performance. The values of $A^d_{j_F,j_R}$, $B^d_{j_F,j_R}$ and $f^d_{j_F,j_R}$ can be found in Appendix B.

$$\begin{aligned}
A_{j,k} &= \begin{bmatrix} -2\frac{a_{Fj}{l_F}^2 + a_{Rk}{l_R}^2}{I_z u} & -2\frac{a_{Fj}l_F - a_{Rk}l_R}{I_z} \\ -2\frac{a_{Fj}l_F + a_{Rk}l_R}{Mu} - 1 & -2\frac{a_{Fj} + a_{Rk}}{M} \end{bmatrix} \\
B_{j,k} &= \begin{bmatrix} \frac{b_{Fj}l_F + B/2}{I_z} & \frac{b_{Fj}l_F - B/2}{I_z} & \frac{-b_{Rk}l_R + B/2}{I_z} & \frac{-b_{Rk}l_R - B/2}{I_z} & \frac{2a_{Fj}l_F}{I_z} \\ \frac{b_{Fj}}{M} & \frac{b_{Fj}}{M} & \frac{b_{Rk}}{M} & \frac{b_{Rk}}{M} & \frac{2a_{Fj}}{M} \end{bmatrix} \\
f_{j,k} &= \begin{bmatrix} 2\frac{c_{Fj}l_F - c_{Rk}l_R}{I_z} \\ 2\frac{c_{Fj} + c_{Rk}}{M} \end{bmatrix}
\end{aligned} \tag{2-5}$$

$$\begin{aligned}
\mathbf{A^d_{j_F,j_R}} &= e^{\mathbf{A_{j_F,j_R}}T_s} \\
\mathbf{B^d_{j_F,j_R}} &= \mathbf{A^{-1}_{j_F,j_R}}\left(\mathbf{A^d_{j_F,j_R}} - \mathbf{I}\right)\mathbf{B_{j_F,j_R}} \\
\mathbf{f^d_{j_F,j_R}} &= \mathbf{f_{j_F,j_R}}T_s
\end{aligned} \tag{2-6}$$

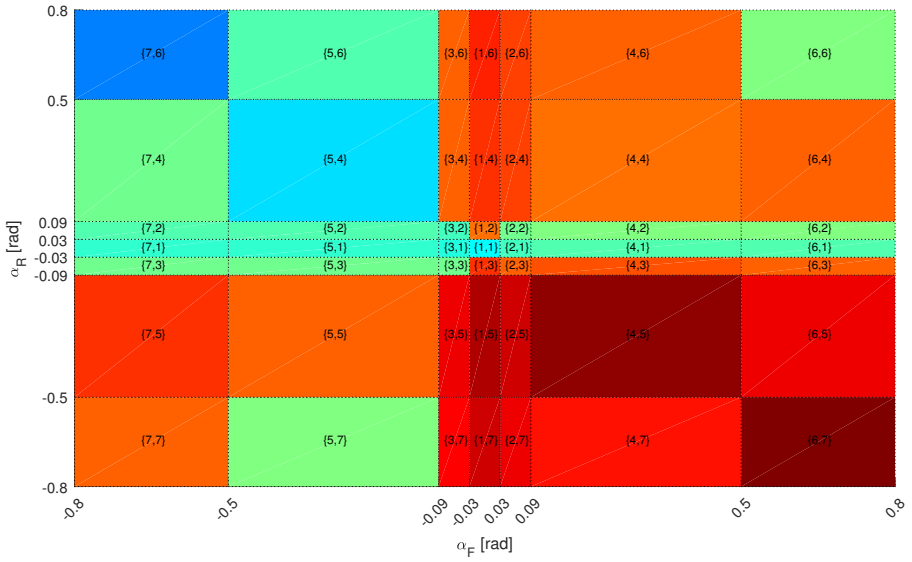

Figure 2-6: Local stability of local dynamics. The first index in the brackets indicates the mode of the front axle, the second the rear axle.

2-3 Analysis on Local Stability

The stability of each mode of local dynamics is of interest, in order to understand how the vehicle behaves. The stability of discrete-time state-space model can be determined by calculating the eigenvalues of the system matrix $\mathbf{A^d}$. If all the eigenvalues have a magnitude smaller than 1, the system is stable. However, the existence of disturbance term $\mathbf{f^d}$ introduced extra complexity. Even if the system matrix scales down the state variables, the disturbance may act in the opposite direction to finally magnify them. Contrarily, if the system matrix tends to scale up the state variables, the disturbance may also counteract to make them decrease. Hence, the disturbance vector should first be expressed by the linear combination of eigenvectors and then be added to the eigenvalues. If the magnitude of this sum is smaller than 1, a mode of local dynamics is considered stable (2-7). By checking the validity for all the modes in the hybrid vehicle model, Figure 2-6 has been obtained, on which the stability of each mode is shown by the color. Stable modes are in blue and green colors. Bluer color means a quicker reduction of the state variables. Unstable modes are in orange and red colors. Redder color means a quicker growth of the state variables.

$$\begin{aligned} &\mathbf{w} = \mathbf{V^{-1}} f^d \\ &\forall q, |\lambda_q + w_q| < 1 \end{aligned} \tag{2-7}$$

Chapter 3

Hybrid Model Predictive Control

With the hybrid vehicle model derived in the previous chapter, the stabilization problem can be formulated as an optimization problem of an objective function. The formulation of the optimization will be introduced first in this chapter, followed with some insight into the solution to such problems implemented in hybrid model predictive control. The hybrid MPC controller is generated and the parameters are tuned to achieve good closed-loop performance. The solution to the hybrid MPC problem is finally calculated offline and stored explicitly, which will later serve as an initial guess to the model predictive control problem using fully nonlinear vehicle model.

3-1 Formulation of Hybrid MPC Problem

Hybrid MPC is a special category of MPC that predicts the object value with a hybrid model and optimizes control input accordingly. In this study, the objective function is given by (3-1). The output of the system y is the side-slip angle of the rear tires α_R (3-2). The choice of using α_R instead of state variables is based on the following considerations. First of all, when the rear tires are saturated in the lateral direction, i.e. α_R is large, the vehicle is inherently unstable, as examined in the previous chapter. Additionally, the stabilization of β requires an increase of r in the opposite direction. This is suggested by the observation that a $-r$ term exists in the expression of $\dot{\beta}$. The determination of prediction and control horizon and weights will be explained in the upcoming section.

$$J = \sum_{i=1}^{N_p} \left(y_{k+i}^T Q y_{k+i} + \mathbf{u_{k+i-1}^T R u_{k+i-1}} \right) \tag{3-1}$$

$$y_k = \alpha_{R,k} = [l_R/v_x, -1]\mathbf{x_k} \tag{3-2}$$

3-2 Hybrid MPC Controller

The hybrid MPC controller determines the optimal control input at each time step by optimizing the objective function. The switching of dynamics in hybrid systems is described by including additional binary variables and the corresponding constraints. The consequent optimization problem is classified as a mixed-integer quadratic programming problem. The solution to this class of problem can be efficiently determined with guaranteed global optimality, provided that the complexity of the system is not excessive.
The objective function is automatically derived into MIQP form with the help of MPT Toolbox 3.0 and the MIQP is solved with Gurobi Solver.

3-3 Tuning of Controller's Parameters

The controller is tuned with closed-loop simulations using the hybrid vehicle model. The weights, sampling time, control horizon and prediction horizon will be determined.

3-3-1 Weights in objective function

Since the controller is developed for safety critical situations, the weights of system output should be set significantly higher than those of control inputs. Steering can generate more yaw dynamics especially when the rear tires are saturated. Thus the weights of steering and braking action are designed in such a way that applying maximum steering has a smaller contribution to the object value when compared to applying maximum braking. The weight of braking is first fixed to 1 and the other weights are adjusted accordingly. The consequent choice of weights can be found in Table 3-1.

3-3-2 Sampling Time and Horizons

For discretizing the continuous-time system, it is commonly recommended that $0.1T_{cl} \leq T_s \leq 0.25T_{cl}$, i.e. the sampling time should lie between 10 % to 25 % of the desired closed-loop response time of the system. The typical response time of vehicle's yaw and side-slip motion is around 0.5 second. Thus, $T_s = 0.05s$ and $T_s = 0.1s$ lie within the recommended range. Using smaller T_s is helpful for quicker control response, while larger T_s reduces numerical complexity when predicting for the same period of time. Hence, the decision will be made after comparing the performance of both options.
Typically, N_p should be chosen such that $T_s N_p \approx T_{cl}$ i.e. the prediction horizon should cover the system's response time. The candidate combinations of T_s and N_p that meet the requirement can be found in Table 3-2. Due to the existence of delays in the actuators as well as the demand for reducing computational complexity, the control horizon N_c is chosen as 1.

Variable	Maximum Value	Weight	Cost at Maximum Value
y	1.2	100	144
F_{bF*}	2.5	1	6.25
F_{bR*}	1.5	1	2.25
δ	0.48	4	0.922

Table 3-1: Weights of control inputs and system output

Controller #	Sampling Time T_s [s]	Prediction Horizon N_p	Accumulated Cost
1	0.05	6	20.0
2	0.05	8	19.2
3	0.05	10	20.5
4	0.05	12	21.8
5	0.1	3	62.7
6	0.1	4	30.6
7	0.1	5	28.5
8	0.1	6	26.8

Table 3-2: Combinations of sampling time and prediction horizon to be examined

3-3-3 Closed-loop Performance of Hybrid MPC

The controllers are examined through closed-loop simulation. The initial state of the simulation is $x_{h0} = [1.0 - 0.4]^T$. The closed-loop performance of controller #1-#8 is compared in Figure 3-1. As can be observed, all but #5 have foreseen the benefit of increasing r negatively on accelerating the stabilization process. The stabilization time of controllers with $T_s = 0.05s$ is around 0.4 second shorter than those using $T_s = 0.1s$. # 1 and #2 stabilized the vehicle the quickest but #2 has a significant advantage regarding the reduction of β between 0.4 and 0.8 s. The accumulated cost was calculated using. The results are given in Table 3-2.

$$J = \sum_{i=1}^{N_{sim}} \left(y_{k+i}^T Q y_{k+i} + \mathbf{u_{k+i-1}^T R u_{k+i-1}} \right) T_s \tag{3-3}$$

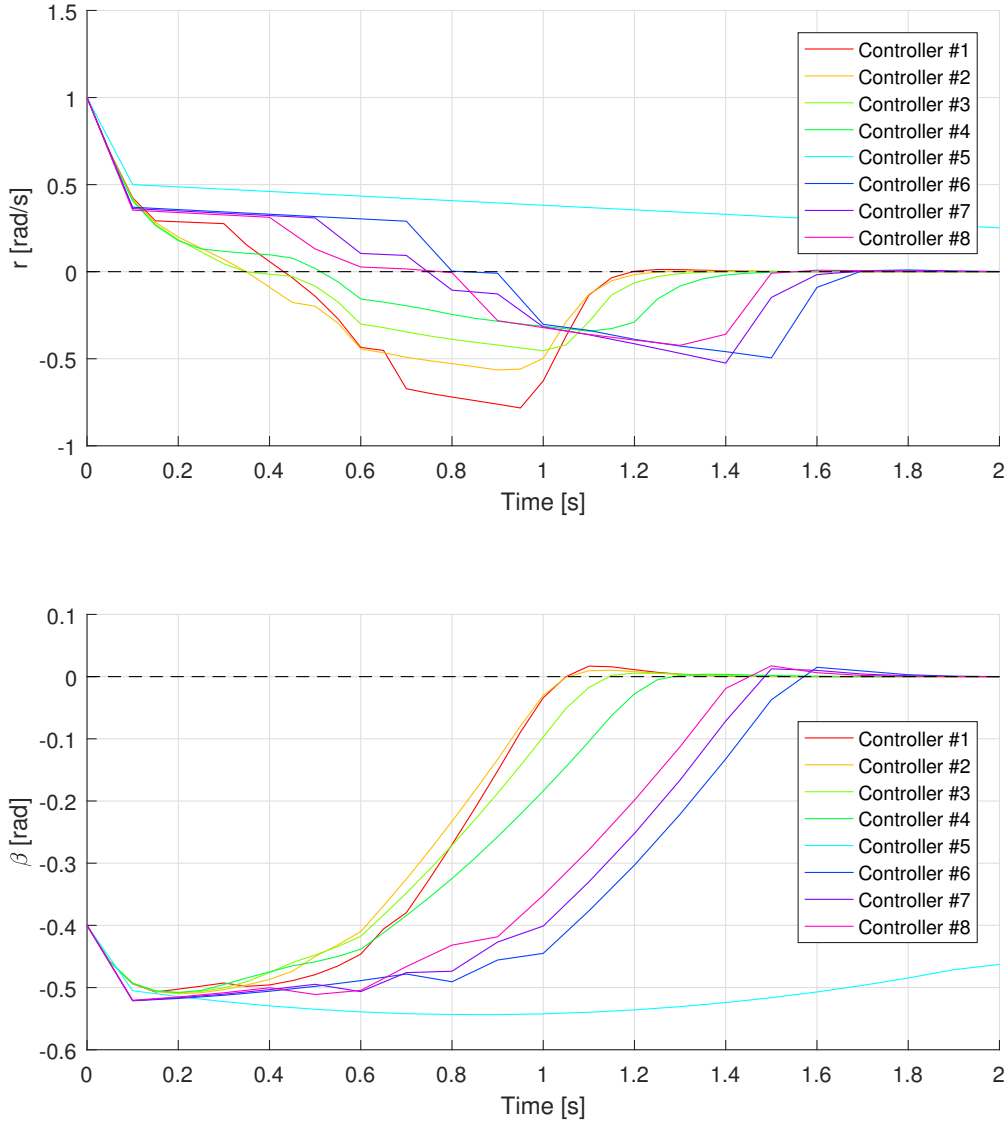

Figure 3-1: Comparison of hybrid MPC controllers using different combinations of T_s and N_p.

3-4 Analysis of Stabilization Process

In this section, the process how controller # 2 stabilizes the vehicle will be explained in detail. The simulation result from the aforementioned initial state is shown in Figure 3-2. As indicated by Figure 3-3, the initial state put the rear tires in mode 4, close to the border to mode 6. The controller decided to apply full counter-steering, putting the front tires in mode 5. It may seem wiser to use mode 3 here, which allows more lateral force with smaller control effort. But the control input is fixed for the upcoming 8 steps. According to the prediction

(the red line), the front tire will enter mode 3 and 1 and α_R will start to drop after the growth that lasts 7 steps. Full counter-steering (combined with braking) remains the optimal control decision until the 9th step of the simulation (the purple line), after which the steering angle start to decrease in order to spare control effort. Until step number 12, it is still predicted that counter-steering achieves the quickest reduction of α_R. However, α_R will already drop below zero and start to grow negatively. Hence, at step number 13, δ jumped to a small positive value, so that the vehicle enters mode $\{4, 4\}$ where α_R reduces less aggressively (the green line). After staying in this mode for 4 time-steps, the vehicle is very close to entering the inherently stable mode (the yellow line).

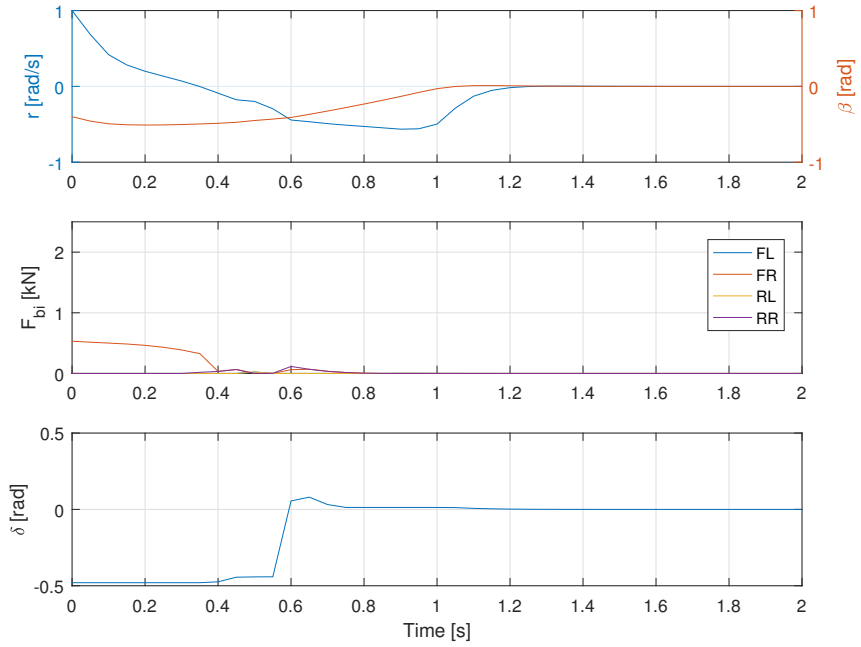

Figure 3-2: Closed-loop simulation of hybrid MPC controller using $T_s = 0.05s$ and $N_p = 8$. The initial state is $x_{h0} = [1.0, -0.4]^T$.

The controller is also examined with a different initial state $x_{h0} = [1.5, -0.2]^T$. As can be observed from Figure 3-4. Figure 3-5 demonstrates the prediction made by the controller through the stabilization process. At the initial state, the vehicle is in a similar mode as in the previous case, where the rear tires are saturated and hence the front tire must be steered to the negative direction to limit the growth of α_R (the yellow line). At step number 6, δ should start to reduce so that α_R will not grow quickly to the other side (the blue line). The jump in δ at step number 9, as has appeared in the previous case, then brings the vehicle on the track to self-stabilization (the purple line). From step number 13, the vehicle acquired autonomous stability (red line), thus the steering and braking actions became negligible. These two cases have clearly shown how the optimization in hybrid MPC is related to the stabilization of the hybrid vehicle model. The stabilization process also demonstrated the validity of using α_R in the objective function, due to the strong connection between α_R and stability of certain vehicle modes. By quickly reducing α_R, the vehicle enters the self-stabilizing path through

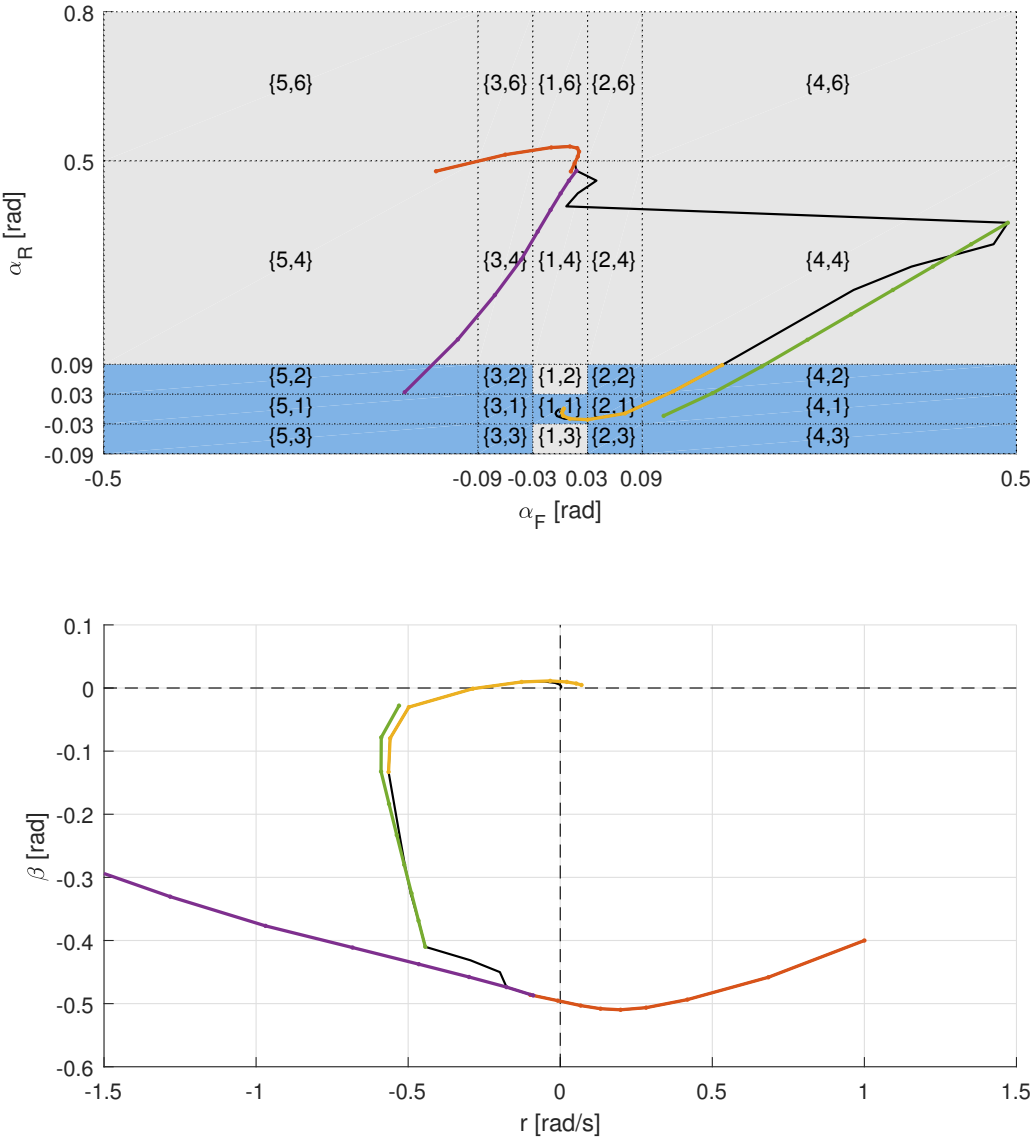

Figure 3-3: Stabilization process demonstrated on $\alpha_F - \alpha_R$ plane. The labels of each partition indicates the mode of behavior that the front and rear tires are put in.

mode $\{4,4\}$, $\{4,2\}$, $\{4,1\}$, $\{2,1\}$ and $\{1,1\}$ (or from the other side through mode $\{5,5\}$, $\{5,3\}$, $\{5,1\}$, $\{3,1\}$ and $\{1,1\}$). In the inherently stable modes, very small control effort is needed for suppressing oscillation and overshoot.

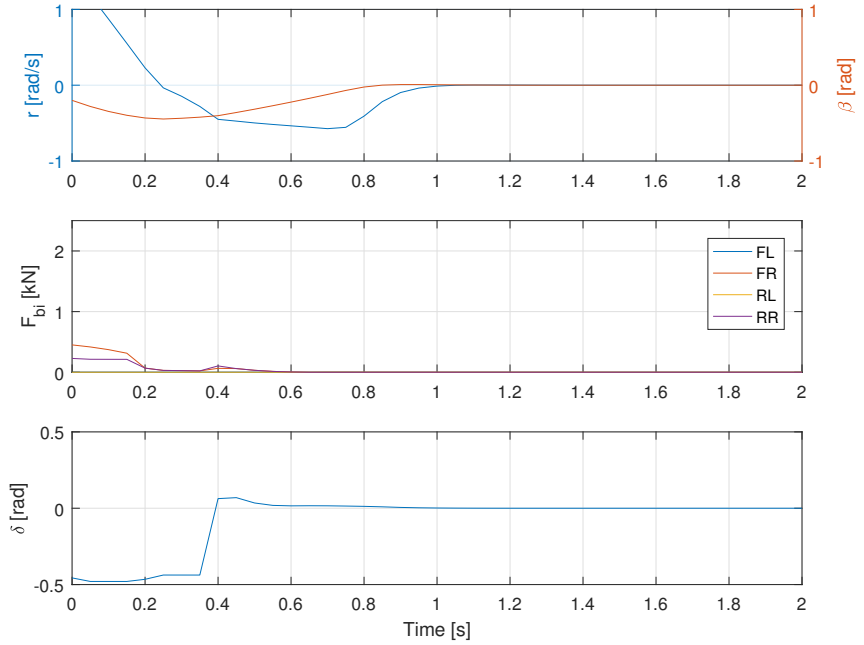

Figure 3-4: Closed-loop simulation of hybrid MPC controller using $T_s = 0.05s$ and $N_p = 8$. The initial state is $x_{h0} = [1.5, -0.2]^T$.

3-5 Explicit Control Map

The control input given by hybrid MPC can be used as the initial guess for solving the nonlinear MPC problem. Meanwhile, it also serves as a back-up controller when the nonlinear MPC fails. By solving the hybrid MPC problem offline at different points in the state space, good initial guesses to the NMPC problem can be generated efficiently, which further reduces online computational effort. The typical approach for this task is using explicit MPC. However, a large number of local dynamics in the hybrid system drastically increases the complexity of calculating the hybrid explicit MPC. The number of partitions in the state space, when using explicit MPC, has a similar scale as $N_{dyn}^{N_c} = 49^8 \approx 3.3 \times 10^{13}$. It means the hybrid MPC problem has to be solved at 3.3×10^{13} points. Moreover, each partition consists of at least 3 linear constraints (for fully defining a partition in 2-dimensional space) and 5 linear functions for evaluating the 5 control variables. Hence, the controller has to store around 2.64×10^{14} floating-point numbers. A double-precision floating-point number takes up 8 bytes, thus the memory of the controller should be around 2.1×10^15 bytes, i.e. 2,100 TB. A hard-drive disk with 12 TB capacity weights 700 grams and takes 0.4 Liter of space. Thus the total mass and volume for storing this explicit information will be at least 122.5 kg and 70 L, which is absolutely impossible for on-board implementation. Therefore, an alternative solution should be found.

The explicit MPC approach preserves the accuracy of the explicit solution as much as possible, which is not necessary since the hybrid model itself is an approximation of the actual dynamics of the vehicle. Moreover, the initial guess only needs to point in the correct direction as it is not directly taken by the vehicle as the control input. The loss of detail in the explicit

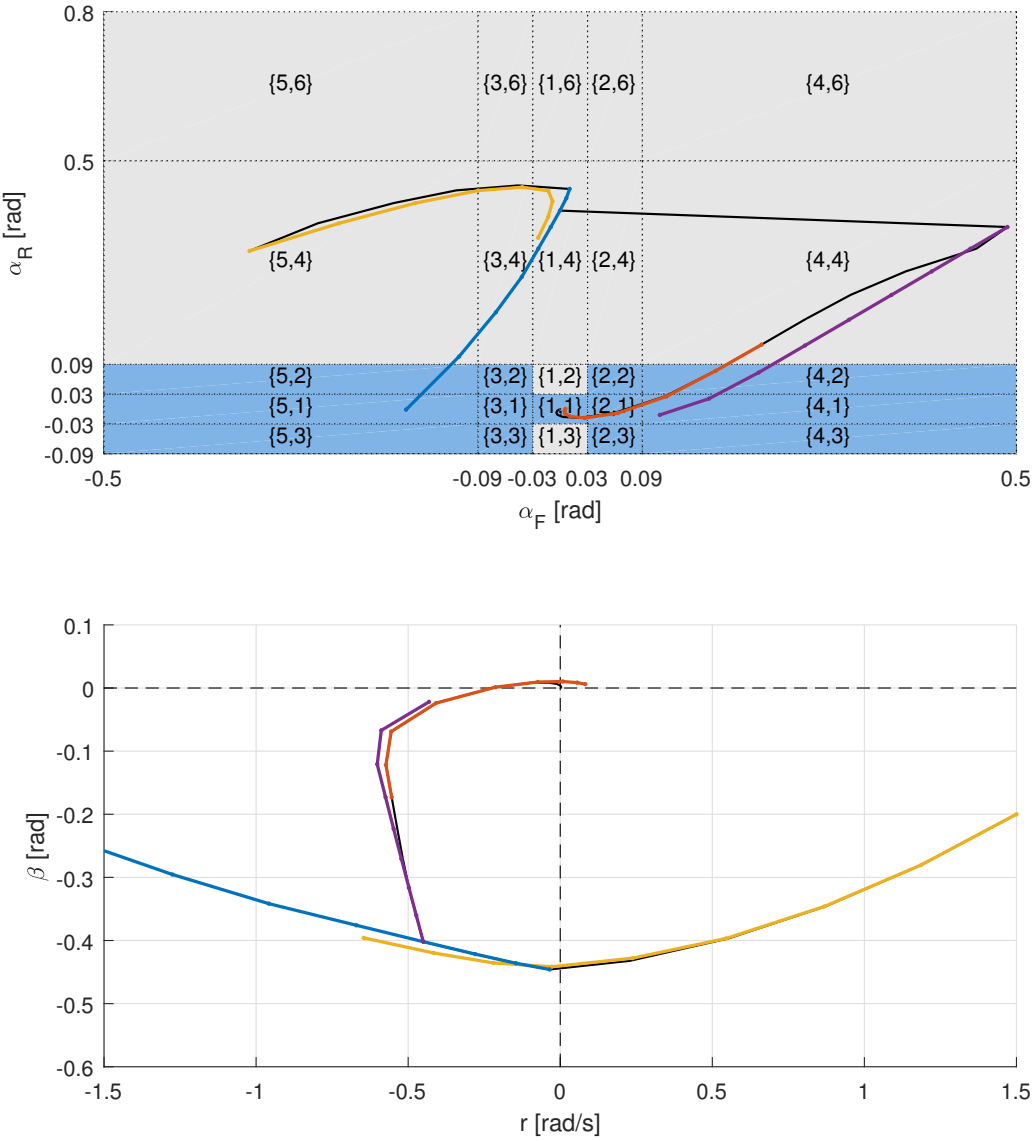

Figure 3-5: Stabilization process demonstrated on $\alpha_F - \alpha_R$ plane. The dashed lines shows the predicted evolution using the optimal input given by the hybrid MPC.

solution can be covered by the online optimization process. Hence, it is a reasonable choice to barely use a rough uniform grid for generating the explicit look-up table. The uniform grid defined by (3-4) is generated first, which contains 133 nodes, dividing the state space into 108 partitions. The hybrid MPC problem is then solved at these nodes to obtain the look-up table, which is visualized in Figure 3-6. Each node contains 7 floating numbers, among which 5 are for control inputs and 2 are the coordinate of the node. Consequently, the controller stores in total 931 floating-point numbers for determining the initial guesses. The storage of

the information takes less than 8 kB of memory.

$$\begin{aligned} r_h &= -1.8 : 0.2 : 1.8 \\ \beta_h &= -0.6 : 0.2 : 0.6 \end{aligned} \tag{3-4}$$

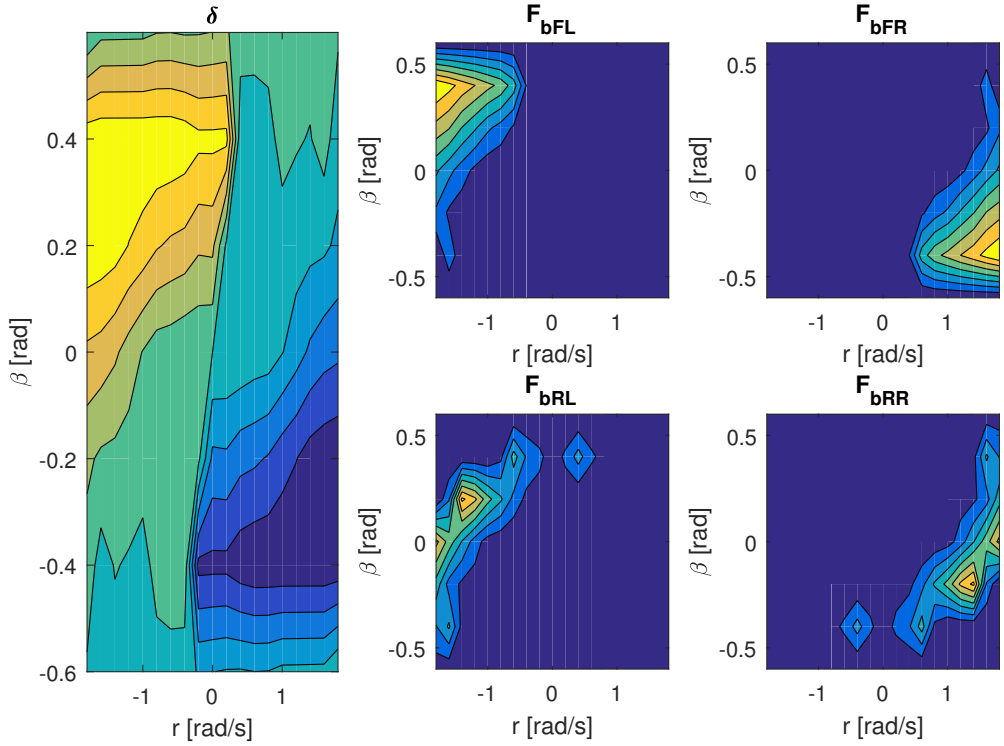

Figure 3-6: Explicit map of initial guess for nonlinear optimization, visualized as 2-D contour.

Chapter 4

Nonlinear Model Predictive Control

4-1 Problem Formulation

The nonlinear MPC solves for the control input by optimizing an objective function as given by (4-1). The objective function has an identical form as the one used in the hybrid MPC developed in Chapter 3, so that the explicit control table can indeed serve as the initial guess to the nonlinear MPC problem. The difference is that the prediction is based on a nonlinear model at this stage, which will be explained in detail in the next section.

$$J = \sum_{i=1}^{N_p} \left(y_{k+i}^T Q y_{k+i} + \mathbf{u_{k+i-1}^T R u_{k+i-1}} \right) \tag{4-1}$$

4-2 Nonlinear Vehicle Model

The original form of the 2-track vehicle model, as already given in the chapter in (2-2), is employed for predicting the vehicle's behavior within the prediction horizon. The tire forces are determined by the modified combined-slip Pacejka's model (4-2). The parameters A_*, B_*, C_*, D_* are functions of side-slip angle α, longitudinal slip ratio κ, vertical load F_z (see (4-3)). The modified model has assumed symmetric behavior and irrelevance to inflation pressure and camber angle.

$$
\begin{aligned}
F_x &= G_x F_{x0} \\
F_y &= G_y F_{y0} \\
F_{x0} &= D_x \sin\left(C_x \tan^{-1}\left(B_x\kappa - E_x\left(B_x\kappa - \tan^{-1}\left(B_x\kappa\right)\right)\right)\right) \\
F_{y0} &= D_y \sin\left(C_y \tan^{-1}\left(B_y\alpha - E_y\left(B_y\alpha - \tan^{-1}\left(B_y\alpha\right)\right)\right)\right) \\
G_x &= \cos\left(C_\alpha \tan^{-1}\left(B_\alpha\alpha - E_x\left(B_\alpha\alpha - \tan^{-1}\left(B_\alpha\alpha\right)\right)\right)\right) \\
G_y &= \cos\left(C_\kappa \tan^{-1}\left(B_\kappa\kappa - E_\kappa\left(B_\kappa\kappa - \tan^{-1}\left(B_\kappa\kappa\right)\right)\right)\right)
\end{aligned}
\tag{4-2}
$$

$$
\begin{aligned}
D_x &= \left(p_{Dx1} + p_{Dx2} df_z\right) F_z \\
C_x &= p_{Cx1} \\
B_x &= \frac{K_{x\kappa}}{C_x D_x} \\
K_{x\kappa} &= F_z \left(p_{Kx1} + p_{Kx2} df_z\right) \exp\left(p_{Kx3} df_z\right) \\
E_x &= p_{Ex1} + p_{Ex2} df_z + p_{Ex3} df_z^2 \\
D_y &= \left(p_{Dy1} + p_{Dy2} df_z\right) F_z \\
C_y &= p_{Cy1} \\
B_y &= \frac{K_{y\alpha}}{C_y D_y} \\
K_{y\alpha} &= p_{Ky1} F_{z0} \sin\left(p_{Ky4} \arctan\left(\frac{F_z}{F_{z0} p_{Ky2}}\right)\right) \\
E_y &= p_{Ey1} + p_{Ey2} df_z \\
df_z &= \frac{F_z - F_{z0}}{F_{z0}} \\
dp_i &= \frac{p_i - p_{i0}}{p_{i0}}
\end{aligned}
\tag{4-3}
$$

To accurately seize the key nonlinear characteristics in large acceleration motion, steady-state load transfer in both longitudinal and lateral direction is incorporated (4-4). It has been taken into consideration in (4-4) how the heights of the front and rear roll centers influence load transfer. The effect of Ackermann steering geometry on the actual steering angle of front tires has also been modelled (4-5). The Ackermann geometry has a strong influence on vehicle's lateral dynamics especially when the lateral acceleration is large and hence the difference between the vertical loads on the inner and outer tires becomes immense.

$$
\begin{aligned}
M_{roll} &= a_y \left(m_{uF} \left(h_{cg} - h_{rF} \right) + m_{uR} \left(h_{cg} - h_{rR} \right) \right) \\
M_{pitch} &= a_x \left(m_{uF} + m_{uR} \right) h_{cg} \\
\Delta F_{zFL} &= \frac{l_R}{2L} mg - \frac{K_{rF}}{K_{rF} + K_{rR}} \frac{M_{roll}}{B} - \frac{M_{pitch}}{2L} \\
\Delta F_{zFR} &= \frac{l_R}{2L} mg + \frac{K_{rF}}{K_{rF} + K_{rR}} \frac{M_{roll}}{B} - \frac{M_{pitch}}{2L} \\
\Delta F_{zRL} &= \frac{l_F}{2L} mg - \frac{K_{rR}}{K_{rF} + K_{rR}} \frac{M_{roll}}{B} + \frac{M_{pitch}}{2L} \\
\Delta F_{zRR} &= \frac{l_F}{2L} mg + \frac{K_{rR}}{K_{rF} + K_{rR}} \frac{M_{roll}}{B} + \frac{M_{pitch}}{2L}
\end{aligned}
\tag{4-4}
$$

$$
\begin{aligned}
\delta_{Ack} &= \tan^{-1} \left(L / \left(\frac{L}{\tan\left(\delta\right)} - B \right) \right) \\
\delta_{in} &= \delta + P_{Ack} \delta_{Ack} / 2 \\
\delta_{out} &= \delta - P_{Ack} \delta_{Ack} / 2
\end{aligned}
\tag{4-5}
$$

After fully defining the nonlinear vehicle model, the prediction can be achieved by means of numerical integration of the equations of motion. Multiple methods are available for numerical integration, including Euler step (4-6), Heun step (4-7), Runge-Kutta 4th order method (4-8), etc.. The accuracy and complexity depend on the order of the method, i.e. the count of evaluation of equations of motion per integration step. In the implementation of nonlinear MPC, the integration interval is relatively short (0.4 s) thus using low order integration method can only introduce limited error. Moreover, there is a strong demand for reducing computational complexity when implementing real-time model predictive control. Thus Euler step is chosen. The integration step length is 0.05 second, identical to the sampling time for discretizing the hybrid vehicle model. The main advantage of this method is the minimized count of evaluation of the cost function. The repeated reduction of step length in line search allows a quicker approach to the optimum at the beginning of each iteration, and a good accuracy when finely searching in the neighborhood of the optimum or approaching the upper or lower bound of control inputs. Parameters in the local optimization algorithm can be found in Table 4-1.

$$
x_{n+1} = x_n + T_i \dot{x} \left|_{x=x_n} \right.
\tag{4-6}
$$

$$
\begin{aligned}
x^*_{n+1} &= x_n + T_i \dot{x} \left|_{x=x_n} \right. \\
x_{n+1} &= x_n + \frac{T_i}{2} \left(\dot{x} \left|_{x=x_n} \right. + \dot{x} \left|_{x=x^*_{n+1}} \right. \right)
\end{aligned}
\tag{4-7}
$$

$$
\begin{aligned}
k_1 &= \dot{x}\,|_{x=x_n} \\
k_2 &= \dot{x}\,\Big|_{x=x_n+\frac{T_i}{2}k_1} \\
k_3 &= \dot{x}\,\Big|_{x=x_n+\frac{T_i}{2}k_2} \\
k_4 &= \dot{x}\,|_{x=x_n+T_i k_3} \\
x_{n+1} &= x_n + \frac{T_i}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right)
\end{aligned}
\tag{4-8}
$$

4-3 Local Optimization With Initial Guess

The NMPC controller finds the optimal control input by optimizing the objective function. The complexity of the nonlinear vehicle model introduces great difficulty into the optimization of the objective function. On one hand, the convexity of the objective function is not guaranteed at all. This means the function can have multiple local minimums. There is no verification whether a local minimum found by the optimization process is the actual global optimum. The position and object value at a local minimum can be far from the global one, leading to invalid control input. One may repeat the optimization process at different starting points and use the local minimum with lowest object value as the so-believed global optimum. Such approach raises the chance of finding the global optimum with brutal force, resulting in a higher computational burden on the controller. This problem is already handled by generating the look-up table for the initial guess. The solution given by the hybrid MPC is expected to be close to the global optimum of the nonlinear objective function since the hybrid model approximates the nonlinear model with good fidelity and the parameters in the hybrid objective function is identical to those in the nonlinear one.
On the other hand, the derivation of the analytic form of the objective function is hardly possible. Most state-of-the-art numerical optimization algorithms exploit the gradient and even Hessian information of the objective function. Such information can only be efficiently determined if the analytic expression of the objective function is available. Although the gradient can be calculated numerically and hence the Hessian can be approximated using the numerical gradient of the objective function, such calculation requires lots of computational effort, not to mention that the evaluation of the objective function is already time-consuming. Therefore, a gradient-free method should be implemented. The perpendicular search method combined with varied-step line search has been developed, as explained in Algorithm 1.

The method aims to minimize the total count of evaluation of the objective function while preserving acceptable accuracy. In the stage of direction determination, the algorithm already takes a valid step in the best direction. Hence, the starting point for line search is closer to the optimum-on-the-line. In addition, the line search automatically adjusts step length. In the beginning, the step length is large, allowing quicker approaching to the optimum-on-the-line, or the constraint that the line may violate. When the object value increases or a constraint is violated, the algorithm moves back to the previous point and retry with a reduced step length. The reduction of step length can be repeated several times until the step length is sufficiently small. At this moment the precision of line search has been maximized, thus the current

Algorithm 1 A gradient-free optimization algorithm

function LocalOptimization($\mathbf{u_0}$)
  if $J(\mathbf{u_0}) \leq J_{act}$ **then**
    return $\mathbf{u_0}$
  else
    for all $n_i = 1$ to N_i **do**
      Find k s.t. $J(\mathbf{u_0} + \mathbf{D_k}) =$
      $\min([J(\mathbf{u_0} + \mathbf{D_1}), ..., J(\mathbf{u_0} + \mathbf{D_N})]$
      if $J(\mathbf{u_0} + \mathbf{D_k}) > J(\mathbf{u_0})$ **then**
        return $\mathbf{u_0}$
      else
        $\mathbf{u_0} = \mathbf{u_0} + \mathbf{D_k}, J_{min} = J(\mathbf{u_0})$
        Reset L_l, n_r
        while $I_r \leq N_r$ **do**
          if $\mathbf{lb} \leq \mathbf{u_0} + L_l\mathbf{D_k} \leq \mathbf{ub}$ **then**
            Evaluate $J(\mathbf{u_0} + L_l\mathbf{D_k})$
            if $J(\mathbf{u_0} + L_l D_k) < J_{min}$ **then**
              $\mathbf{u_0} = \mathbf{u_0} + \mathbf{D_k}$
              $J_{min} = J(\mathbf{u_0})$
            else
              $L_l = L_l/3$
              $I_r = I_r = +1$
            end if
          else
            $L_l = L_l/3$
            $I_r = I_r = +1$
          end if
        end while
        return $\mathbf{u_0}$
      end if
    end for
  end if
end function

Parameter	J_{act}	N_i	N_r	L_l	$\lvert\mathbf{D_k}\rvert$
Value	10	10	5	5	0.01

Table 4-1: Parameters of local optimization.

point is indeed the optimum-on-the-line or the farthest possible point within the available range. Parameters in the local optimization algorithms are listed in Table 4-1. The online optimization is only activated when necessary, i.e. the object value is above 10. No more than 10 iterations is allowed, in order to prevent timeout error in real-time implementation. Of course, the algorithm can terminate when a local minimum is already found, i.e. the object value increases in all available directions. The step length for line search is 0.05 after reset and can reach 2.06×10^{-4} at the end of the line search.

Chapter 5

Simulation with Multibody Model

The controller developed in this will be tested under real-world situations. As stated in Chapter 3, the initial guess to the NMPC problem should be a sub-optimal control input itself. This statement will also be examined by directly feeding the output of the explicit lookup table to the vehicles as the control input.

5-1 Multibody Model of Vehicle Dynamics

The simulation will be performed using a multibody model of the vehicle. The model has in total 9 degrees of freedoms and was built with SimMechanics. The roll-axis model of vehicle dynamics is implemented, which incorporates the pitch and roll dynamics in addition to the planar motions, i.e. longitudinal, lateral and yaw motion. In the roll-axis model, the vehicle is divided into several rigid bodies connected to each other with joints. The main vehicle body is connected to the front and rear rigid axle with a rotational joint on each side. The position of the joint is defined by the roll-center in suspension design. These two joints constructed the axis about which the vehicle body's roll motion is allowed. Rotational springs and dampers are attached to these joints, representing the roll stiffness and roll damping by the suspension. The front and rear axles are each attached with two wheels. The tire dynamics is modelled using Delft-Tire 6.2 with a Magic Formula steady-state slip model describing nonlinear slip forces and moments. The relaxation behavior is linear using empirical relations for the relaxation lengths. Actuator dynamics are also taken into consideration in the multibody model. The steering dynamics is simplified and represented by a second-order transfer function with time delay. The Ackerman steering geometry is added to describe the difference between the inner and outer wheels. An experimentally calibrated model of the electric-hydraulic braking system is included, which uses a pressure-relevant delay plus a second-order transfer function. The initial parameters of the model are obtained from the manufacturer's brochures, vehicle design manuals as well as documented measurements. With the help of some experimental data obtained from previous studies, the parameters were finely tuned to guarantee that the

model accurately resembles the actual vehicle. Detailed model validation procedures are included in Appendix C. The validation was accomplished in cooperation with another M.Sc student L. Zhao.

5-2 Simulated Scenario

The advantage of incorporating the strong nonlinear behavior in vehicle dynamics in the controller can be clearly demonstrated when the rear tires are saturated and the vehicle suffers side-slip motion. The vehicle may enter side-slip motion due to multiple causes:

- **Aggressive maneuver**. The driver may have to perform emergency evasive maneuver under certain situations such as a sudden appearance of wild animals or debris on the road. The delay in lateral load transfer, due to the rather slower response in the roll dynamics, may cause the rear tires to slide. Several standard vehicle tests including double-lane-change, moose test and sine-with-dwell are inspired by this cause.

- **Low-friction road surface**. Hydroplaning (or aquaplaning) is extremely dangerous for any ground vehicles. When driving on the wet surface with high speed, the tire may not be able to stay in contact with the road surface but lifted by the water in between. The recovery of such motion is far beyond the capability of ordinary drivers, who usually become panicked and instinctively apply a full braking, which is a very wrong choice.

- **Collision with other vehicles**. No driver can guarantee that he or she will not be crashed by another one. The vehicle is easily destabilized when hit laterally on the rear end. The impact directly causes rear tires to slide and also introduces massive yaw moment, which makes the recovery even more difficult. This situation can happen when passing crossroads and changing lanes on the highway.

The first two causes of instability can be avoided by state-of-the-art active safety systems. A quick response can prevent the vehicle from entering the unstable motion at all. However, a collision with another vehicle may not be handled by conventional systems. The excessive impact force can force the vehicle to spin within 0.1 s with a yaw moment far beyond the capability of yaw moment generation by applying braking forces. Hence this is the ideal scenario to examine the capability of the nonlinear MPC controller. As stated in Chapter 3, the explicit lookup table can serve as backup controller, the performance of such explicit control is also tested through simulation. By comparing the performance of NMPC and explicit control, the significance of applying the online optimization of the control input can be well demonstrated.
In the simulation, the friction resembles dry asphalt road surface ($\mu = 1$). This can be even more challenging for the controller as the friction forces are the strongest and the vehicle dynamics is the quickest. Thus the controller's action must be quick and accurate. The

lateral impact is exerted at 1.8 meters behind the center of gravity, between the rear axle and the rear end of the vehicle. As mentioned above, the lateral collision can happen due to multiple causes. In this study, the collisions at a crossing and on the highway are investigated (Figure 5-1). Crossings without the regulation of traffic lights can be dangerous. In practice, such crossings are regulated by the priority rules informed with marks on the road. The speed limit at this kind of crossing is typically 50 km/h (13.9 m/s) for the directions with priority. Vehicles approaching from other directions are expected to a with lower speed to prepare for giving priority. If the driver on the latter vehicle sees the former one too late, he may not be able to fully stop the vehicle and hence a perpendicular collision will happen. The speed of impact is assumed 14 km/h (3.89 m/s) and the collision is assumed to be fully non-elastic, i.e. the momentum of the latter vehicle is evenly shared by both of them. If the exchange of momentum finishes in 0.1 s, the consequent magnitude of the lateral impact is 30,917 N. Another case of interest is changing lanes on the highway. If a driver changes to the left lane without checking the blind point, he may hit the vehicle in the adjacent lane that is overtaking him. The maximum speed limit is 130 km/h in many countries. The vehicle causing the collision may be driving at 90 km/h if blocked by a truck (for which the speed limit is 90 km/h). Ordinary drivers prefer driving less aggressively on the highway. When performing a lane change, the lateral acceleration hardly exceeds 0.3 g. Given the speed and lateral acceleration, the average curvature radius of the trajectory can be estimated as 212.4 meters. The lane width of highways is 3.5 meters according to relevant standards and the impact is expected to happen when the vehicle moves 2 meters in the lateral direction. Thus the angle of impact can be determined as 0.137 rad or 7.87 degrees. Again, the equal sharing of the lateral momentum is assumed as well as the duration of 0.1 s, resulting in a magnitude of 27,208 N.

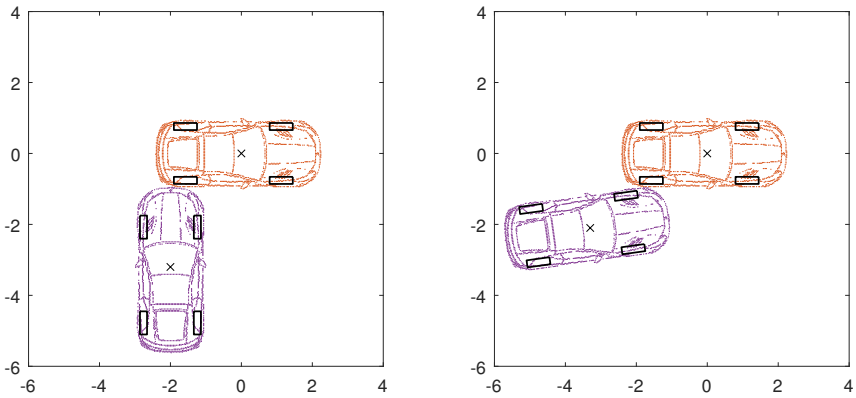

Figure 5-1: Postures of the lateral collisions that will be investigated. The posture on the left corresponds to the crossroad case and the posture on the right corresponds to the highway case.

5-3 Simulation Results

5-3-1 Perpendicular Crossing

The trajectories and attitudes of the vehicle when without control, explicitly controlled and controlled by NMPC during the post-collision recovery is shown in Figure 5-2. The yellow vehicle represents the uncontrolled case, where the loss of stability is clearly demonstrated by how the vehicle spun and swept to the right side of the original path. The blue vehicle was controlled by directly using the initial guess to the NMPC problem as the final control input. The terminal attitude suggests that the vehicle was stabilized properly. The red vehicle controlled by NMPC shows even better performance, as the deviation in heading angle is smaller. Figure 5-3 provides information about the vehicle states during the stabilization process and Figure 5-4 shows the reaction of the controller. The lateral impact introduced huge yaw motion in 0.1 s and hence the vehicle started to spin and slide. The yaw rate remained high in the uncontrolled case until $T = 5s$ where the energy has been dissipated through friction. The jump in β at $T = 2.2s$ indicates that the vehicle was already reversing. The explicit control and NMPC both brought down the yaw rate to 0 in 0.3 s by the comprehensive utilization of braking and steering actions, while α_R reached a maximum of around 0.5 rad. After that, the yaw rate is further increased in the negative direction, which allowed quicker descent of α_R and β. The exclusion of influence of v_x causes mismatches between the actual values of α_F and α_R and the approximated one the hybrid MPC sees. Hence, the explicit control input at $T = 1.4s$ caused discontinued decrease of yaw rate. And since the rotational transformation of forces of front tires has been omitted in the hybrid vehicle model, the explicit control tends to apply more braking force, which is not as effective in generating yaw moment when the front tires are counter-steered. The incorporation of the rotation matrix in the nonlinear vehicle model, on the other hand, enabled the controller to correctly coordinate the steering and braking. The advantage of accurate modelling can also be shown in the period from $T = 2.0s$ to $T = 3.0s$. The side-slip motion has already been suppressed at $T = 2.0s$ but the residual yaw motion still existed. The NMPC controller eliminated the residual oscillation effectively while the explicit control caused the yaw rate to increase again to a peak value of 0.6 rad/s. The accumulated cost through the entire recovery maneuver is calculated. NMPC led to 11.9 % less cost than explicit control, which further proved how NMPC enhances the control performance.

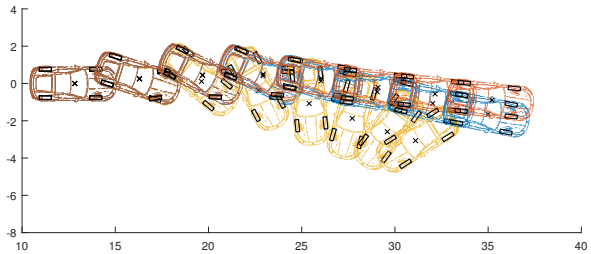

Figure 5-2: Vehicle attitude and trajectory during the recovery phase of simulation, with an initial speed of 14 m/s or 50 km/h. Blue for uncontrolled, orange for explicit control and yellow for NMPC.

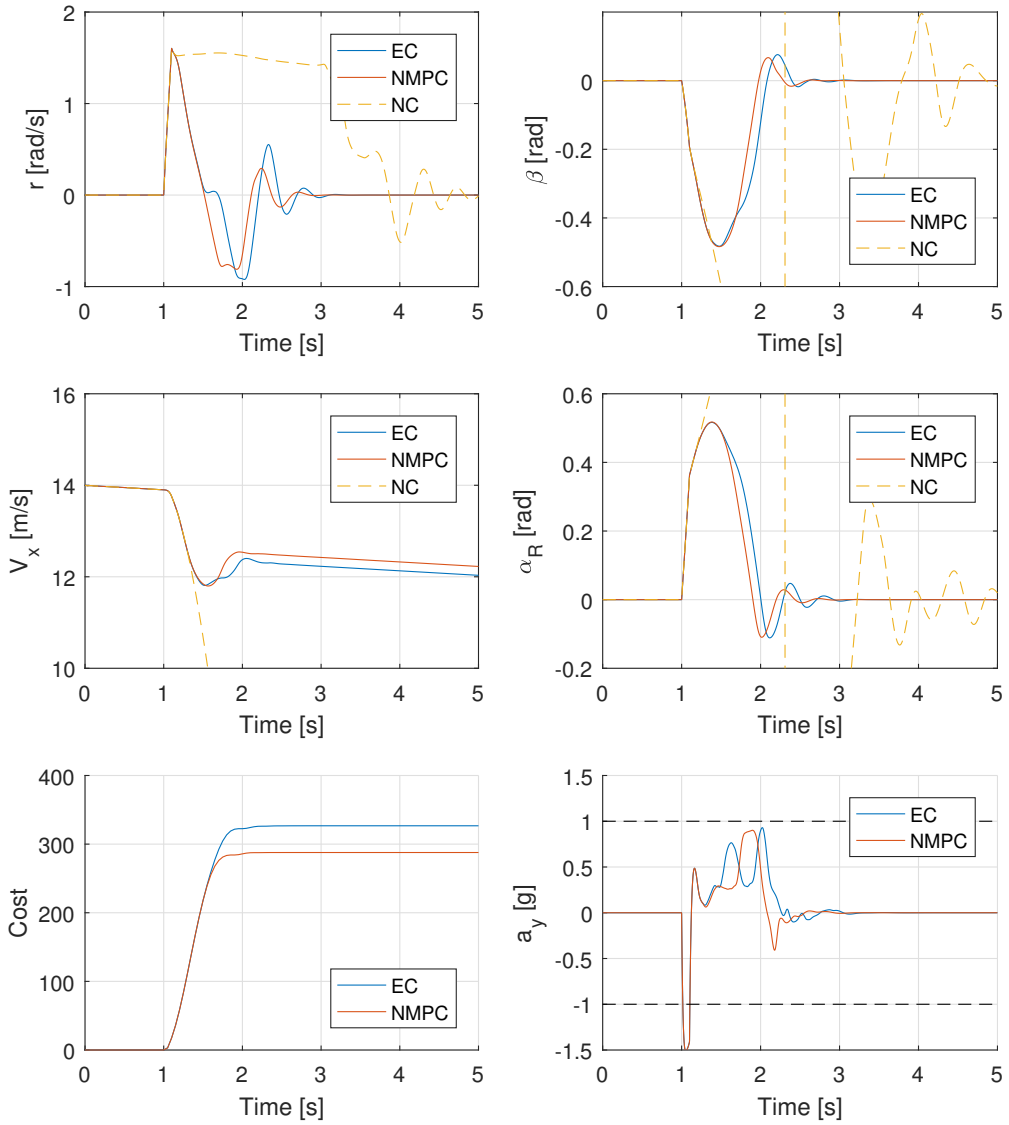

Figure 5-3: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 14 m/s or 50 km/h. EC for explicitly controlled, NMPC for nonlinear model predictive control and NC for no control.

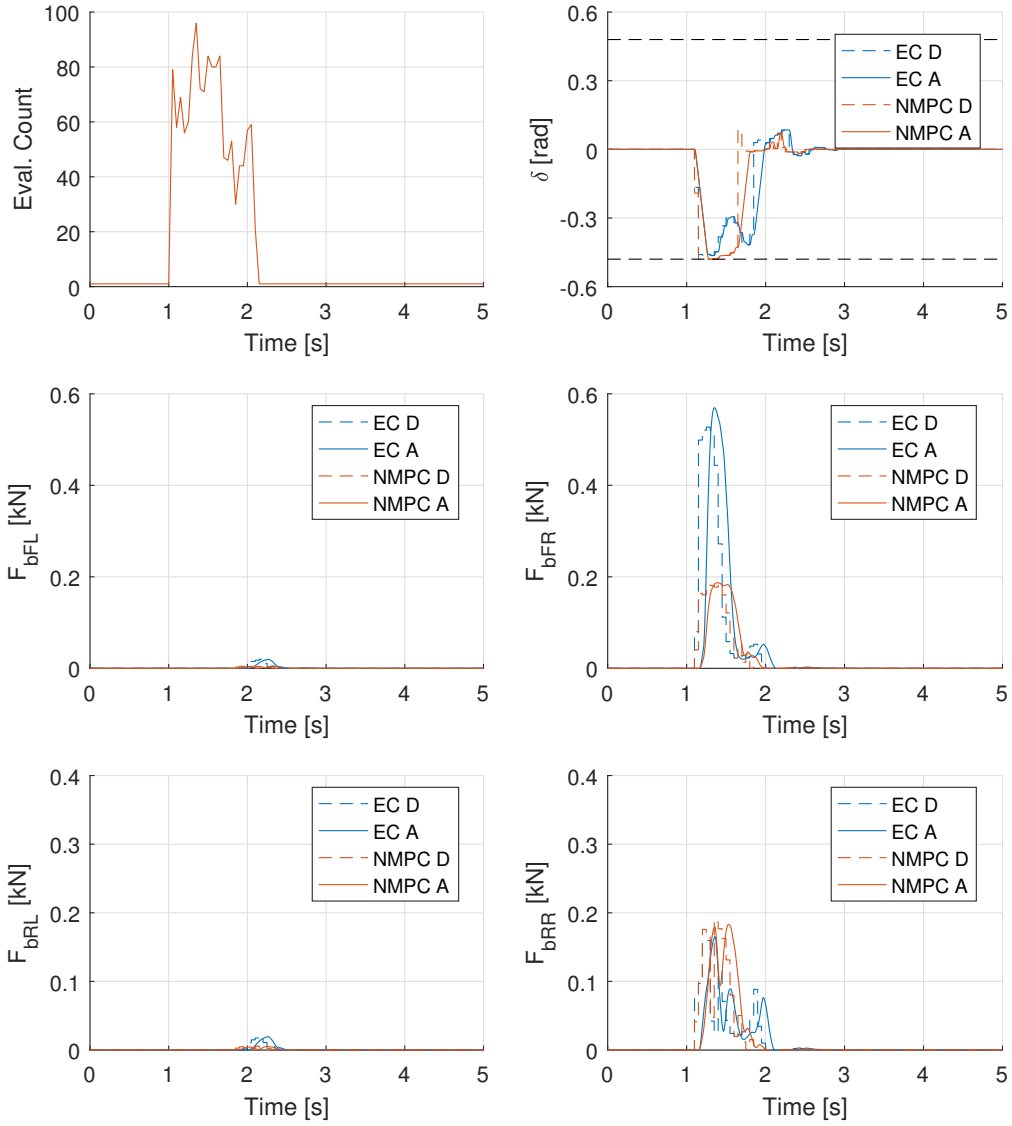

Figure 5-4: Control inputs during lateral collision recovery, with an initial speed of 14 m/s or 50 km/h. Dashed lines for control demand (D) and solid lines for actual response of actuators (A). EC for explicitly controlled, NMPC for nonlinear model predictive control and NC for no control. The total count of evaluation of the object function in the NMPC per time step is also presented.

5-3-2 Changing Lanes On Highway

The vehicle's attitude and trajectory in the recovery phase of the simulation are shown in Figure 5-5. Again, the uncontrolled vehicle lost stability and slide to the right side. The difference between explicit control and NMPC is less significant in this case because the magnitude of the impact is not as large. Both were able to quickly bring the vehicle back to stable straight-line driving. Vehicle states and controller actions are given in Figure 5-6 and Figure 5-7. NMPC outperformed the explicit control by achieving quicker reduction of α_R and suppressing the overshoot in yaw motion. The NMPC also resulted in 15.3 % less accumulated cost as compared to explicit control.

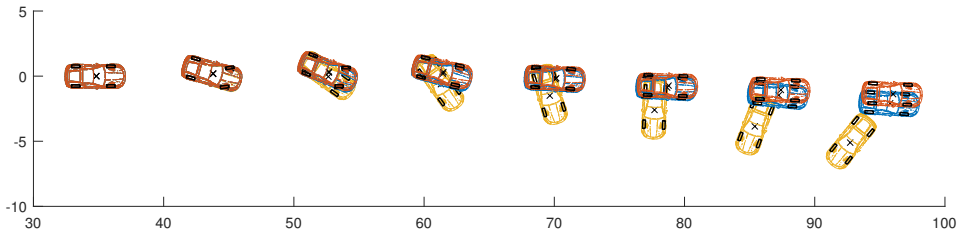

Figure 5-5: Vehicle attitude and trajectory during the recovery phase of simulation, with an initial speed of 36 m/s or 130 km/h. Blue for uncontrolled, orange for explicit control and yellow for NMPC.

5-4 Analysis on Friction Utilization

The NMPC accurately incorporates the nonlinear behavior of the tires, thus is expected to achieve highly efficient utilization of the friction. The friction limit of the tire used in this study is demonstrated using the friction ellipse in Figure 5-8. The longitudinal and lateral forces are normalized by dividing the vertical load. The friction ellipse can be well approximated by the ellipse expressed in Figure 5-8. In the simulation, the forces are measured with the reaction forces exerted by the wheel hub. It is noteworthy that the gravity force of tire and wheel is not exerted on the hub thus they should be added to the measured vertical load. Finally, the longitudinal and lateral forces are divided by this vertical load. Figure 5-9 shows how each tire uses the potential of friction during the recovery. The availability of regulating both α and F_b of the front tires enabled full utilization of friction in order to generate yaw moment. The lateral force of rear tires are not freely controlled but the utilization of friction was still improved by applying proper braking force. The acceleration of vehicle body also shows how the friction is efficiently used. As presented by Figure 5-10, the peak of vehicle's planar acceleration is very close to 1 g.

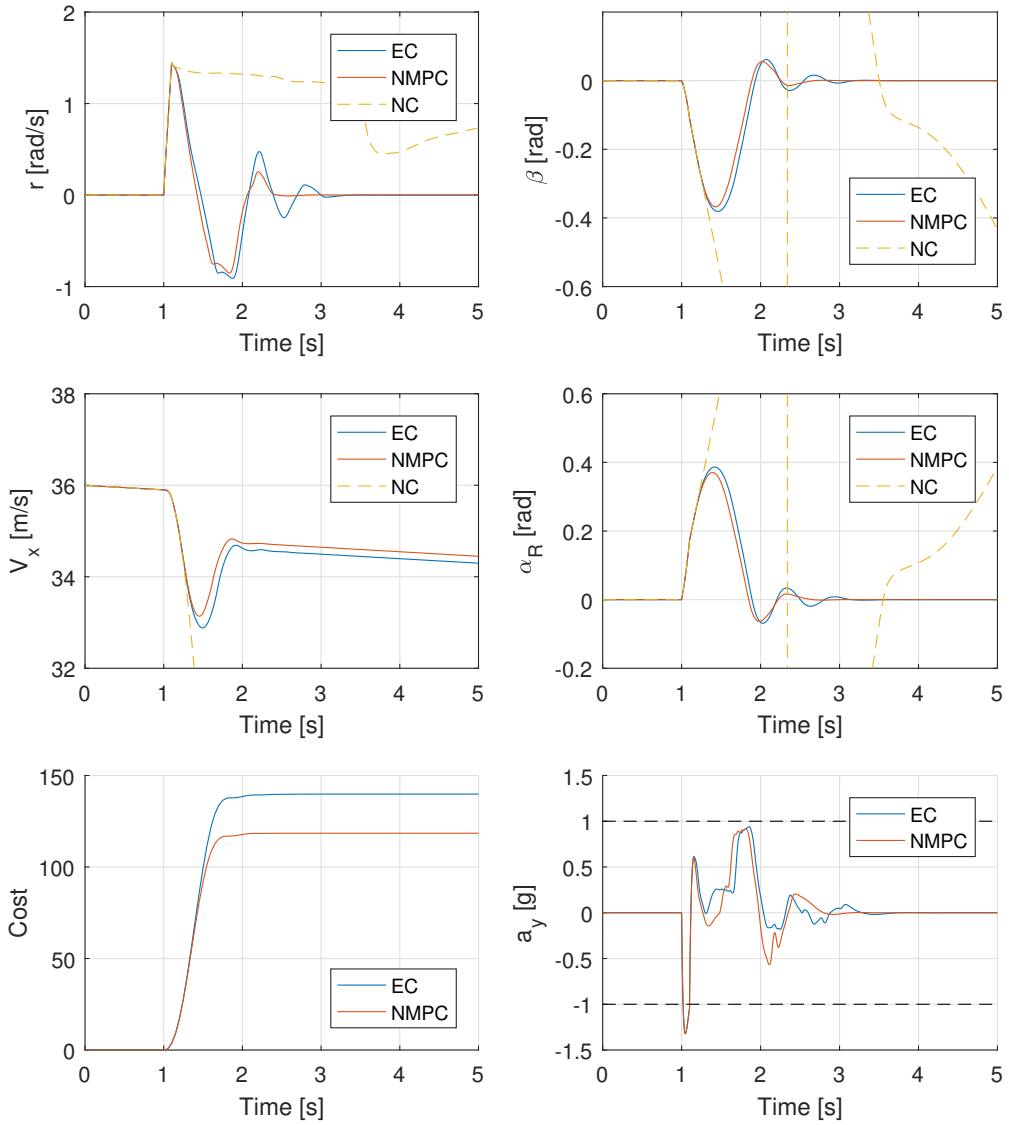

Figure 5-6: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 36 m/s or 130 km/h. EC for explicitly controlled, NMPC for nonlinear model predictive control and NC for no control.

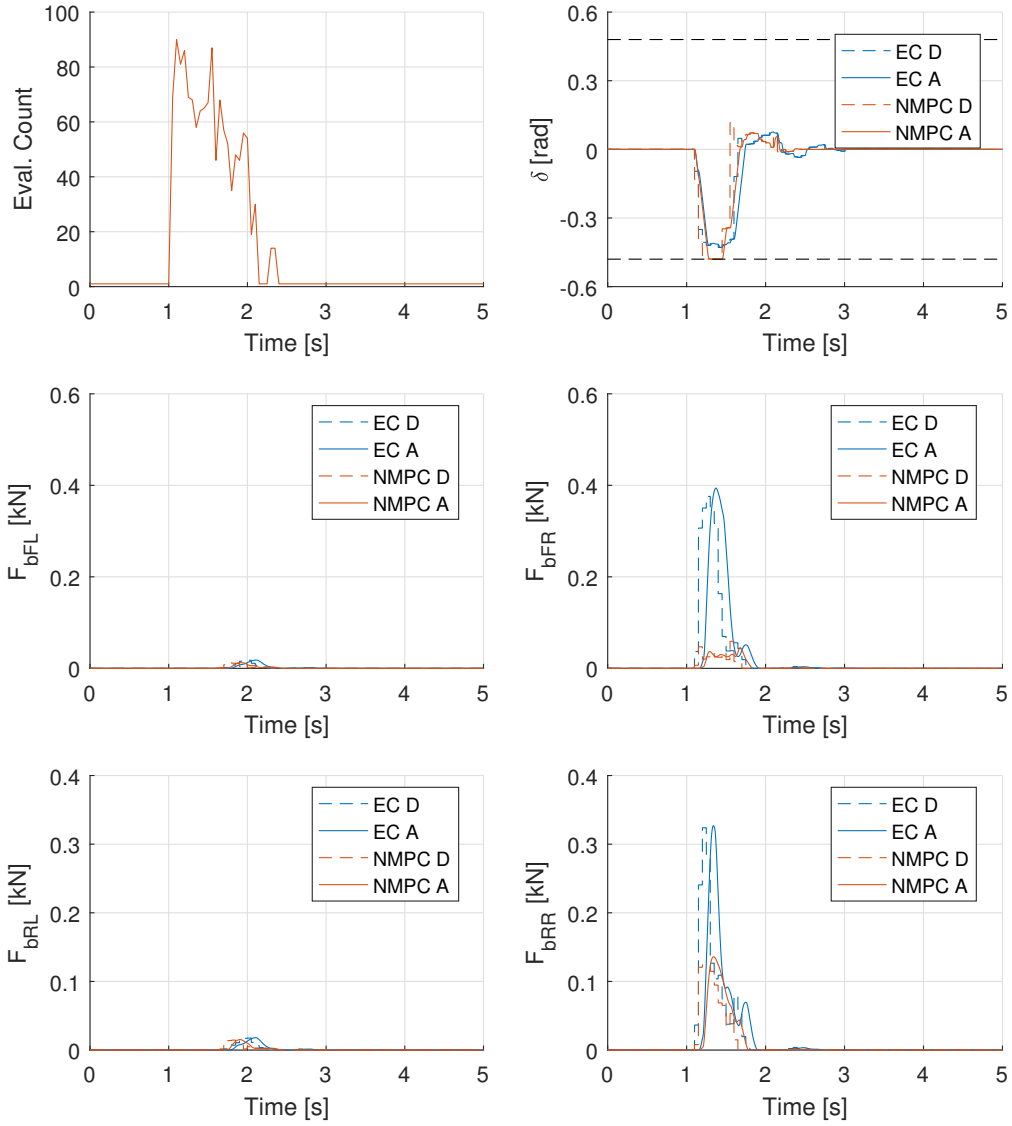

Figure 5-7: Control inputs during lateral collision recovery, with an initial speed of 36 m/s or 130 km/h. Dashed lines for control demand (D) and solid lines for actual response of actuators (A). EC for explicitly controlled, NMPC for nonlinear model predictive control and NC for no control. The total count of evaluation of the object function in the NMPC per time step is also presented.

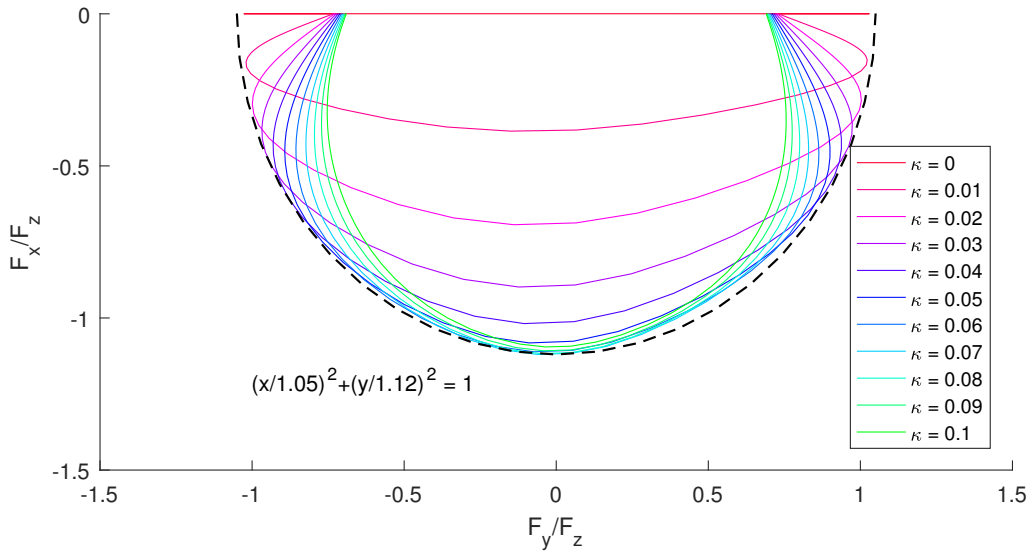

Figure 5-8: Friction ellipse of the tire model used in this study.

5-5 Maximum Performance of NMPC

The NMPC has proved its capability in handling the designated situations based on real-world cases. This section will further explore the limit of the NMPC's performance by varying the longitudinal speed where the collision happens as well as the magnitude of the lateral impact. First of all, the speeds are chosen according to the typical speed limits given by traffic rules. The speed limit in built-up areas are typically 50 km/h. Suburban open roads usually have a speed limit of 80 km/h. The speed limit for highways can vary per area and 100 km/h and 130 km/h are most commonly seen. Since there are highway sections in Germany where no speed limit is applied, the controller will also be tested at the top speed that an economy-class passenger car can reach, which is chosen as 180 km/h. By iteratively increasing the magnitude of the impact at these speeds until the stabilization cannot be properly achieved, the controller's performance limit can be determined. Figure 5-11 shows the maximum performance of NMPC compared to explicit control. The detailed results of simulation at the limit of controller's performance can be found in Appendix D.

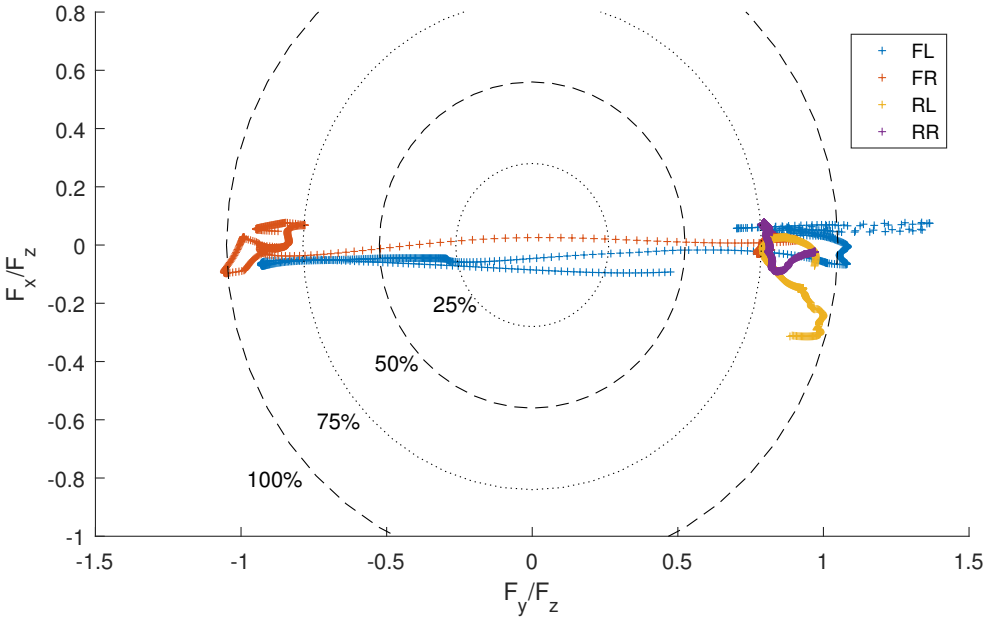

Figure 5-9: Scatter plot of normalized friction forces on the tires.

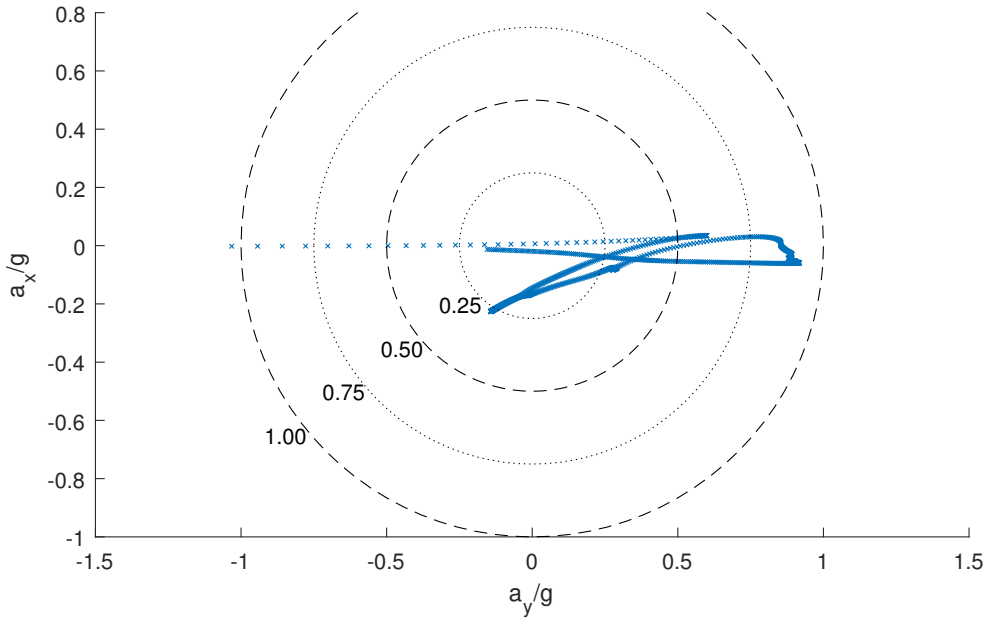

Figure 5-10: Scatter plot of normalized acceleration of the vehicle body.

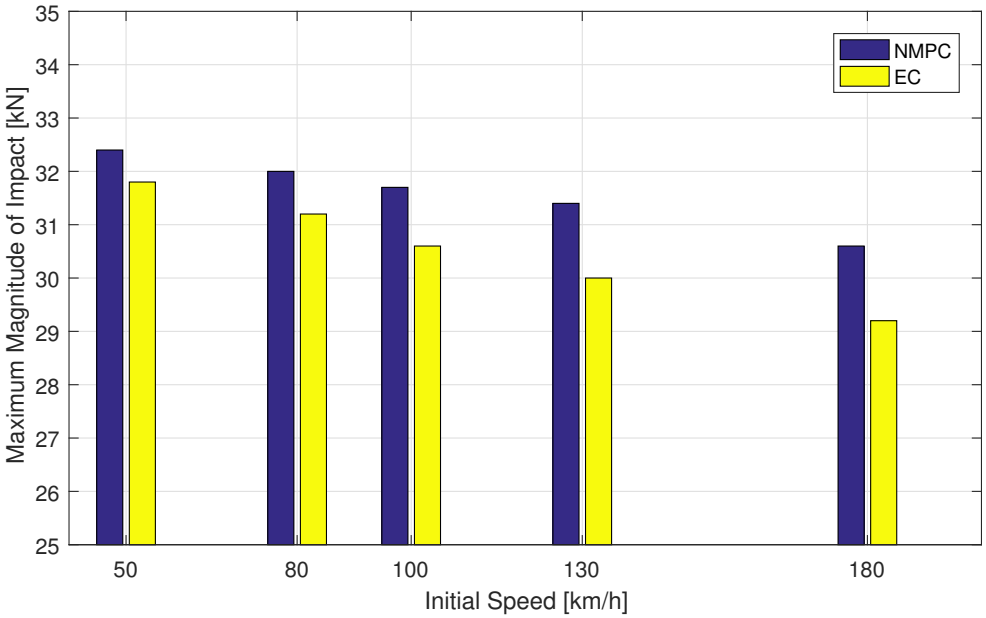

Figure 5-11: Maximum magnitude of lateral impact that the controller can sustain.

Chapter 6

Verification of Numerical Performance

As presented in the previous chapter, the NMPC controller has demonstrated its capability of stabilizing the vehicle in extreme cases. This chapter will look into the numerical performance of the proposed method, and verify if the controller can be implemented in real-time.

6-1 Benchmarking With MATLAB Global Optimization Toolbox

To prove the validity of the initial guess and efficiency of the proposed method, the functions *Global Search* and *Pattern Search* from Global Optimization Toolbox of MATLAB without the initial guess map was adopted as a benchmark. The algorithms are tested at randomly generated points in the critical range of motion, where loss of stability is happening for sure without proper control. The range is defined based on the observation of the simulation results. The motion is considered critical if the accumulated cost grows quickly. Hence, the range described with Table 6-1 is chosen.

With the explicit map, an initial guess for minimizing the objective function is determined using the first two variables (β_0 and v_{x0}). Local optimization starting from the initial guess is then performed, which should find the local optimum closest to the initial guess. The functions *Global Search* and *Pattern Search* from Global Optimization Toolbox of MATLAB were utilized afterwards to set up a benchmark. The object value obtained with each method

State	Lower Bound	Upper Bound
Yaw Rate (rad/s)	0.3	0.8
Body Slip Angle (rad)	-0.4	-0.15
Longitudinal Speed (m/s)	15	40

Table 6-1: Range of motion where the random state is generated for the benchmark test on the numerical performance.

(noted as C_m) are calculated and recorded. The efficiency is evaluated with two indicators: the solution time per case, and the relative difference D_m from the real optimal. The minimum cost among all algorithms is believed to be the actual global minimum and the optimal cost given by other algorithms are compared to this global minimum (6-1).

$$\begin{aligned} C_{opt} &= \min([C_{IG}, C_{IG+LO}, C_{GS}, C_{PS}]) \\ D_m &= C_m/C_m - 1 \\ m &\in \{IG, IG+LO, GS, PS\} \end{aligned} \tag{6-1}$$

In total 500 tests are performed on a laptop with Intel i7-7700HQ (quad-core 2.8GHz) and 32 GB DDR4 RAM. The histogram of the two indicators is shown in Figure 6-1 and Fig. 6-2 and the statistics are summarized in Table 6-2. As an indication of the instability of the random initial state, the cost value with all control inputs equalling zero is also calculated and presented in the table as NC (for 'no control'). *Global Search* requires the longest computation time and finds the global optimum in most cases. Exceptions can be observed with a very low frequency (2 out of 500) but the local minimums that trapped the global search were far from the actual ones, with a cost value over twice as much. *Pattern Search* is around 8 times faster than global search. The performance is, however, less consistent, as it found the wrong local minimum in 26.4% of the tests. The chance is certainly too high for safety-critical applications. The initial guess has shown promising consistency and efficiency, as it gives control inputs that in the worst case lead to 13.5% of performance loss using on average 2 ms. The local optimization can significantly reduce performance loss, as is obviously indicated by the statistics. Thanks to the quality of the initial guess, local optimization has always converged to the actual optimum. The proposed method requires, in most cases, less than 150 times of evaluation of the cost function to find a minimum. Consequently, the mean computation time is only 1.53% of what is needed for *Global Search* to converge. The accuracy benefits from the repetitive reduction of step length in the line search as expected. The difference from actual optimal cost is negligible even in the worst case.

Method	Maximum D_m	Average D_m	Maximum Comp. Time	Average Comp. Time
IG	13.5 %	3.24 %	3.99 ms	1.77 ms
IG+LO	0.107 %	0.007%	0.271 s	0.149 s
GS	137 %	0.706 %	69.0 s	7.32 s
PS	81.3 %	17.2 %	1.64 s	1.10 s
NC	224 %	117 %	-	-

Table 6-2: Summarized result of numerical tests.

6-2 Insights to the Optimization Process

Some of the 500 tests is picked out as examples to show how the method proposed in this study improves computational efficiency. In the global optimization of a non-convex nonlinear

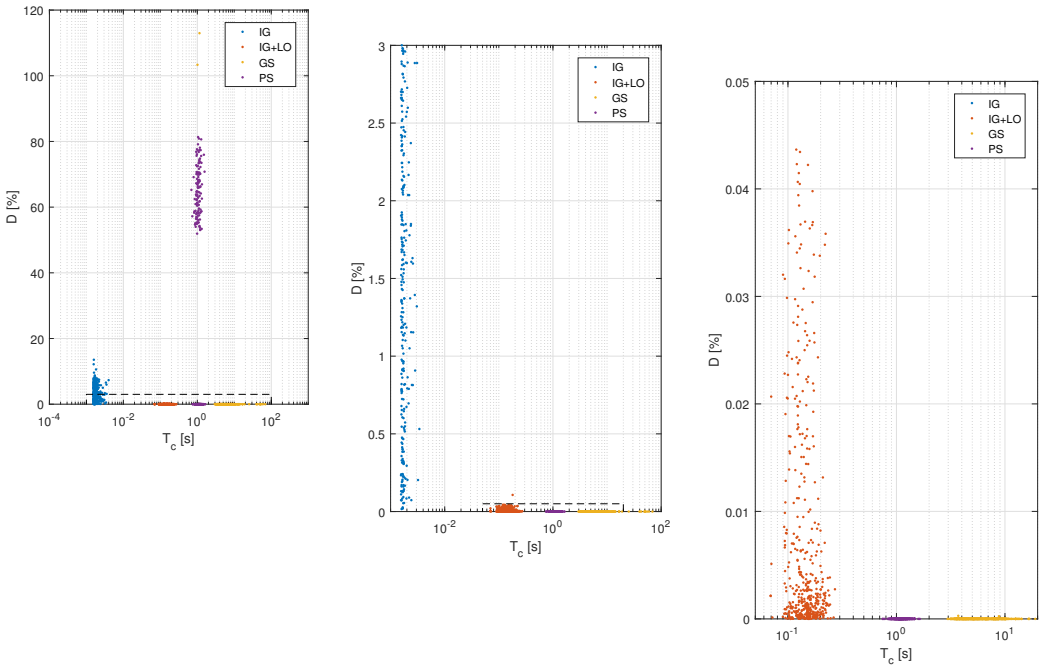

Figure 6-1: Direct demonstration of result of random test on proposed optimization algorithm, compared to MATLAB Global Optimization Toolbox. IG for initial guess, IG+LO for initial guess plus local optimization, GS for MATLAB *Global Search* and PS for MATLAB *Pattern Search*.

function, there are a great number of local optimums where the optimization may converge. To increase the chance of finding the actual global optimum, state-of-the-art methods usually perform optimization at multiple random starting points. Nevertheless, there is no guarantee whether all the local minimums can be found. Hence it is still impossible to verify whether a local minimum is the global minimum. This problem has already been exposed by the results presented in the previous section, where the MATLAB *Global Search* converged to a local minimum far from the actual global minimum. The explicit look-up table generated by solving the hybrid MPC problem offline, on the other hand, showed high consistency in providing a good initial guess that is close to the global optimum. In the first example, all three methods successfully found the global optimum. As can be seen from Figure 6-3, the initial guess help the local optimization to start from a point that gives an object value close to the optimal one. Without an initial guess, the *Global Search* (Figure 6-4) and *Pattern Search* (Figure 6-4) already need to evaluate the objective function a lot of times to get close to the global optimum. And even when a good object value has been found, *Global Search* still have to wait until all the remaining local solvers converge, in order to check whether the current optimum is global. *Pattern Search*, as another gradient-free optimization method, clearly shows how the numerical evaluation of a function's gradient increases computational burden.

An example of cases where *Pattern Search* fails is given below. The proposed method and *Global Search* method performed similarly (Figure 6-6 and Figure 6-7). *Pattern Search*, however, converged to a wrong local optimum (Figure 6-8). From iteration 14, the method found

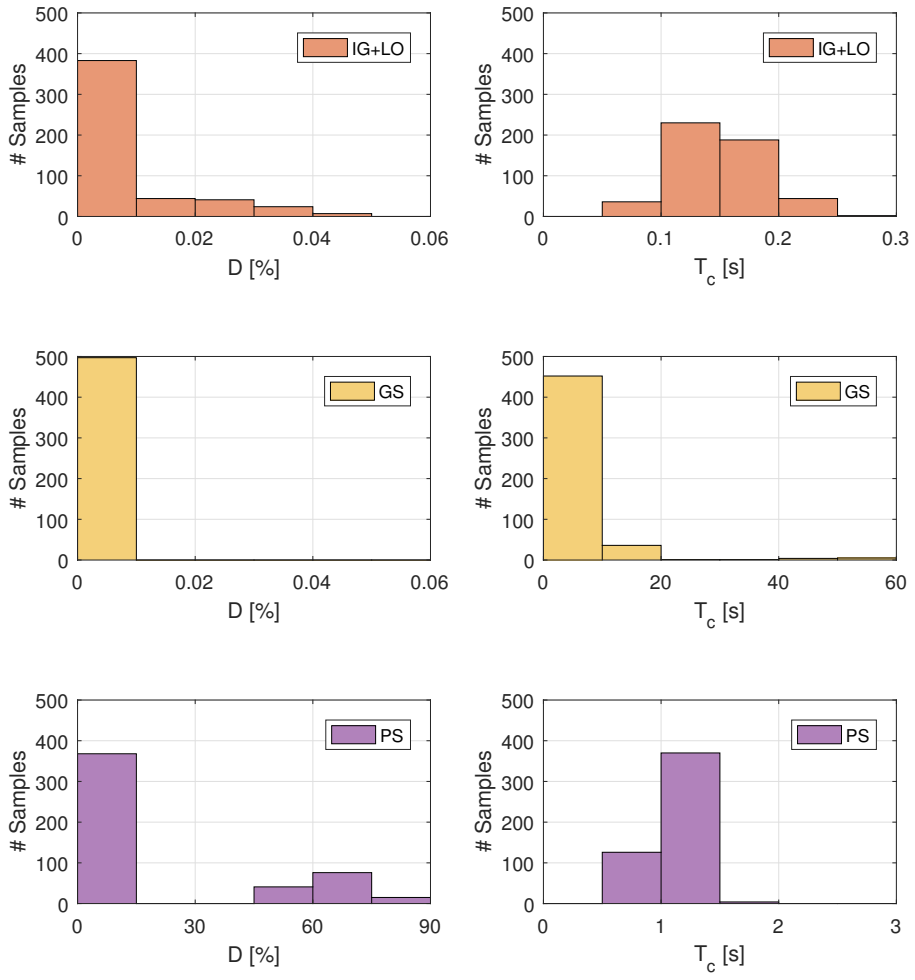

Figure 6-2: Histogram of result of random test on proposed optimization algorithm, compared to MATLAB Global Optimization Toolbox. IG+LO for initial guess plus local optimization, GS for MATLAB *Global Search* and PS for MATLAB *Pattern Search*.

no better points than the current one. *Global Search* has also encountered this problem through the 500 tests although the chance is very rare. In fact, *Global Search* includes an algorithm to find possible starting points. *Pattern Search* does not include such algorithm but always starts from the origin. Hence, this situation is more likely to happen when the optimum is close to the lower or upper bound of the variable and far from the origin. This difference, again, demonstrated how numerical optimization can benefit from good initial guesses.

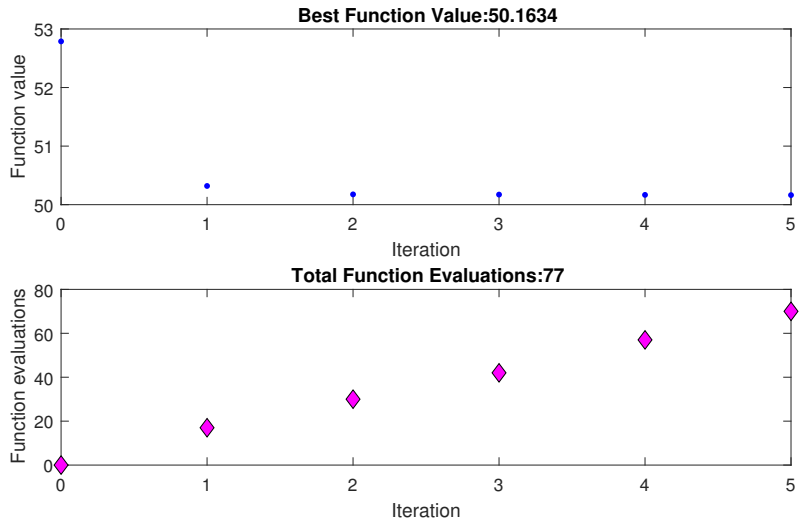

Figure 6-3: Demonstration of the optimization process given by the proposed initial guess plus local optimization method.

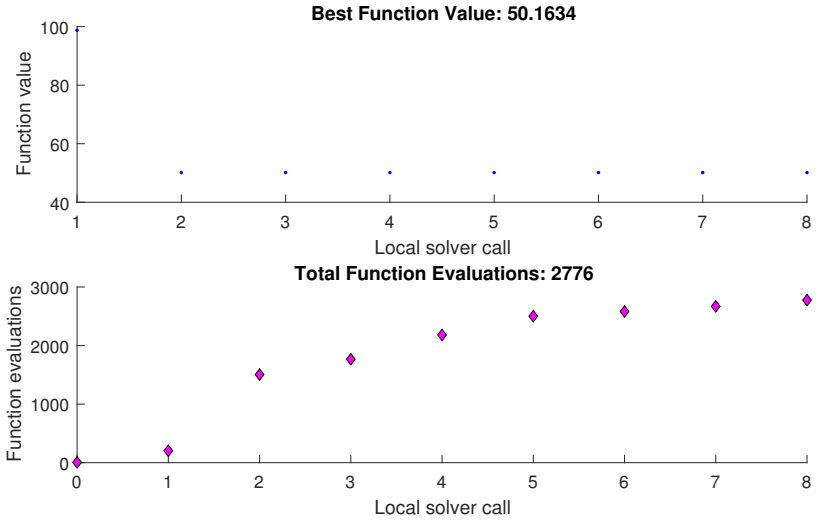

Figure 6-4: Demonstration of the optimization process given by the MATLAB Global Search Function.

6-3 Real-Time Verification on dSPACE

In order to obtain a direct proof that the method is capable of real-time implementation, experiments are performed on dSPACE. The processing board DS 1006 with AMD Opteron quad-core 2.8 GHz and 1 GB RAM was adopted. The layout of the dSPACE experiment can be found in Figure 6-9. In order to exclude the computation load contributed by the simulation itself, the multibody model and the controller are compiled on separate cores. The multibody model runs at 1,000 Hz and the controller runs at 20 Hz. It is admissible if the multibody model causes a time-out because there will be no computation for this part when

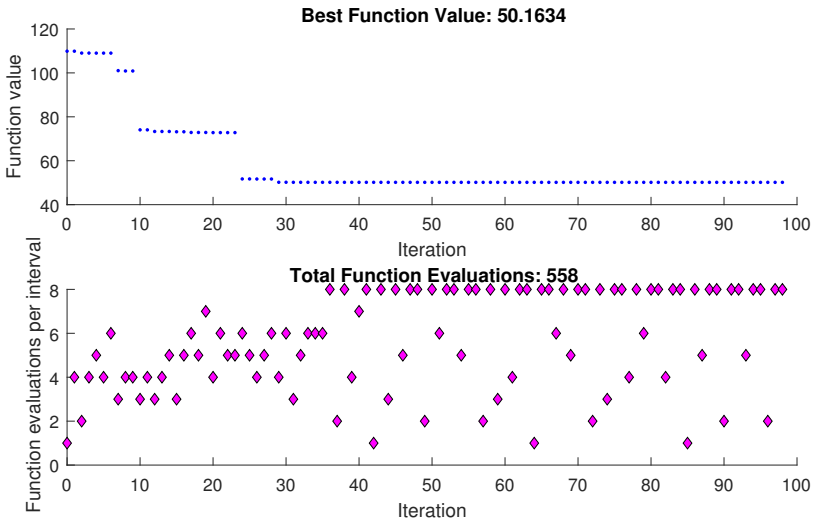

Figure 6-5: Demonstration of the optimization process given by the MATLAB Global Search Function.

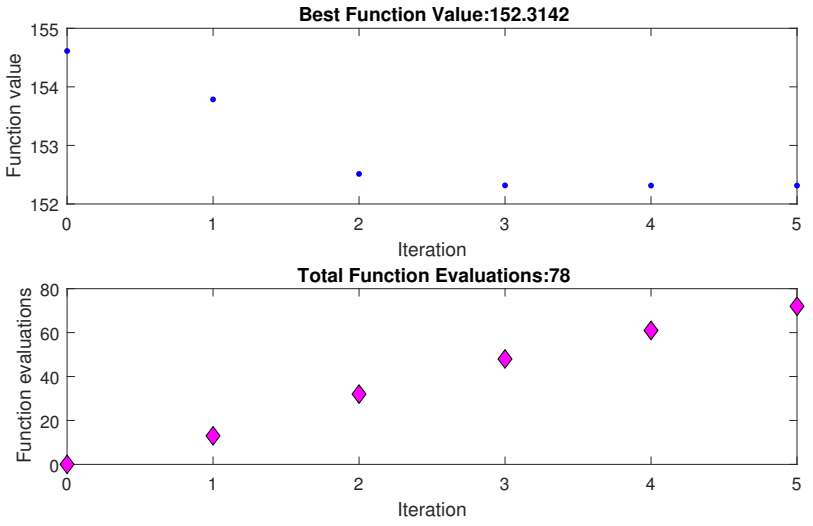

Figure 6-6: Demonstration of the optimization process given by the proposed initial guess plus local optimization method.

implemented in real world. In contrast, the simulation will be terminated if the controller fails to complete the computation within 50 ms. The failure may put the safety of passengers at risk if it happens in real world. Fortunately, the initial guess can be used as backup controller although the performance will deteriorate slightly. The computation time was recorded and plotted in Figure 6-10. The time needed for determining the initial guess is between 3 μs and 4 μs. When the NMPC is activated, the computation time grew to slightly under 0.2 ms (after $T = 1.0s$ when the impact was applied. Compared to the sampling time T_s of 50 ms, it is clear how the proposed method is capable of real-time implementation with a huge margin.

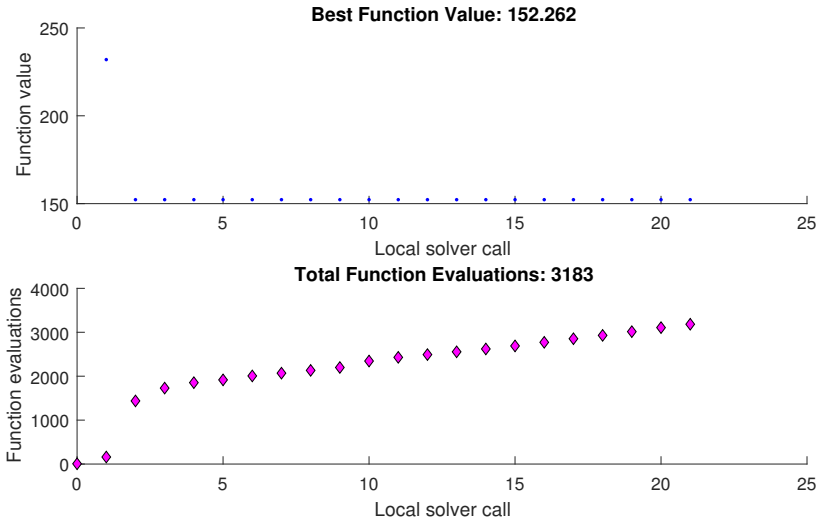

Figure 6-7: Demonstration of the optimization process given by the MATLAB Global Search Function.

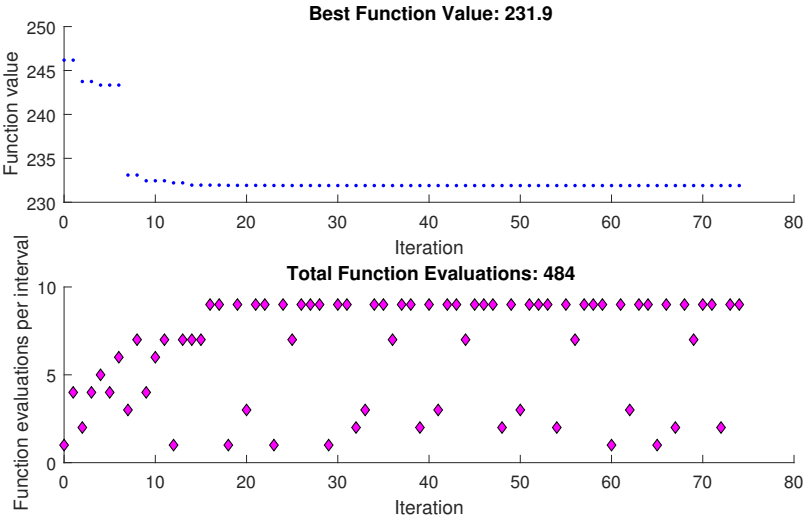

Figure 6-8: Demonstration of the optimization process given by the MATLAB Global Search Function.

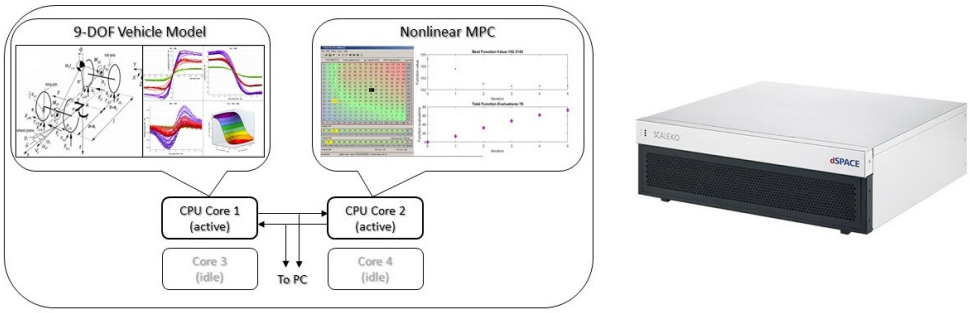

Figure 6-9: Schematic drawing of the experiment setup on dSPACE hardware.

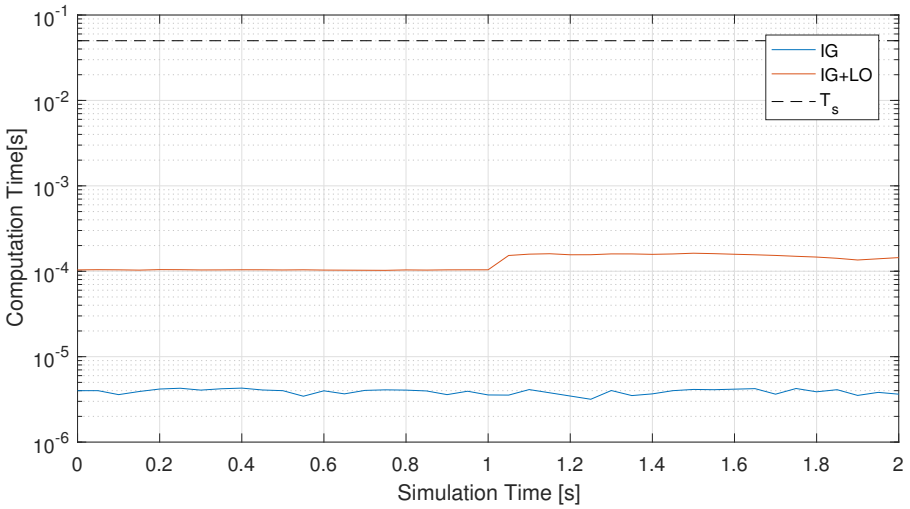

Figure 6-10: Computation time for the determination of initial guess and optimization of control input, when compiled on dSPACE. The dashed line shows the sampling time of the controller.

Chapter 7

Conclusions

7-1 Main Contributions

The thesis proposed a method for improving the numerical efficiency in solving nonlinear MPC problem, in order to enable real-time implementation of nonlinear MPC in vehicle dynamics control. The method first modified the Pacejka's combined-slip model into a function of braking force and side-slip angle to reduce model complexity. Then the corresponding PWA model of vehicle dynamics was derived by neglecting several nonlinear factors. With the PWA model, a hybrid MPC controller was generated to solve for an explicit map containing the initial guesses to the nonlinear optimization problem that corresponds to the NMPC control problem. Consequently, the global optimization in solving nonlinear MPC problem can be replaced by a local optimization performed on top of the explicitly stored initial guess. To further get rid of the computational burden introduced by numerical evaluation of function's gradient, a perpendicular search plus varied-step line search method has been developed.
The resulted nonlinear MPC controller was tested under real-world situations. The controller successfully stabilized the vehicle even when directly forced into side-slip motion by lateral impact up to 32.5 kN with a yaw moment of 58.5 kNm under the speed of 50 km/h. Even when tested under the top speed that normal economy-class passenger vehicles can reach, the NMPC controller can still recovery from a lateral impact up to 30.5 kN exerted behind the rear axle. Meanwhile, the initial guess itself, based on the hybrid MPC, is also capable of beyond-the-limit stabilization, although the maximum magnitude slightly decreases. This implies the high accuracy of the initial guess.
Afterward, the numerical performance of the proposed method is tested in comparison to two methods provided by the MATLAB Global Optimization Toolbox. Significance improvement has been indicated regarding both speed and quality. The proposed method is 7.33 times as fast as the Pattern Search method and 49 times as fast as the Global Search. With the help of the properly generated initial guess, the proposed method found the correct global optimum in all of the tests and the loss of accuracy is completely negligible. Global Search, despite the highly intensive computational effort, converged to a wrong local optimum in 2 out of 500 cases and the Pattern Search suffered the same problem quite frequently. When

compiled in C code on dSPACE processing board, the proposed method can return the optimal control input within 0.2 ms hence allowed real-time implementation with a huge margin.

7-2 Limitations and Recommendations

It must be admitted that nonlinear MPC is strongly dependent on the modelling accuracy. In this case, the control performance is not actually optimal when tested with the multibody model. There are multiple towards more accurate modelling of the vehicle dynamics. In the development of the PWA tire model, the parameters can be optimized to reduce the overall mismatch from the nonlinear one. The wheel dynamics, roll dynamics and response of the actuators can be incorporated in the vehicle model in NMPC and probably also in the hybrid model. Incorporation of additional degrees of freedom and the corresponding dynamics may result in higher computational load. Nevertheless, the real-time margin has allowed plenty of space for this. The computational load is still worthy if it is accompanied by improved control performance and passengers' safety.
Besides, the thesis only investigated the stabilization of vehicle's lateral motion at $r = \beta = 0$. In practical application, there should be a reference value for the controller to track. This can be done by manipulating the object function but the time span of the thesis is not sufficient for further investigation into that. The proposed controller can be easily extended to control a vehicle equipped with four-wheel-steering and in-wheel electric machines. By also incorporating the kinematics model of planar motion, trajectory control is another possible extension from the current study.
The framework of exploiting hybrid model to accelerate NMPC may also be implemented in other systems with quick dynamics such as moving-based robot and aircraft. It is also possible to add learning-based features by means of, for example, updating the lookup table for initial guess online.

Appendix A

Parameters of the PWA Tire Model

Parameter	FL and FR	RL and RR
a_{i1}	96667	75000
a_{i2}, a_{i3}	26667	18333
a_{i4}, a_{i5}	-3049	-2317
a_{i6}, a_{i7}	-250	-200
$b_{i1} - b_{i7}, c_{i1}, d_{i1,3} - d_{i7,3}, d_{i1,4}, d_{i7,4}, e_{i1,1} - e_{i7,1}, e_{i1,2} - e_{i7,2}, f_{i1,4} - f_{i7,4}$	0	0
c_{i2}	2100	1700
c_{i3}	4774	3559
c_{i4}	3375	2500
c_{i5}	-2100	-1700
c_{i6}	-4774	-3559
c_{i7}	-3375	-2500
$d_{i1,1} - d_{i7,1}, e_{i1,3} - e_{i7,3}, e_{i4,5}, e_{i5,5}$	1	1
$d_{i1,2} - d_{i7,2}, e_{i1,4} - e_{i7,4}$	-1	-1
$d_{i4,5}$	6.58	4.29
$d_{i5,5}$	-6.58	-4.29
$f_{i1,1}, f_{i1,2}$	0.03	0.03
$f_{i2,2}, f_{i3,1}$	-0.03	-0.03
$f_{i2,1}, f_{i5,2}$	0.09	0.09
$f_{i3,1}, f_{i4,2}$	-0.09	-0.09
$f_{i4,1}, f_{i7,2}$	0.5	0.5
$f_{i5,1}, f_{i6,2}$	-0.5	-0.5
$f_{i6,1}$	1.5	1.5
$f_{i7,1}$	-1.5	-1.5
$f_{i1,3} - f_{i5,3}$	2.5	1.5
$f_{i6,4}, f_{i7,4}$	ϵ	ϵ
$f_{i4,5}, f_{i5,5}$	3.29	2.14

Table A-1: Parameters of PWA tire model

Appendix B

State-space Matrices of the Hybrid 2-Track Model

j_F, j_R	$\mathbf{A^d_{j_F,j_R}}$		$\mathbf{B^d_{j_F,j_R}}$					$\mathbf{f^d_{j_F,j_R}}$
1,1	0.575	0.199	0.0103	−0.0103	0.0103	−0.0103	2.987	0.0000
	−0.028	0.578	−0.0002	0.0002	−0.0002	0.0002	0.169	0.0000
1,2	0.809	−2.086	0.0119	−0.0119	0.0119	−0.0119	3.060	−0.0947
	−0.046	0.760	−0.0003	0.0003	−0.0003	0.0003	0.164	0.0053
1,4	0.923	−3.226	0.0126	−0.0126	0.0126	−0.0126	3.092	−0.1983
	−0.055	0.848	−0.0004	0.0004	−0.0004	0.0004	0.162	0.0112
1,6	0.910	−3.100	0.0126	−0.0126	0.0125	−0.0125	3.088	−0.1393
	−0.054	0.839	−0.0004	0.0004	−0.0004	0.0004	0.162	0.0079
1,3	0.809	−2.086	0.0119	−0.0119	0.0119	−0.0119	3.060	0.0947
	−0.046	0.760	−0.0003	0.0003	−0.0003	0.0003	0.164	−0.0053
1,5	0.923	−3.226	0.0126	−0.0126	0.0126	−0.0126	3.092	0.1983
	−0.055	0.848	−0.0004	0.0004	−0.0004	0.0004	0.162	−0.0112
1,7	0.910	−3.100	0.0126	−0.0126	0.0125	−0.0125	3.088	0.1393
	−0.054	0.839	−0.0004	0.0004	−0.0004	0.0004	0.162	−0.0079
2,1	0.639	2.149	0.0109	−0.0109	0.0109	−0.0109	0.964	0.0833
	−0.025	0.687	−0.0002	0.0002	−0.0002	0.0002	0.056	0.0066
2,2	0.870	−0.032	0.0125	−0.0125	0.0124	−0.0124	0.986	−0.0114
	−0.044	0.869	−0.0003	0.0003	−0.0003	0.0003	0.054	0.0119
2,4	0.983	−1.127	0.0132	−0.0132	0.0131	−0.0131	0.996	−0.1150
	−0.052	0.957	−0.0004	0.0004	−0.0004	0.0004	0.053	0.0178
2,6	0.971	−1.006	0.0132	−0.0132	0.0131	−0.0131	0.995	−0.0560
	−0.052	0.947	−0.0004	0.0004	−0.0003	0.0003	0.053	0.0145
2,3	0.870	−0.032	0.0125	−0.0125	0.0124	−0.0124	0.986	0.1781
	−0.044	0.869	−0.0003	0.0003	−0.0003	0.0003	0.054	0.0013
2,5	0.983	−1.127	0.0132	−0.0132	0.0131	−0.0131	0.996	0.2816
	−0.052	0.957	−0.0004	0.0004	−0.0004	0.0004	0.053	−0.0046

2,7	0.971	−1.006	0.0132	−0.0132	0.0131	−0.0131	0.995	0.2226
	−0.052	0.947	−0.0004	0.0004	−0.0003	0.0003	0.053	−0.0013
4,1	0.675	3.197	0.0112	−0.0112	0.0112	−0.0112	−0.118	0.1895
	−0.023	0.747	−0.0002	0.0002	−0.0002	0.0002	−0.007	0.0150
4,2	0.905	1.066	0.0128	−0.0128	0.0127	−0.0127	−0.121	0.0947
	−0.042	0.929	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0204
4,4	1.017	−0.008	0.0135	−0.0135	0.0134	−0.0134	−0.122	−0.0088
	−0.051	1.017	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0262
4,6	1.005	0.111	0.0135	−0.0135	0.0134	−0.0134	−0.122	0.0501
	−0.050	1.008	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0229
4,3	0.905	1.066	0.0128	−0.0128	0.0127	−0.0127	−0.121	0.2842
	−0.042	0.929	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0097
4,5	1.017	−0.008	0.0135	−0.0135	0.0134	−0.0134	−0.122	0.3878
	−0.051	1.017	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0038
4,7	1.005	0.111	0.0135	−0.0135	0.0134	−0.0134	−0.122	0.3288
	−0.050	1.008	−0.0003	0.0003	−0.0003	0.0003	−0.007	0.0072
6,1	0.671	3.092	0.0112	−0.0112	0.0111	−0.0111	−0.010	0.1339
	−0.023	0.741	−0.0002	0.0002	−0.0002	0.0002	−0.001	0.0106
6,2	0.901	0.956	0.0128	−0.0128	0.0127	−0.0127	−0.010	0.0392
	−0.042	0.923	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0160
6,4	1.014	−0.120	0.0135	−0.0135	0.0134	−0.0134	−0.010	−0.0644
	−0.051	1.011	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0218
6,6	1.001	−0.001	0.0134	−0.0134	0.0133	−0.0133	−0.010	−0.0054
	−0.050	1.001	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0185
6,3	0.901	0.956	0.0128	−0.0128	0.0127	−0.0127	−0.010	0.2287
	−0.042	0.923	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0053
6,5	1.014	−0.120	0.0135	−0.0135	0.0134	−0.0134	−0.010	0.3322
	−0.051	1.011	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0006
6,7	1.001	−0.001	0.0134	−0.0134	0.0133	−0.0133	−0.010	0.2732
	−0.050	1.001	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0028
3,1	0.639	2.149	0.0109	−0.0109	0.0109	−0.0109	0.964	−0.0833
	−0.025	0.687	−0.0002	0.0002	−0.0002	0.0002	0.056	−0.0066
3,2	0.870	−0.032	0.0125	−0.0125	0.0124	−0.0124	0.986	−0.1781
	−0.044	0.869	−0.0003	0.0003	−0.0003	0.0003	0.054	−0.0013
3,4	0.983	−1.127	0.0132	−0.0132	0.0131	−0.0131	0.996	−0.2816
	−0.052	0.957	−0.0004	0.0004	−0.0004	0.0004	0.053	0.0046
3,6	0.971	−1.006	0.0132	−0.0132	0.0131	−0.0131	0.995	−0.2226
	−0.052	0.947	−0.0004	0.0004	−0.0003	0.0003	0.053	0.0013
3,3	0.870	−0.032	0.0125	−0.0125	0.0124	−0.0124	0.986	0.0114
	−0.044	0.869	−0.0003	0.0003	−0.0003	0.0003	0.054	−0.0119
3,5	0.983	−1.127	0.0132	−0.0132	0.0131	−0.0131	0.996	0.1150
	−0.052	0.957	−0.0004	0.0004	−0.0004	0.0004	0.053	−0.0178
3,7	0.971	−1.006	0.0132	−0.0132	0.0131	−0.0131	0.995	0.0560
	−0.052	0.947	−0.0004	0.0004	−0.0003	0.0003	0.053	−0.0145

5,1	0.675	3.197	0.0112	−0.0112	0.0112	−0.0112	−0.118	−0.1895
	−0.023	0.747	−0.0002	0.0002	−0.0002	0.0002	−0.007	−0.0150
5,2	0.905	1.066	0.0128	−0.0128	0.0127	−0.0127	−0.121	−0.2842
	−0.042	0.929	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0097
5,4	1.017	−0.008	0.0135	−0.0135	0.0134	−0.0134	−0.122	−0.3878
	−0.051	1.017	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0038
5,6	1.005	0.111	0.0135	−0.0135	0.0134	−0.0134	−0.122	−0.3288
	−0.050	1.008	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0072
5,3	0.905	1.066	0.0128	−0.0128	0.0127	−0.0127	−0.121	−0.0947
	−0.042	0.929	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0204
5,5	1.017	−0.008	0.0135	−0.0135	0.0134	−0.0134	−0.122	0.0088
	−0.051	1.017	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0262
5,7	1.005	0.111	0.0135	−0.0135	0.0134	−0.0134	−0.122	−0.0501
	−0.050	1.008	−0.0003	0.0003	−0.0003	0.0003	−0.007	−0.0229
7,1	0.671	3.092	0.0112	−0.0112	0.0111	−0.0111	−0.010	−0.1339
	−0.023	0.741	−0.0002	0.0002	−0.0002	0.0002	−0.001	−0.0106
7,2	0.901	0.956	0.0128	−0.0128	0.0127	−0.0127	−0.010	−0.2287
	−0.042	0.923	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0053
7,4	1.014	−0.120	0.0135	−0.0135	0.0134	−0.0134	−0.010	−0.3322
	−0.051	1.011	−0.0003	0.0003	−0.0003	0.0003	−0.001	0.0006
7,6	1.001	−0.001	0.0134	−0.0134	0.0133	−0.0133	−0.010	−0.2732
	−0.050	1.001	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0028
7,3	0.901	0.956	0.0128	−0.0128	0.0127	−0.0127	−0.010	−0.0392
	−0.042	0.923	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0160
7,5	1.014	−0.120	0.0135	−0.0135	0.0134	−0.0134	−0.010	0.0644
	−0.051	1.011	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0218
7,7	1.001	−0.001	0.0134	−0.0134	0.0133	−0.0133	−0.010	0.0054
	−0.050	1.001	−0.0003	0.0003	−0.0003	0.0003	−0.001	−0.0185

Table B-1: State-space matrices in the PWA vehicle model

Appendix C

Calibration of Vehicle Dynamics Model

C-1 Introduction

The project aims to calibrate a vehicle model to fit the dynamic characteristics of 3rd generation Toyota Prius, so that it can later be used in the design of path-following controller of this vehicle.

The model we started with was a roll-axis model calibrated for BMW 5 series. Several changes were made due to the difference between 5 series and Prius: the driving force was applied on the front wheels instead of rear wheels as Prius is a front wheel drive vehicle; the cruise control module was tuned to fit the output of the hybrid power-train of Prius. We also tried to make the model more realistic by introducing Ackermann angle to the steering system.

The parameters of the model were updated from multiple sources, including official publications from manufacture, previous measurements, calculation from experimental data, and estimation. Some tuning were required to fit the model to experimental data.

The calibrated model was validated by comparing the simulation results to experimental data. The model has shown good matching in steady state, step steer, and sinusoidal maneuvers.

C-2 Structure of Model

C-2-1 Tire Modelling

The Delft-tire model is applied to determine the forces on the contact patch. Magic formula is adopted for evaluation of forces. Combined slip and relaxation behavior are included.

C-2-2 Roll Dynamics

The dynamics of body's roll motion is modelled with SimMechanics. Three rigid bodies are included in the model, namely the sprung body and front and rear axle. Either axle is

attached to the sprung body with a prismatic joint to enable vertical motion, and a revolute joint to enable roll motion.

C-2-3 Cruise Control

The longitudinal driving/braking torques are regulated by the cruise control module to track the velocity profiles recorded during experiments. A PI controller is used to apply torques to front wheels based on the error in longitudinal velocity, which inevitably produced some delay in velocity tracking. We expect that using the accelerator pedal position together with a model of the power train could yield better result, however the Move Box system failed to record the accelerator pedal position, so we shall leave this as a suggestion for future research.

C-3 Determination of Parameters

Setting the parameters properly was one of the most challenging part of this project since some of the parameters are not published and measuring them requires special equipment which we do not have access to. Therefore, we first adopted the published parameters, such as the wheel base and overall dimension of the vehicle; then we checked measurements conducted in previous researches and applied them if they are consistent with our calculation, such as the axle weight and roll gradient; some parameters were calculated from experimental data or tuned to fit the model to the experimental data, such as the steering ratio and roll stiffness; finally, a few other parameters were estimated based on common practices in automotive industry.

C-3-1 Inertia Terms

Due to the limitation mentioned above, the rotational inertia components are estimated according to recommendations given by Duane et al [23], as shown in equation C-1. Duane et al collected the inertia data from large amount of measurements and fitted to vehicle dimensions including vehicle mass m, total length L, wheelbase$l$, wheel track t, roof height h_{roof}, etc. The resulted inertia terms are: $I_{xx} = 623[kgm^2]$, $I_{yy} = 2728[kgm^2]$, $I_{zz} = 2830[kgm^2]$.

$$\begin{aligned} I_{xx} &= 0.1274m(h_{roof} + h_{cg})t \pm 6.5\% \\ I_{yy} &= 0.1425mlL \pm 5.7\% \\ I_{zz} &= 0.1478mlL \pm 4.8\% \end{aligned} \tag{C-1}$$

The masses of front and rear axle, or the unsprung masses, are chosen according to recommendations from textbook [24] as a certain proportion of the total front and rear mass (see equation C-2, C-3). The resulted axle masses are: $m_{U,f} = 85[kg]$, $m_{U,r} = 65[kg]$.

$$m_{U,f} = i_{m,f} m_{V,f}/(1 + i_{m,f}) \tag{C-2}$$

$$m_{U,r} = i_{m,r} m_{V,r}/(1 + i_{m,r}) \tag{C-3}$$

C-3-2 Tire Properties

The tire property parameters are borrowed from a previously tested 225/50R17 tire from BMW 5-series, while the actual tire on the Prius is 215/45R17. The characteristics can be further modified, if necessary, by changing the scaling factors. The tire properties lead to a longitudinal stiffness of 126.9 kN and cornering stiffness of 92.4 kN/rad. The tire characteristics can be found in figure C-1.

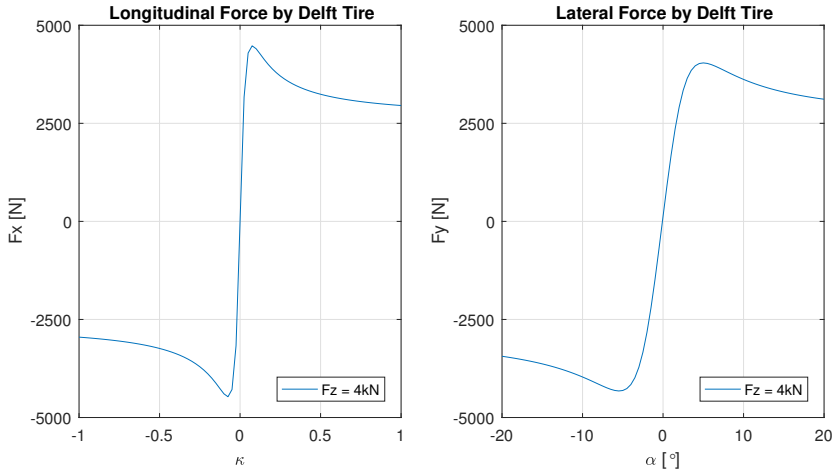

Figure C-1: Longitudinal and lateral forces

C-3-3 Steering Mechanism

The steering parameters were found to be most influential to steady state responses. The key parameters of steering system are steering ratio k_s, Ackermann factor p_{Ac}, and steering wheel angle offset δ_{off}. These parameters have not been published and direct measurement was unfortunately not possible due to equipment and time limitations.

The steering ratio k_s is the ratio between the average steering angle on both front wheels δ_{avg} and the steering wheel angle δ_w. An alternative approach was proposed to calculate k_s from the experimental data of steady state maneuver. We assume that the side slip angles of all tires are negligible at very low speed, then the motion of the vehicle is decided by only the geometry of the steering system and the steering wheel angle δ_w. The turning radius of the vehicle r_{turn} was calculated in 2 methods: fitting the trajectory data points to a circle with the method given by Pratt [25], or calculated from the longitudinal velocity u and yaw rate r as equation C-4. The trajectory of the maneuver, u, and r are measured and logged by the GNSS/INS system with the sample rate of 25Hz.

$$r_{turn} = u/r \tag{C-4}$$

The first approach requires the trajectory used for fitting to be sufficiently long, otherwise the fitting error will become very large. However, the fitting process gives only the average radius of the selected trajectory. So some transient change of radius might be lost due to averaging. After some tuning, we found that averaging over 4 seconds of trajectory (from

2 seconds before current time step till 2 second later) provides the best trade-off between fitting error and averaging effect. A comparison between different averaging length is shown in Figure C-2.

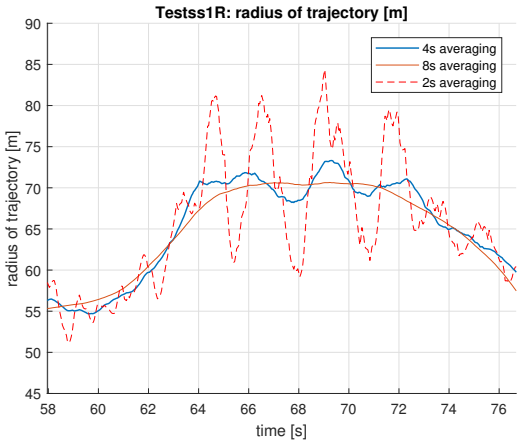

Figure C-2: Radius of trajectory with different averaging length

The second approach was affected by the noise in the measurement of velocities. The longitudinal velocity u are generally free of noise, while the yaw rate r was more noisy due to the rather high sample rate of the GNSS/INS system at $25Hz$. A low pass Fourier filter with cut-off frequency of $1Hz$ was used to eliminate the high frequency noise. An example of original and filtered data are shown in the first plot of Figure C-3.

The resulted turning radius from both method are generally consistent with each other as shown in Figure C-4.

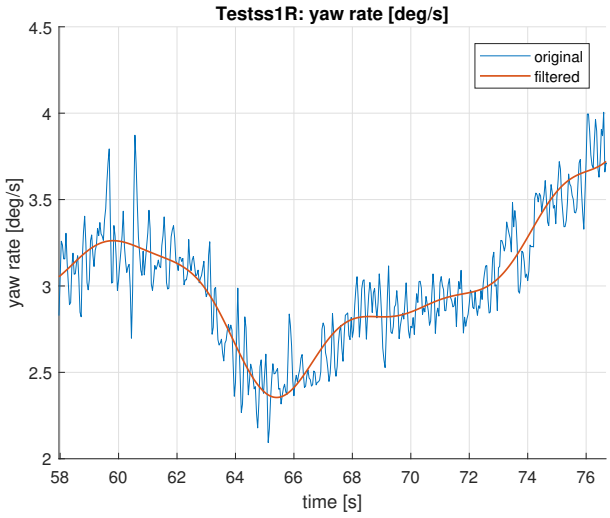

Figure C-3: Fourier filtering of yaw rate

While the turning radius r_{turn} is know, the average steering angle of front wheels δ_{avg} is calculated using the wheel base L:

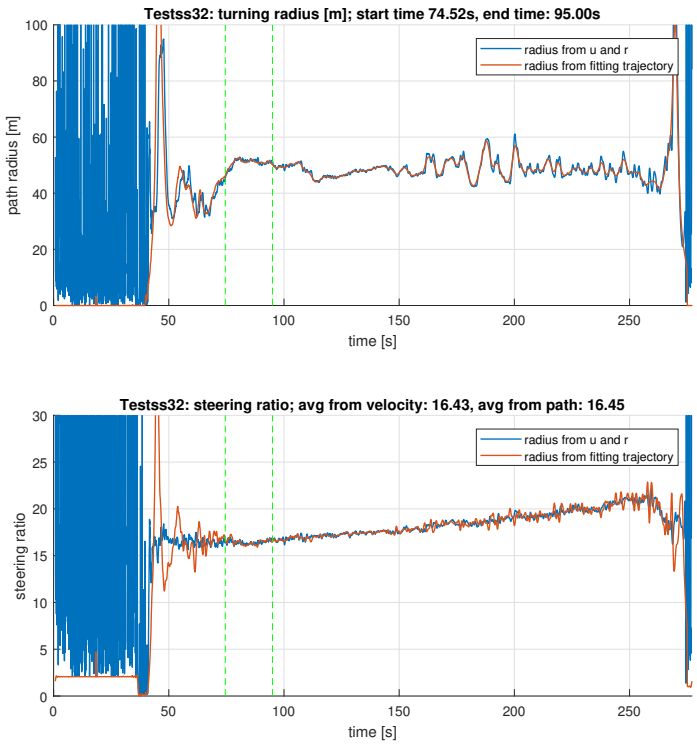

Figure C-4: The turning radius of the vehicle, calculated by both methods

$$\delta_{avg} = arctan(L/r_{turn}) \tag{C-5}$$

Then k_s can be calculated directly as δ_w is measured by the on board steering wheel angle sensor and logged by the Move Box.

$$k_s = \delta_{avg}/\delta_w \tag{C-6}$$

As mentioned above, we only consider the resulted steering ratio at very low velocity (2 to 4 m/s), which was marked by the 2 dashed lines in Figure C-4. Averaging over the selected section gives the steering ratio, as shown in Table C-1. Clearly there is a big difference between the left hand and right hand turn. We believe such asymmetry is mainly due to the steering wheel angle sensor of the vehicle is not perfectly aligned with the steering wheel, therefore created a static error, i.e. the sensor would give a none zero value when the steering wheel is at center position. We suspect that the steering wheel angle sensor was not perfectly aligned with the center position of the steering wheel due to manufacturing imperfections, as the Move Box system was using the stock steering wheel angle sensor that was not calibrated prior to the experiment, no external steering wheel sensor was used.

The steering wheel angle offset δ_{off} was introduced to compensate for this misalignment as shown in equation C-7. After some tuning, the offset of 0.09rad (5.16 deg) matched the left hand and right hand steering ratio at about 16.5, as shown in Table C-2.

$$k_s = \delta_{avg}/(\delta_w + \delta_{off}) \tag{C-7}$$

Left hand turn		Right hand turn	
Run 1	14.7	Run 1	18.4
Run 2	14.7	Run 2	18.3

Table C-1: Steering ratio from steady state maneuver (without offset)

Left hand turn		Right hand turn	
Run 1	16.5	Run 1	16.4
Run 2	16.4	Run 2	16.5

Table C-2: Steering ratio from steady state maneuver (with offset of 0.09rad)

After δ_{off} and K_s are set, the Ackermann factor p_{Ac} was tuned to match the yaw rate response and lateral acceleration response in steady state maneuver. Generally a higher Ackermann factor enables the front wheels to produce higher lateral forces that lead to higher yaw rate and lateral acceleration. We have found that $p_{Ac} = 50\%$ provided a good match between the simulation results and experimental data, as shown in Figure C-9.

Additionally, the offset in steering wheel angle can be found statistically from the experimental data. By selecting all the samples of steering wheel angle corresponding to a yaw rate smaller than 0.01 [rad/s] from the return journey, a histogram as shown in figure C-5 can be obtained. An obvious deviation from zero can be seen, suggesting an offset between 2 and 4 degrees or equally 0.035 and 0.070 rad. The difference from the choice of 0.090 rad might be introduced from the roads' typical cross slope (for drainage) of 2 percent.

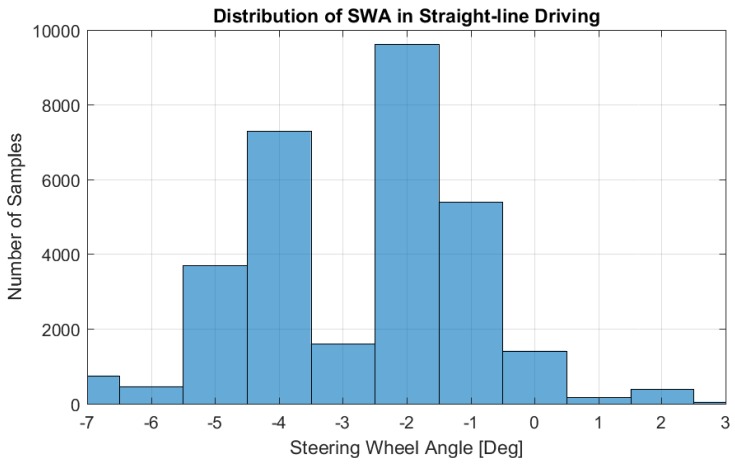

Figure C-5: Histogram of steering wheel position in straight-line driving

C-3-4 Suspension

The vertical stiffness and roll stiffness of front and rear axle are the key parameters in the roll-axis model of dynamics. In addition, the distribution of roll stiffness between the front and rear axle plays an important role in the model's later characteristics, especially when driving close to the limit of handling. The parameters are determined with following steps.
The vertical stiffness is calculated such that the natural frequency of vertical motion on each axle is in a proper range of slightly over 1 Hz. Meanwhile, the frequency of rear axle is expected to be 10-20 percent higher than the front [24]. By setting the vertical stiffnesses to 52 / 48 kN/m, the natural frequencies become 1.25 / 1.43 [Hz] (front / rear). The vertical damping are then decided to achieve a damping ratio between 0.25 and 0.35 [26]. Therefore the damping coefficients are 4.0 / 3.2 [kNs/m].
The distribution of roll stiffness between the front and rear axles plays an important role in the steering characteristics of a vehicle, especially for close-to-limit cases. For safety concern, family sedans expect more load transfer on the front axle, i.e. a more roll stiffness distributed on the front axle. The weight distribution is approximately 1.4:1 for front:rear. Therefore an initial guess of 1.54:1, i.e. the front axle is 10 percent stiffer than the rear, is implemented and to be further verified.The total roll stiffness can be first roughly estimated with simple mechanical model, where two torsional springs are attached to a bar, and the centrifugal force is applied to the bar with an offset equal to the difference between the height of gravity center and the height of front or rear roll center. The height of gravity is reported from experimental measurement. The height of front roll center are chosen from recommended range for McPherson suspension, while height of rear roll center is equal to half of the wheel's rolling radius. The reported value of roll gradient is 0.45 [degs^2/m] thus the estimated total roll stiffness is 89 [kNm/rad].

C-3-5 Powertrain

The power characteristics is obtained from online resource which offered a plot of wheel power versus longitudinal velocity (see figure C-6). The curve was measured with the hybrid system operating with its embedded logic and gives an insight of the maximum torque available at each velocity.

C-4 Model Validation

The calibrated model is validated through comparison to experiments performed previously. The experiments included steady-state cornering, step input and sine wave test. The velocity profile and steering wheel angle were recorded and the lateral acceleration and yaw velocity are available for comparison.

C-4-1 Steady-state Cornering

The experimental test of steady-state cornering was carried out as the steering wheel was fixed while the longitudinal velocity increased from 0 to 15 m/s. Steady-state cornering is

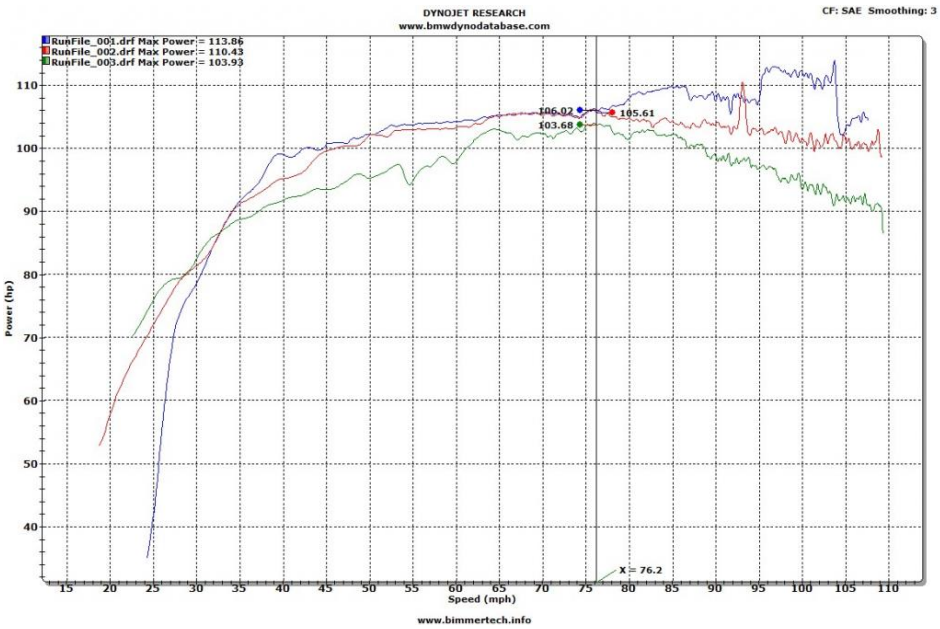

Figure C-6: Dyno test results of Toyota Prius

adopted to to examine the model parameters including steering ratio and total roll stiffness. Firstly, the rough estimate of total roll stiffness lead to obvious difference in roll gradient. As shown in figure C-7, the resulted roll gradient is approximately 0.66 [degs^2/m]. By increasing the total roll stiffness by roughly 40 percent to 124 [kNm/rad], the expected roll gradient is achieved (see figure C-8). The roll gradient data shown in figure C-7 and C-8 were calculated with lateral acceleration filtered up to 1Hz and yaw rate filtered up to 0.25Hz. The matching of steady-state gain of yaw rate and lateral acceleration, as can be observed from figure C-9, confirms the validity of steering ratio equal to 16.5. The amount of of outliers in the box plot is considerably large in the high acceleration range, despite that the same filter (cut-off frequency of 1 Hz) has been applied to experimental data. This could be a consequence of road's poor flatness, as the dynamics becomes sensitive to interruption in vertical load when the tires are close to saturation.

C-4-2 Step Steer Test

The vehicle's transient steering response is firstly tested with step steer, where the longitudinal velocity is fixed and a step input given to the steering wheel. The comparison of results from experiment and simulation verified the choice of distribution of roll stiffness, as can be seen from figure C-10, C-11 and C-12. No convincing explanation for the mismatch in amplitude and phase lag around the second peak has yet been found as the dynamics there are extremely complicated.

In principle, data from step steering tests should be further processed in order to compare criteria including rise time, settle time, overshoot, etc. However, the experimental tests failed to meet the standard of ISO 7401 [27] as the steering wheel input was manually actuated in the experimental test. Firstly, the steering wheel was not turned rapidly enough to be

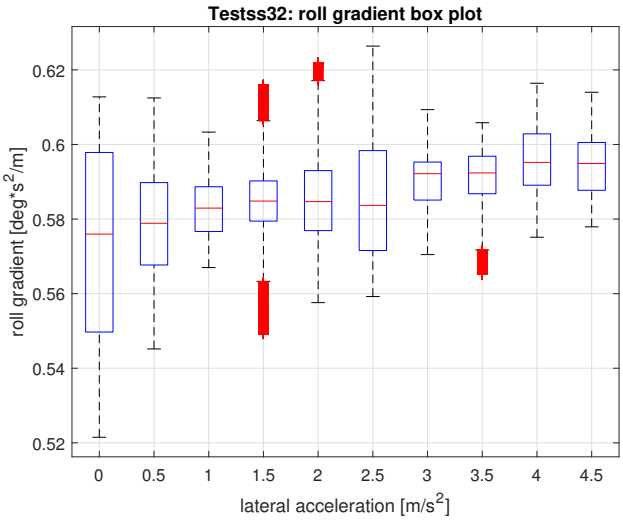

Figure C-7: Roll gradient of model with proposed total roll stiffness

considered as step input. The rise times of steering input are 0.84s, 0.88s and 0.44s for Run 01, 02 and 03, respectively (figure C-13), while the standard requires a rise time of no longer than 0.15s. Moreover, the steering wheel was not fixed for a sufficient duration so that the yaw rate and lateral acceleration failed to settle before the steering wheel angle changed again. Therefore, the detailed matching of step-response criteria was unavailable. The matching of profiles' shape and amplitude is achieved for the first step anyway. The deviation in the second period is caused by multiple uncertain factors and cannot yet be eliminated as limited experimental tests are available.

C-4-3 Continuous sinusoidal test

According to ISO standards [27], the test should start with a certain longitudinal velocity and the throttle position should be fixed despite possible drop of speed. Meanwhile the amplitude of steering wheel input should generate a lateral acceleration of 2, 4 or 6 $[m/s^2]$. The steering input roughly met the second requirement but the longitudinal velocity was set completely wrong (figure C-15). The matching of lateral acceleration and yaw rate is still checked anyway. The model gives higher amplitude in sine response as can be seen in figure C-14. The mismatch is possibly caused by the longitudinal load transfer during acceleration. The model does not include anti-squat parameter in suspension design and the vertical parameters are only chosen according to recommendations.

C-5 Conclusion

The calibrated model has shown good accuracy as it closely matched with the experimental data of Toyota Prius, especially in a moderate maneuvers. The modified parameters are listed below. As compared to the initial choices, the most significant change is observed in the front roll stiffness, which is reduced by 42 percent. This change is based on the matching of both transient steering response and steady-state roll gradient.

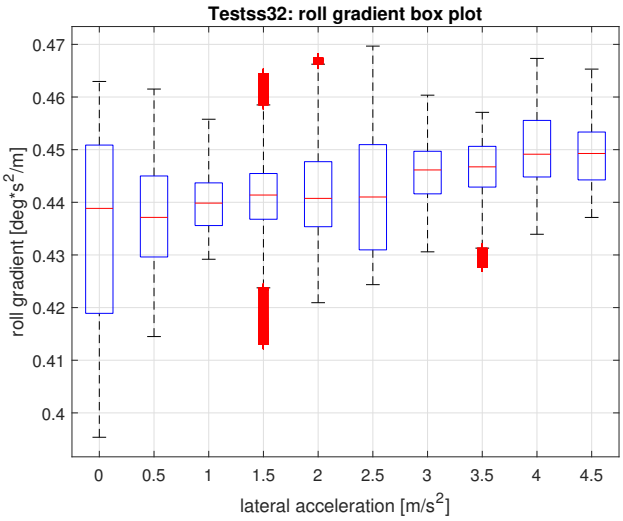

Figure C-8: Roll gradient of model with modified total roll stiffness

The parameters of the steering system (steering ratio, steering wheel offset, Ackermann factor) were found to be most influential to steady-state cornering characteristics. However, due to the lack of actual measurements, these parameters are calculated in a indirect way that involves multiple approximations and filtering, which might leads to large errors. A thorough measurement of the steering system should be carried out in order to validate these parameters before applying the model to any controller.

Also the stock steering wheel angle sensor has a resolution of $1.72deg$, which we believe is not sufficient for path-following tasks, especially at highway driving when the steering wheel angles are generally small. A steering wheel sensor with better resolution should be installed, and also calibrated so that it is aligned with the center position of the steering wheel in order to minimize static error.

Although some inertia terms and damping coefficients are also changed by a large proportion, their effect were not shown clearly in the maneuvers of this experiment. These parameters mainly affect the longitudinal and vertical dynamics characteristics while the experiment was focused on lateral dynamics. These parameters are not validated as related experiments were not conducted.
In the end, it is strongly recommended that the experiments can be repeated with strict compliance with ISO 7401 standard, in order to provide data with higher quality for more accurate parameter calibration.

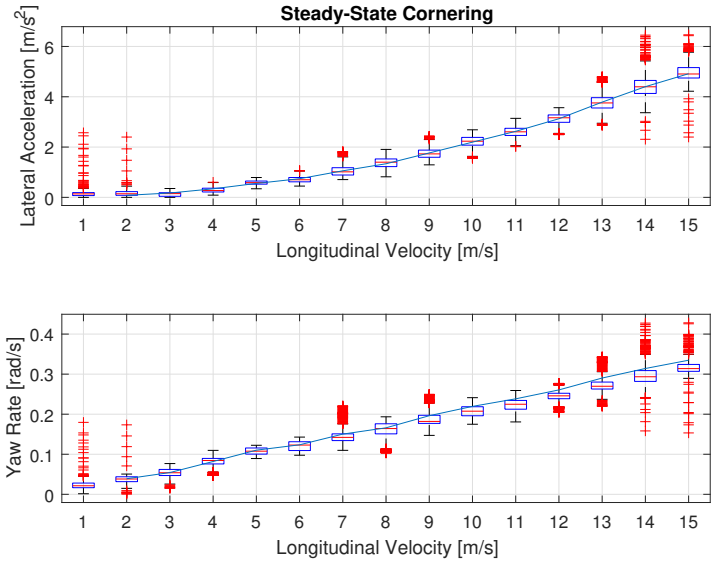

Figure C-9: Comparison of steady-state steering characteristics in box-plot

Parameter [unit]	Initial Value	Final Value	Source
Full Mass [kg]	1590	1590	Published
Roll Moment of Inertia [kgm2]	700	623	Estimated
Pitch Moment of Inertia [kgm2]	3200	2728	Estimated
Yaw Moment of Inertia [kgm2]	2787	2830	Estimated
Front Axle Mass [kg]	120	85	Estimated
Rear Axle Mass [kg]	90	65	Estimated
Height of Front Roll Center [m]	0.10	0.06	Estimated
Height of Rear Roll Center [m]	0.10	0.15	Estimated
Front Vertical Stiffness [kN/m]	52.0	52.0	Estimated
Rear Vertical Stiffness [kN/m]	40.0	48.0	Estimated
Front Vertical Damping [kNs/m]	3.0	4.0	Estimated
Rear Vertical Damping [kNs/m]	3.0	3.2	Estimated
Front Roll Stiffness [kNm/rad]	130.0	75.2	Data Fitting
Rear Roll Stiffness [kNm/rad]	40.0	48.8	Data Fitting
Front Roll Damping [kNms/rad]	2.0	3.3	Estimated
Rear Roll Damping [kNms/rad]	2.0	2.0	Estimated
Steering Ratio	15.65	16.50	Calculated
Ackermann Factor	0	0.5	Estimated
Steering Wheel Angle Offset [deg]	0	5.16	Calculated

Table C-3: Model parameters

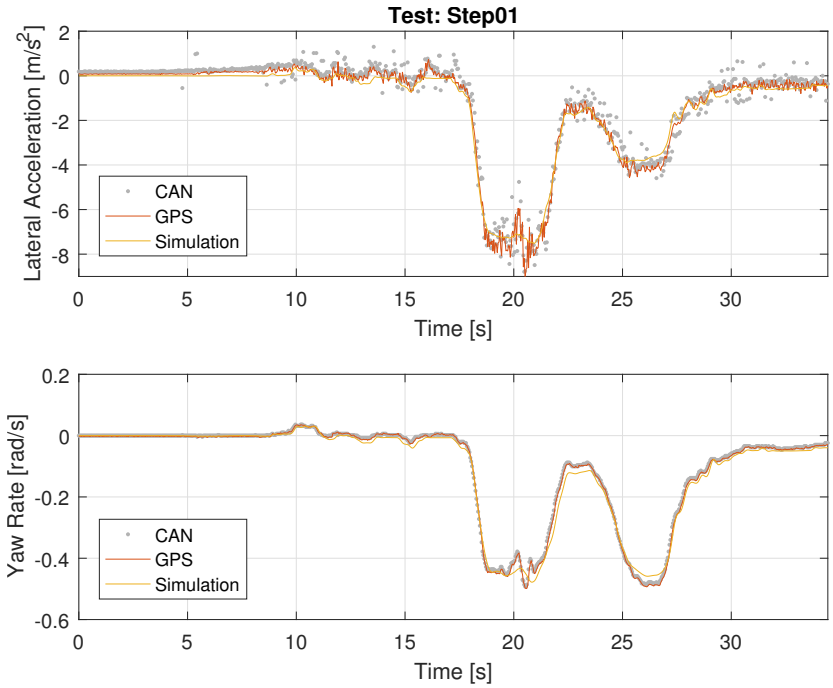

Figure C-10: Comparison of step response, Run01

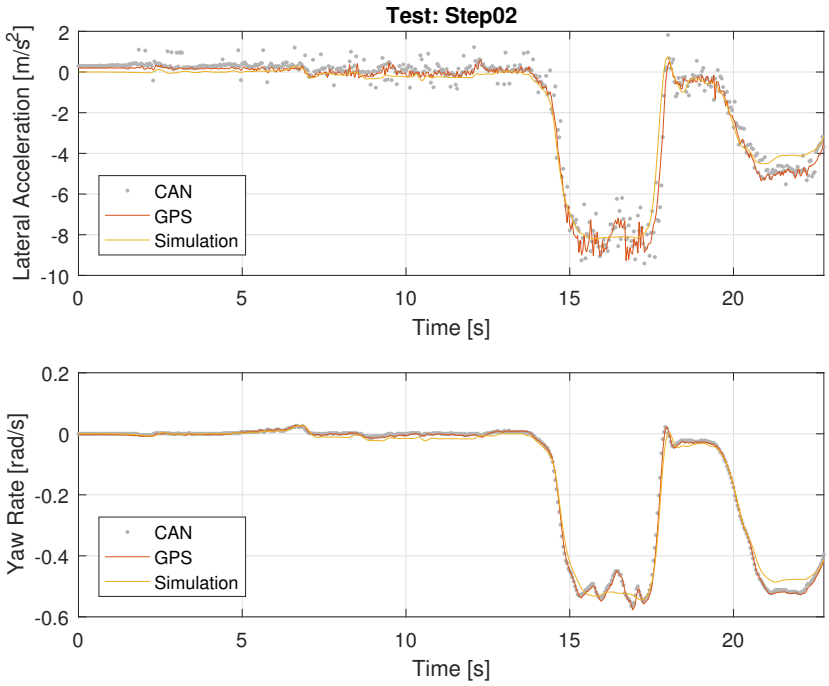

Figure C-11: Comparison of step response, Run02

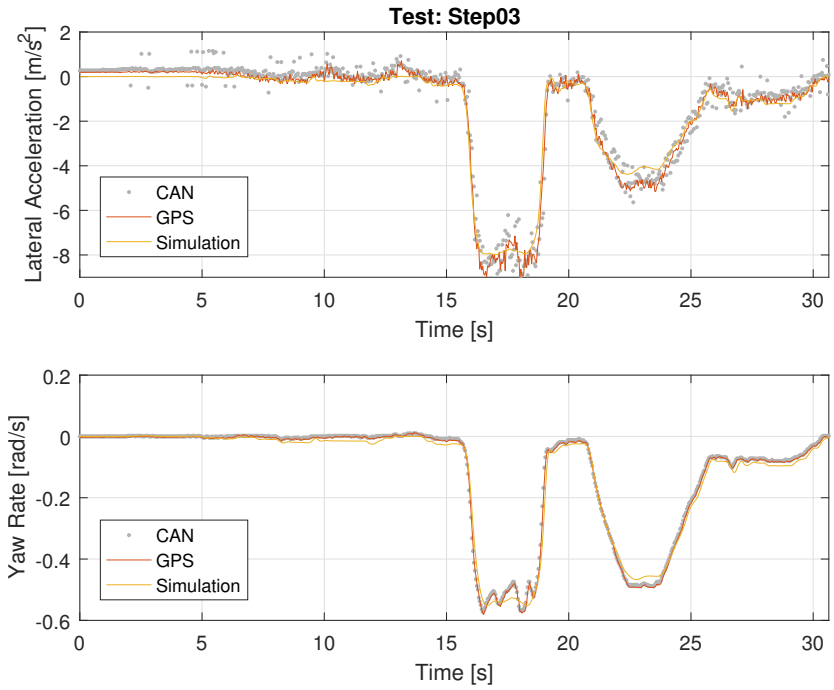

Figure C-12: Comparison of step response, Run03

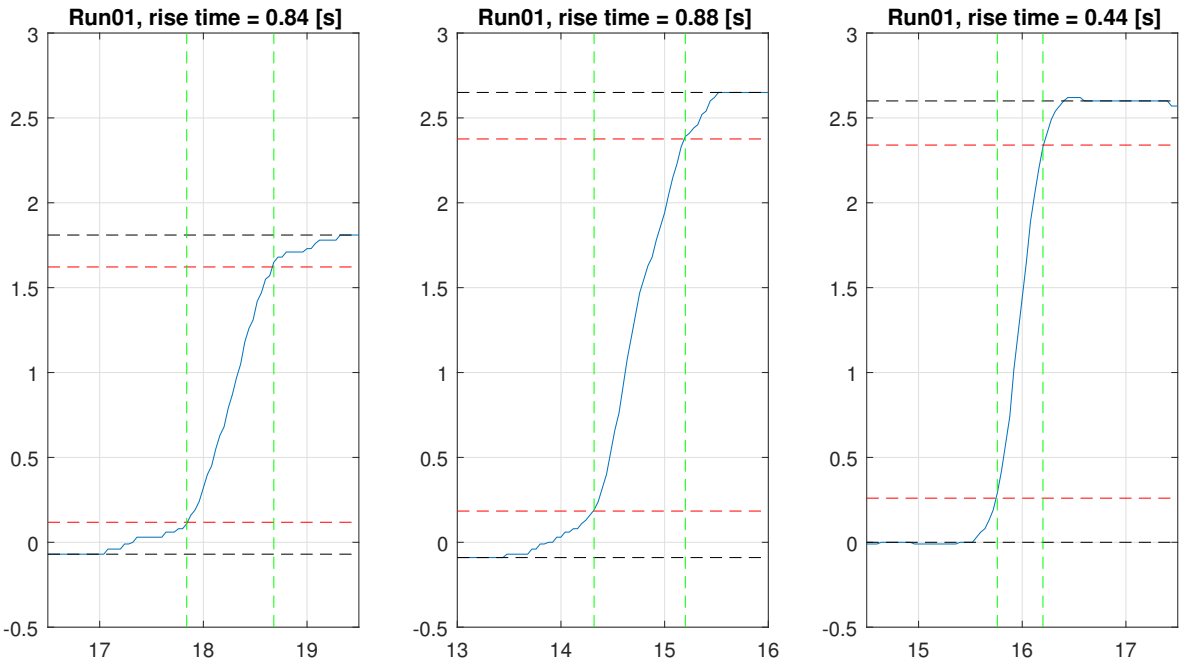

Figure C-13: Steering wheel input from experiments

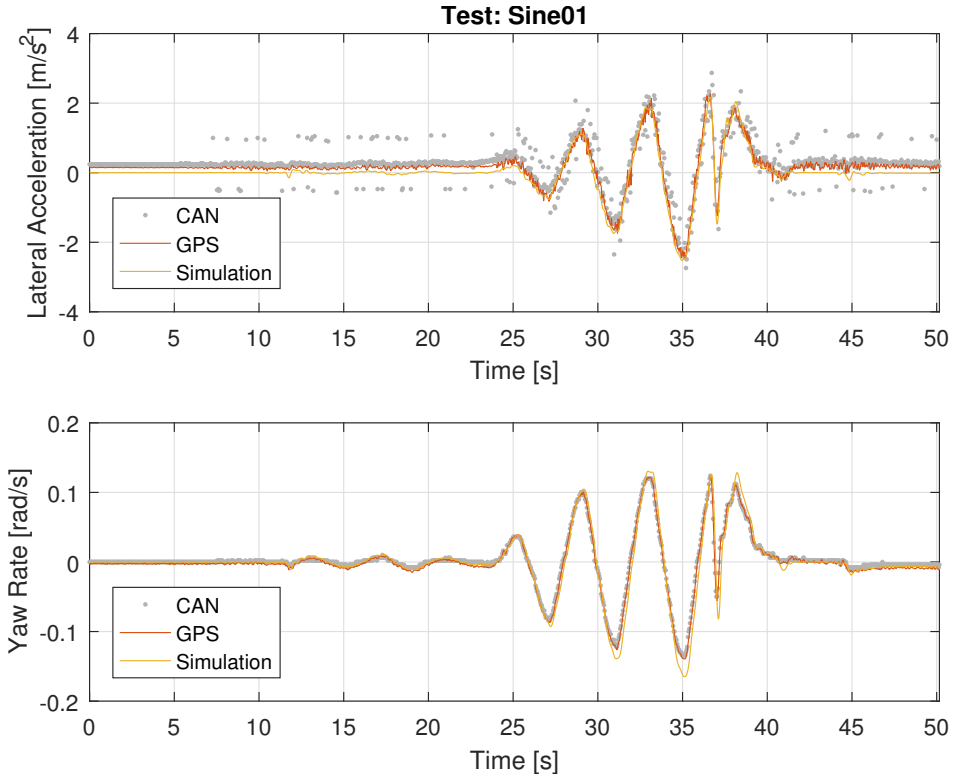

Figure C-14: Comparison of sine wave response

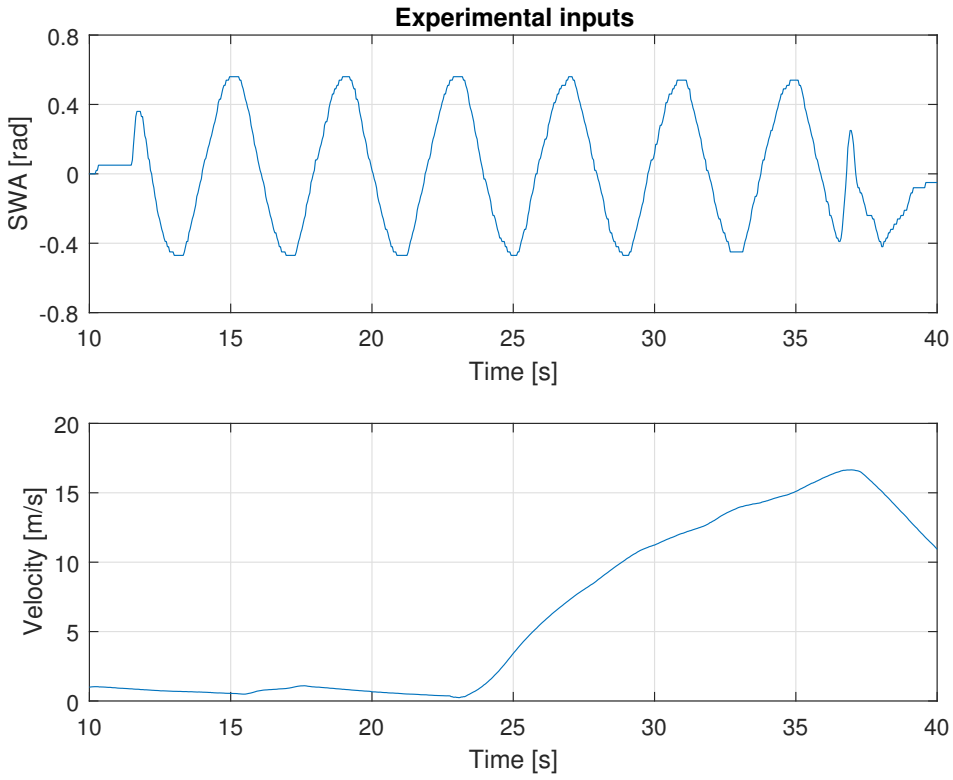

Figure C-15: Comparison of sine wave response

Appendix D

Simulation Results for the Performance Limit

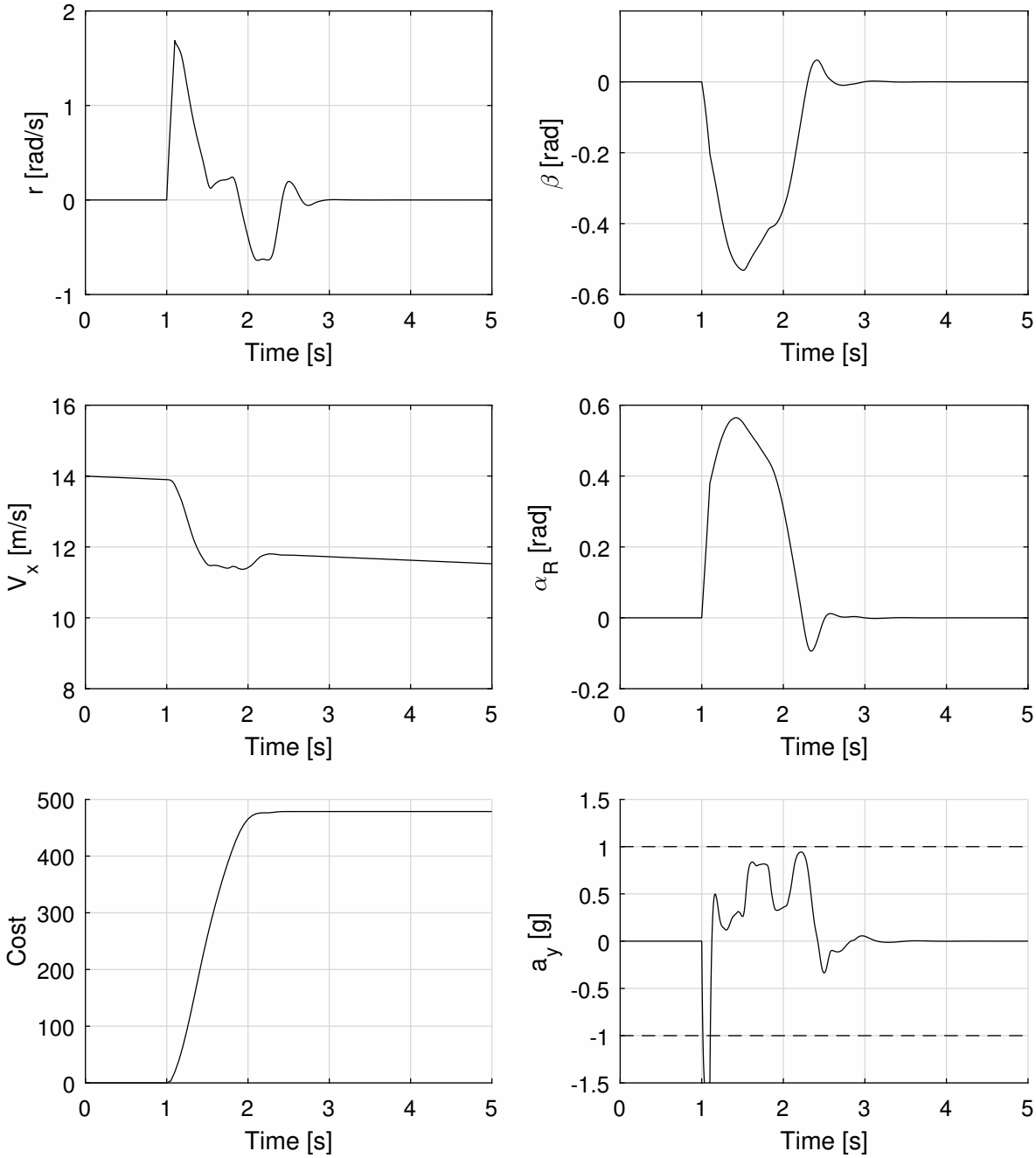

Figure D-1: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 14 m/s or 50 km/h.

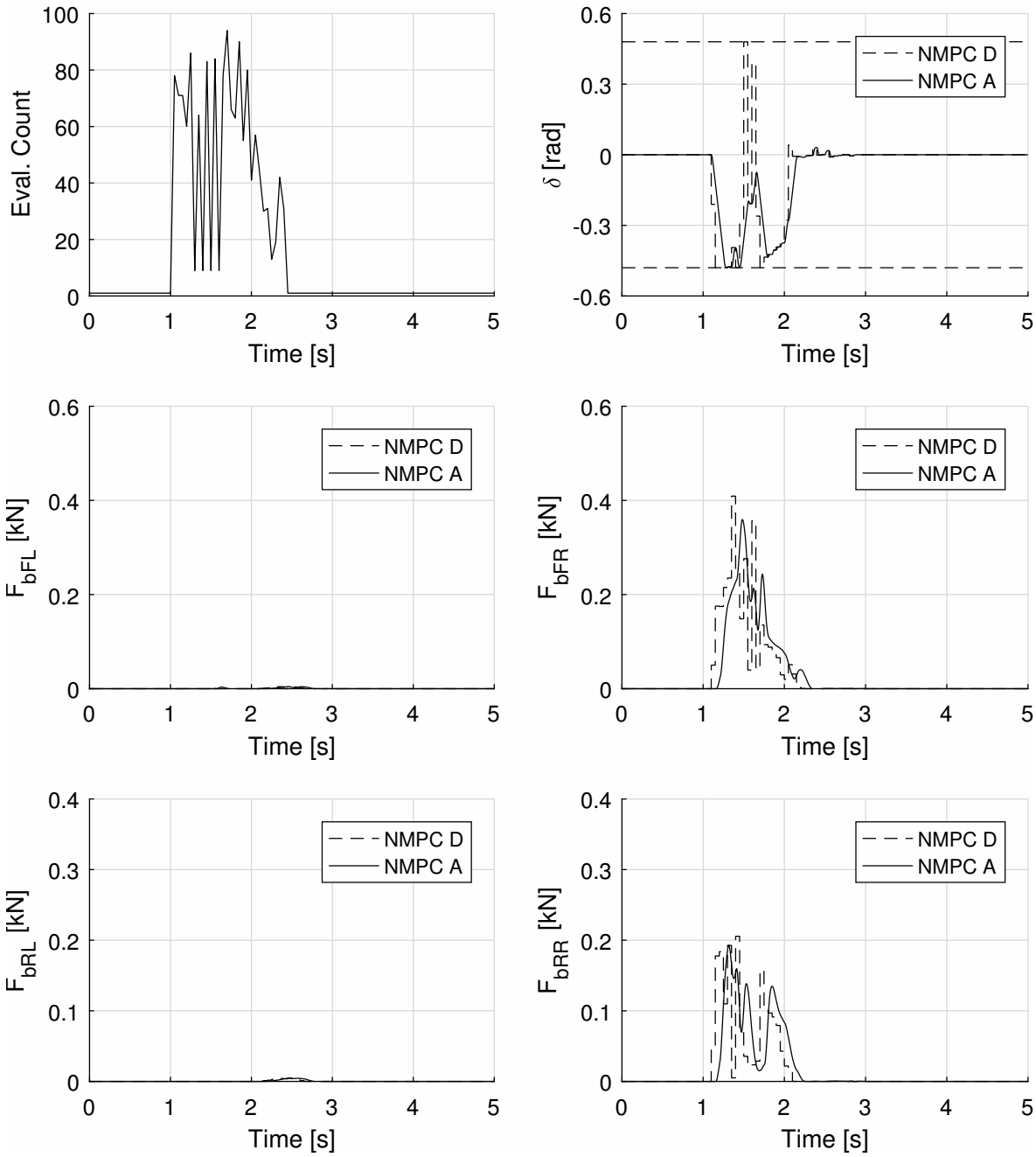

Figure D-2: Control inputs by NMPC during lateral collision recovery, with an initial speed of 14 m/s or 50 km/h. Dashed lines for control demand and solid lines for actual response of actuators. The total count of evaluation of the object function in the NMPC per time step is also presented.

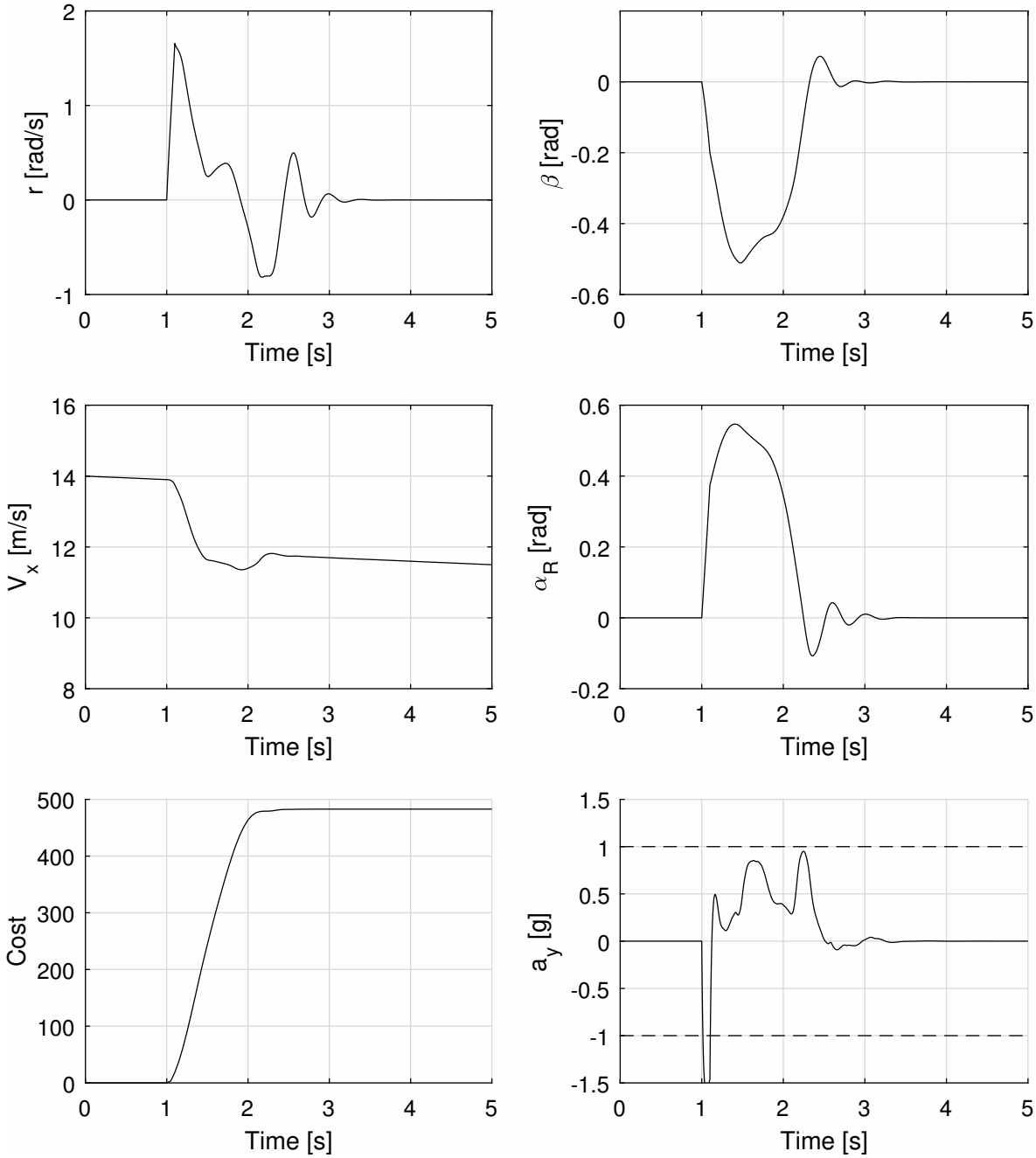

Figure D-3: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 14 m/s or 50 km/h.

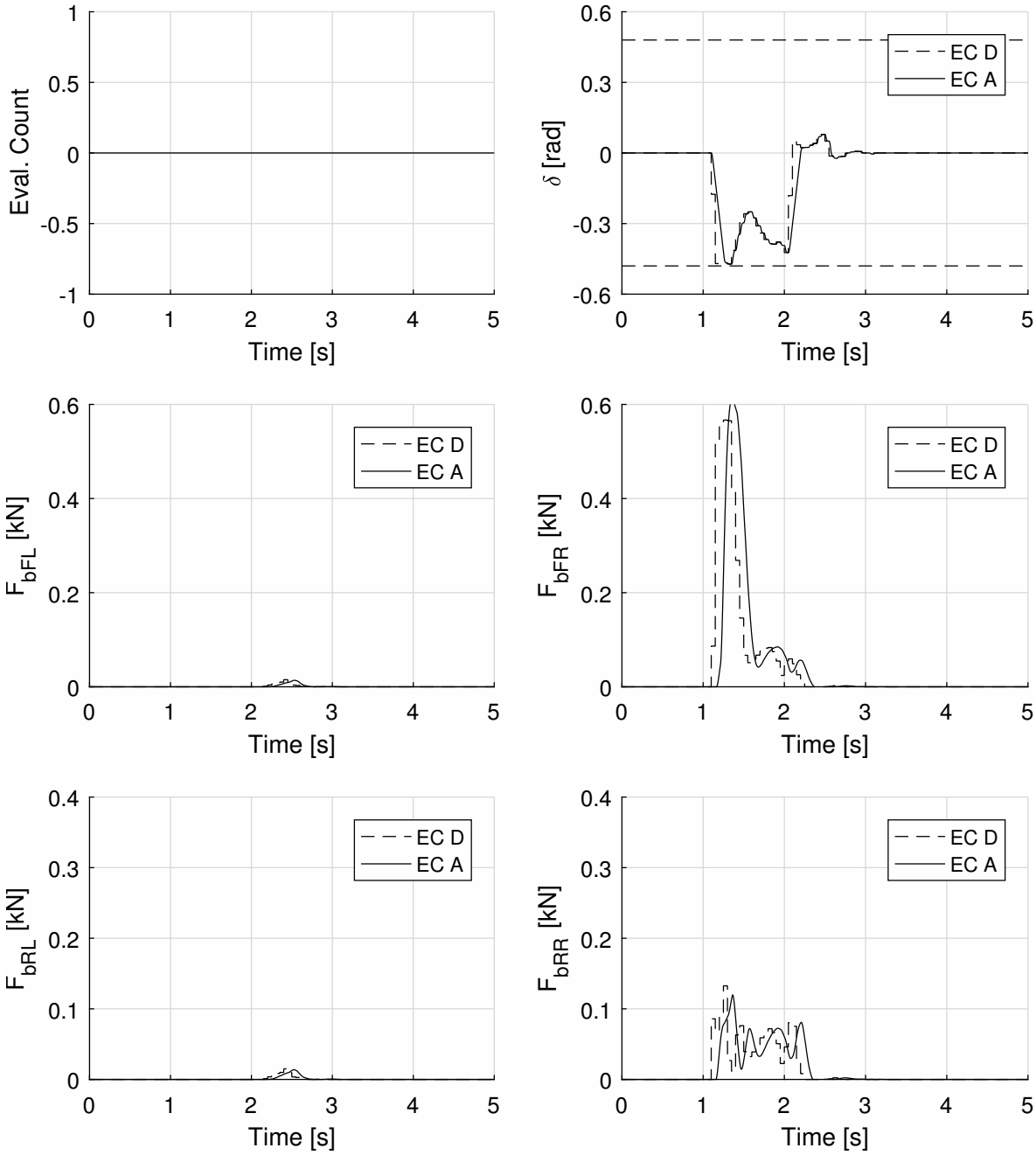

Figure D-4: Control inputs by explicit control during lateral collision recovery, with an initial speed of 14 m/s or 50 km/h. Dashed lines for control demand and solid lines for actual response of actuators.

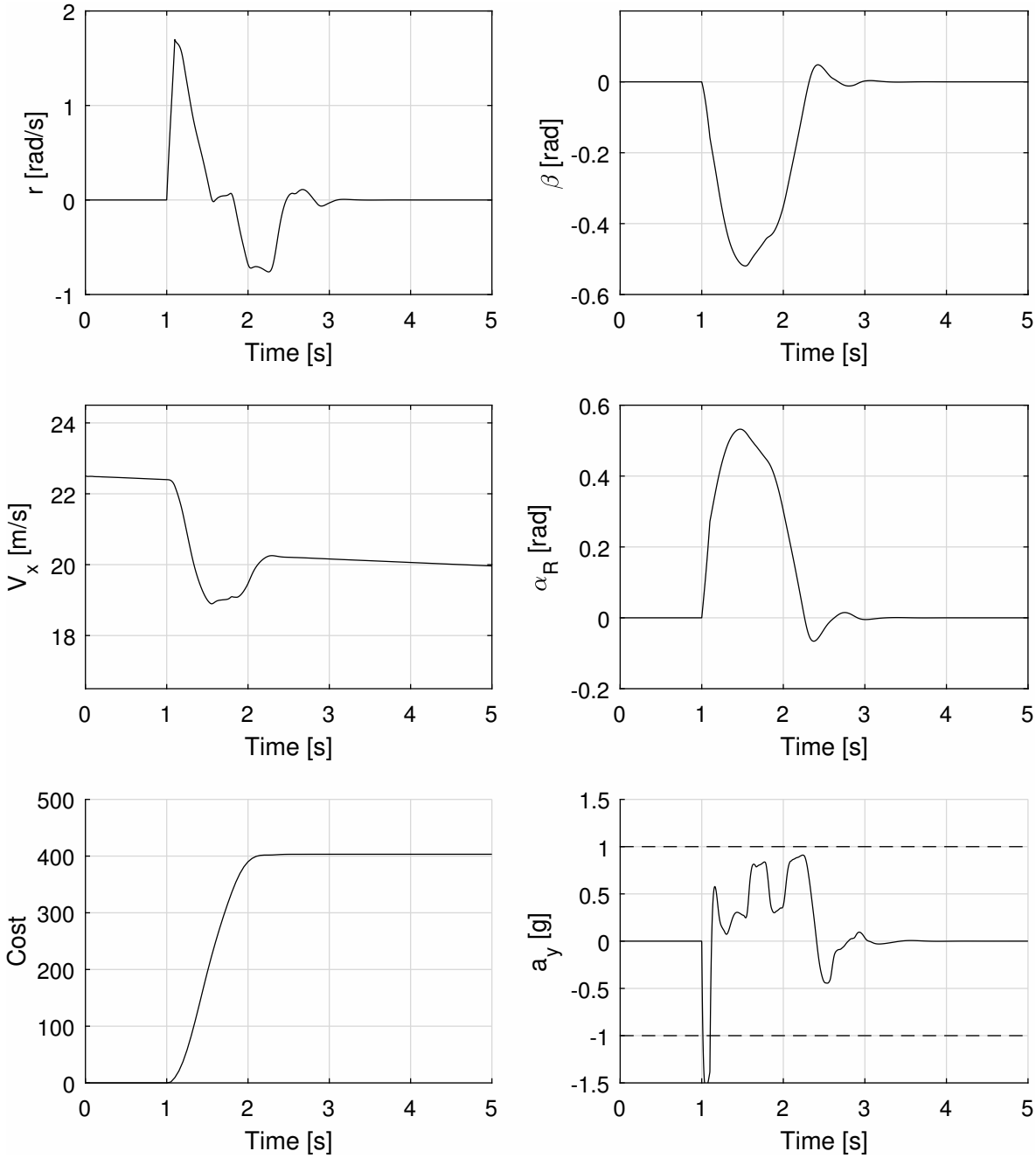

Figure D-5: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 22.5 m/s or 80 km/h.

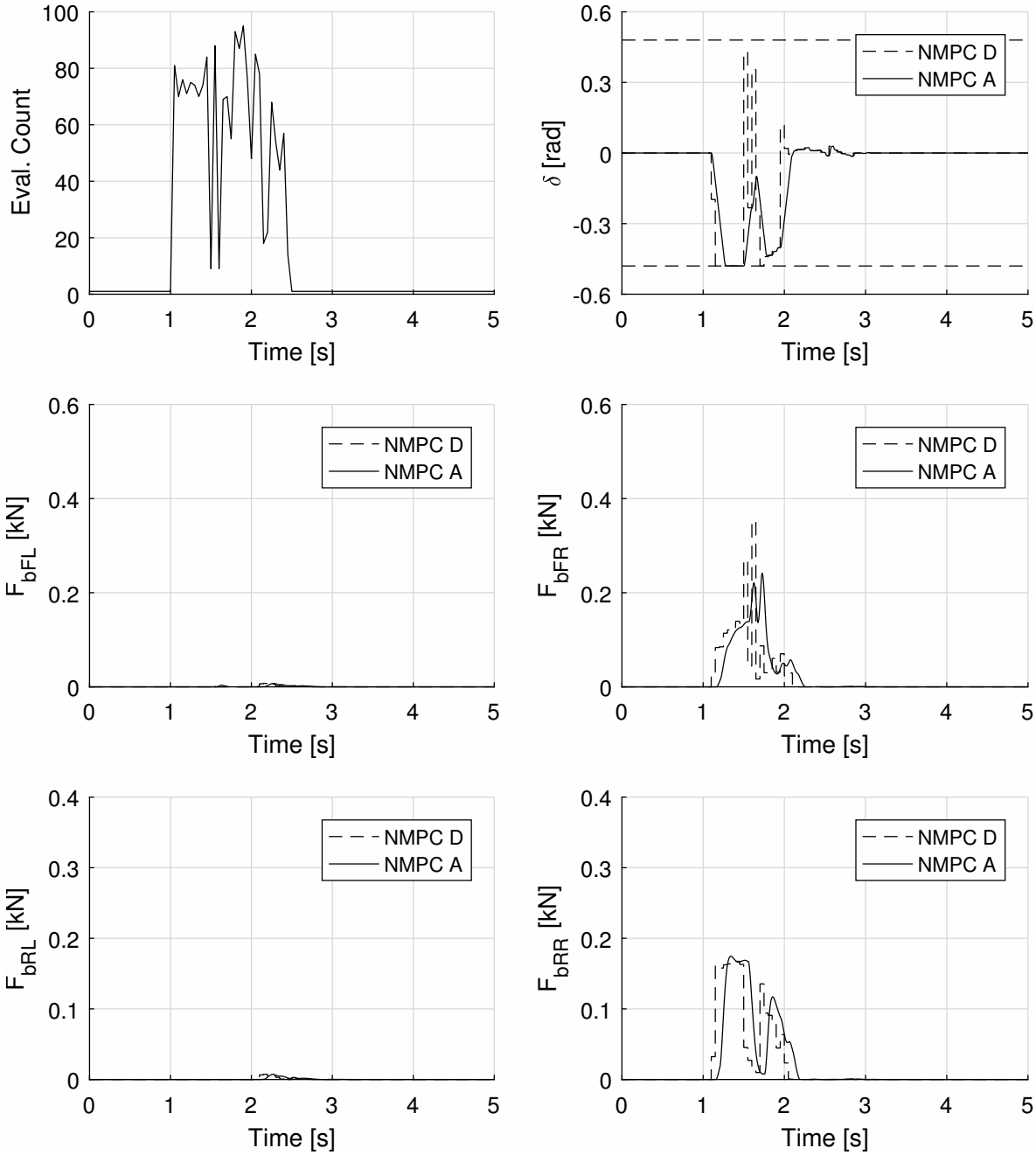

Figure D-6: Control inputs by NMPC during lateral collision recovery, with an initial speed of 22.5 m/s or 80 km/h. Dashed lines for control demand and solid lines for actual response of actuators. The total count of evaluation of the object function in the NMPC per time step is also presented.

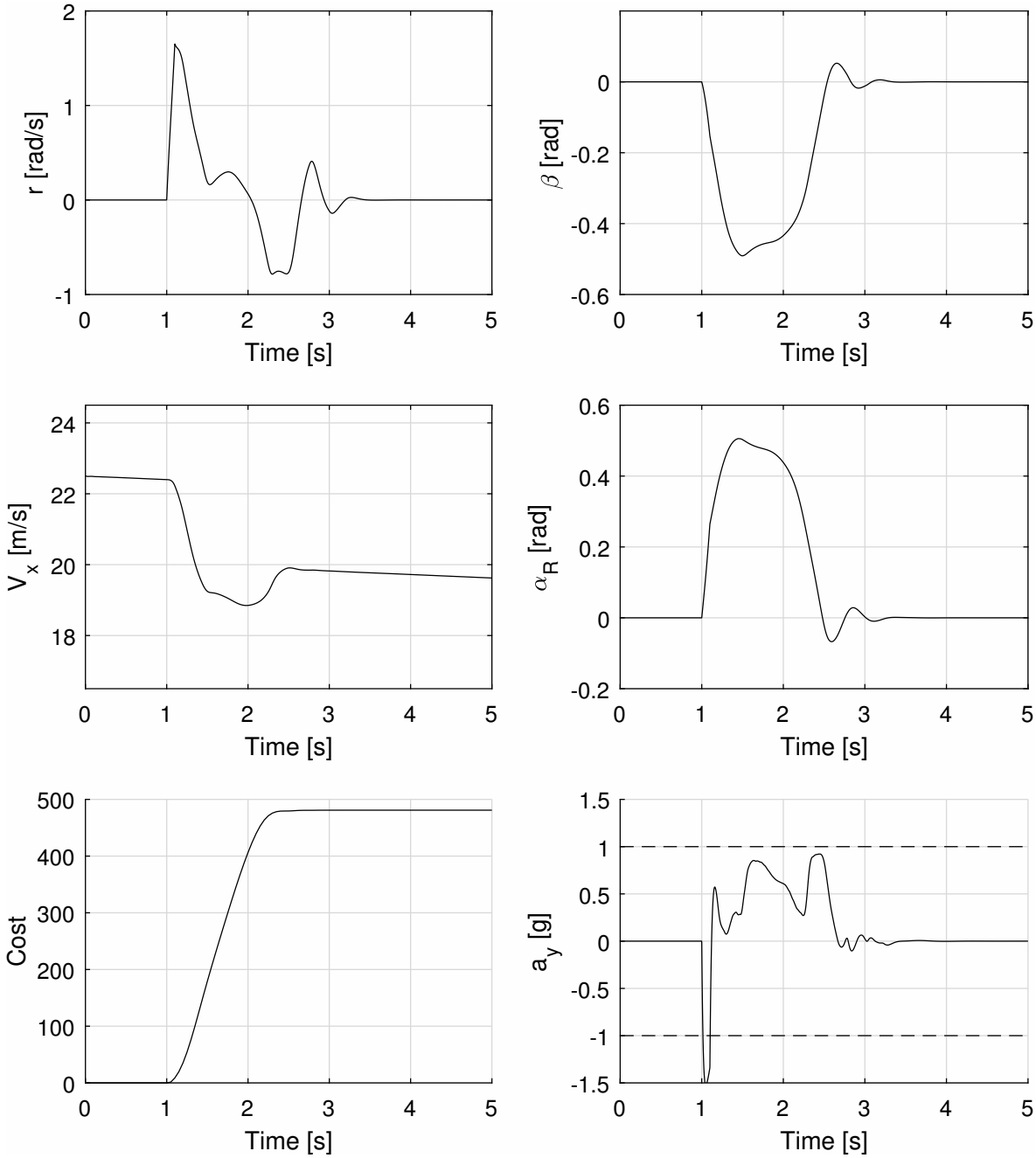

Figure D-7: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 22.5 m/s or 80 km/h.

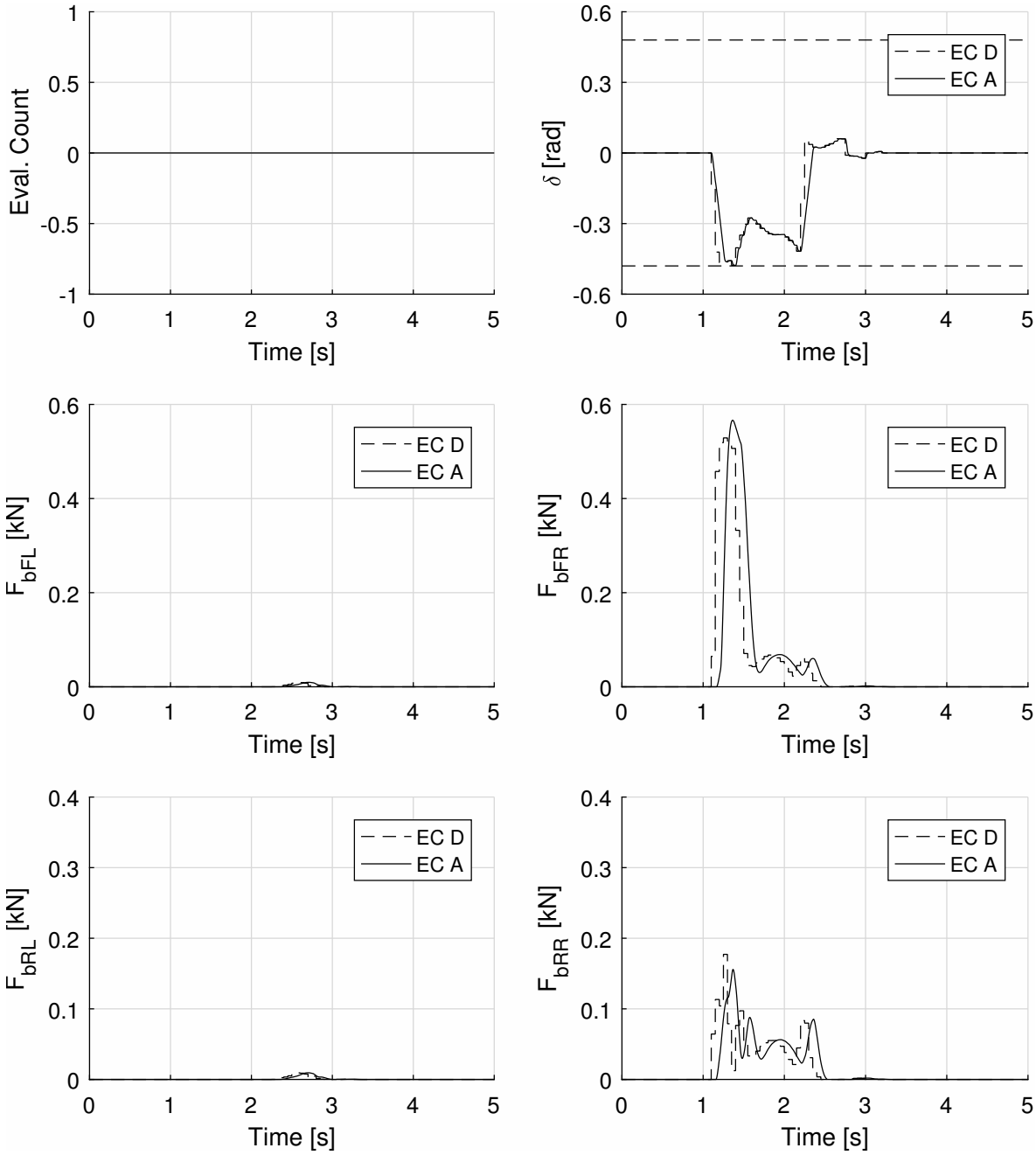

Figure D-8: Control inputs by explicit control during lateral collision recovery, with an initial speed of 22.5 m/s or 80 km/h. Dashed lines for control demand and solid lines for actual response of actuators.

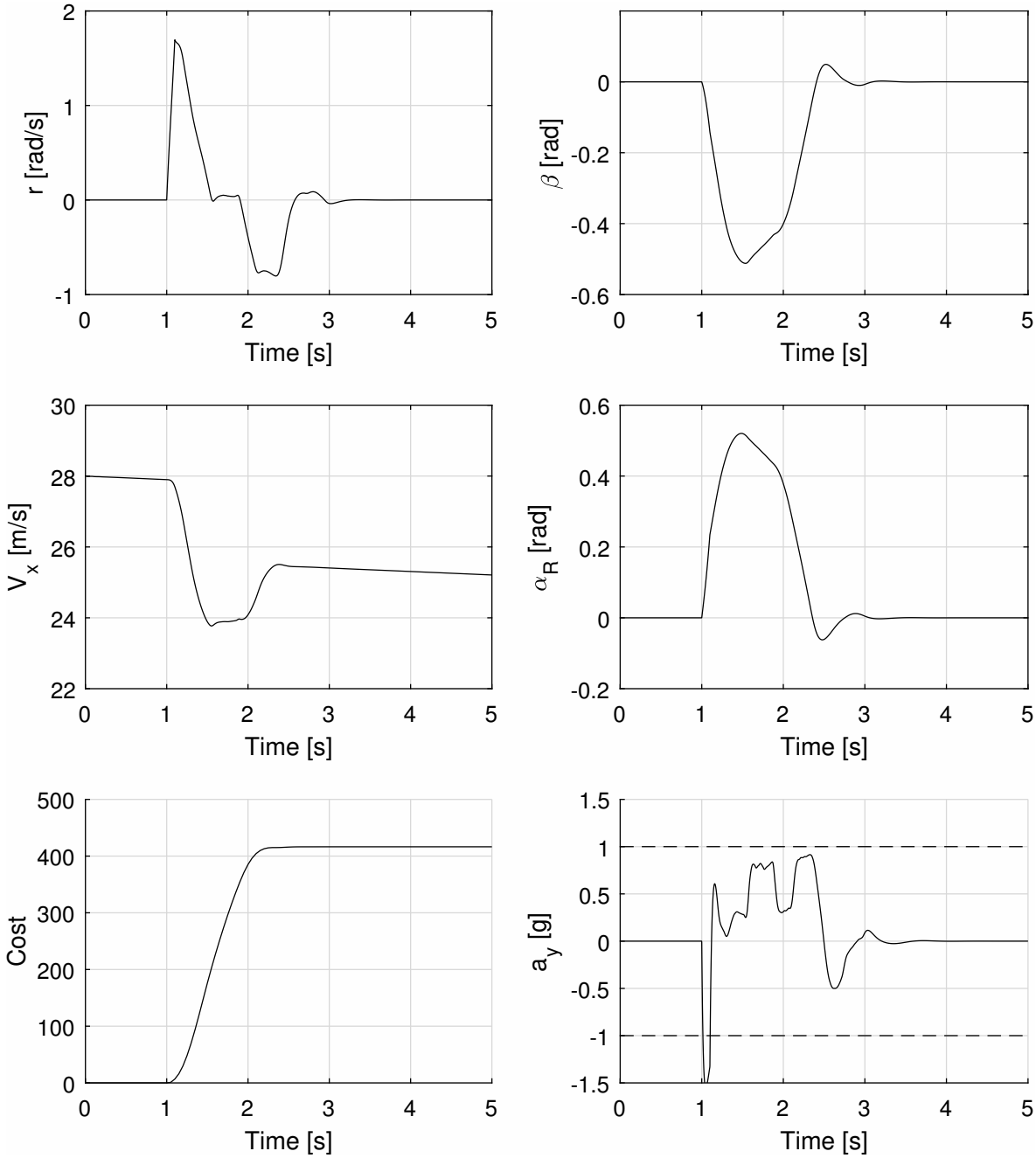

Figure D-9: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 28 m/s or 100 km/h.

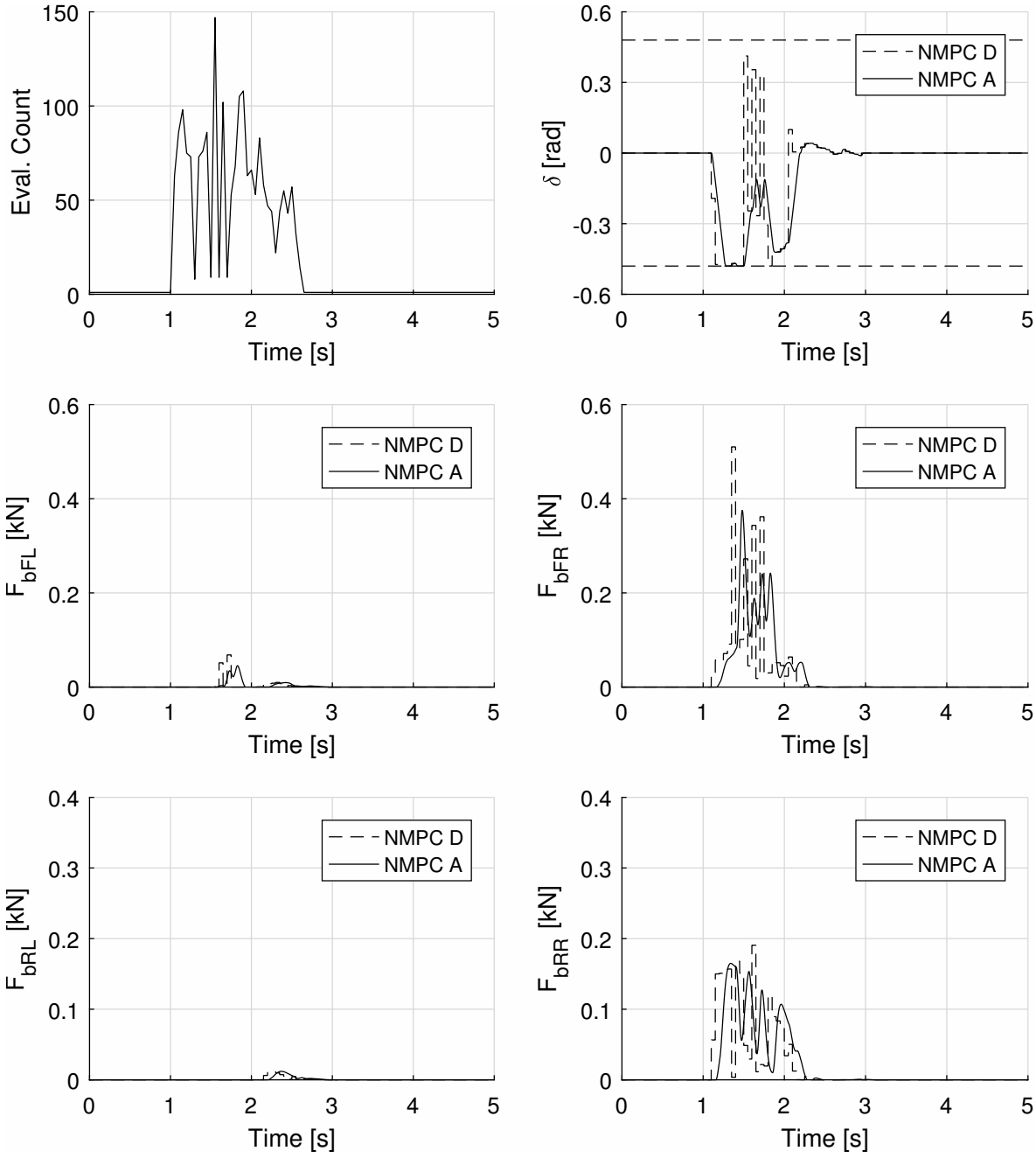

Figure D-10: Control inputs by NMPC during lateral collision recovery, with an initial speed of 28 m/s or 100 km/h. Dashed lines for control demand and solid lines for actual response of actuators. The total count of evaluation of the object function in the NMPC per time step is also presented.

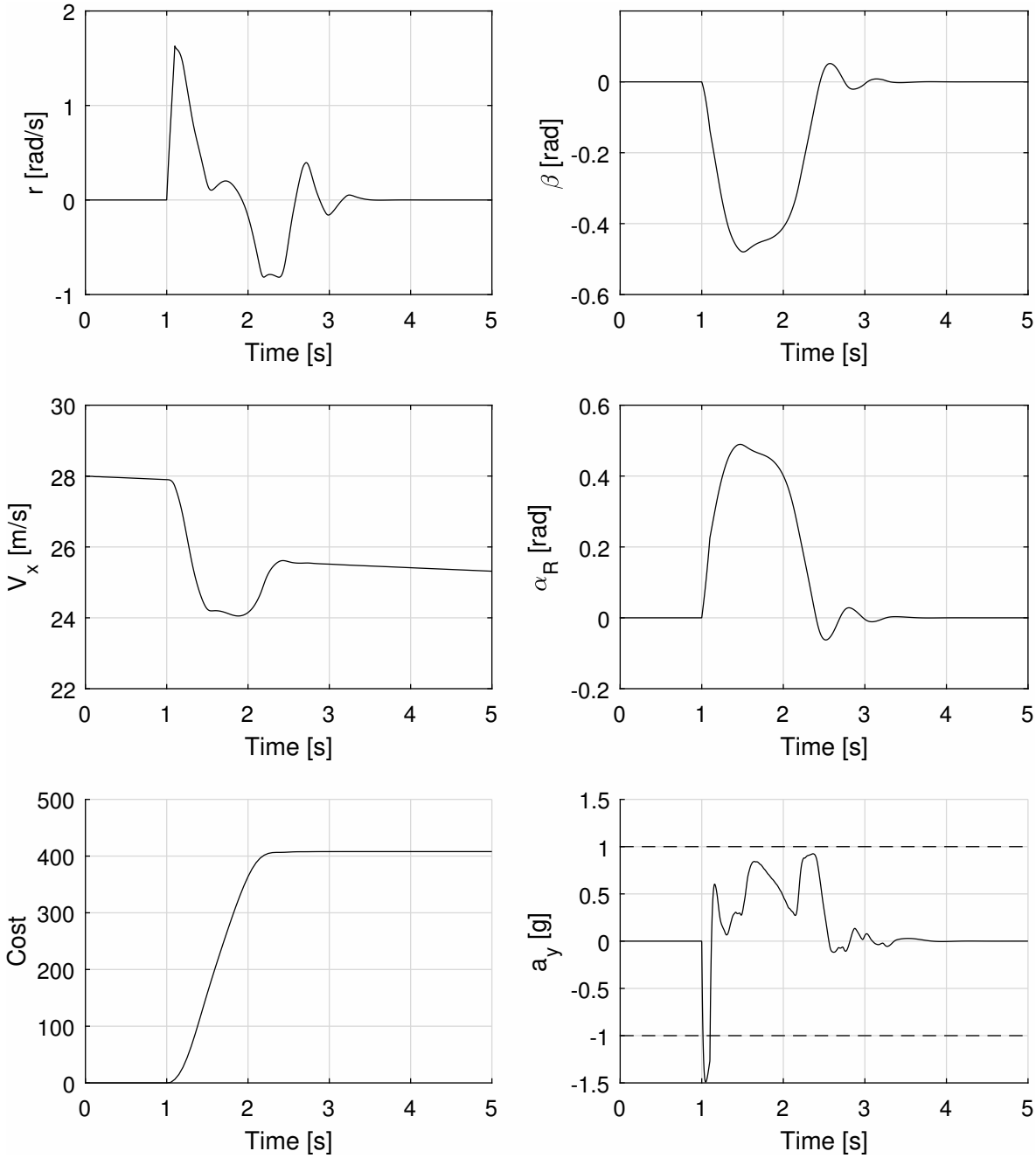

Figure D-11: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 28 m/s or 100 km/h.

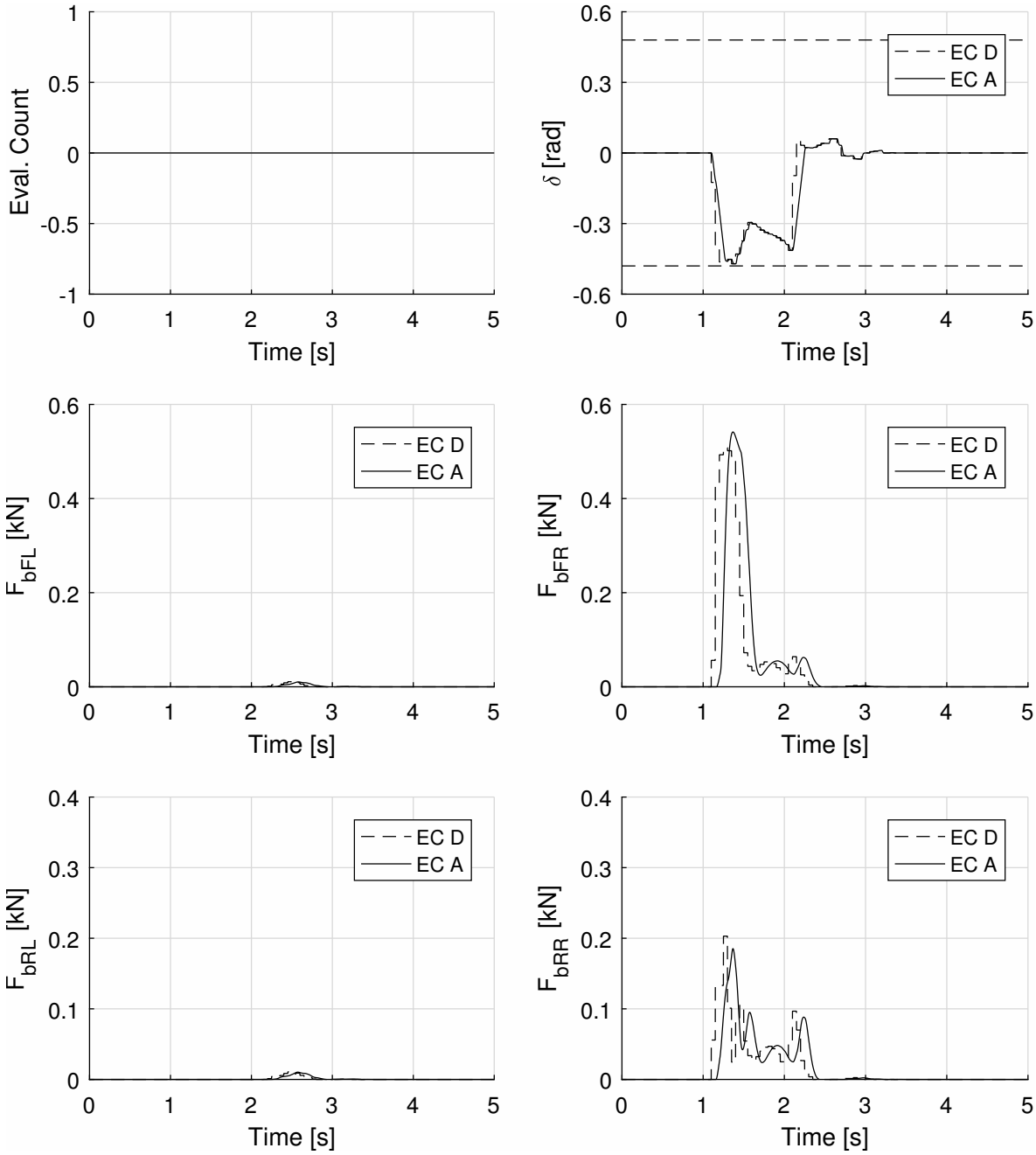

Figure D-12: Control inputs by explicit control during lateral collision recovery, with an initial speed of 28 m/s or 100 km/h. Dashed lines for control demand and solid lines for actual response of actuators.

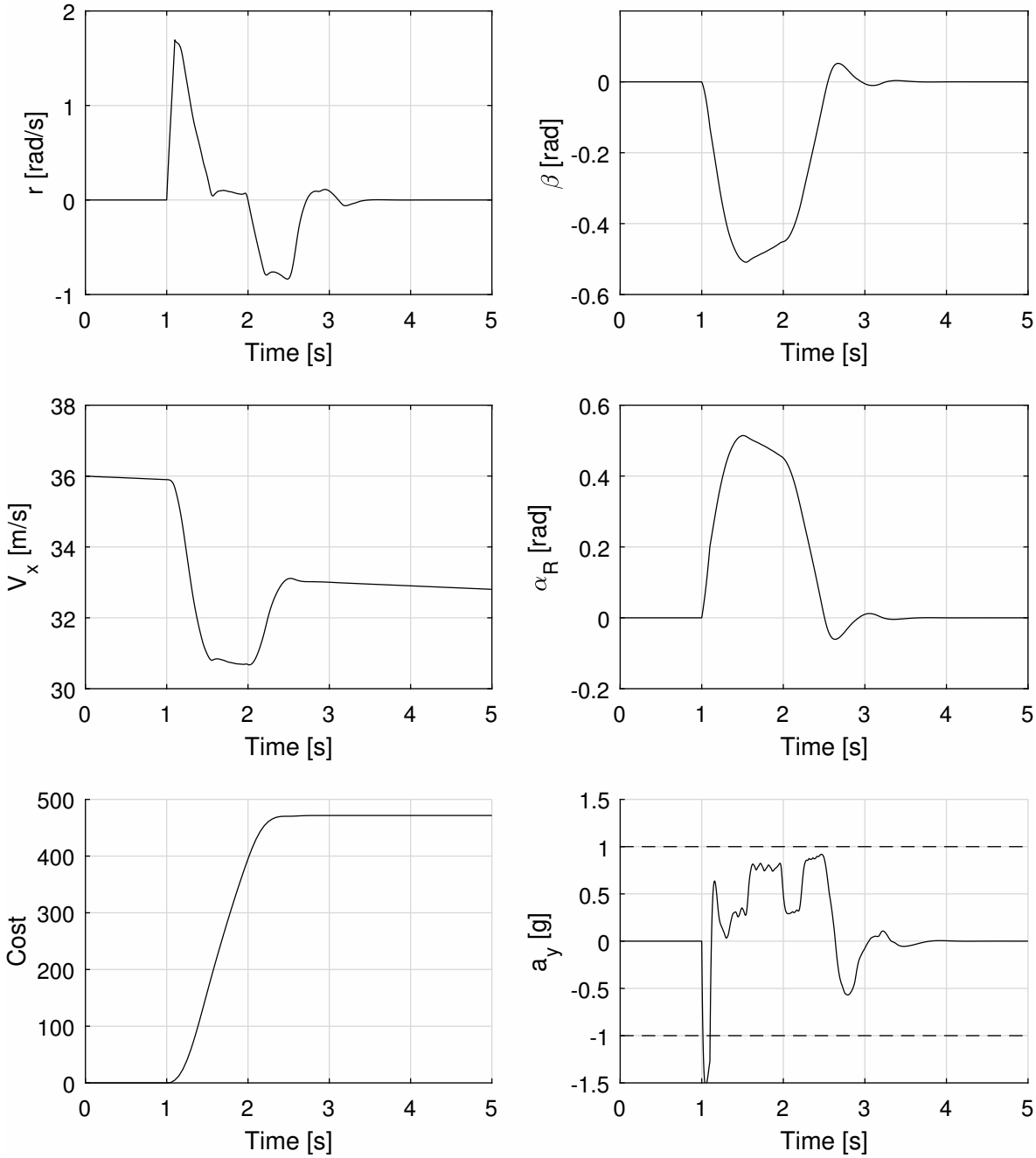

Figure D-13: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 36 m/s or 130 km/h.

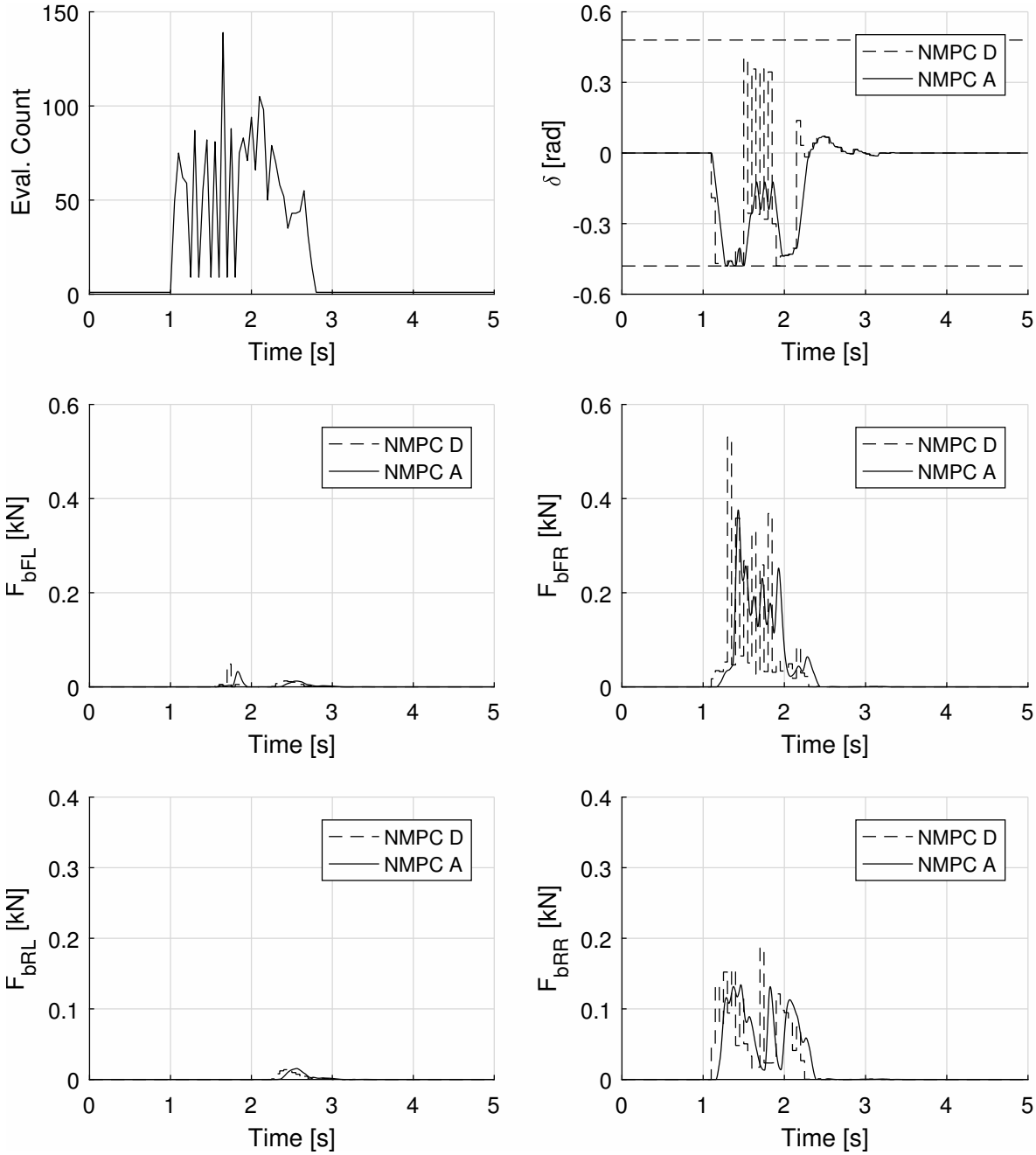

Figure D-14: Control inputs by NMPC during lateral collision recovery, with an initial speed of 36 m/s or 130 km/h. Dashed lines for control demand and solid lines for actual response of actuators. The total count of evaluation of the object function in the NMPC per time step is also presented.

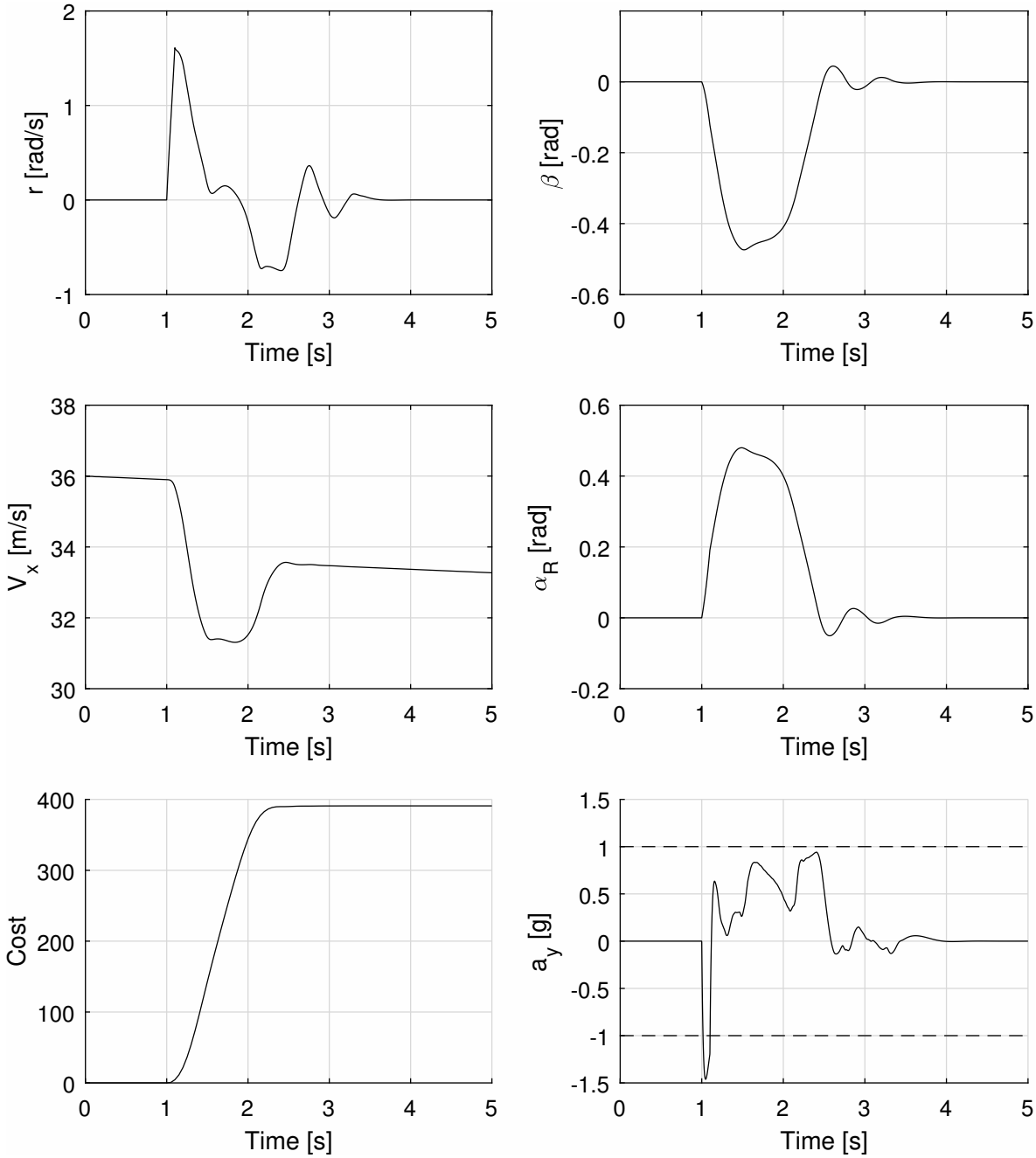

Figure D-15: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 36 m/s or 130 km/h.

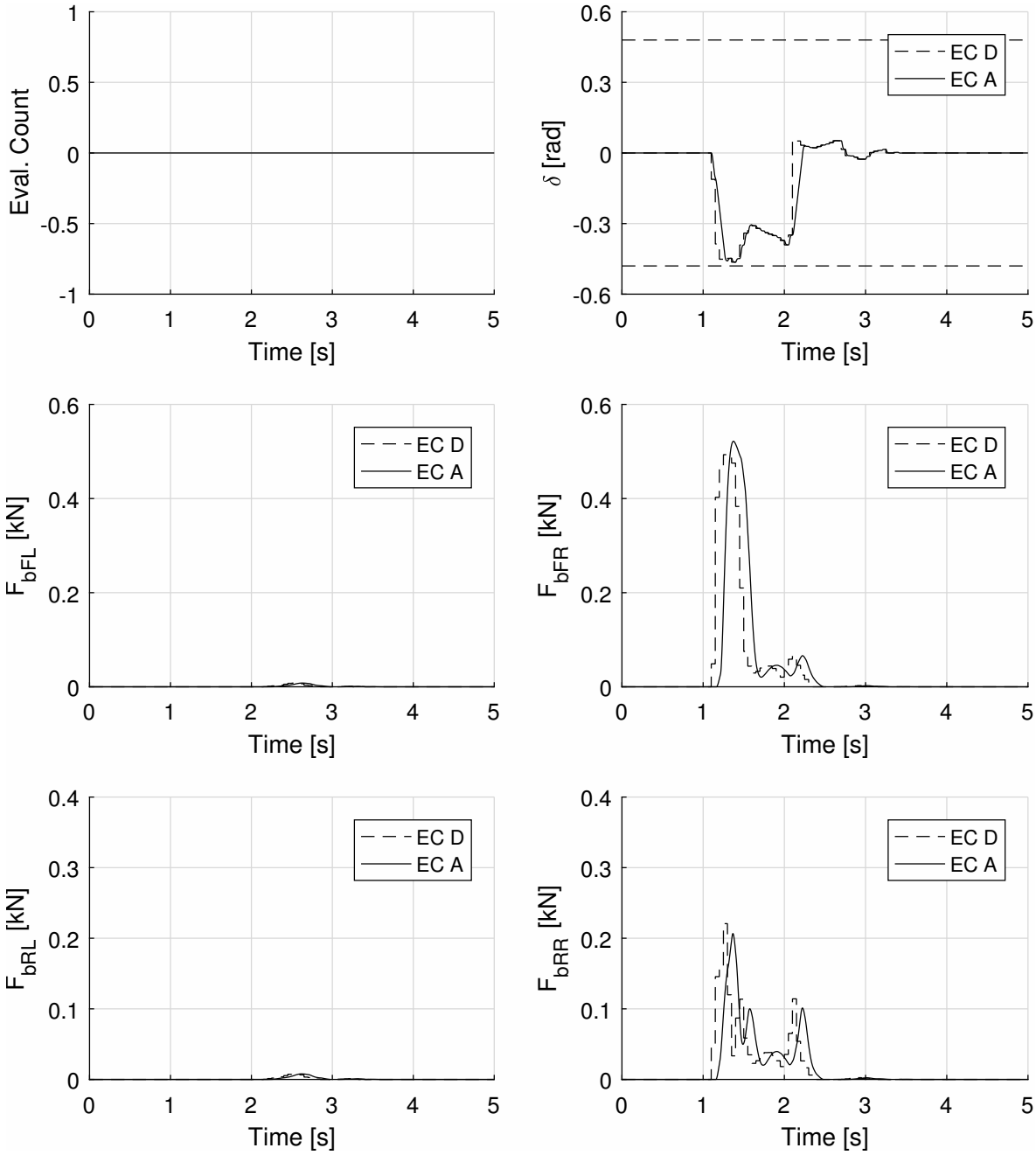

Figure D-16: Control inputs by explicit control during lateral collision recovery, with an initial speed of 36 m/s or 130 km/h. Dashed lines for control demand and solid lines for actual response of actuators.

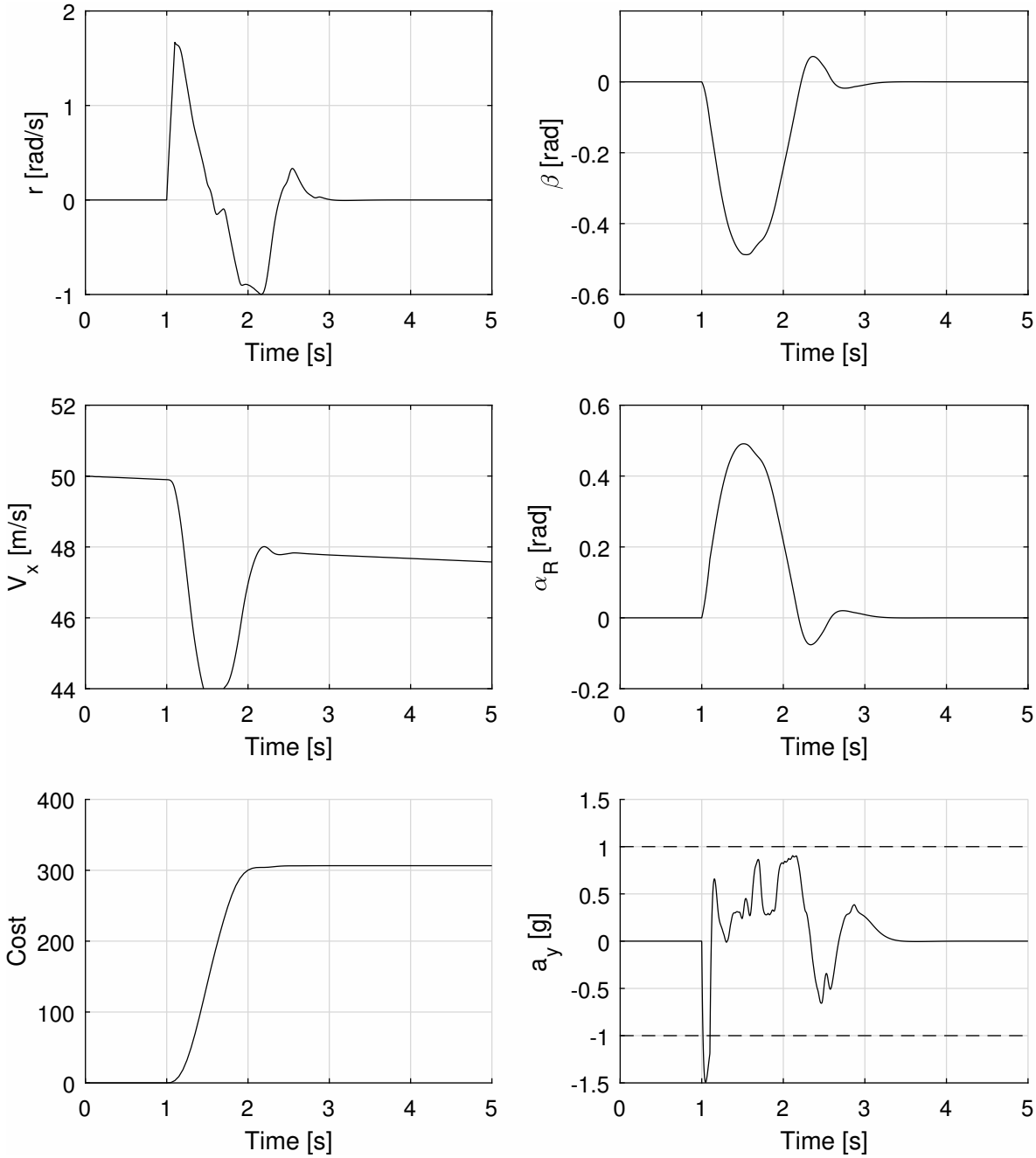

Figure D-17: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 50 m/s or 180 km/h.

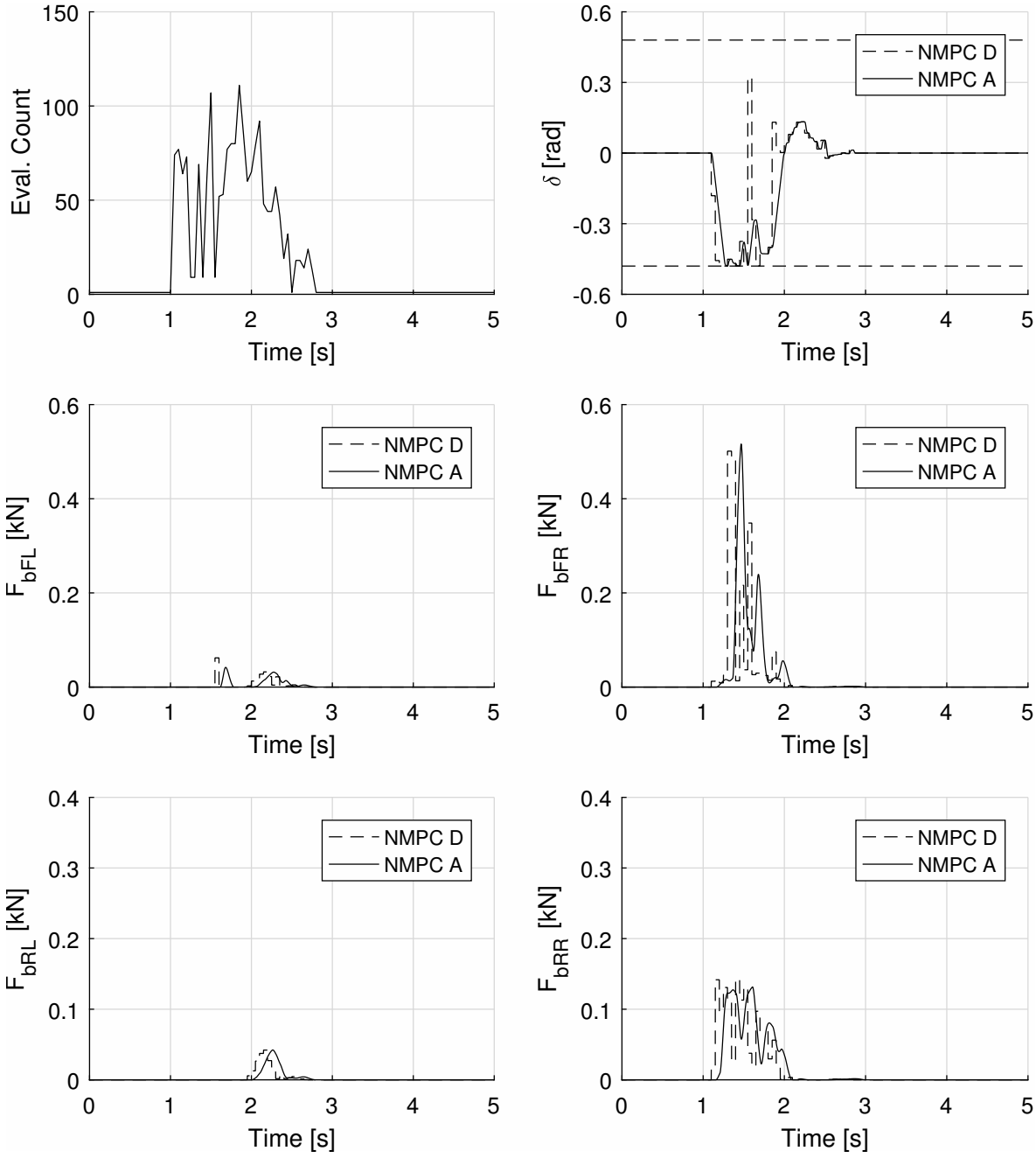

Figure D-18: Control inputs by NMPC during lateral collision recovery, with an initial speed of 50 m/s or 180 km/h. Dashed lines for control demand and solid lines for actual response of actuators. The total count of evaluation of the object function in the NMPC per time step is also presented.

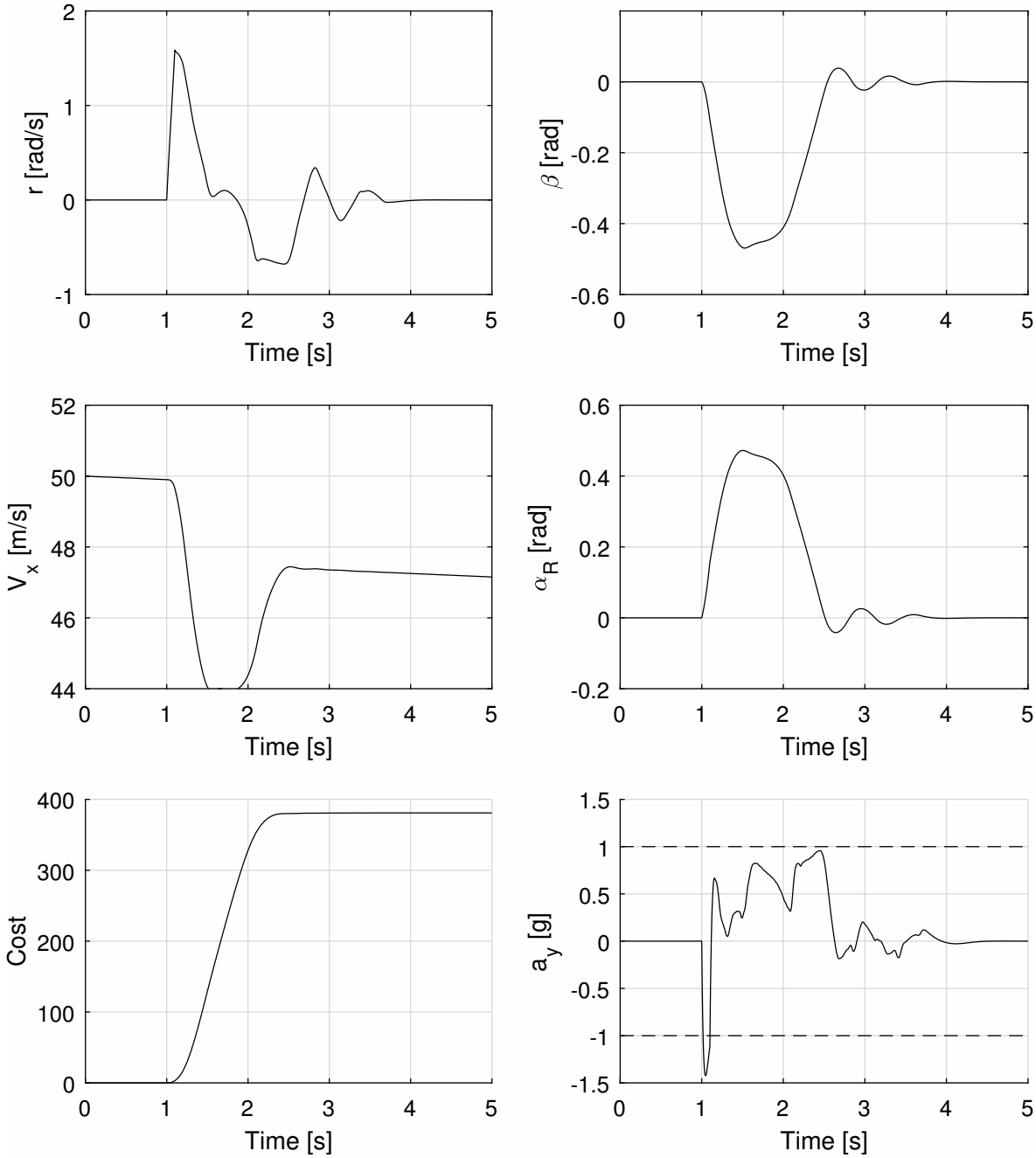

Figure D-19: Vehicle states, side-slip angle of the rear tires and the accumulated cost during the post-collision recovery, with an initial speed of 50 m/s or 180 km/h.

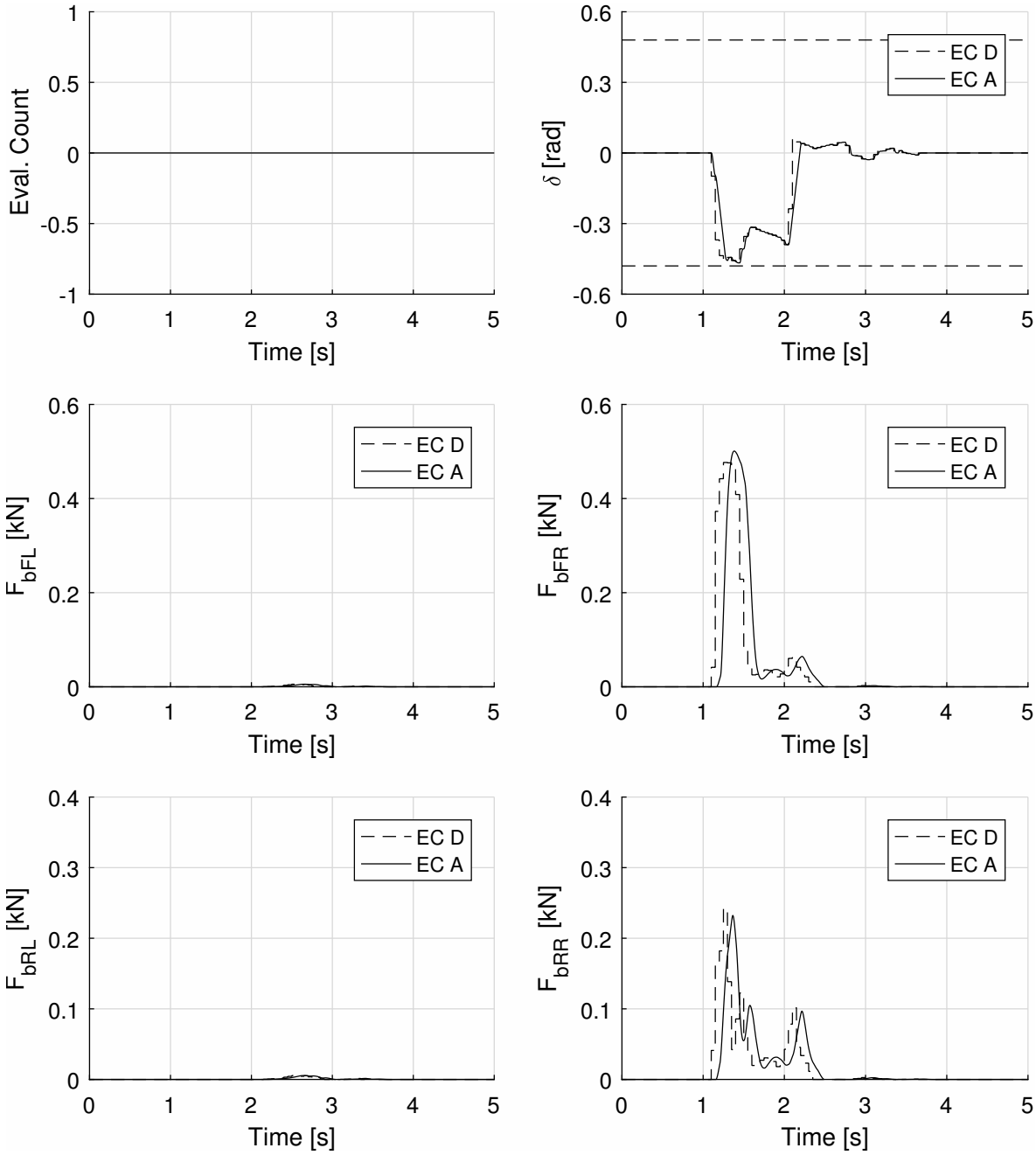

Figure D-20: Control inputs by explicit control during lateral collision recovery, with an initial speed of 50 m/s or 180 km/h. Dashed lines for control demand and solid lines for actual response of actuators.

Bibliography

[1] D. A. GmbH, "Dekra road safety report 2017," 2017.

[2] ITF, "Road safety annual report 2017," 2017.

[3] E. Commission, "Annual accident report 2016," 2016.

[4] P. A. Pisano, L. C. Goodwin, and M. A. Rossetti, "U.s. highway crashes in adverse road weather conditions," in *24th Conference on International Interactive Information and Processing Systems for Meteorology*, 2008.

[5] B. C. Tefft, "Motor vehicle crashes, injuries, and deaths in relation to weather conditions, united states, 2010-2014," 2016.

[6] J. N. Dang, "Statistical analysis of the effectiveness of electronic stability control (esc) system," 2007.

[7] A. T. Van Zanten, "Bosch esp systems: 5 years of experience," tech. rep., SAE Technical Paper, 2000.

[8] R. Lozano, M. Naghavi, K. Foreman, S. Lim, K. Shibuya, V. Aboyans, J. Abraham, T. Adair, R. Aggarwal, S. Y. Ahn, *et al.*, "Global and regional mortality from 235 causes of death for 20 age groups in 1990 and 2010: a systematic analysis for the global burden of disease study 2010," *The lancet*, vol. 380, no. 9859, pp. 2095–2128, 2012.

[9] NHTSA, "Traffic safety facts 2015," 2017.

[10] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat, "Mpc-based approach to active steering for autonomous vehicle systems," *International Journal of Vehicle Autonomous Systems*, vol. 3, no. 2-4, pp. 265–291, 2005.

[11] P. Falcone, H. Eric Tseng, F. Borrelli, J. Asgari, and D. Hrovat, "Mpc-based yaw and lateral stabilisation via active front steering and braking," *Vehicle System Dynamics*, vol. 46, no. S1, pp. 611–628, 2008.

[12] E. Siampis, E. Velenis, S. Gariuolo, and S. Longo, "A real-time nonlinear model predictive control strategy for stabilization of an electric vehicle at the limits of handling," *IEEE Transactions on Control Systems Technology*, 2017.

[13] C. E. Beal and J. C. Gerdes, "Model predictive control for vehicle stabilization at the limits of handling," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1258–1269, 2013.

[14] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat, "Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation," *International journal of robust and nonlinear control*, vol. 18, no. 8, pp. 862–875, 2008.

[15] D. Bernardini, S. Di Cairanoz, A. Bemporad, and H. Tsengz, "Drive-by-wire vehicle stabilization and yaw regulation: A hybrid model predictive control design," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 7621–7626, IEEE, 2009.

[16] S. Di Cairano, H. E. Tseng, D. Bernardini, and A. Bemporad, "Vehicle yaw stability control by coordinated active front steering and differential braking in the tire sideslip angles domain," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1236–1248, 2013.

[17] P. Tondel and T. Johansen, "Control allocation for yaw stabilization in automotive vehicles using multiparametric nonlinear programming," in *American Control Conference, 2005. Proceedings of the 2005*, pp. 453–458, IEEE, 2005.

[18] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[19] M. Gertz, J. Nocedal, and A. Sartenar, "A starting point strategy for nonlinear interior methods," *Applied mathematics letters*, vol. 17, no. 8, pp. 945–952, 2004.

[20] M. N. Zeilinger, C. N. Jones, and M. Morari, "Real-time suboptimal model predictive control using a combination of explicit mpc and online optimization," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1524–1534, 2011.

[21] D. Yang, T. J. Gordon, B. Jacobson, and M. Jonasson, "Quasi-linear optimal path controller applied to post impact vehicle dynamics," *IEEE transactions on intelligent transportation systems*, vol. 13, no. 4, pp. 1586–1598, 2012.

[22] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surveys in Operations Research and Management Science*, vol. 17, no. 2, pp. 97–106, 2012.

[23] D. D. MacInnis, W. E. Cliff, and K. W. Ising, "A comparison of moment of inertia estimation techniques for vehicle dynamics simulation," tech. rep., SAE Technical Paper, 1997.

[24] J. Reimpell, H. Stoll, and J. Betzler, *The automotive chassis: engineering principles*. Butterworth-Heinemann, 2001.

[25] V. Pratt, "Direct least-squares fitting of algebraic surfaces," in *ACM SIGGRAPH computer graphics*, vol. 21, pp. 145–152, ACM, 1987.

[26] B. Heißing and M. Ersoy, *Chassis handbook: fundamentals, driving dynamics, components, mechatronics, perspectives*. Springer Science & Business Media, 2010.

[27] "Road vehicles – lateral transient response test methods – open-loop test methods," 2011.