CRANFIELD UNIVERSITY

KONSTANTINOS ZARKADIS

MODEL PREDICTIVE TORQUE VECTORING CONTROL WITH ACTIVE TRAIL-BRAKING FOR ELECTRIC VEHICLES

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Transport Systems

MSc by Research
Academic Year: 2016–2017

Supervisor: Dr E. Velenis and Dr S. Longo
February 2018

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Transport Systems

MSc by Research

Academic Year: 2016–2017

KONSTANTINOS ZARKADIS

Model Predictive Torque Vectoring Control with Active Trail-Braking for Electric Vehicles

Supervisor: Dr E. Velenis and Dr S. Longo
February 2018

This thesis is submitted in partial fulfilment of the requirements for the degree of MSc by Research.

Abstract

In this work we present the development of a torque vectoring controller for electric vehicles. The proposed controller distributes drive/brake torque between the four wheels to achieve the desired handling response and, in addition, intervenes in the longitudinal dynamics in cases where the turning radius demand is infeasible at the speed at which the vehicle is traveling. The proposed controller is designed in both the Linear and Nonlinear Model Predictive Control framework, which have shown great promise for real time implementation the last decades. Hence, we compare both controllers and observe their ability to behave under critical nonlinearities of the vehicle dynamics in limit handling conditions and constraints from the actuators and tyre-road interaction. We implement the controllers in a realistic, high fidelity simulation environment to demonstrate their performance using CarMaker and Simulink.

Keywords

torque vectoring; MPC; nonlinear predictive control; FEV.

Contents

List of Figures

List of Tables

List of Abbreviations

4WS	Four Wheel Steering
ABS	Anti-lock Braking System
AFS	Active Front Steering
ATTS	Active Torque Transfer System
AWD	All-Wheel Drive
AYC	Active Yaw Control
CC	Cruise Control
CM	Centre of Mass
DoF	Degrees of Freedom
DYC	Direct Yaw Control
EM	Electric Motor
EoM	Equations of Motion
ESP	Electronic Stability Program
EV	Electric Vehicle
FWD	Front-Wheel Drive
HEV	Hybrid Electric Vehicle
ICE	Internal Combustion Engine
LTV	Linear Time Varying
LQG	Linear Quadratic Gaussian

LQR	Linear Quadratic Regulation
MF	Magic Formula
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
PDIP	Primal-Dual Interior Point
PID	Proportional Integral Derivative
QP	Quadratic Program
RHC	Receding Horizon Control
RWD	Rear-Wheel Drive
SATM	School of Aerospace, Transport and Manufacturing
SH-AWD	Super Handling All-Wheel Drive
SMC	Sliding Mode Control
SUV	Sport Utility Vehicle
TCS	Traction Control System
UKF	Unscented Kalman Filter

Acknowledgements

I would first like to thank my supervisors Drs. Efstathios Velenis and Stefano Longo for introducing me into the vehicle dynamics and MPC world respectively as well as for all the guidance through this research work. I would also like to thank Drs. Siampis, Sofian who were always by my side supporting my research with their valuable knowledge and experience.

Finally, I express my thanks to my family and my girlfriend "K" for supporting and encouraging me through the process of my research and writing this thesis. This accomplishment would not have been possible without them.

A special gratitude to the Autolads. Thank you all.

Chapter 1

Introduction

1.1 Motivation

Chassis control systems have been in the centre of research many decades, controlling the longitudinal, lateral and vertical motion of the vehicle in order to improve handling and acceleration\braking behavior [21]. This subject area growing fast nowadays due to the increased safety concerns rising from the increasing number of automobiles on the road in combination with higher performance. Additionally, the rapid development in the microprocessor computing field is offering faster and cheaper platform solutions for chassis control deployment.

In 1978, the introduction of the Anti-lock Braking System (ABS) was one major breakthrough and now is a standard basic feature of every vehicle. A few years later another chassis control system was introduced, the Traction Control System (TCS), ex-

panding the ABS by including slip control during acceleration [34]. Later on, more control systems came to light such as Four Wheel Steering (4WS), active suspension and braking systems like Electronic Stability Program (ESP) [58].

Losing control of a car in a corner is dangerous. Ideally, a car should be able to negotiate the corner under control with neither excessive oversteer or understeer. However, taking a corner too fast, performing an emergency maneuver, bad weather, defective road surfaces or poor maintenance of the car may result in a loss of grip leading to a loss of control. Systems that control the lateral dynamics, in such scenarios, mainly focus on improving the steerability of the vehicle and preventing the driver from losing control in limit handling conditions. On the other hand, longitudinal vehicle control is commonly regulated under the command of the driver with systems such as Cruise Control (CC) in which, recently, safety functions for vehicle speed regulation have been integrated, to keep a safe distance from the front vehicle. However, it is recognised that active control of longitudinal dynamics can improve the vehicle's stability in terminal understeer situations.

Understeer, oversteer and neutral steer are terms used to describe the vehicle's response to steering inputs. Due to the complexity in the relation between the steering angle on the wheels and the response of the vehicle, the concept of the understeer gradient K_{und} has been introduced. Assuming a single-track model under steady state cornering with all tyres at their linear region operation, the understeer gradient gives an indication of the natural behavior of the vehicle under a constant steering input from [49]

$$\delta = \frac{L}{R} + K_{und} a_y, \tag{1.1}$$

where δ is the front wheels' steering angle, L is the vehicle's wheelbase, R is the vehicle's path radius and $a_y = \frac{V_x^2}{R}$ is the lateral acceleration at the vehicle's Centre of Mass (CoM).

The three cases are:

- $K_{und} = 0$, neutralsteer, no need to vary the steering angle ,
- $K_{und} > 0$, understeer, the steering angle has to be increased according to the second term of (1.1) in order to keep a constant radius path, with a characteristic speed V_{char} being the speed at which the steering angle is double the Ackerman angle $\delta_{acker} = \frac{L}{R}$ [23] as shown in Fig1.1
- $K_{und} < 0$, oversteer, the steering angle has to be decreased while the speed increases until the vehicle reaches the critical velocity V_{crit} and the steering angle is zero

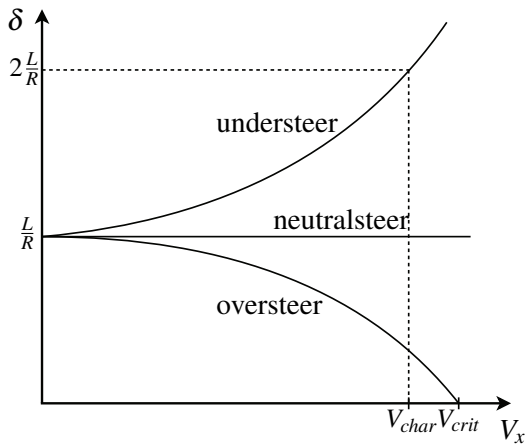

Figure 1.1: Understeer gradient characteristic

Despite the fact that the understeer gradient can quantify the natural tendency of the vehicle to follow a path radius or not, it is important to note that the vehicle's behavior can change while cornering due to the its drivetrain topology and use of acc\brake pedal. This derives from the longitudinal and lateral tyre force coupling effect in which the lateral tyre force reduces when the longitudinal tyre force increases.

Understeer usually occurs when the front wheels reach the tyre's cornering stiffness limit and lose grip earlier than the back wheels, resulting in the car continuing straight

instead of turning. Most Front Wheel Drive (FWD) cars tend to understeer when accelerating out of a bend, mostly because the front tyres do the job of acceleration-deceleration and steering. Approaching a corner faster than what the tyres can support, causes the front tyres to struggle to keep the car in line, and try to steer the car in a direction you're pointing it to.

Oversteer is the opposite of the understeer where the rear tyres lose grip while the front wheels remain bellow the limit of adhesion. Rear Wheel Drive (RWD) cars, on the other hand are less prone to understeer, because the front wheels do the steering and the rear ones the driving. Accelerating hard out of a bend in a RWD car could cause it to oversteer due to the rear wheels running out of grip from the power being delivered and the turning of the car.

Therefore, returning to the limit handling scenarios, terminal understeer refers to a vehicle in which the front tyres potential lateral force is at a maximum due to excessive vehicle speed while cornering. In [68] the importance of a velocity regulation is mentioned as a performance requirement for the development of ESP system by Bosch.

Electric Vehicles (EV) have gained a lot of popularity the last years not only for their important role as environmentally friendly transportation but also for their increased performance in traction and stability systems. Having the ability of different propulsion system configurations, such as independent motors for each wheel, electric vehicles allow us to implement more efficient safety algorithms [7, 64].

One of the most trending algorithms is Torque Vectoring (TV) which controls the wheel torque distribution respecting as closely as possible the drivers steering wheel and throttle/brake commands to improve the passengers safety. To be more precise, TV systems aim at controlling the lateral dynamics of the vehicle by tracking the yaw rate and occasionally the side slip angle reference signals while at the same time following closely the torque demand input given by the driver using the pedals. As shown in Fig.1.2 the

vehicle consists of four independent electric motors, each one driving a wheel. Since the driver's intention is to turn left, as shown from the yaw moment M_z, the torque vectoring control algorithm sets the appropriate torque values on each wheel. Therefore the wheel torques on the outer side of the turning vehicle are positive and on the inner side are negative. Torque Vectoring is used in sublimit cases to deliver customisable handling behavior and can act as stability control as the tyres reach their adhesion limit.

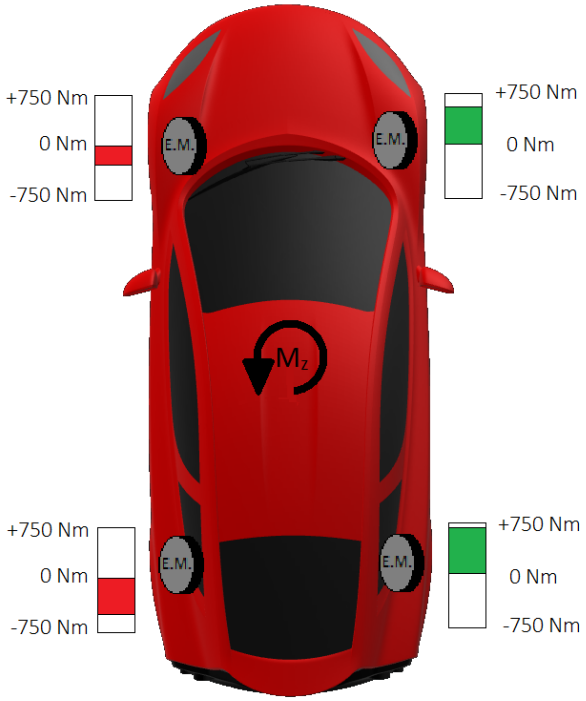

Figure 1.2: Torque Vectoring technique

1.2 Literature Review

The literature of the most important vehicle chassis control solutions in both the automotive industry and academia, is being split into lateral and longitudinal dynamics control systems.

1.2.1 Active Chassis Control

In AVEC '92 a number of papers were presented for the first time refering on the use of left-right tyre force distribution to control the vehicle's lateral dynamics [21]. One of the studies which made a debut was the β-method emphasising the role of side-slip angle under acceleration\braking while cornering [59]. The results not only showed that yaw moment gain decreases while sideslip angles increase thus influencing the vehicle's maneuverability, but also that the yaw moment "shifting" to high values can change a vehicle's behavior from neutralsteer to understeer during acceleration and the opposite during deceleration. A final note in [59] indicates that yaw moment gain under steady-state cornering can be expressed as a function of both longitudinal and lateral acceleration and through the use of a hypothetical external yaw moment the influence of that acceleration and deceleration can be eliminated. This method is called Direct Yaw Control (DYC) and was applied on an All Wheel Drive (AWD) vehicle where the external yaw moment was expressed as a distribution of the traction and braking forces on the rear wheels, while keeping the front-rear distribution constant. The results showed the effectiveness of the method and popularised the brake stability systems by the late 90s.

The most popular types of DYC application so far are [43]:

- lateral braking control, which uses independent braking between the left and right side of the vehicle to generate a yaw moment

- torque distribution control, which splits the engine's torque between left and right wheels resulting in a driving torque difference between them hence a yaw moment generation

- torque vectoring control, which transfers torque from left side wheels to right side ones or vice versa, in order to create a braking torque on one wheel while at the

same time transferring an equal amount of driving torque to the other side wheel.

Using braking to control the lateral dynamics of the vehicle, hence ”lateral” braking control, is effective across a wide range of vehicle operating conditions, making it widely used in limit handling scenarios where stability is more important than comfort, but creates a negative feeling on the driver due to the deceleration of the vehicle [43]. The most successful system under this category is Bosch's ESP [40] with other car manufacturers following their example such as Ford [67] and BMW [38]. Other studies that include lateral dynamics braking control are [6], where a H_∞ controller uses Active Front Steering (AFS) and and differential braking to achieve the yaw rate and sideslip angle targets, [66], which uses a Sliding Mode Control (SMC) strategy for yaw rate and sideslip control while taking into account variations in the longitudinal dynamics, and [31] which uses throttle control and differential braking to manipulate the slipping condition of the rear tyres according to a yaw rate target on a RWD vehicle.

The last two DYC techniques mentioned above, quickly gained popularity against the lateral braking control due to their less intrusive character in sub-limit conditions. Systems with lateral torque distribution are mainly active differentials that regulate the direction of torque to the left and right wheels under both limit and sub-limit conditions but their main disadvantage is that they cannot generate a corrective yaw moment when the engine torque in not large enough, for example when the vehicle is decelerating [43], or when the engine torque is zero [53]. The most successful example is the Active Torque Transfer System (ATTS) from Honda [57] which shows improved stability and handling during a combination of steering and acceleration\deceleration inputs. In the case of torque vectoring control, the torque transmitted between the wheels does not conflict with the driver's acceleration and braking commands although it can have a negative effect on the vehicle's steering wheel action when it's applied on the front axle. The most popu-

lar example is the Active Yaw Control (AYC) system from Mitsubishi and its successor the Super AYC [74]. They use a mechanism which transfers and controls the rear wheel torques under different driving conditions, limiting the vehicle's yaw moment and enhance its cornering performance. AYC can also act like a limited slip differential by containing rear wheel slip to improve traction.

Torque distribution, nowadays, has also found its way into the AWD vehicles. In [46, 47], Piyabongkarn showed that front-rear torque distribution can change the understeer characteristiccs of the vehicle, although it is not as effective as left-right distribution. A representative example is a series of papers from Ricardo developing a centre differential for a Sports Utility Vehicle (SUV), where experiments in a BMW X5 showed mixed results [72] and led to a left-right differential change instead in [71]. Distribution of the torque to all four wheels gives better traction compared to a FWD\RWD solution and the cornering performance can be improved without interfering with the driver's throttle\brake inputs [55]. The most characteristic example is the Super Handling AWD (SH-AWD) system implemented by Honda which combines a set of electromagnetic clutshes to vary the front-rear distribution and an improved variant of the ATTS to vary the left-right distribution in a single unit at the rear axle. Experimental results showed less understeering behavior when the SH-AWD system was used but when the vehicle was off-throttle it was not possible to transfer torque between wheels [36].

1.2.2 Electric Motor as Control Actuator

The rapid development of both Electric Vehicles (EV) and Hybrid Electric Vehicles (HEV) has already presented exciting new possibilities on the vehicle dynamics area. Both EVs and HEVs have attracted attention not only as response to the increasing fuel prices and the growing environmental concerns but also because EMs deliver both tractive and brak-

ing torque. Depending on vehicle topology we can distribute torque between front/rear axles, left/right wheels of one axle, or all 4 wheels therefore eliminating the distinction between the different control strategies documented above (braking, torque distribution and torque vectoring) thanks to the use of the electric motor.

Most of the research has been focused on the energy management and powertrain technology challenges [11]. However, the electirc motor has some distinct advantages over the conventional drivelines [26, 28, 52]:

- extremely quick and accurate response and can be controlled according to speed or torque demand
- dual operation, can be used as a motor or a generator with almost equal efficiency
- high energy efficiency up to 90%
- in the case of in-wheel motors the powertrain architecture is simplified with less mechanical parts giving also way to new passenger cell designs

Despite the advantages mentioned above, there are a few risky questions left. While the fundamentals of vehicle dynamics do not need to be redefined, certain challenges come to light when the powertrain is changed from a conventional Internal Combastion Engine (ICE) to an electric one. The most important ones are the increased sprung mass and packaging constraints related to the necessary inclusion of the battery and the increased unsprung mass and suspension packaging in the case of in-wheel motors, both investigated from Crolla and Cao [12]. Extra load from the battery can impact roll stability, ride vibration and comfort while the increase of unsprung mass makes the vertical wheel motion more challenging. It is obvious that there are clear advantages and disadvantages using an electric motor as the main actuator and both academia and automotive

industry are actively looking at appropriate solutions along with the increased government interest over the years.

A meaningful amount of torque vectoring examples on EVs can be found in literature. In [51] a SMC strategy is used with the driver steering input modeled as a disturbance, and in [75] a Linear Quadratic Gaussian (LQG) controller, used to enhance steerability within a given yaw and sideslip control region or maneuverability outside it. In the case of AWD EVs one of the earliest examples is presented from Hori Laboratory in Tokyo University [26] where they implemented ABS and TCS on an EV and later extended to yaw rate tracking [22, 44]. Other drivetrain topologies and control methodologies found in literature include an integrated torque control of a rear electric motor and the electro-hydraulic brake system using a fuzzy logic controller in [33], an adaptive controller on a system with independent rear in-wheel motors and AFS [8] and a rather unique EV concept developed by the Technical University of Munich called MUTE, where apart from the main electric motor there is also a second one superimposed in the rear differential to obtain torque vectoring capabilities [25]. Another study controlling the lateral vehicle dynamics can be found in [45] where they use a Proportional-Integrated-Derivative (PID) controller to calculate the requests on the two rear axle electric motors for yaw rate and sideslip angle error minimisation from target values set by a bicycle model.

A characteristic example of torque vectoring control can be found from the 7FP EU project E-VECTOORC [3] where they employed a control allocation scheme for torque vectoring of a four electric motors pure EV. The main aim of the project was to create a fun-to-drive vehicle while at the same time improving energy efficiency using torque modulation for brake energy recuperation, ABS and TCS functionality. In [27] an appropriate cost function is presented for the control of the vehicle dynamics while in [13] Novellis et al. focus on the control allocation problem using an offline optimisation algorithm using a range of different cost functions based on performance and power usage

criteria. The authors conclude that slip-based cost functions are highly recommended for control allocation of the wheel torques in EV applications. Finally they extended their previous work adding a sideslip angle control strategy which activates a sideslip-based yaw moment contribution when the sideslip angle value exceeds a pre-defined threshold [14].

Until now, the longitudinal dynamics control of the vehicle has largely remained under the full authority of the driver being restricted in systems such as the CC for comfort reasons and in autonomous vehicle control applications. In addition, braking systems for DYC that decelerate the vehicle are mostly viewed as depreciating on the driving experience [46, 54]. Although it is well known that the driver should remain at the centre of the longitudinal dynamics control, later research has proven that active control can improve the stability in limit handling situations [24, 39]. Terminal understeer arises when an overspeeding vehicle enters a turn and its turning radius cannot be decreased to match the minimum turn-radius given by its velocity and understeer gradient. One of the earliest studies which explored this idea is [35] where they noted that stability and path tracking is improved with the combination of a corrective yaw moment and braking through appropriate brake control of the four wheels. More recently, Rajamani and Piyabongkarn [50] concluded that a reduction of lateral acceleration by decreasing the velocity of the vehicle before entering a sharp turn provides a better cornering performance and rollover mitigation than a typical yaw rate controller. Reduction of lateral acceleration results in reduction of slip angle at the tyres and lower chances of exceeding the limit of adhesion.

A more interesting EV implementation of active longitudinal dynamics control can be found in [61]. Here, a combined solution of yaw stabilisation and velocity regulation for terminal understeer mitigation is presented. The controller is an extension of a previous work [69] which consisted of a Linear Quadratic Regulator (LQR) with the steering angle and the angular rate of the rear wheels as inputs. In addition, they extended the control architecture to contain a rear axle torque vectoring configuration, considering independently

driven rear wheels and taking into account the requested turning radius in agreement with the velocity of the vehicle.

1.2.3 Model Predictive Control in Chassis Control

Model Predictive Control (MPC) models are mainly solving complex dynamical systems. This complexity occurs due to large time delays and high-order dynamics where PID controllers have a difficulty solving. Recently, the emergence of MPC and efficient numerical algorithms, has made it possible for such techniques to be used in vehicle chassis control applications, delivering optimal solutions and incorporation of critical constraints in the calculation of the control action.

In the automotive research and development sector, a variety of MPC solutions can be found in the literature, ranging from steering control [18] to semi-active suspension control [10], longitudinal following control of autonomous vehicles to achieve vehicle platoons [48] and emission regulation [56]. Looking closer in our area of interest, the area of vehicle dynamics control, we distinguish two major MPC application areas, autonomous\semi-autonomous vehicle control and active safety control systems. However, due to the rapid improvement of sensor technologies and sensor fusion algorithms, the distinction between those two application fields is becoming ambiguous nowadays.

From the autonomous vehicle applications perspective, a series of papers have been presented from Borrelli and Falcone [9, 17–20] and Keviczky [32], exploring the application of MPC for trajectory tracking in an autonomous vehicle application using the AFS system with/without differential braking and traction control. In [9, 32] both authors implemented a nonlinear MPC (NMPC) strategy for tracking a determined trajectory using the AFS of an autonomous vehicle which, according to the authors, sets the benchmark for future sub-optimal strategies. Since the problem with the NMPC strategy was that it

could not be implemented in real-time, in [18], Falcone et al. presented a Linear Time Varying (LTV) MPC linearising the NMPC problem from [9, 32] about the operating point. Simulation and experimental results show no infeasibility issues in high velocities but are suffering in tracking performance compared to the NMPC formulation. In the next two papers [17,19] they integrated more functions in the MPC including independent wheel braking and active front and rear differentials. Another distinction from the previous work is the replacement of the bicycle model by a four-wheel vehicle model in [19], although the effect of load transfer is still not taken into account. The goal of the MPC is to follow a predefined trajectory as before but also follow a given velocity reference. The double lane change simulations on a low-μ road surface show a comparison of different drivetrain topologies, one which includes AFS with braking and traction control, another which neglects traction control and finally one including AFS only. As for the test results, it is interesting to note that the reference velocity used is set equal to the initial velocity of the vehicle, thus a speed decrease is observed due to the vehicle reaching a terminal understeer condition. From the above results, the authors concluded the solution that combines AFS with braking and traction control has the best overall performance but the best lateral position tracking is achieved by the solution that uses AFS with braking control only.

In the final work of the series [20] the authors constructed two NMPC strategies, one employing a four-wheel vehicle model with wheel dynamics and control inputs the front steering and individual wheel brake torques and another one that uses a bicycle model instead, with a direct yaw moment along with AFS as control inputs. Although simulation tests on a double-lane change show promising results, once again the main problem for both controllers remains to be the high computational cost, making them impossible for real-time implementation. For that reason, a third controller is developed which uses a linearisation of the first, more complex, controller about the operating point and tested on

a vehicle with rather good path tracking results. All three controllers showed some trade-offs exhibiting certain advantages and disadvantages, although a recurring topic seems to be the importance of good tuning.

In the field of active safety control systems there is plenty of literature too, varying from yaw stability controllers using independent braking of the four wheels based on a LTV-MPC strategy [5] and hybrid MPC [15], to slip controller using torque blending of both electric and hydraulic brake torque [7] and torque vectoring algorithm to achieve minimum time performance maneuvering [64]. Having a closer look at Basrah et al. [7], they employed both linear and nonlinear MPC strategies integrating a slip controller and torque blending between both electric and hydraulic braking actuators. The internal model used in the MPC algorithm is a single-wheel model consisted of longitudinal acceleration, angular wheel rate and the total amount of wheel torque equal to the summation of electric and hydraulic torque. For the calculation of the tyre force they use a simplified version of Pacejka's Magic Formula (MF) which contains the controlled longitudinal slip to compute the longitudinal force, and neglect the lateral movement of the vehicle. Simulation results show that the linear MPC suffers from poor performance at low speeds compared to the nonlinear one, but both controllers track the slip reference target in a similar way for most of the braking maneuver. A solution to that poor performance is proposed by reducing the sampling time of the linear MPC, however the computational performance is worsened making the controller not implementable in real-time. One interesting observation was made under the split μ simulation maneuver where the vehicle maintains stability and steerability throughout the braking with sufficient countersteering by the CarMaker driver model.

1.3 Gap in knowledge

A series of torque allocation techniques including predictive control have been done by Khajepour et al. from the University of Waterloo in collaboration with General Motors (GM). In [30] they deployed a linear MPC technique using recursive linearisation of the vehicle dynamics, to achieve the desired handling response (yaw rate) by distributing the torque demand in the four wheels of an electric vehicle. Later, in [29] they expanded their work by adding a velocity estimation method which treats acceleration measurement noises and road conditions as uncertainties, implementing an Unscented Kalman Filter (UKF). Longitudinal and lateral tyre forces are assumed to be known from the Kalman Filter estimation without requiring the road friction coefficient. The MPC calculates the appropriate wheel torque according to the yaw rate, the yaw moment of the lateral tyre forces and the wheel speed tracking errors. Then they feed the torque change for each wheel from the current driver's torque demand, based on the accelerator pedal position, to the vehicle. The real-time results in both [29] and [30] show effective handling and stability performances tested on a four electric motor wheel vehicle under several driving scenarios.

In [63] a combined yaw stabilization and velocity regulation is presented to mitigate terminal under-steer using rear axle electric torque vectoring. The vehicle model incorporates nonlinear tyre characteristics and coupling of the longitudinal and lateral tyre forces and linear MPC designs are presented using recursive linearisation of the vehicle dynamics. Recently, the control design from [63] has been extended to nonlinear MPC in [62] and compared with previous linear approaches, both in terms of control objective achievement and demanded computational resource. It is worth noting that the control scheme of both [63] and [62] does not take into consideration any torque demand by the driver. The controller aims at stabilizing the lateral dynamics of the vehicle and tracking a speed ref-

erence determined from the steering input of the driver in order for the requested turning radius to be feasible. In addition, this approach does not consider the modification of the handling behavior of the vehicle, as for instance in [14] where the understeer gradient of the controlled vehicle is actively modified by the torque vectoring system.

In this research study we present the development of a torque vectoring controller to distribute the requested drive/brake torque to the four wheels of an electric vehicle to control the longitudinal and lateral dynamics. In addition to the approaches of [30] and [14], which are close to the classic torque vectoring, the controller is designed to intervene to the longitudinal dynamics of the vehicle in cases of overspeeding in a cornering maneuver, thus having a protection on terminal understeer. The controller delivers an on-demand modified lateral dynamics response and aims to deliver the torque demand set by the driver in addition to the approach of [63] and [62]. We employ a NMPC design which accounts for vehicle dynamics nonlinearities and actuator limitations. In addition, the longitudinal intervention is integrated into the control design by introducing a feasible velocity constraint, rather than tracking a velocity reference as in [63] and [62]. The controller is implemented in a high fidelity simulation environment, IPG Carmaker [2], where its performance is demonstrated and the real time implementation capability is discussed.

1.4 Objectives and Methodology

The aim of this research is to develop a real-time implementable torque vectoring control algorithm for electric vehicles with a limit case scenario expansion for limit handling conditions. The attention is focused on a specific vehicle, the Delta Motorsport E-4 Coupe which is a AWD electric vehicle with four electric motors each commanding one wheel. The final solution will be able to stabilise the vehicle under any sub\limit handling condition including oversteer cases.

In order to meet this objective, some contributions must be achieved first:

- develop a computationally simple yet accurate vehicle model to produce the reference signals needed for the controller to follow
- develop an unconstrained linear optimal control to observe the torque vectoring behavior
- develop a constrained version of the previous controller showing the importance of the velocity regulation under limit case scenarios
- expand the constrained controller to nonlinear and analyse both advantages and disadvantages
- compare both linear and nonlinear optimal controllers' computational computer real time performance

It is important to note at this point that all the developed strategies will be systematically assessed in terms of real-time feasibility, since in the context of vehicle dynamics control strategies like the ones presented here it is important to make sure that all solutions are real-time implementable. The simulation studies are made on a laptop computer (i7-4710HQ at 2.50 GHz with 16 GB of RAM memory).

Chapter 2

Vehicle Modelling

2.1 Vehicle Dynamics

In this section we provide the vehicle dynamics model which is used to calculate the optimal control inputs by the Model Predictive Controller presented in the following section. As mentioned in the introduction, the controller is aimed to intervene in both longitudinal and lateral dynamics and hence longitudinal, lateral speed and yaw rate are the selected state variables of the model.

A three Degrees of Freedom vehicle model is used in this study where its Equations of Motion (EoM) are expressed in a coordinated frame attached to the center of mass as shown in Fig. 2.1. Similar to common practice [60, 70], in order to reduce the complexity of the model certain assumptions are made, neglecting:

- the Ackerman Principle, therefore both front wheels steer with the same angle
- the rolling resistance
- the suspension dynamics
- pitch and roll motion of vehicle
- the transmission and brake system characteristics
- the aerodynamic forces

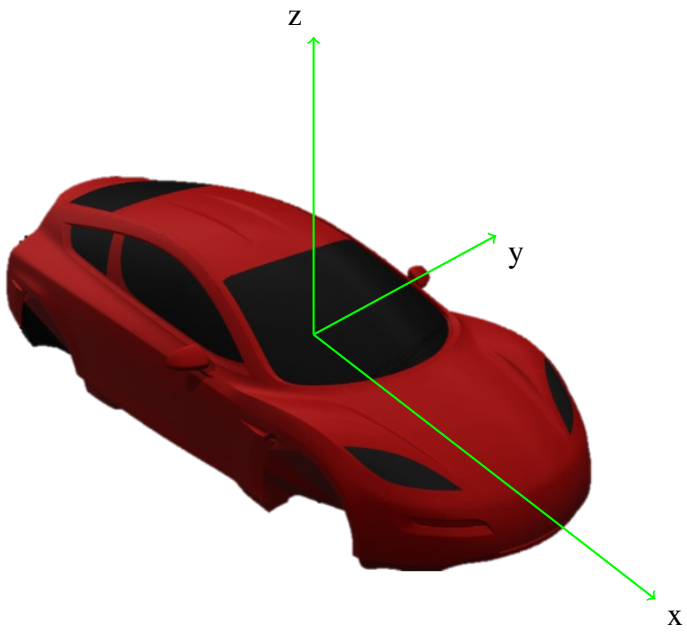

Figure 2.1: Vehicle coordinate system

Using Newton's 2nd Law in the longitudinal and lateral direction we derive the Equations of Motion:

$$\begin{aligned} ma_x &= \sum f_x, \\ ma_y &= \sum f_y, \end{aligned} \tag{2.1}$$

where a_x, a_y are the longitudianl and lateral accelerations respectively and can be expressed in terms of the corresponding velocities V_x and V_y and the yaw rate r as follows:

$$\begin{aligned} a_x &= \dot{V}_x - rV_y, \\ a_y &= \dot{V}_y + rV_x \end{aligned} \tag{2.2}$$

Additionally, including the rotational part of the Neuton-Euler equations and the angular rate dynamics of the four wheels, the EoM for the four wheel vehicle model are:

$$m\dot{V}_x = (f_{FL_x} + f_{FR_x})\cos\delta - (f_{FL_y} + f_{FR_y})\sin\delta + f_{RL_x} + f_{RR_x} + mrV_y, \tag{2.3a}$$

$$m\dot{V}_y = (f_{FL_x} + f_{FR_x})\sin\delta + (f_{FL_y} + f_{FR_y})\cos\delta + f_{RL_y} + f_{RR_y} - mrV_x, \tag{2.3b}$$

$$\begin{aligned} I_z\dot{r} = {} & l_F[(f_{FL_x} + f_{FR_x})\sin\delta + (f_{FL_y} + f_{FR_y})\cos\delta] - l_R(f_{RL_y} + f_{RR_y}) \\ & + w_L(-f_{FL_x}\cos\delta + f_{FL_y}\sin\delta - f_{RL_x}) + w_R(f_{FR_x}\cos\delta - f_{FR_y}\sin\delta + f_{RR_x}), \end{aligned} \tag{2.3c}$$

where m is the mass of the vehicle, δ is the steering angle on both the front wheels, I_z is the vehicle's moment of inertia about the vertical axis and $\dot{r}$ is the vehicle's yaw moment. The longitudinal and lateral tyre forces are denoted by f_{ij_k} where $i = F, R$ (Front, Rear), $j = L, R$ (Left, Right) and $k = x, y$. Finally, the distances l_F, l_R, w_L, w_R determine the location of the center of each wheel with respect to the CoM as shown in Fig.2.2.

In the EoM above we consider the steering angle as a parameter provided by the driver. The longitudinal tyre forces are calculated from wheel torque rate control inputs and vertical wheel loads whereas the lateral tyre forces are provided as functions of the corresponding tyre slip angle as described in the following section. All the parameters used for the vehicle and tyre model can be seen in Table 2.1.

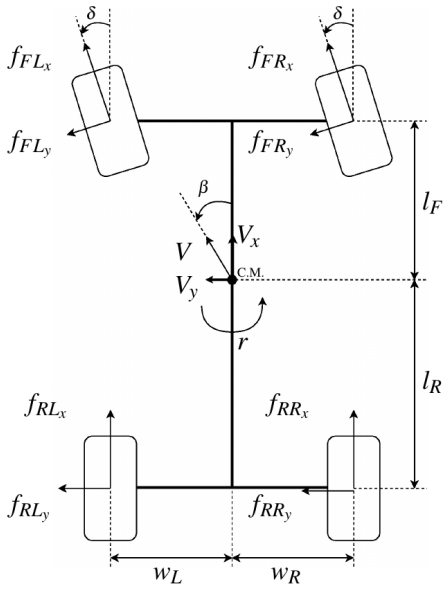

Figure 2.2: Four wheel vehicle model.

Table 2.1: Vehicle properties.

Parameter (Unit)	Description	Value
m (kg)	mass	1137
L (m)	wheelbase	2.5
R_w (m)	wheel radius	0.298
h (m)	height of C.M.	0.317
I_z (kg·m^2)	yaw inertia of vehicle	1174
l_F (m)	front axle to C.M. distance	1.187
l_R (m)	rear axle to C.M. distance	1.313
w_L (m)	left wheels to C.M. distance	0.687
w_R (m)	right wheels to C.M. distance	0.687

2.2 Tyre Forces

By neglecting the pitch and roll rotation along with the vertical motion of the sprung mass of the vehicle, the vertical force f_{ij_z} on each wheel is calculated using the static load

distribution and longitudinal and lateral weight transfers [69]:

$$\begin{aligned} f_{FL_z} &= f^0_{FL_z} - \Delta f^x_L - \Delta f^y_F, \\ f_{FR_z} &= f^0_{FR_z} - \Delta f^x_R + \Delta f^y_F, \\ f_{RL_z} &= f^0_{RL_z} + \Delta f^x_L - \Delta f^y_R, \\ f_{RR_z} &= f^0_{RR_z} + \Delta f^x_R + \Delta f^y_R, \end{aligned} \tag{2.4}$$

$$\begin{aligned} \Delta f^y_F &= \frac{mhl_R}{(l_F + l_R)(w_L + w_R)} a_y, \\ \Delta f^y_R &= \frac{mhl_F}{(l_F + l_R)(w_L + w_R)} a_y, \\ \Delta f^x_L &= \frac{mhw_R}{(l_F + l_R)(w_L + w_R)} a_x, \\ \Delta f^x_R &= \frac{mhw_L}{(l_F + l_R)(w_L + w_R)} a_x, \end{aligned} \tag{2.5}$$

where $f^0_{ij_z}$ is the static vertical force distribution, Δf^y_F and Δf^y_R the weight transfer along the lateral body axis resulting from lateral acceleration a_y and Δf^x_L and Δf^x_R the weight transfer along the longitudinal body axis resulting from longitudinal acceleration a_x.

As mentioned before the longitudinal forces in EoM (2.3a) - (2.3c) are considered as the control inputs of the vehicle dynamics. In the literature TV controllers deliver wheel force (or torque) commands. A low level controller then is applied to provide the commanded wheel forces, controlling the wheel dynamics. In the proposed method we assume knowledge of the friction coefficient between the tyre and road (μ) and use this as a constraint on the individual wheel force command. Hence the demanded force is always feasible, although there will be a delay between applied torque and wheel force associated with the wheel inertia. For simplicity and computational efficiency we do not implement a low level controller for wheel dynamics, however the controller is validated

in the following using a rich vehicle model including wheel dynamics as well as the electric motor's dynamics. Assume we can control longitudinal forces of the tyre directly we can associate the normalised longitudinal tyre forces μ_{ij_x} with the applied wheel torques as follows [30]:

$$\mu_{ij_x} = \frac{T_{ij}}{f_{ij_z} R_w}, \tag{2.6}$$

where T_{ij} is the torque of each wheel and R_w the wheel's radius.

Next we introduce a model for the calculation of the lateral tyre forces, which includes the dependence on tyre slip angle (cornering stiffness), a linear dependency on the normal force and the coupling with the longitudinal tyre forces.

Using the friction circle concept, the maximum lateral tyre force coefficient $\mu_{ij_y}^{\max}$ is given by the tyre-road friction coefficient mu and the controlled longitudinal tyre force coefficient μ_{ij_x}:

$$\mu_{ij_y}^{\max} = \sqrt{\mu^2 - \mu_{ij_x}^2}. \tag{2.7}$$

Neglecting wheel dynamics and tyre force dependency on slip, we assume direct control of longitudinal forces. We are still able to incorporate the friction circle constraint. The lateral force coefficient is then calculated as a linear function of the tyre slip angle, saturated by the above $\mu_{ij_y}^{\max}$ limit.

$$\mu_{ij_y} = sign(\alpha_{ij}) \cdot min(\mu_{ij_y}^{\max}, n_{ij}|\alpha_{ij}|), \tag{2.8}$$

where α_{ij} is the slip angle at each of the four tyres, and n_{ij} is the cornering stiffness coefficient of each tyre defined as the ratio of tyre cornering stiffness divided by the normal force at each tyre.

In order to avoid the non-smoothness at the point of saturation in equation (2.8), we propose to use an approximation of this expression using the Logistic function [37] as follows:

$$\mu_{ij_y} = \frac{2\mu_{ij_y}^{\max}}{1+e^{-kn_i\alpha_{ij}}} - \mu_{ij_y}^{\max}. \tag{2.9}$$

In the above equation k is the steepness of the curve and is tuned according to the cornering stiffness coefficient of the tyre. To be more precise, we simulated the tyre model under the same road friction coefficient but different understeer gradients, found the best match for the variable k in each case and then calculated a curve that fits those points using MATLAB's "*fit*" command. The k as a function of μ_{ij}^{max} is:

$$k(\mu_{ij}^{max}) = (p_1 \cdot (\mu_{ij}^{max})^2 + p_2 \cdot \mu_{ij}^{max} + p_3) \tag{2.10}$$

where $p_1 = 5.179$, $p_2 = -12.37$ and $p_3 = 9.429$.

As we can see in Fig.2.3 the saturating equation (2.9) not only follows the boundaries but is smoother than the one corresponding to (2.8). Finally, the longitudinal and lateral tyre forces are calculated using the normal force at each tyre as follows:

$$f_{ij_x} = f_{ij_z}\mu_{ij_x} \quad \text{and} \quad f_{ij_y} = f_{ij_z}\mu_{ij_y} \tag{2.11}$$

2.3 Powertrain

In a vehicle, the powertrain portrays the fundamental components that generate power and drive it to the road surface. This incorporates the motor, transmission, drive shafts,

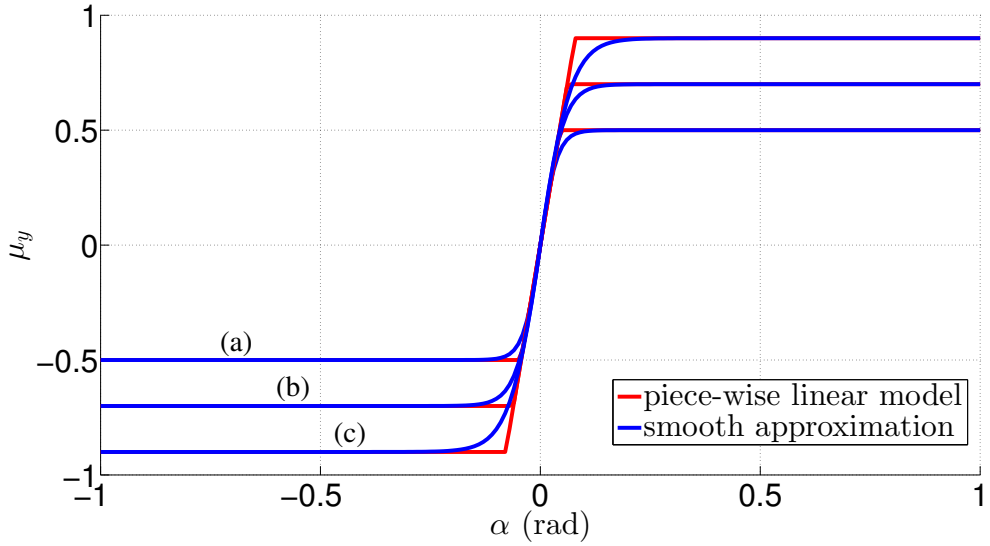

Figure 2.3: Lateral road coefficient approximation; (a) $\mu_{max} = 0.9$, (b) $\mu_{max} = 0.7$, (c) $\mu_{max} = 0.5$

differentials, and the final drive. In the last decade more elements have been added to the powertrain group due to the emergence of electric and hybrid vehicles, such as the battery, the electric motor and the control algorithm.

The powertrain used in this study is based on an electric vehicle with four individual electric motors (EM). Each EM delivers torque to one wheel limited by the producer's specs. In our case the electric vehicle used for the simulations is based on the Delta E4-coupe, a prototype vehicle manufactured by Delta Motorsport [1], a company based in Silverstone circuit. Their vehicle has four individual YASA electric motors with maximum and minimum torques $T_{max} = 750Nm$ and $T_{min} = -500Nm$ respectively. The vehicle is also equipped with every necessary measurement sensors all road vehicles include.

The torque vectoring control algorithm we employ is deeply focused on pure electric vehicle control, however the vehicle features hydraulic brakes on each wheel for safety reasons. As shown in Fig.2.4 the static torque of the YASA-750 electric motor is not constant but changes according to its rotational speed, making it important to include torque

constraints in the controller. The control scheme we are implementing is respecting the EM torque limits and can be easily reconfigured to work with different brand manufacturers without losing its robustness.

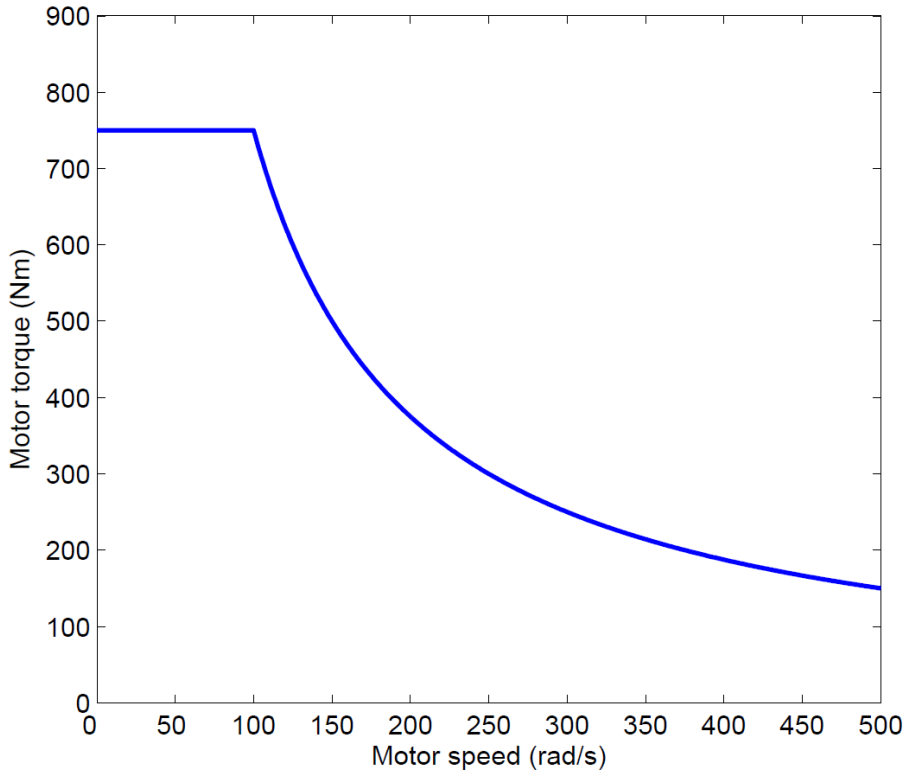

Figure 2.4: Static torque map of YASA-750 motor. [60]

Chapter 3

Model Predictive Control Fundamentals

Model Predictive Control is an optimisation based control law which utilises an interior model of the procedure, a measured history of the past control inputs and an optimisation cost function J over a receding prediction horizon. MPC, is also called Receding Horizon Control (RHC). This control algorithm computes the necessary control inputs for the plant every time step. To be more precise, it solves an omptimisation problem for the current prediction horizon, then applies the first value of the computed control sequence to the plant and finally gets the system state and recomputes at the next time step. This conceptual idea of the so-called receding horizon policy is shown in a simple graph presentation in Fig.3.1.

While MPC has attracted both automotive industry and academia research, especially when using constraints, it has some drawbacks which need careful consideration when designing the controller:

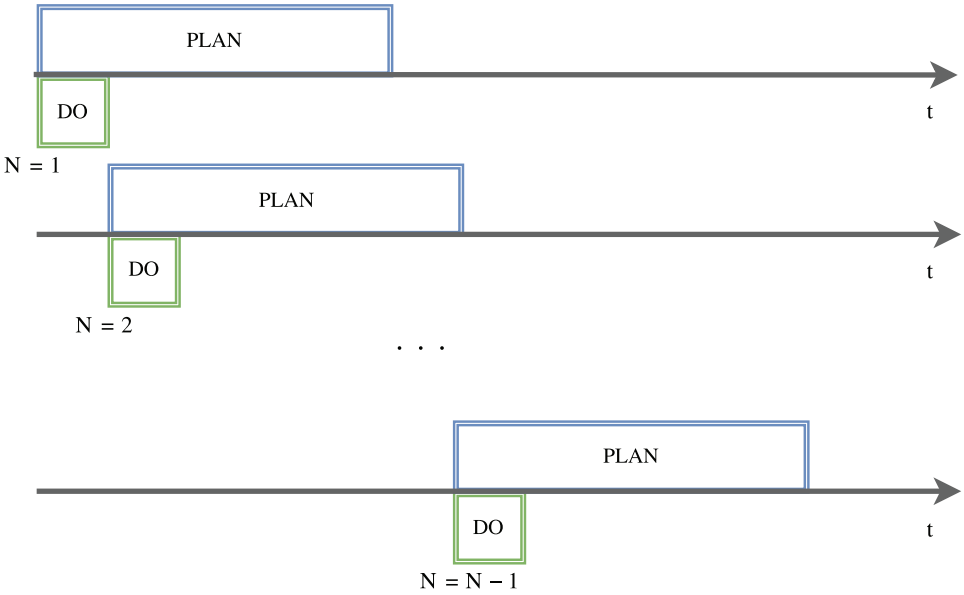

Figure 3.1: Receding Horizon Control policy.

- intenral model: a large, nonlinear model can rapidly increase the number of optimisation variables and the complexity of the problem to be solved

- sampling time: a longer sampling time can reduce the number of optimisation variables in a fixed horizon but may result in slower, ineffective control actions

- prediction and control horizons: short horizons for a fixed sampling time can also reduce the number of optimisation variables but may result in inadequate control actions too

- constraints: a large number of constraints and nonlinear state and control input constraints, increases the problem complexity, whereas linear constraints may fail to capture the nature of the original limits

- weighting parameters: tuning parameters to be chosen which relate to the minimisation of the cost function like in any other optimal control problem

Therefore, there is a clear trade-off between performance and computational effort attached to both the internal model of the MPC choice and the tuning parameters, which in the case of a vehicle control strategy with fast dynamics, need to be carefully chosen.

MPC is an optimization based control law, and the cost function almost always consists of quadratic terms. The objective is to find the optimal control input u that minimises the cost J under a finite prediction horizon N, by defining positive definite matrices $Q = Q^T > 0$ and $R = R^T > 0$ which are the weighting matrices on the state error and control input respectively.

The simple MPC regulation problem is:

$$\min_{x,u} \sum_{k=0}^{N-1} (x_k - x_{ref})^T Q_d (x_k - x_{ref}) + (u_k - u_{ref})^T R_d (u_k - u_{ref}), \tag{3.1a}$$

$$\text{subject to:} \quad x_{k+1} = Ax_k + Bu_k, \quad k = 0,1,...,N-1 \tag{3.1b}$$

$$x_k \in \mathscr{X}, \quad k = 1,...,N \tag{3.1c}$$

$$u_k \in \mathscr{U}, \quad k = 0,1,...,N-1 \tag{3.1d}$$

$$x_0 = x(t). \tag{3.1e}$$

where (3.1a) is the cost function to minimise with x and u being the states and control inputs respectively, (3.1b) are the affine discrete system dynamics and (3.1c) - (3.1d) are the state and input inequality constraints respectively with $\mathscr{X}$ and $\mathscr{U}$ the corresponding boundaries. Finally the (3.1e) sets the initial state x_0 equal to the current state.

Based on the above observations we construct a linear and nonlinear MPC approach as described in the following sections.

3.1 Cost Function

The main objective of the controller is to track a yaw rate reference and respect the drivers torque demand while at the same time regulating the velocity so that it stays within a feasible region. Therefore, the objective function is defined as:

$$J = \sum_{k=0}^{N-1} q_r(r - r_{ref})^2 + q_{T_{dmd}}(T_{dmd} - T_{veh})^2 + q_u u^2 + \rho_r e_r + \rho_V e_V, \tag{3.2}$$

where q_r, q_T and q_u in front of each term are their corresponding weights, r is the current yaw rate of the vehicle, T_{dmd} is the driver's total torque demand, u corresponds to the wheel torque control inputs and e_r and e_V are used in the yaw rate and velocity constraints as described in Section 3.2.

The first term in the cost function relates to the yaw rate tracking error. As common practice suggests [14], we use a linear steady-state bicycle model with the desired handling characteristic (understeer gradient) to create the reference yaw rate:

$$r_{ref} = \delta \frac{V_x}{L + K_{und} V_x^2}, \tag{3.3}$$

where L is the length of the vehicle and K_{und} is the desired understeer gradient. The second term in (3.2) is defined such that the controller meets the drivers torque demand, where T_{veh} is the summation of all four wheel torques T_{ij}. The third term introduces penalisation of the control inputs. As explained in the following section, the last two terms are used to soften the yaw rate and velocity constraint respectively.

3.2 State Constraints

As discussed in the introduction, the aim of the proposed controller is to intervene in the longitudinal dynamics, when the requested lateral acceleration is infeasible for the given velocity of the vehicle. Accordingly, we chose the state vector for the controller $x = [V_x \ V_y \ r]$.

From the steady-state equations of the bicycle model the reference yaw rate corresponds to a reference lateral acceleration, which is limited by the available tyre-road friction coefficient (grip)

$$a_{y_{ref}} = V r_{ref} \leqslant \mu g, \tag{3.4}$$

where g is the acceleration of gravity and V is the current vehicles total speed [49]. The above can be interpreted as a limit of the vehicle's speed such that the requested yaw rate (or lateral acceleration) is feasible:

$$V_{lim} = g\frac{\mu}{r_{ref}}. \tag{3.5}$$

This translates to a state constraint:

$$\left|\frac{V_x}{cos\beta}\right| \leqslant V_{lim} + e_V, \tag{3.6}$$

which is implemented as a soft constraint with slack variable $e_V \geqslant 0$. It is important to note that we implement a soft constraint at this stage because the yaw rate reference is generated by the driver's steering input and there are no guarantees that the constraint will be respected. The control action to reduce the speed (brake) in order to satisfy the constraint (3.6) is herein referred to as Active Trail-Braking.

Similar to [62, 63] we impose a yaw rate constraint to increase the robustness of the controller:

$$|r| \leqslant r_{lim} + e_r, \tag{3.7}$$

where $e_r \geqslant 0$ is the corresponding slack variable and r_{lim} is calculated using the current total vehicle velocity V:

$$r_{lim} = g\frac{\mu}{V}. \tag{3.8}$$

3.3 Input Constraints

The driver's torque demand is another parameter our controller focuses on. Electric vehicles have many precautions when it comes to safety especially regarding the input voltage and current of their electric motors since EMs cannot operate all the time at maximum power. Because of those limitations, each EV has an accurate system protection implemented between the control systems and the electric motors. From this perspective, taking into account the motor's torque map features, we feedback the rotational wheel speeds from the motors back to the controller, calculate the wheel torque constraints using the motor's torque map as shown in Fig.3.2 and also use the maximum torque limits to derate the driver's torque command, Fig.3.3.

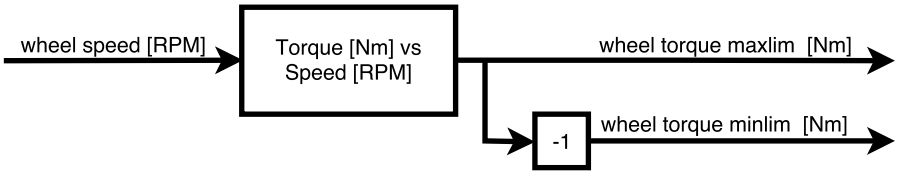

Figure 3.2: Wheel torque limit calculation block.

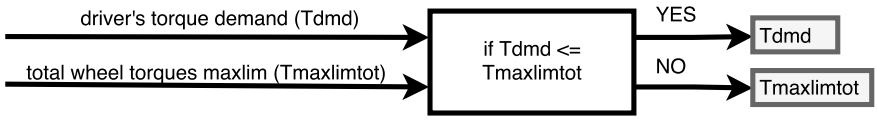

Figure 3.3: Torque demand limit calculation block.

As control inputs for both linear and nonlinear formulation we use the wheel torques which are fed to the vehicle as demanded torques. Considering the limits on the torque capacity of each electric motor, we constraint the control inputs as:

$$T_{ij}^{min} \leqslant T_{ij} \leqslant T_{ij}^{max}, \tag{3.9}$$

where T_{ij}^{min} and T_{ij}^{max} are calculated as described above and shown in Fig.3.2 which only corresponds to the front-left wheel.

In the nonlinear control development section we also test a different formulation where the control input changes to wheel torque rates instead of wheel torques and the wheel torques become states which are also constrained as in (3.9). Additionally, constraints are set on the control inputs ΔT_{ij} so that the rates of torque on the wheels never exceed the maximum allowable torque change for both motor and battery safe operations

$$|\Delta T_{ij}| \leqslant \Delta T_{safe} T_s, \tag{3.10}$$

where ΔT_{safe} is given by the electric motor supplier. All the motor parameters which were taken under consideration for our simulations can be seen in Table 3.1.

Table 3.1: Motor specs in simulation. [1]

Parameter (Unit)	Description	Value
ΔT_{safe} (Nm/s)	motor/battery torque rate limit	10000
T_{max} (Nm)	maximum torque	700
T_{min} (Nm)	minimum torque	-500

Chapter 4

Linear Torque Vectoring Control Development

Based on the standard linear MPC problem (3.1) a dense MPC formulation using soft constraints on the states is used in this chapter to avoid infeasibility problems [41], with the necessary A and B matrices updated at each time step according to the current steering command from the driver and the current vehicle velocity. In addition, we chose the state vector for the controller $x = [V_x \ V_y \ r]$ and as control input the torque for each wheel $u = [T_{ij}]$. The resulting Quadratic Programming (QP) problem is then solved using a specialised solver FORCES Pro solver [16] in MATLAB which employs the Primal-Dual Interior Point (PDIP) method [73, 76].

In order to create the mandatory formulation for the linear controller, first we have to linearise the continuous system as described in Chapter 2, using the continuous vehicle

dynamics (2.3). There is plenty of research work done around linearisation techniques in the literature. For our research we used a complex variable derivative estimation technique which is faster and easier to implement [65]. Let $F(z)$ be an infinitely differentiable and smoothly extended into the complex plane function, x_0 be a point on the real axis and h be a real parameter. Expanding $F(z)$ in a Taylor series off the real axis we get:

$$F(x_0+ih) = F(x_0) + ihF'(x_0) - h^2\frac{F''(x_0)}{2!} - ih^3\frac{F^{(3)}(x_0)}{3!} + \dots \tag{4.1}$$

Dividing the imaginary part by h

$$F'(x_0) = \frac{Im(F(x_0+ih))}{h} + O(h^2) \tag{4.2}$$

gives an approximation to the value of the derivative, $F'(x_0)$, that is accurate to order $O(h^2)$. In our linearisation we choose $h = 10^{-16}$.

An MPC controller requires a discrete form of the internal prediction model. Hence, after computing the linearised formulation of the system

$$\dot{x} = A_c x + B_c u \tag{4.3}$$

we descretise the A_c and B_c matrices using the Euler's approximation [42]:

$$A_d = I + T_s A_c \tag{4.4a}$$

$$B_d = T_s B_c \tag{4.4b}$$

where I is the identity matrix and T_s is the controller's sampling time. Then the descretised system becomes

$$\widetilde{x}_{k+1} = A_d\widetilde{x}_k + B_d\widetilde{u}_k. \tag{4.5}$$

4.1 Simulation Results

In this section we compare the MPC strategies to one another as well as against an uncontrolled vehicle with equal wheel torque split using the high fidelity vehicle model and the driver model available in CarMaker. The vehicle model in CarMaker is naturally understeer and was granted by Delta Motorsport. The testing maneuvers are a Step-Steer and Double Lane Change scenario.

4.1.1 Step Steer

In the Step-Steer maneuver we assume a constant torque demand by the driver and apply a steering wheel input $\delta = 90°$ after 2 seconds. This scenario is considered fundamental for the controller as the weights of the cost function are tuned related to it and are kept the same for the purposes of comparison. The range of the driver's torque demand can vary from 0Nm to 3000Nm, although due to the extreme handling conditions while the torque increases as well as the chosen controller's tuning parameters, the vehicle obeys the driver's torque command up to 1800Nm.

4.1.1.1 Importance of velocity constraint

The following simulations show the necessity of adding a velocity regulation feature in the control scheme. The comparison is done between an uncontrolled vehicle (UnCtrl), where there is no Torque Vectoring control and the driver's torque demand is equally split to the 4 wheels, a vehicle that includes TV implemented with a LMPC including a yaw rate constraint inside the yaw rate reference signal and control input constraints on the wheel torques but without state constraints (LMPC Unconstr), similar to [30], and a fully TV controlled vehicle with velocity regulation feature as implemented in this research (LMPC Constr).

As shown in the velocity graph in Fig.4.1, the uncontrolled vehicle, green line, doesn't regulate its speed during cornering whereas the controlled ones respond better. The blue line which corresponds to a vehicle similar to [30], has no hard constraint on both yaw rate and velocity although due to the constraining of the yaw rate reference signal we observe a much better velocity reduction than the uncontrolled case. On the other hand, the controller implemented for this research adjusts the vehicle's velocity quickly in less than 2 seconds and then remains constant during the maneuver slightly under the constraint. The soft constraint on the velocity allows the controller to find a solution even though the constraint itself is not respected, depending on the value of its weighting parameter inside the cost function, in our case the p_V in (3.2). All the details regarding the weights are mentioned in the Appendix section.

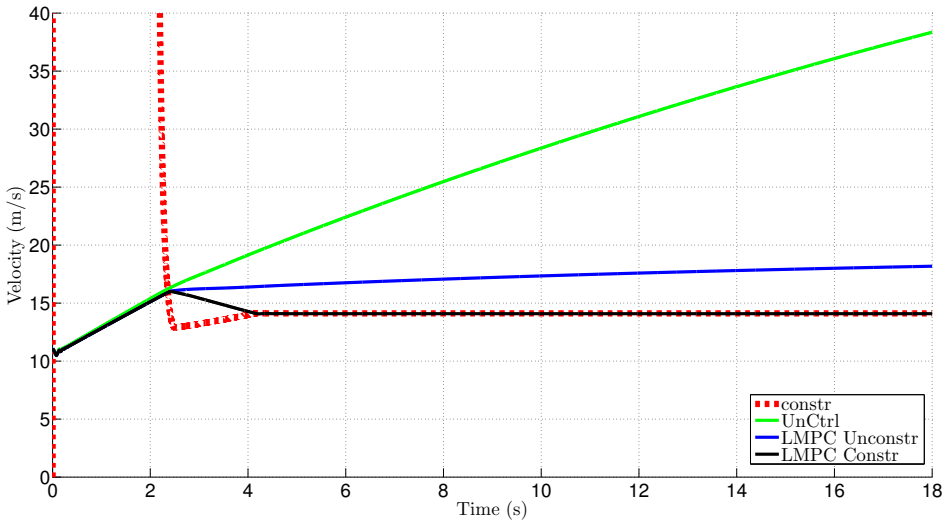

Figure 4.1: Velocity, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

Figure 4.2 shows the yaw rate responses of the three vehicles being compared. Once again, it is easily observed that the vehicle with equal torque split to all four wheels doesn't go into a steady-state phase. The LMPC unconstrained vehicle has no hard constraints on velocity as proved on the previous graph nor on yaw rate, but the yaw rate response has a lower slope and respects the yaw rate reference with a minor error. The only vehicle that goes into steady-state is the constrained LMPC. While there is a minor overshoot at the start of the step-steer between the seconds 2 and 3, which is allowed by the yaw rate soft constraint similar to the velocity one, the controller respects the yaw rate constraint quicker than the velocity case and then follows the reference signal with an error of $1.5°/s$. Moreover, all the trajectories of the vehicles are shown in Fig.4.3 where the fully constrained vehicle, LMPC Constr, completes a circle whereas the other vehicles are following a more spiral path.

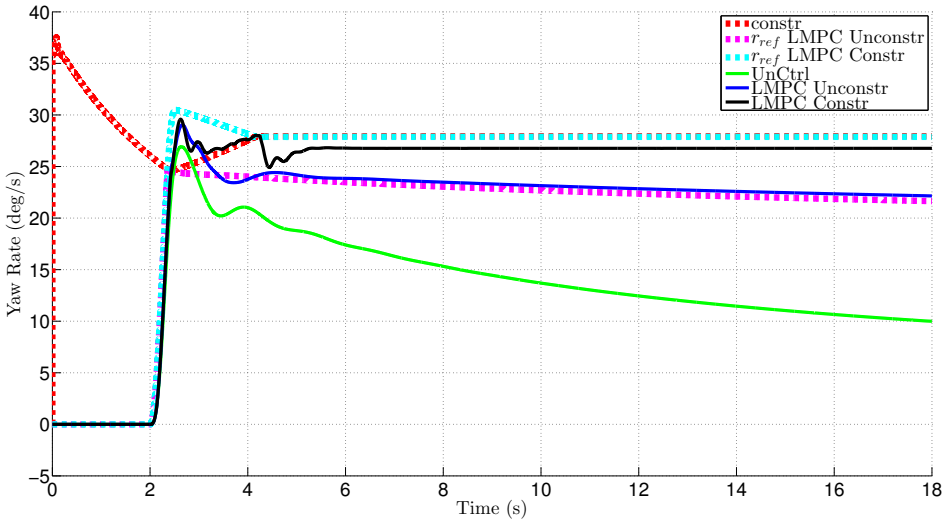

Figure 4.2: Yaw rate, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

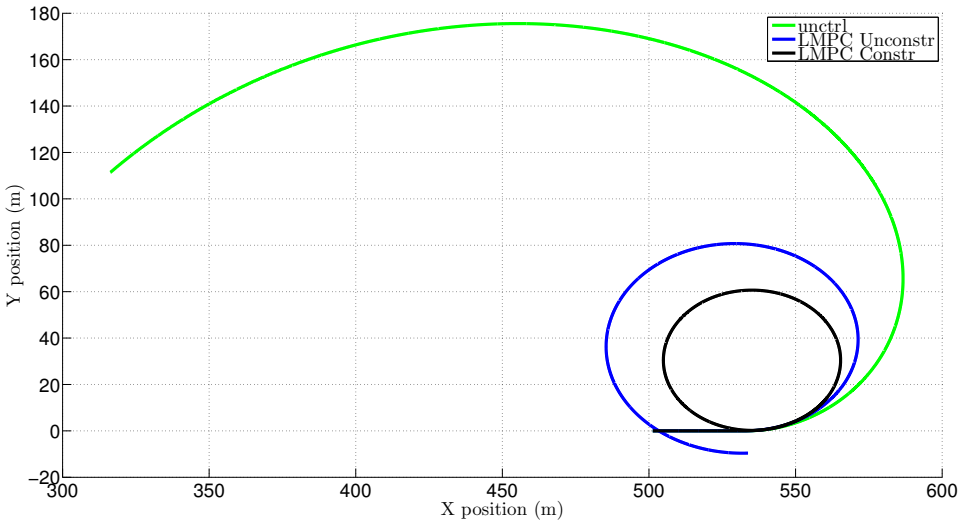

Figure 4.3: Position, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

According to the system protection which all EVs have as mentioned in Chapter 3, the wheel torques of the same step-steer example ($1000Nm$ and $V_{in} = 40kph$) shown in Fig.4.4 don't reach the expected $250Nm$ torque each, in the first 2 seconds due to the torque derates of the motor's driving current limitations inside the system protection block. Furthermore after the 2 seconds have passed, we observe a torque reduction on the inner left wheels and a complementary increase on the outer right wheels of the vehicle, until the velocity constraint is hit where the vehicle starts braking. The braking torques are configured based on the $q_{T_{dmd}}$ weighting parameter under the overspeeding case (see Appendix for further details). Under the chosen tuning parameters the controller brakes the outer wheels of the vehicle more than the inner ones trying to maintain stability and then enters the steady-state where all wheel torques are positive but the outer wheels have slightly more torque than the inner wheels.

Finally, to show the robustness of the controller working under a different range of driver's torque demands, we observe on Fig.4.5 that all cases reach the steady-state. In this graph the dashed and continuous lines represent the driver's torque demand and the

vehicle's total torque respectively whereas the different line colors differentiate the scenarios between them.

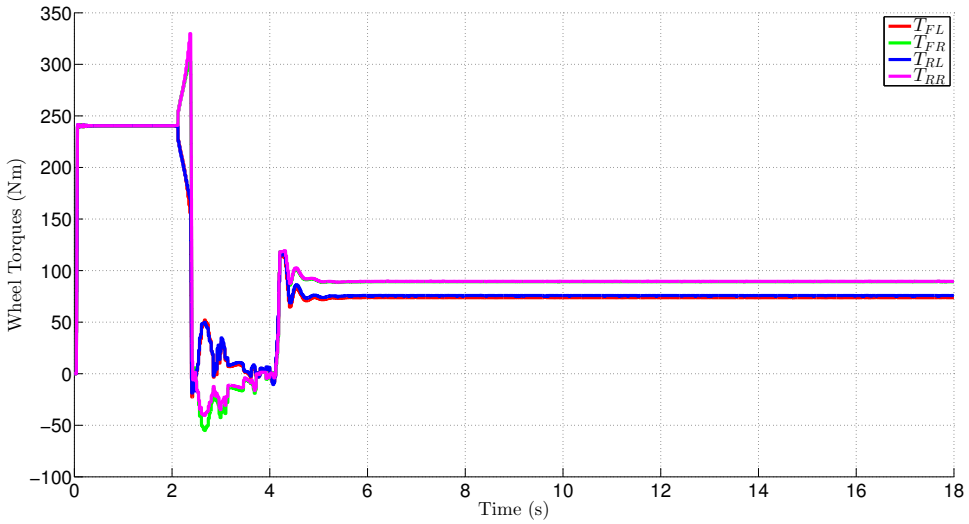

Figure 4.4: Wheel torques, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

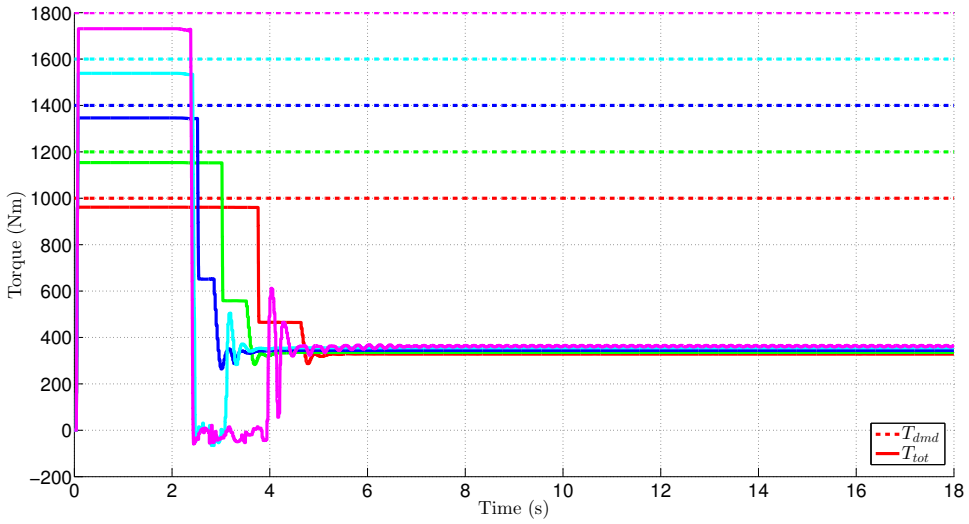

Figure 4.5: Total torque, step-steer at $V_{in} = 40kph$ with a variation of T_{dmd}.

4.1.1.2 Understeer gradient tuning performance

As shown in equation (3.3) the K_{und} has a crucial role on the yaw rate reference generation. Having the ability to independently tune the understeer gradient of the controller's internal vehicle model, gives us the freedom to extensively control the turning radius of the vehicle's trajectory. The understeer gradient used for the tuning purposes is based on an understeer vehicle with $K_{und} = 0.0017 rad/s = 1°/g$. Furthermore the robustness of the controller was tested for different values of $K_{und} = 0$ and $K_{und} = -0.0017 rad/s$ which correspond to a neutralsteer and oversteer vehicle respectively.

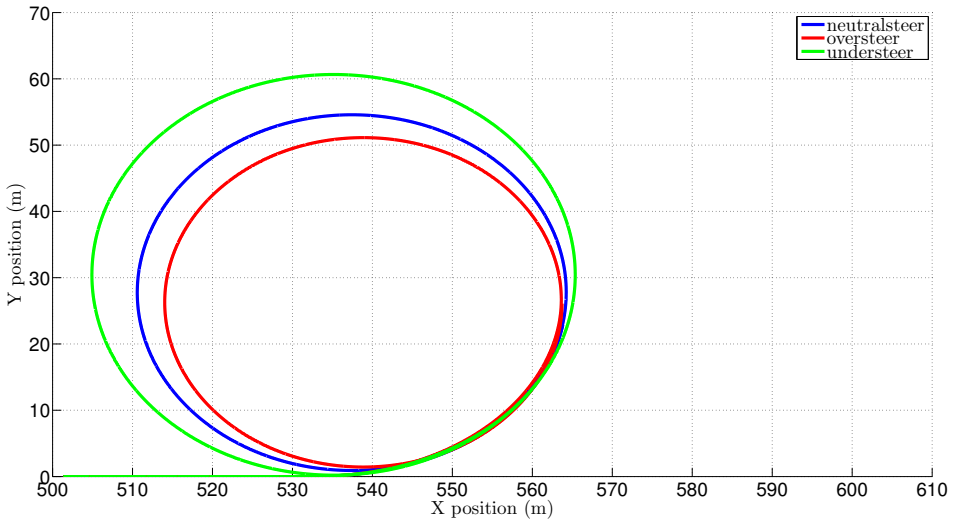

Figure 4.6: Position responses of different K_{und}, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

4.1.2 Double Lane Change

The double lane change maneuver we used is the "Double Lane Change ISO" provided by CarMaker. The CarMaker driver parameters used for the lateral dynamics are shown in Fig.4.7 and the longitudinal dynamics are set manually through Simulink by the constant

driver's torque demand. The vehicle was tested under different road friction coefficients assuming a constant torque demand from the driver and $K_{und} = 0.0017 rad/sec$, as assumed in the previous testing scenario also. One more important thing to mention is that our internal model has a configured constant road friction coefficient $\mu = 0.7$. This value was chosen in order to test the controller's robustness on a variety of different surfaces as it is demonstrated bellow.

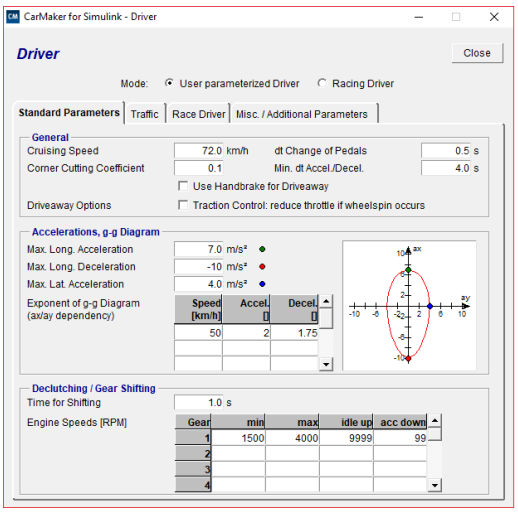

Figure 4.7: Lateral dynamics CarMaker driver parameters.

4.1.2.1 Double Lane Change $\mu = 0.5$

As we can see from both yaw rate and position figures, Fig.4.8 and Fig.4.9 respectively, the uncontrolled vehicle on low road friction coefficient $\mu = 0.5$ is starting to lose control from a total torque demand of $700Nm$ and above whereas the LMPC can successfully complete the maneuver. However, the maximum torque demand on the LMPC is limited up to $720Nm$ before the controlled vehicle is unable to maintain stability for the given set of tuning parameters. Although this can be increased by changing the tuning parameters of the controller, the weights inside the cost function, appropriately.

It is important to note that the yaw rate doesn't respect the constraint all the time

because the driver is the one who controls the steering wheel which has a large effect on the excitement of the vehicles yaw rate. Although the controller continues to find an optimal solution as it is shown on later graphs, because of the soft constraints used as mentioned in Section 3.

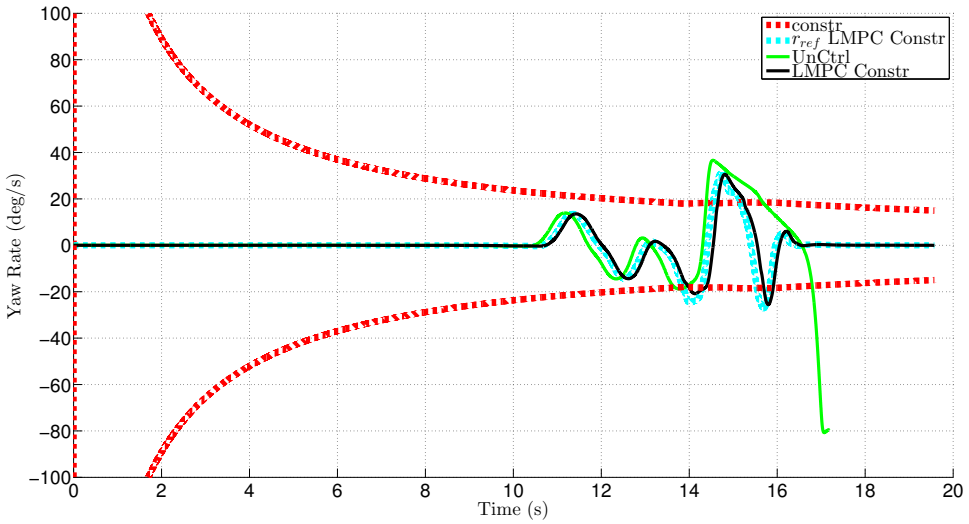

Figure 4.8: Yaw rate, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

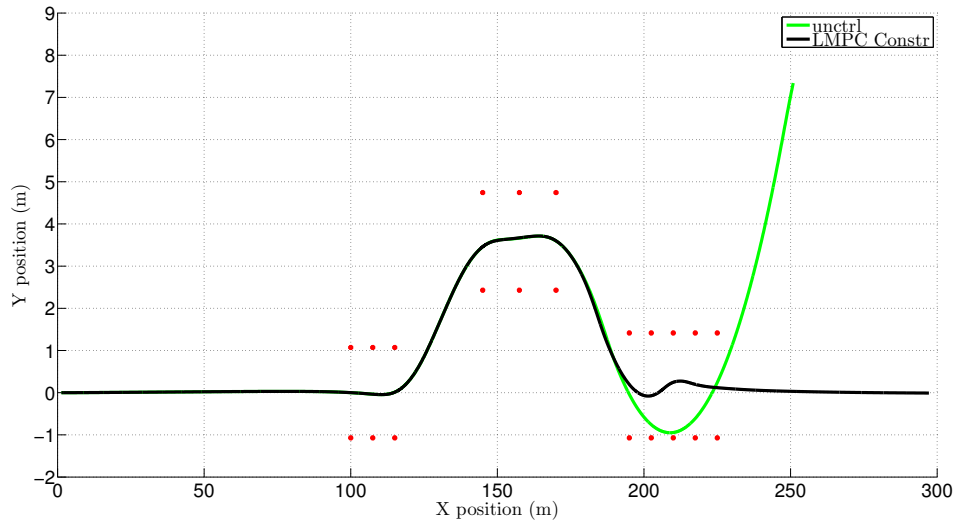

Figure 4.9: Position, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

The velocity graph, in Fig4.10, is zoomed in and presents the velocity regulation comparison between our controller and the uncontrolled vehicle. As it is shown the velocity is regulated as soon as it fails to respect the constraint thus keeping the car inside the maneuver track width. The slope difference in the acceleration phase occurs due to the controller's weighting factor on the driver's torque demand. The red dotted line corresponds to the value of V_{lim} referred in equation (3.5) and has an upper limit of 80mps as the vehicle's maximum speed.

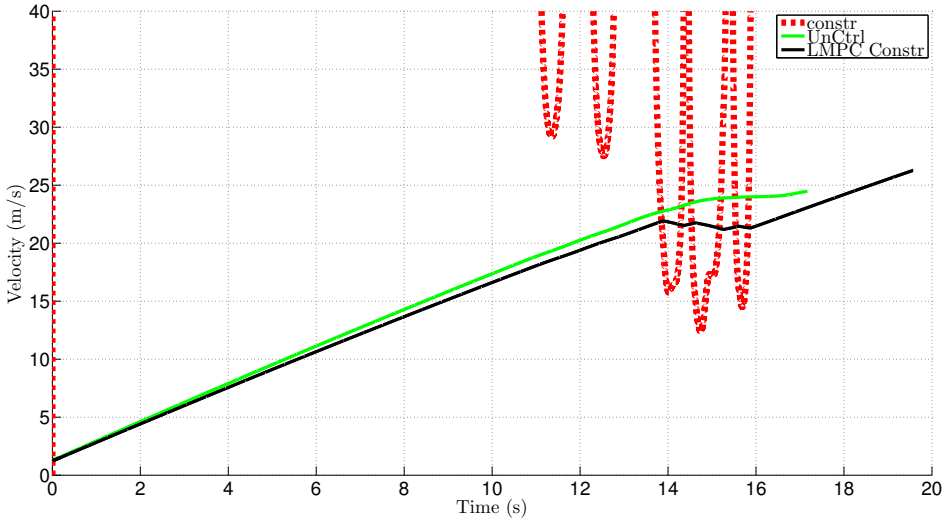

Figure 4.10: Velocity, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

4.1.2.2 Double Lane Change $\mu = 0.7$

Increasing the simulations road friction coefficient to wet asphalt $\mu = 0.7$ the driver's torque demand range where the vehicle remains under his control, is increased up to 950Nm without a TV controller. The maximum torque demand before the controlled vehicle loses stability for the given set of tuning parameters is limited to 1150Nm on a medium wet road.

The yaw rate in Fig.4.11 and top view position of the car in Fig.4.12 show a limit

handling comparison of the uncontrolled car and the TV car where the later completes the maneuver successfully. Both graphs show quite similar response with the previous double lane change configuration. The velocity graph, Fig.4.13 ,is also similar where the velocity is decreased every time it overcomes the constraint and increased when yaw rate goes to zero after the vehicle takes the last lane change.

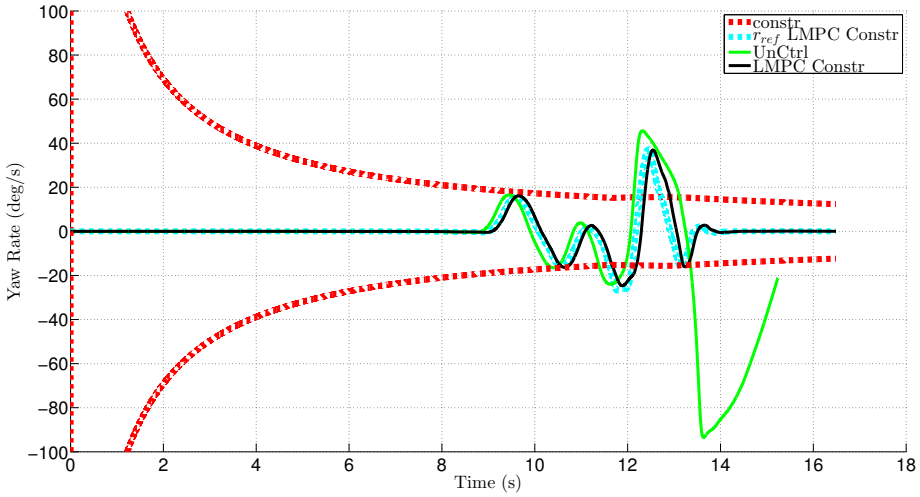

Figure 4.11: Yaw rate, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

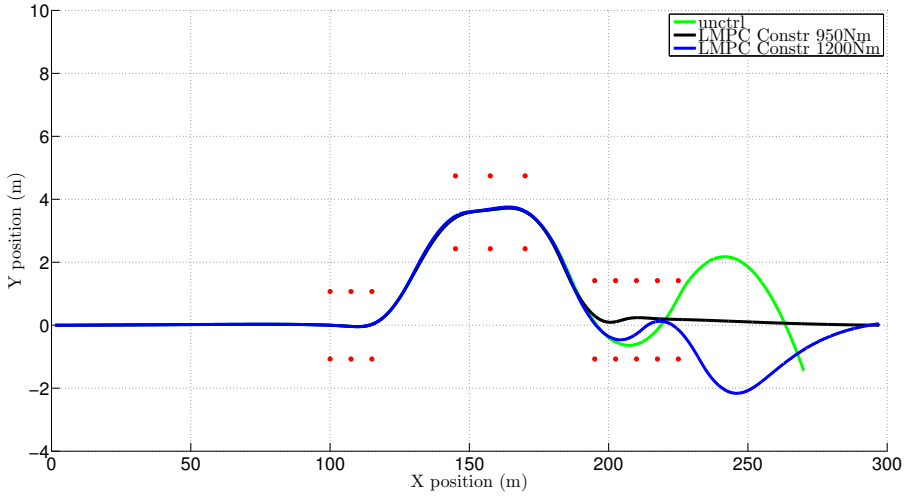

Figure 4.12: Position, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

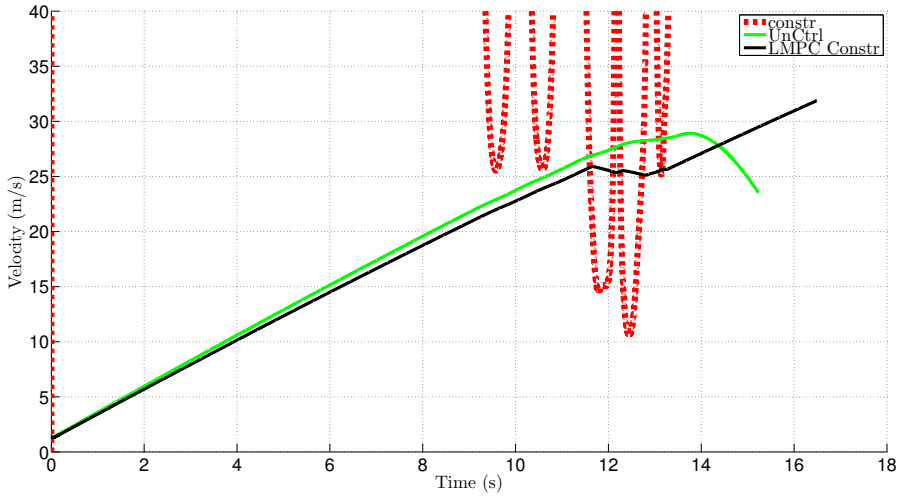

Figure 4.13: Velocity, double lane change at $\mu = 0.7$ and $V_{in} = 8kph$.

The total torque calculated by the LMPC is lower than the driver's torque demand as discussed previously in Chapter 3 due to the motor torque map derate. As we can inspect on Fig.4.14 the torque delivered to the vehicle through the wheels, the powertrain torque (T_{tot} PT), is also lower than the demanded torque computed by the LMPC which makes sense considering the system protection block, located between the controller and the electric motors, contains all the required limitations so that the EV operates according to safety instructions.

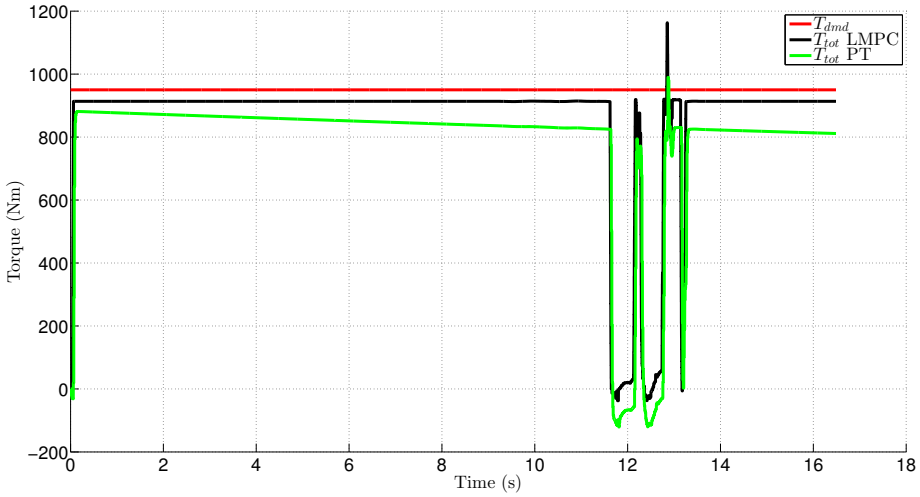

Figure 4.14: Total torque, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

4.1.2.3 Double Lane Change $\mu = 0.9$

Changing the friction of the road to dry asphalt $\mu = 0.9$ increases the vehicle's drive-ability. Figures 4.15-4.17 show the inability of the uncontrolled car to take such a maneuver while the controlled vehicle doesn't lose stability with a maximum driver's torque command up to 1700Nm based on the current tuning configuration and Fig.4.18 shows the total torque derived from the LMPC solution. All these figures have similar results with those from the previous road friction configurations. One important note, is that once again the tuning parameters play a crucial role on the optimisation and solving ability/inability of the controller.

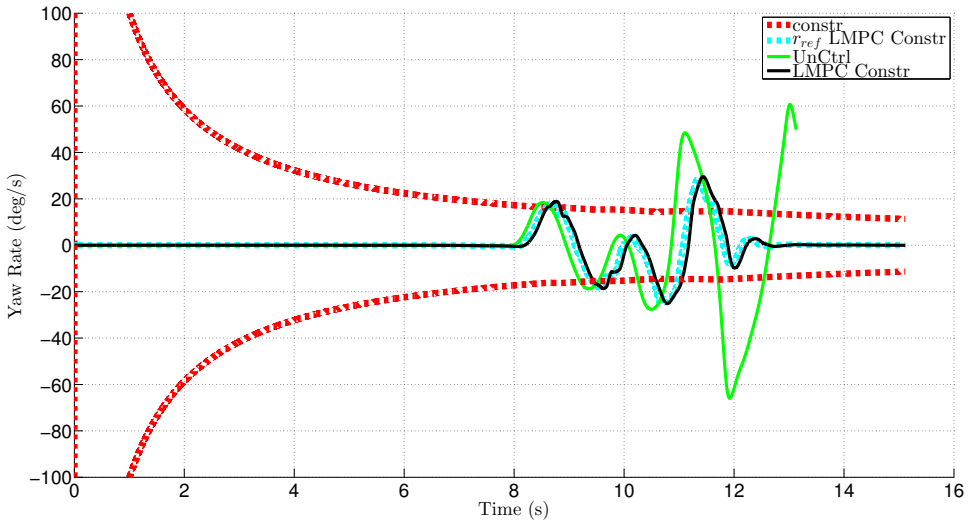

Figure 4.15: Yaw rate, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

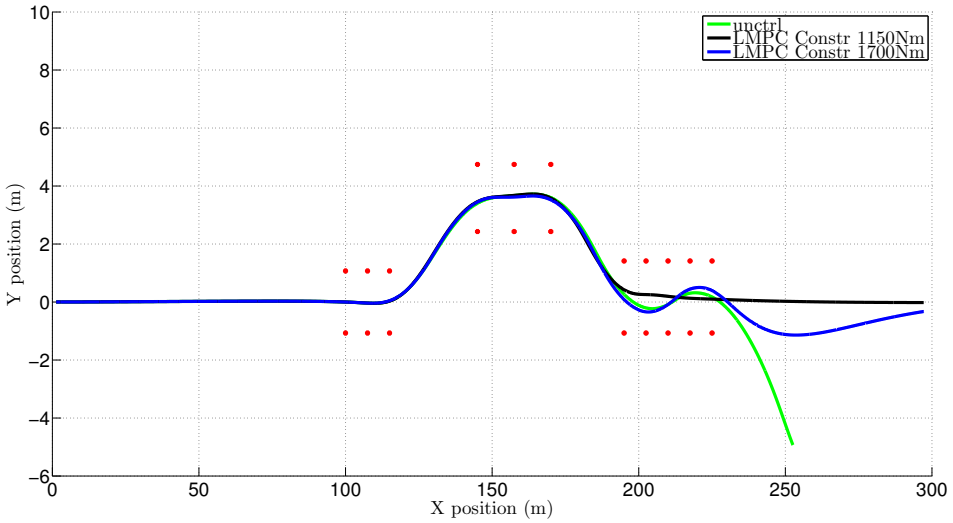

Figure 4.16: Position, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

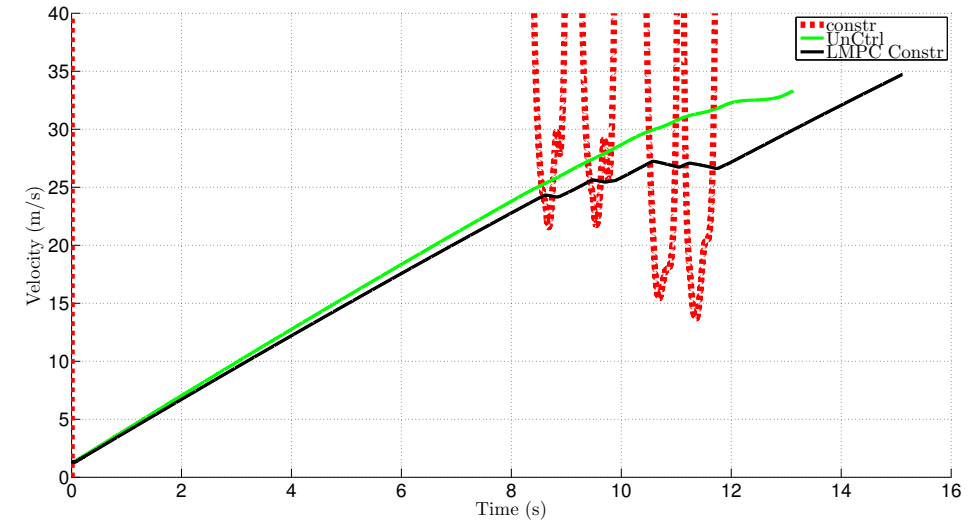

Figure 4.17: Velocity, double lane change at $\mu = 0.9$ and $V_{in} = 8kph$.

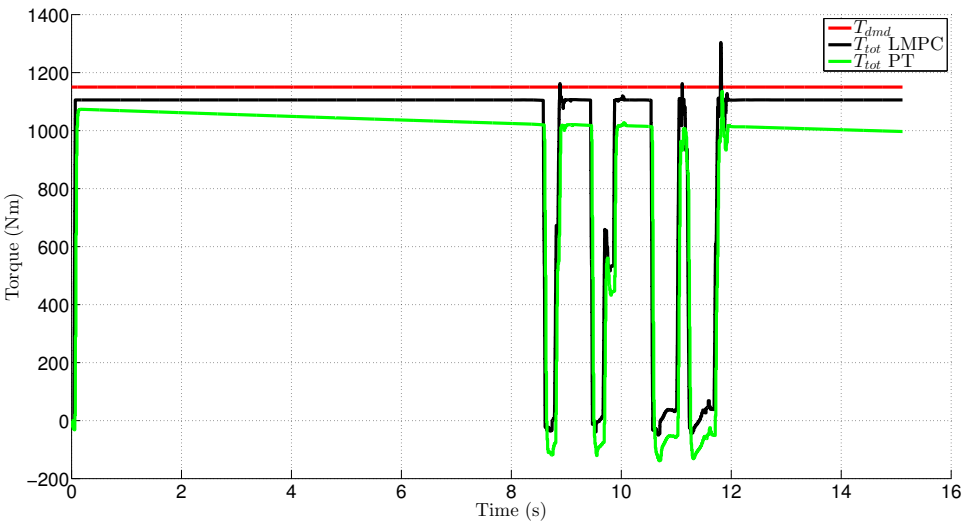

Figure 4.18: Total torque, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

4.2 Computational Performance

The performance of the controller is tested in real time on the laptop computer machine used for the current research. In this section we present some plots showing the performance of the linear controller. The 'solvetime' is the time the controller takes to find a

solution and the 'exitflag' determines whether the solver has exited without an error[1]. In our formulation the number of iterations the solver overtook to find or not a solution, is capped at a maximum number of 300.

In Fig.4.19 we demonstrate the results of the same step-steer maneuver tested in the velocity constraint importance subsection of this chapter. The LMPC solves at around 2ms each sampling time always giving an optimal solution. The number of iterations needed are also quite low making the the controller promising for this particular scenario for real vehicle implementation.

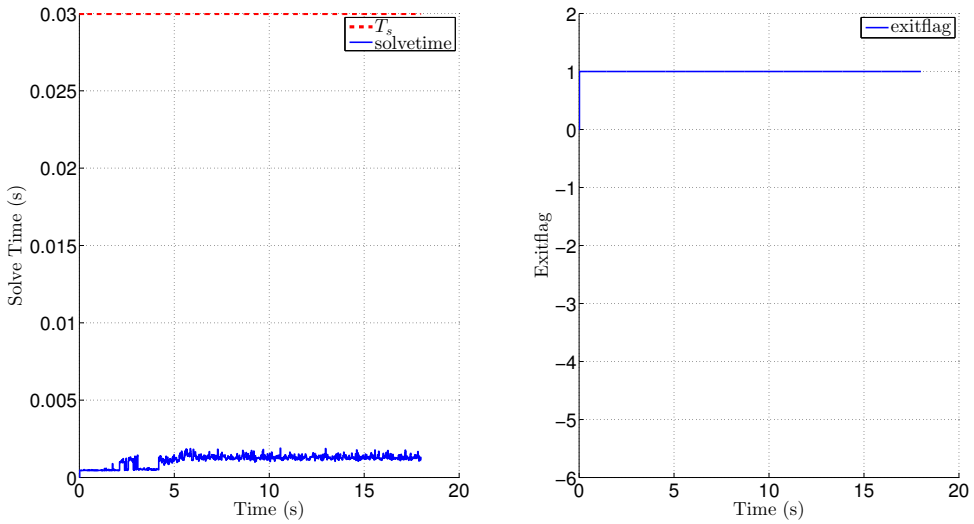

Figure 4.19: LMPC performance, step-teer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

Performance results in the Double Lane Change maneuver are analogous with the above one. The solve time of the controller remains bellow the sampling time in every scenario which means once more that it solves in computer real-time giving optimal solution, without depending on the road friction coefficient change.

[1] 1 - optimal solution found to the requested accuracy; 0 - reached maximum number of iterations or maximum computation time, the returned solution is the best found so far; -6 - NaN or INF occurred during evaluation of functions and derivatives; -7 - The convex solver could not proceed. The problem might be infeasible.

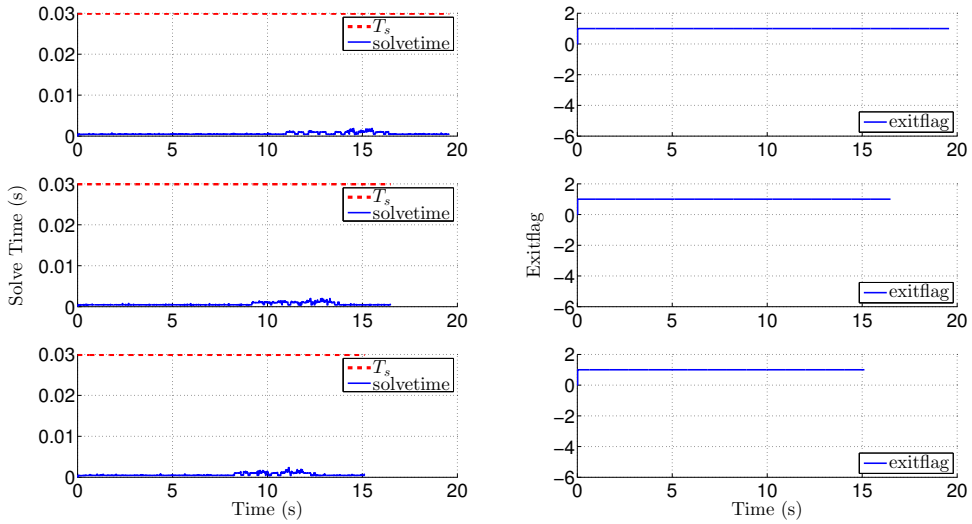

Figure 4.20: LMPC performance, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

4.3 Discussion

In this chapter we have presented the linear MPC strategy for torque vectoring and compared it to an uncontrolled vehicle, which is basically a vehicle with equal torque split on all four wheels, and also to another formulation of linear MPC as found in [30], which has no constrains on yaw rate and velocity but only on the wheel torque control inputs. Both MPC controllers use the same PDIP method to solve the subsequent QP problem.

After comparing the three strategies against each other it was shown that the linear MPC not only tries to respect the yaw rate and velocity constraints but also performs better regarding the yaw rate and velocity response and is a promising control method for real time implementation under a fine tuning. Furthermore, the controller was robust in a range of different torque demands and road friction coefficients tested on a 90° step-steer and a Double Lane Change ISO maneuver proving also the capability of the Torque Vectoring systems.

One final note, is that the chosen sampling time used for the LMPC can be easily

decreased to 10ms as seen from the performance graphs, but the 20ms was the best compromise for later comparison purposes with the nonlinear MPC. The results with a sampling time of 10ms would be slightly more responsive regarding the yaw rate error and thus respecting the velocity constraint quicker with the appropriate torque demand by the controller.

Chapter 5

Nonlinear Torque Vectoring Control Development

5.1 Nonlinear MPC formulation

Nonlinear Model Predictive Control, or NMPC, is a variation of Model Predictive Control (MPC) that is defined by the use of nonlinear models as internal model in the prediction horizon. As in the case of linear MPC, NMPC also requires the iterative solution to be optimal on a finite prediction horizon. While these problems are convex in linear MPC, in nonlinear MPC they are not convex. This poses challenges for both NMPC stability theory and numerical solution [4].

The main difference between the NMPC and LMPC is the internal vehicle model. In the NMPC case we use the nonlinear vehicle dynamics, as opposed to the LMPC

where we use a linearised vehicle model, using the Runge Kutta 4^{th} order as discretisation method for the continuous internal model.

For comparison reasons we used the same cost function and state and input constraints as those on the previous LMPC construction section. Therefore, the optimal control for the nonlinear continuous-time system $\dot{x} = f_c(x,u)$ with states and control inputs x and u respectively is:

$$\min_{x,u} \quad J(x,u), \tag{5.1a}$$

$$\text{subject to:} \quad x_{k+1} = f_d(x_k,u_k), \quad k = 0,1,...,N-1 \tag{5.1b}$$

$$x_k \in \mathscr{X}, \quad k = 1,...,N \tag{5.1c}$$

$$u_k \in \mathscr{U}, \quad k = 0,1,...,N-1 \tag{5.1d}$$

$$x_0 = x_{in}. \tag{5.1e}$$

5.2 Simulation Results

The testing maneuvers used for the NMPC results are identical to those the LMPC was tested too, using the same sampling time and prediction horizon.

5.2.1 Step Steer

In this step-steer maneuver we assume a different constant torque demand by the driver $T_{dmd} = 1200Nm$ and keep the initial velocity at 40kph. The results from the velocity graph in Fig.5.1 show a minor difference on the velocity reduction slope where the linear controller seems to respond faster to the velocity constraint limitation. On the other hand, in Fig.5.2 the yaw rate is respected more often by the nonlinear other than the linear. Furthermore, both controllers' vehicle position paths, Fig.5.3, show significant improvement

not only against the uncontrolled car but also against the unconstrained one as mentioned in the previous section.

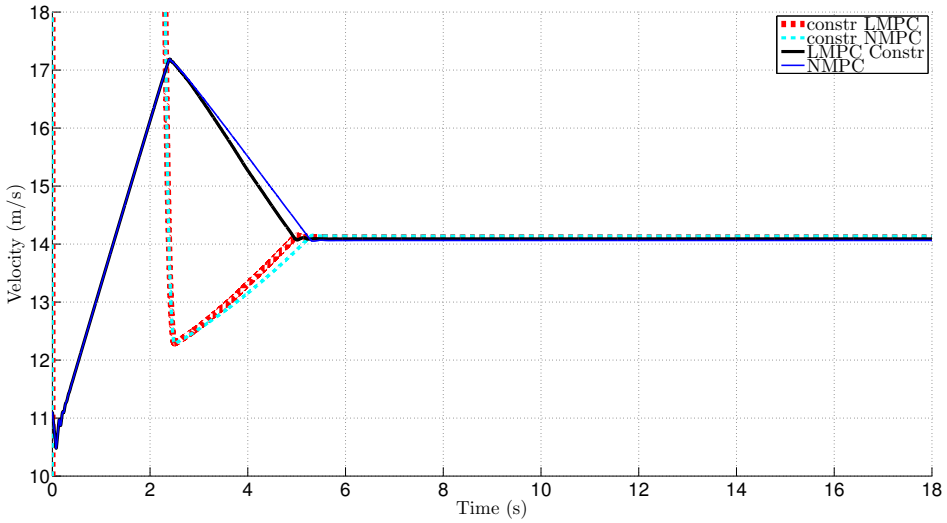

Figure 5.1: Velocity, step-steer at $T_{dmd} = 1200Nm$ and $V_{in} = 40kph$.

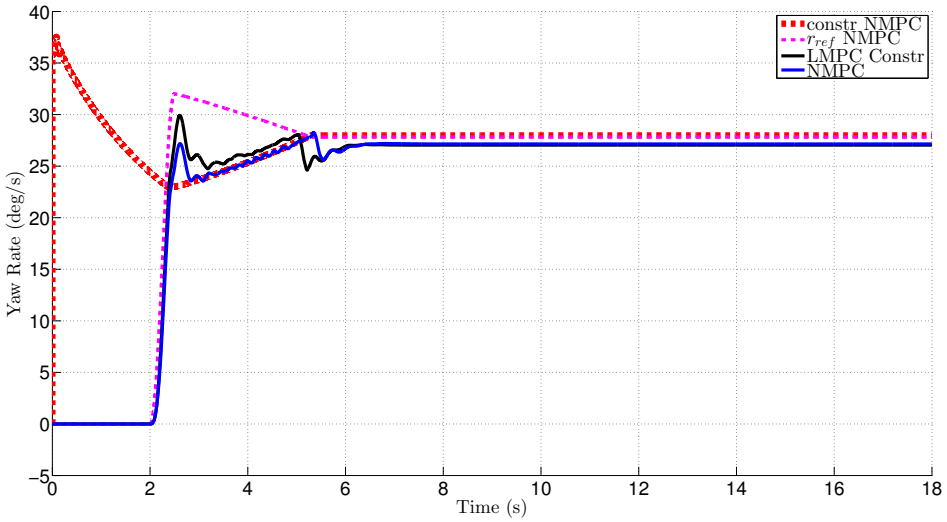

Figure 5.2: Yaw rate, step-steer at $T_{dmd} = 1200Nm$ and $V_{in} = 40kph$.

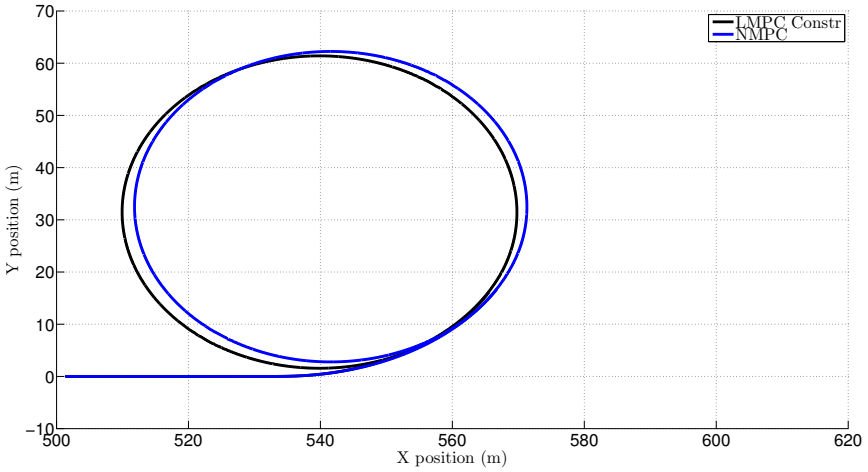

Figure 5.3: Position, step-steer at $T_{dmd} = 1200Nm$ and $V_{in} = 40kph$.

Finally in Fig.5.4 we compare the respectability of the driver's torque demand under both linear and nonlinear MPC formulation. In general the nonlinear controller configuration shows an expected better response than the linear one. This can be justified by the internal vehicle model used for the prediction which is a nonlinear model as opposed to the linearised version used in the LMPC, having the ability of better accuracy.

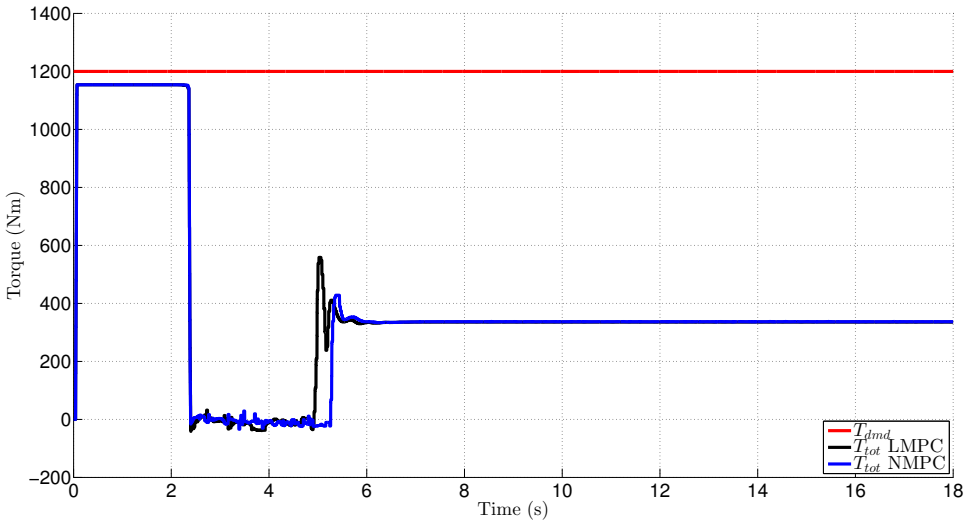

Figure 5.4: Total torque, step-steer at $T_{dmd} = 1200Nm$ and $V_{in} = 40kph$.

5.2.2 Double Lane Change

In the Double Lane Change maneuver we make the same assumptions like in the linear simulation results testing the nonlinear MPC under different road surfaces. In this section we only compare the NMPC with LMPC which has been already compared with both the uncontrolled and unconstrained vehicle before.

5.2.2.1 Double Lane Change $\mu = 0.5$

In low friction surface the nonlinear controller doesn't respond as well as the linear one. This is basically because the tuning process was mainly focused on a compromise between both controllers' performance. Different weight tuning in the cost function might increase the nonlinear controller's computational performance but on the other hand can influence the LMPC's stability on finding an optimal solution.

The velocity as presented in Fig.5.5 follows the same line as the linear case until the vehicle reaches the final lane change where the nonlinear controller has difficulty solving as will be discussed later on the performance subsection. In any case it responds to the velocity constraint all the time. Furthermore, in Fig.5.6 the yaw rate has a similar response with the linear controller but loses a little bit of grip on the last lane change for the same reason the velocity does.

The top-view position comparison between the two controllers is clear from what was mentioned earlier in Fig.5.7. The NMPC struggles to keep the vehicle on track but eventually the driver doesn't lose control of the vehicle compared to the uncontrolled case examined and compared in Chapter 4.

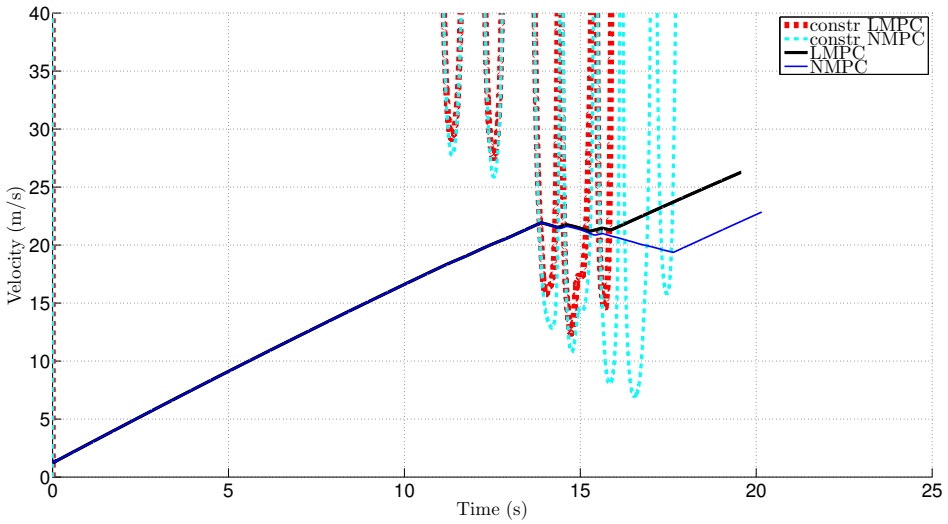

Figure 5.5: Velocity, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

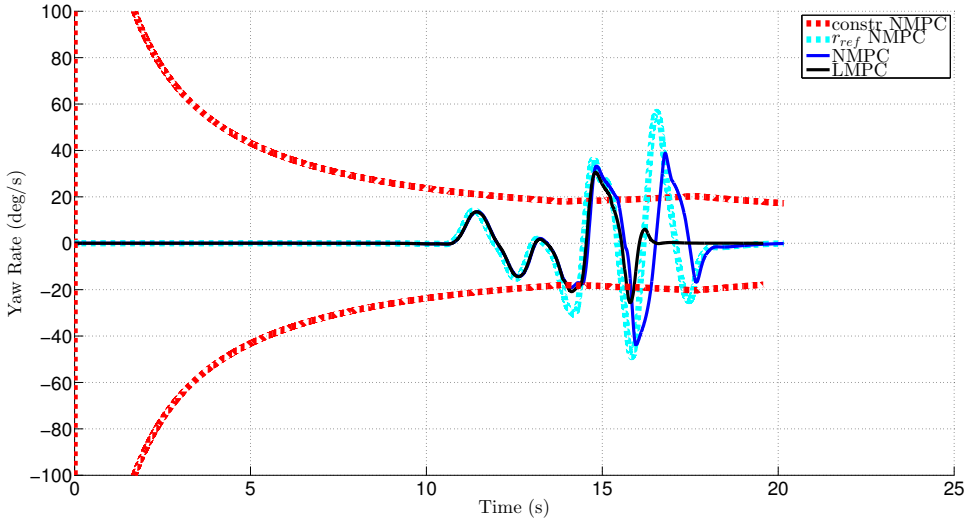

Figure 5.6: Yaw rate, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

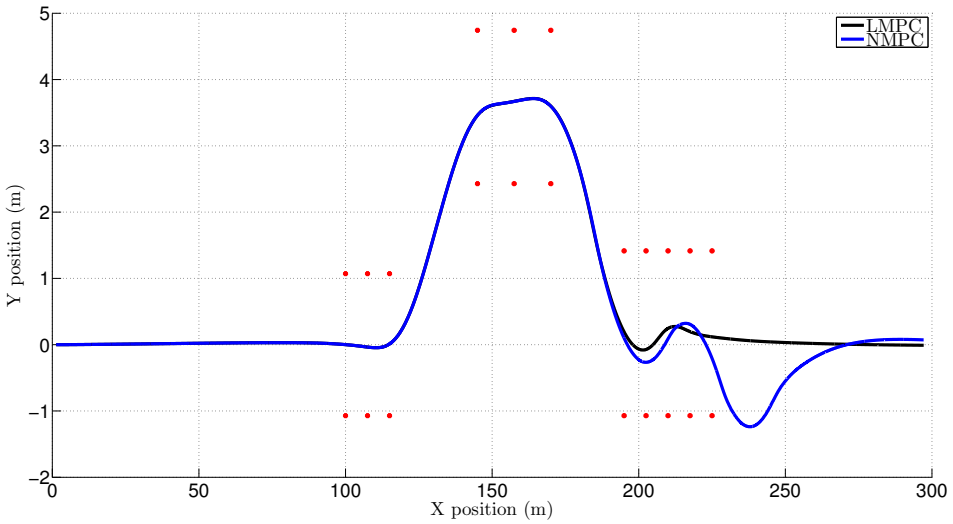

Figure 5.7: Position, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

The total torque of the nonlinear controller compared to linear one is almost identical with minor difference while braking on the first lane change as presented in Fig.5.8. On the second lane change after the 15^{th} second the nonlinear controller doesn't find an optimal solution on the current prediction horizon thus has a much different response from the LMPC. The reason behind this issue is because the NMPC was tuned based on the LMPC weight matrices for comparison purposes. Different tuning weight parameters in the cost function of the NMPC formulation can give a lower cost function closer to zero making the solver find an optimal solution faster.

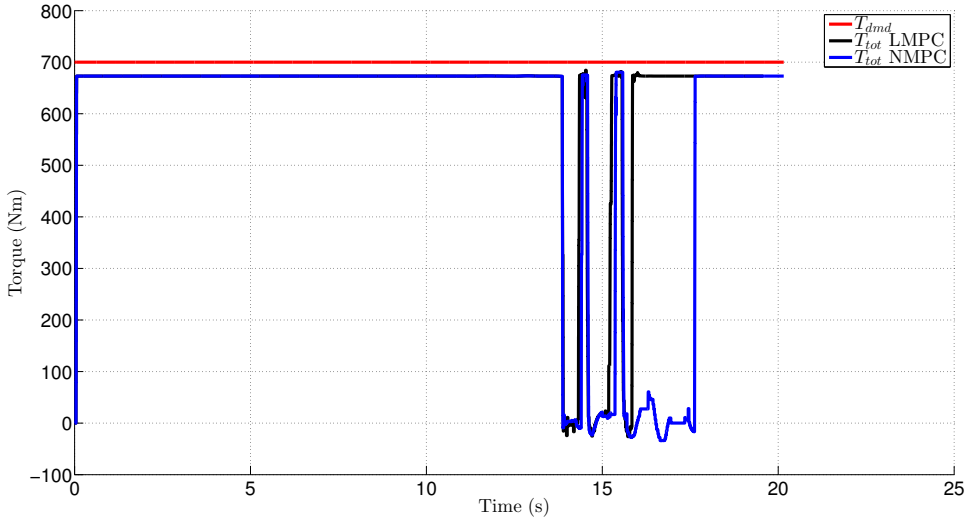

Figure 5.8: Total torque, double lane change at $\mu = 0.5$, $T_{dmd} = 700Nm$ and $V_{in} = 8kph$.

5.2.2.2 Double Lane Change $\mu = 0.7$

Matching the road friction coefficient to the internal vehicle model of the controller has more promising results for the nonlinear than before. Taking a quick look at both the velocity and yaw rate graphs, Fig.5.9 and Fig.5.10 respectively, we notice a quicker velocity regulation on nonlinear formulation which on one hand results on slower exit velocity of the vehicle but on the other hand the yaw rate response is quicker after the final lane change than the linear controller.

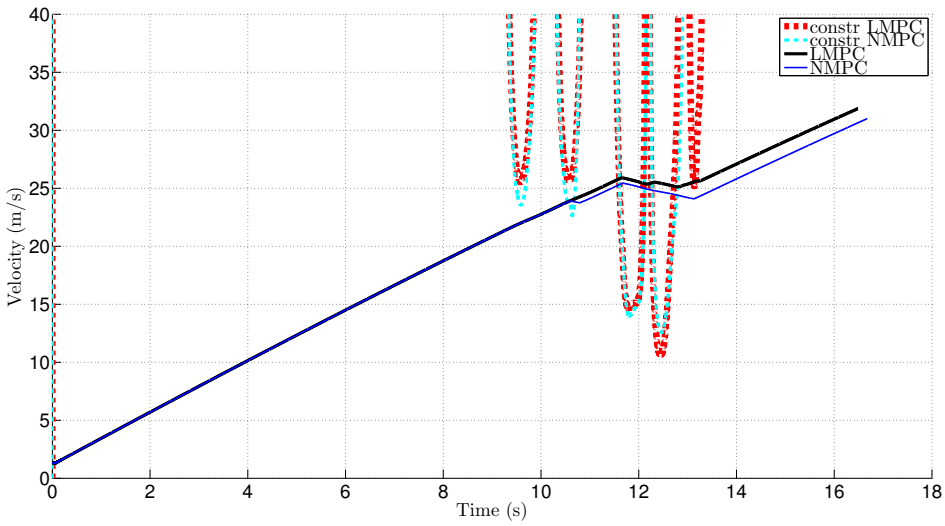

Figure 5.9: Velocity, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

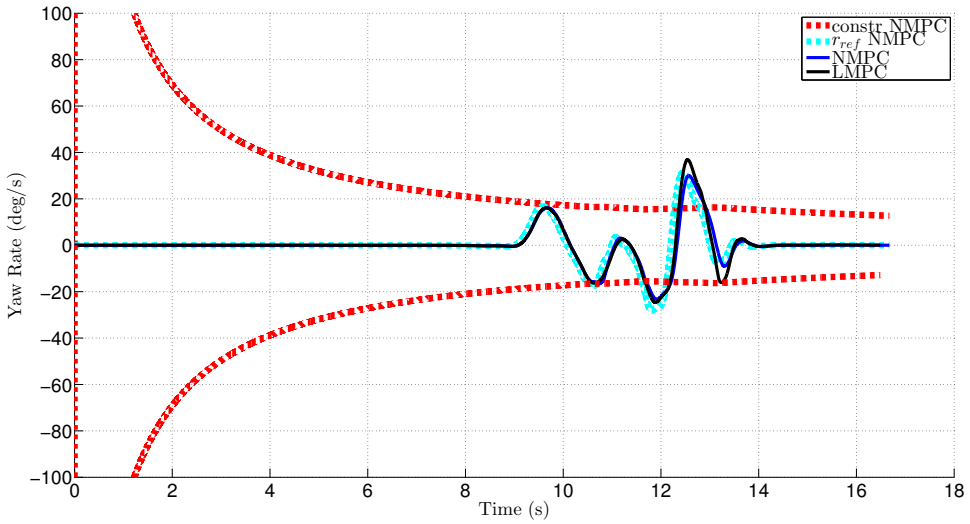

Figure 5.10: Yaw rate, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

The above observation is also detectable clearly in the position comparison Fig.5.11 where the blue line which corresponds to the NMPC has a better grip than the linear control scheme.

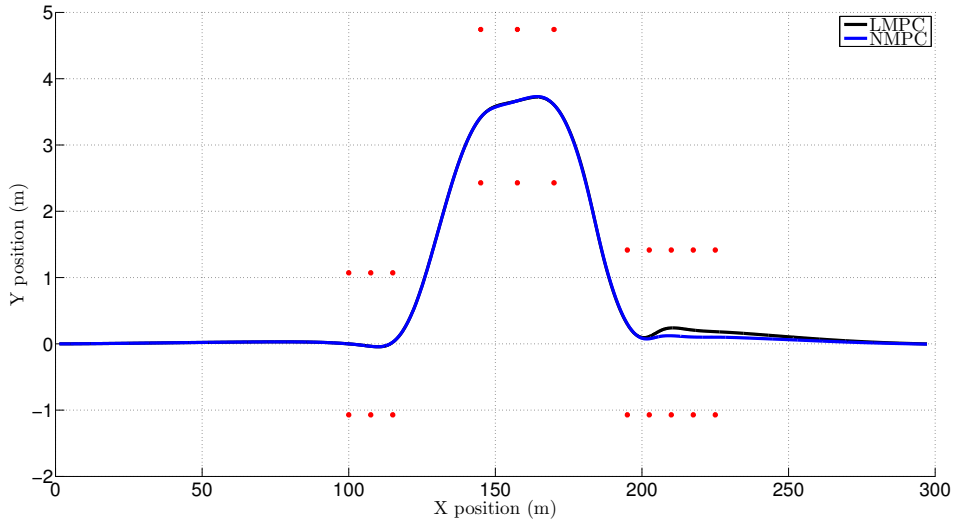

Figure 5.11: Position, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

Finally, getting a closer look at the torque demand graph, Fig.5.12, we notice a re-

sponse difference between the nonlinear and the linear controller. The nonlinear controller has more often less total torque but that gives a quicker response to the driver to react by adjusting the steering wheel. One more thing to point out is that the nonlinear internal model respects the torque demand by the driver throughout the maneuver in contrast with the linearised model which once more exceeds it at some point depending on the cost function value the controller has to minimise. The torque spikes are result of the torque vectoring penalization in the cost function. Depending on the weight factors the controller adjusts the wheel torques accordingly so that it can minimise the whole cost function in the chosen prediction horizon.

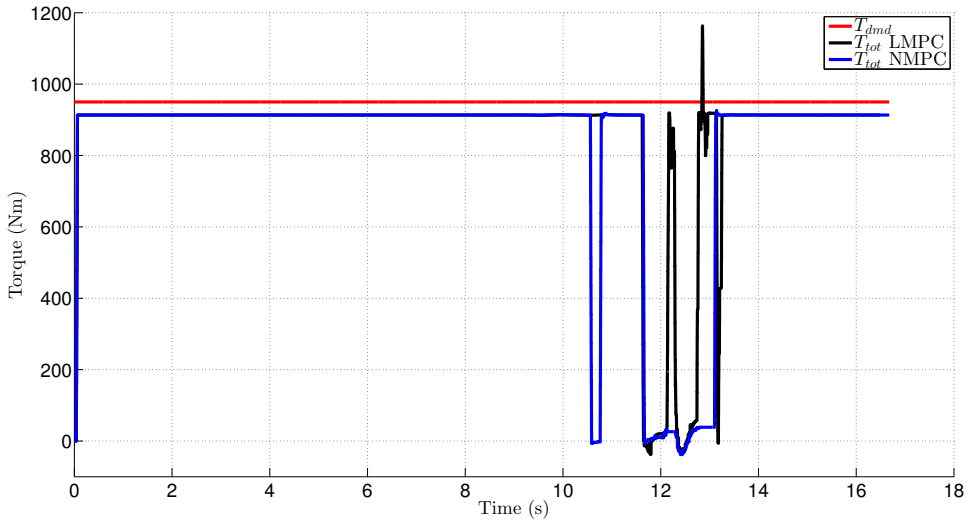

Figure 5.12: Total torque, double lane change at $\mu = 0.7$, $T_{dmd} = 950Nm$ and $V_{in} = 8kph$.

5.2.2.3 Double Lane Change $\mu = 0.9$

The results of the nonlinear MPC formulation are more visible in the case where the road friction coefficient is increased to $\mu = 0.9$. The solver has also difficulty solving as in the case of $\mu = 0.7$ but this time the comparison is distinct between the controllers as

the driver's torque demand increases, with the nonlinear gaining advantage. The velocity graph in Fig.5.13 has a similar comparison response as the Fig.4.13 where it follows the linear for most of the maneuver but losing track in the end of the last lane change for almost 1 second and finally accelerating after exiting. Similar to the velocity graph, the yaw rate graph in Fig.5.14 shows slightly better response of the NMPC compared to the LMPC after the first lane change.

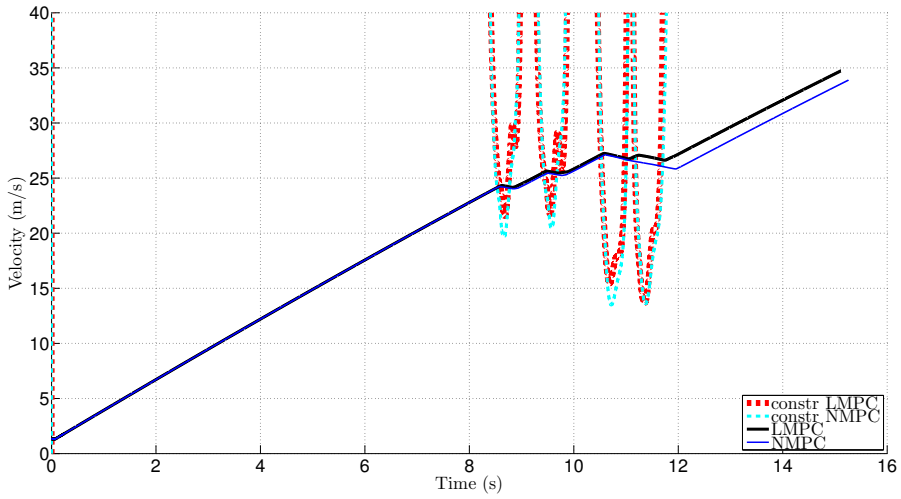

Figure 5.13: Velocity, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

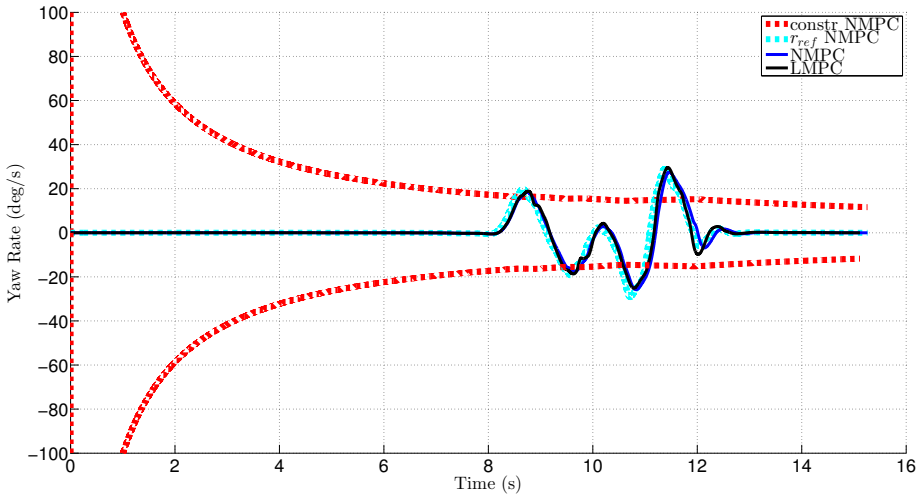

Figure 5.14: Yaw rate, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

Looking at Fig.5.15 we notice a quicker return to the straight line of the nonlinear controller as opposed to the linear as a result of what was mentioned above. Finally, in Fig.5.16 the total torque demanded by the NMPC obeys the driver's torque demand throughout the maneuver and never exceeds it, whereas the LMPC exceeds it even more than in previous scenarios. This happens mainly because of the increase of the T_{dmd} and its weighting factor which remains constant regardless the driver's torque command.

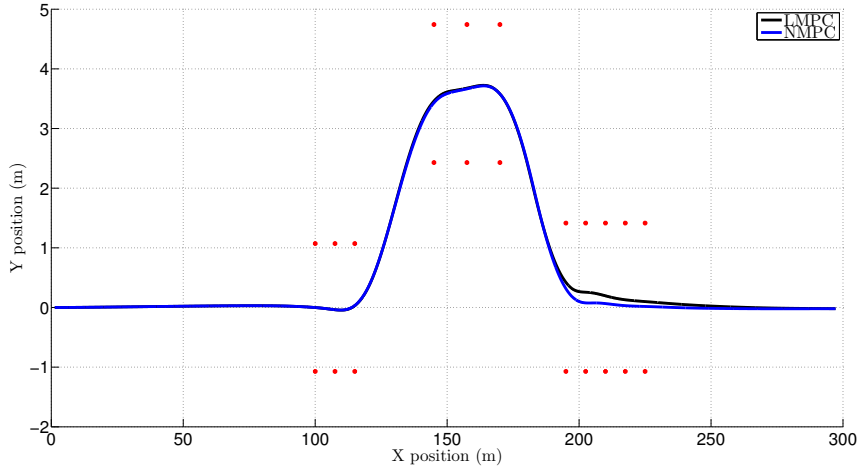

Figure 5.15: Position, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

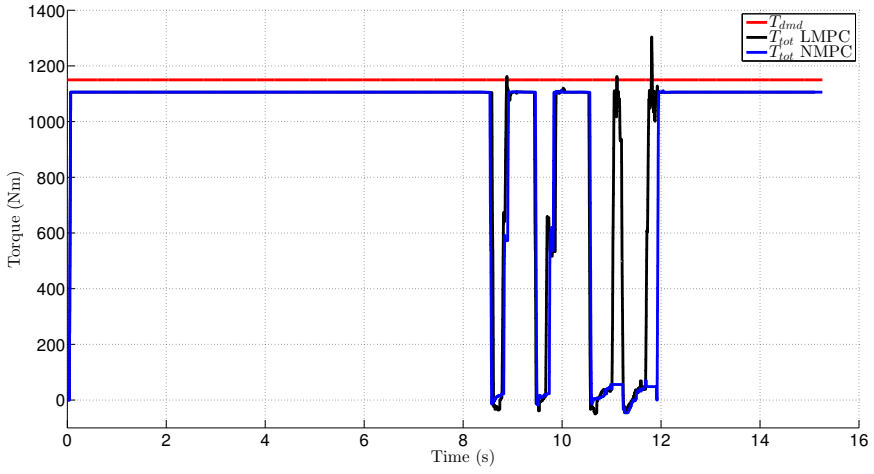

Figure 5.16: Total torque, double lane change at $\mu = 0.9$, $T_{dmd} = 1150Nm$ and $V_{in} = 8kph$.

5.3 Torque rate expansion

After having a complete look at both LMPC and NMPC we decided to take the NMPC a step further changing its formulation from wheel torques to wheel torque rates as control inputs of the controller in order to include the motor's torque rate constraints (see Table 3.1).

5.3.1 Step Steer Results

From a first glance, one can notice the smoothness of increment and decrement on the velocity over acceleration and deceleration respectively, opposed to the NMPC with wheel torque inputs in Fig.5.17. Both controllers meet the steady-state almost at the same time and although the nonlinear with the torque rate implementation hits the constraint at a lower speed and has one oscillation before reaching it.

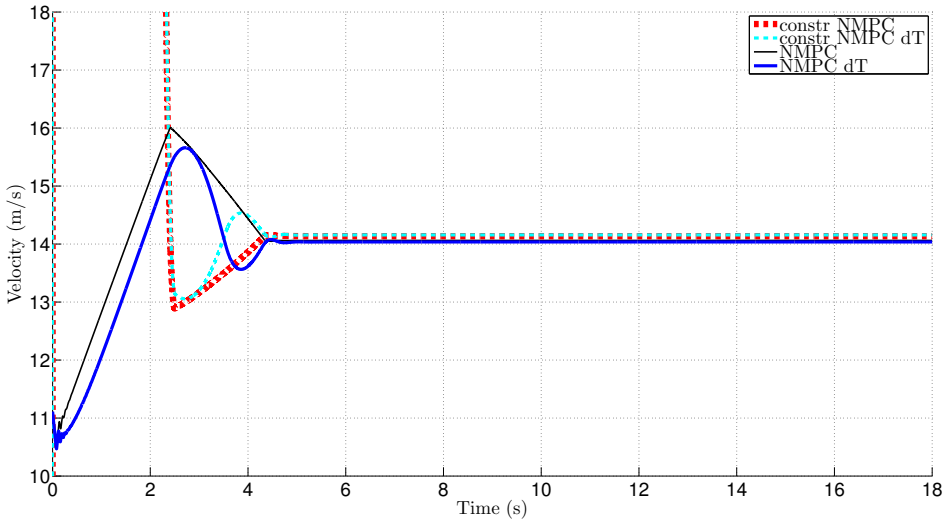

Figure 5.17: Velocity, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

The yaw rate presented in Fig.5.18 has a comparable response with the velocity graph because of the relation between those two variables. Even though the NMPC with torque

rate control inputs has a wide oscillation around the 4^{th} second, reaches the steady-state yaw rate quicker than the wheel torque control input nonlinear formulation. The top-view position in Fig.5.19 of the vehicle's centre of mass shows very similar results on both controllers with slight error between them.

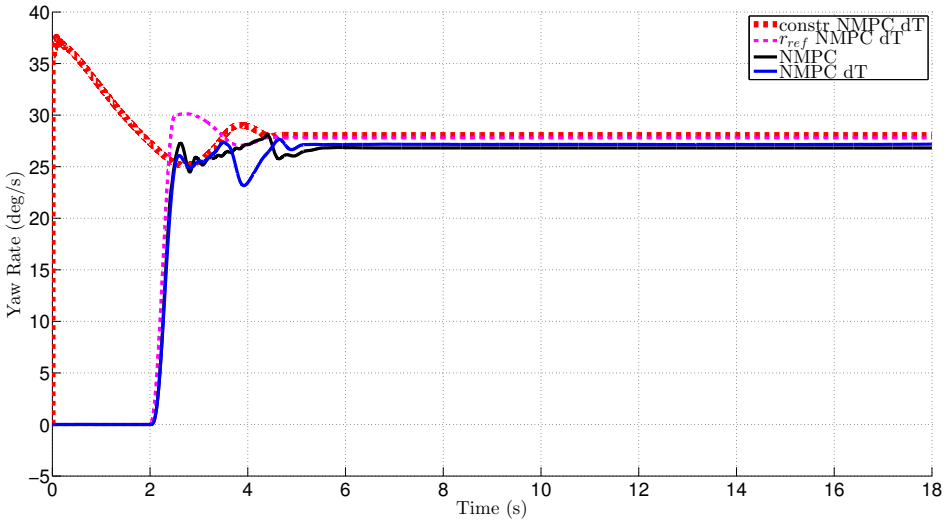

Figure 5.18: Yaw rate, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

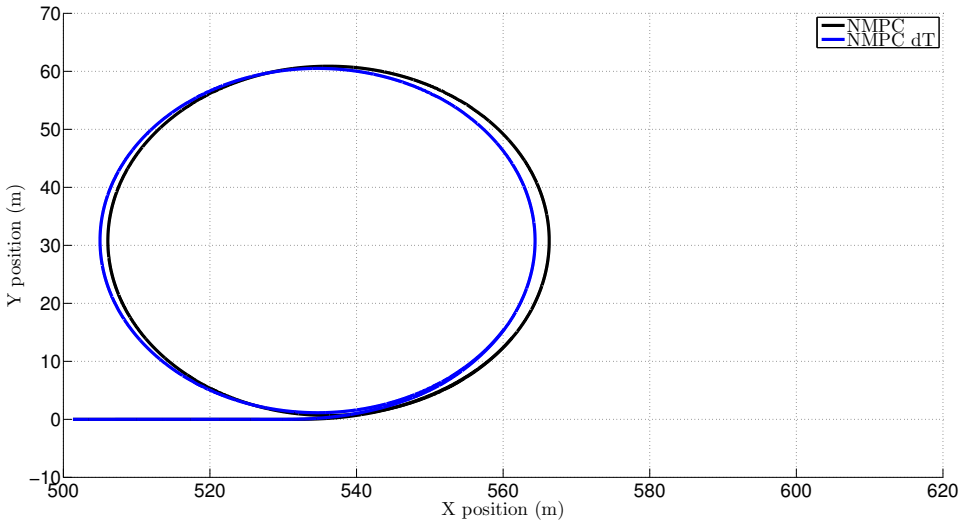

Figure 5.19: Position, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

Finally, the biggest change is observed in the total torque graph in Fig.5.20 where the difference between the two controllers' formulation is distinct. In that mater, the expanded controller (blue line), shows a significant brake intervention while decelerating in order to meet the velocity constraint which refers to the case of active trail-braking. During the acceleration and deceleration the torque change is limited not only due to the torque rate control input constraints but also due to their selected weights in order to satisfy our controller's purpose. An important note is that the weighting parameters chosen for the NMPC with torque rates have been changed because the whole problem has also been changed. For a closer look at the weights see Appendix.

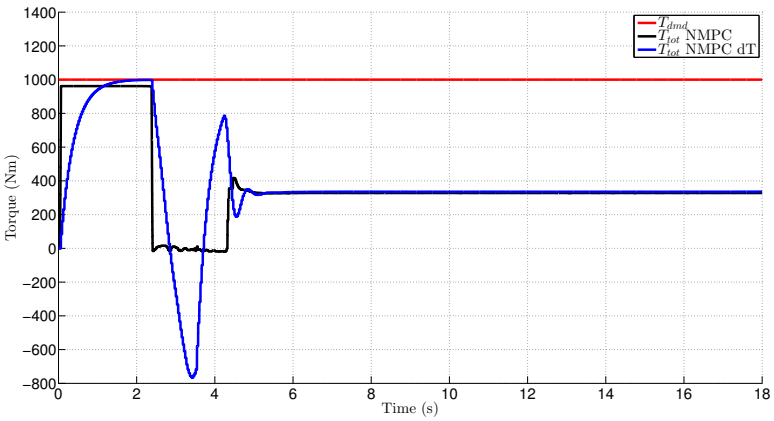

Figure 5.20: Total torque, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$.

5.3.2 Double Lane Change Results

The nonlinear torque rate control input expansion strategy is also tested in a Double Lane Change maneuver. We define a variety of different road friction coefficients similar to the previous simulations with an initial velocity $V_{in} = 8kph$. In Fig.5.21 the same velocity characteristics are seen as mentioned on the step-steer maneuver, where it has a smoother increase and decrease during acceleration and braking and whenever it hits the velocity constraint immediately starts to adjust.

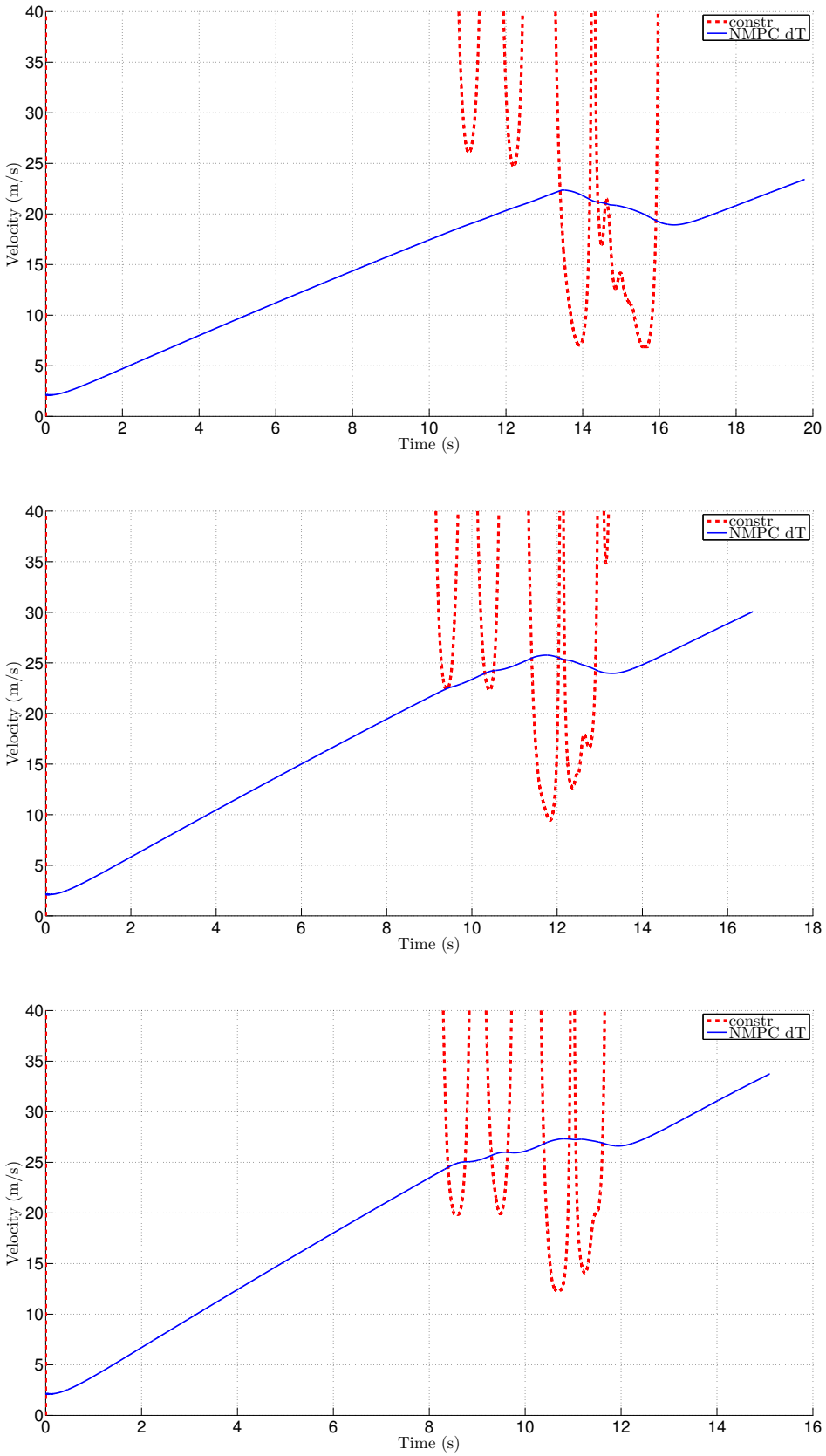

Figure 5.21: Velocity, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

As seen in the yaw rate response graphs in Fig.5.22 , on low road $\mu = 0.5$ the vehicle loses grip at the end of the final lane change whereas in higher road friction coefficients responds as close to the reference signal while at the same time trying to respect the yaw rate constraint. As mentioned again, both yaw rate and velocity can exceed their constraints in order the solver to find an optimal solution, because the constraints have been softened.

Having a closer look at Fig.5.23 the position of the vehicle's CoM in low friction surface goes sideways and hits a red cone on the exit of the final lane change but soon the driver regains control. In all three maneuvers the vehicle follows the same path until the driver reaches the end of the first lane change and starts the last one, at the point where the blue line meets the horizontal line Y=1. As the friction coefficient between the road and the tyre increases, it is easier for the driver to follow the route as seen in the middle and bottom plots of the figure. Of course all these results are taken considering the driver's characteristics described in Chapter 4.

Finally, the total torque demanded by the controller is examined in Fig.5.24 where a curved increment on the acceleration is noticed just like on the step-steer scenario. On the other hand, the deceleration has a more steep slope but compared to the wheel toque control input case is less. On $\mu = 0.5$ the controller demands more braking torque than both the other two road friction scenarios and on high friction $\mu = 0.9$ the demanded torque is decreased, after the end of the double lane change where the driver accelerates in a straight line, because of the safety protection block and its torque, voltage and current limitations.

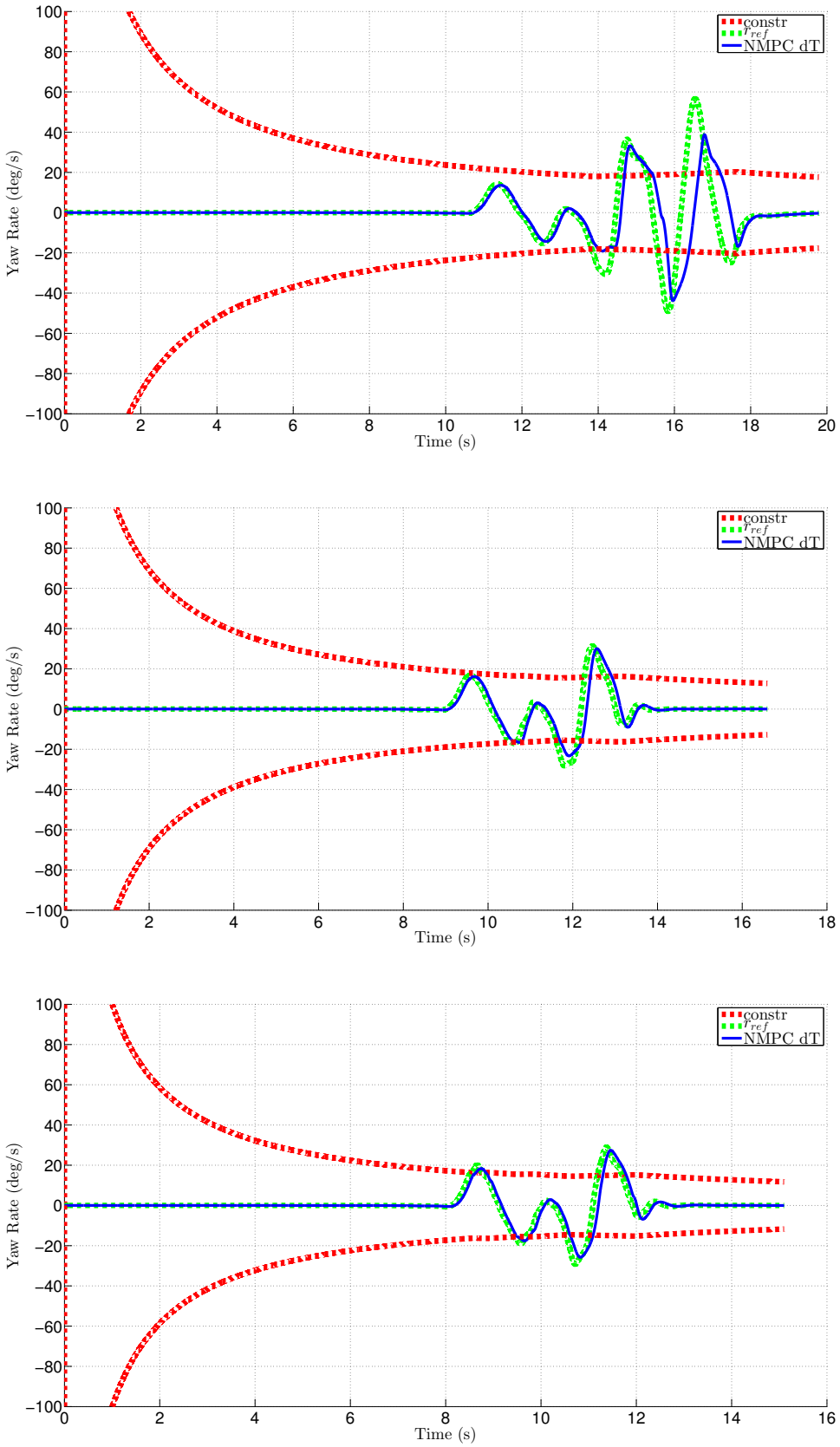

Figure 5.22: Yaw rate, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

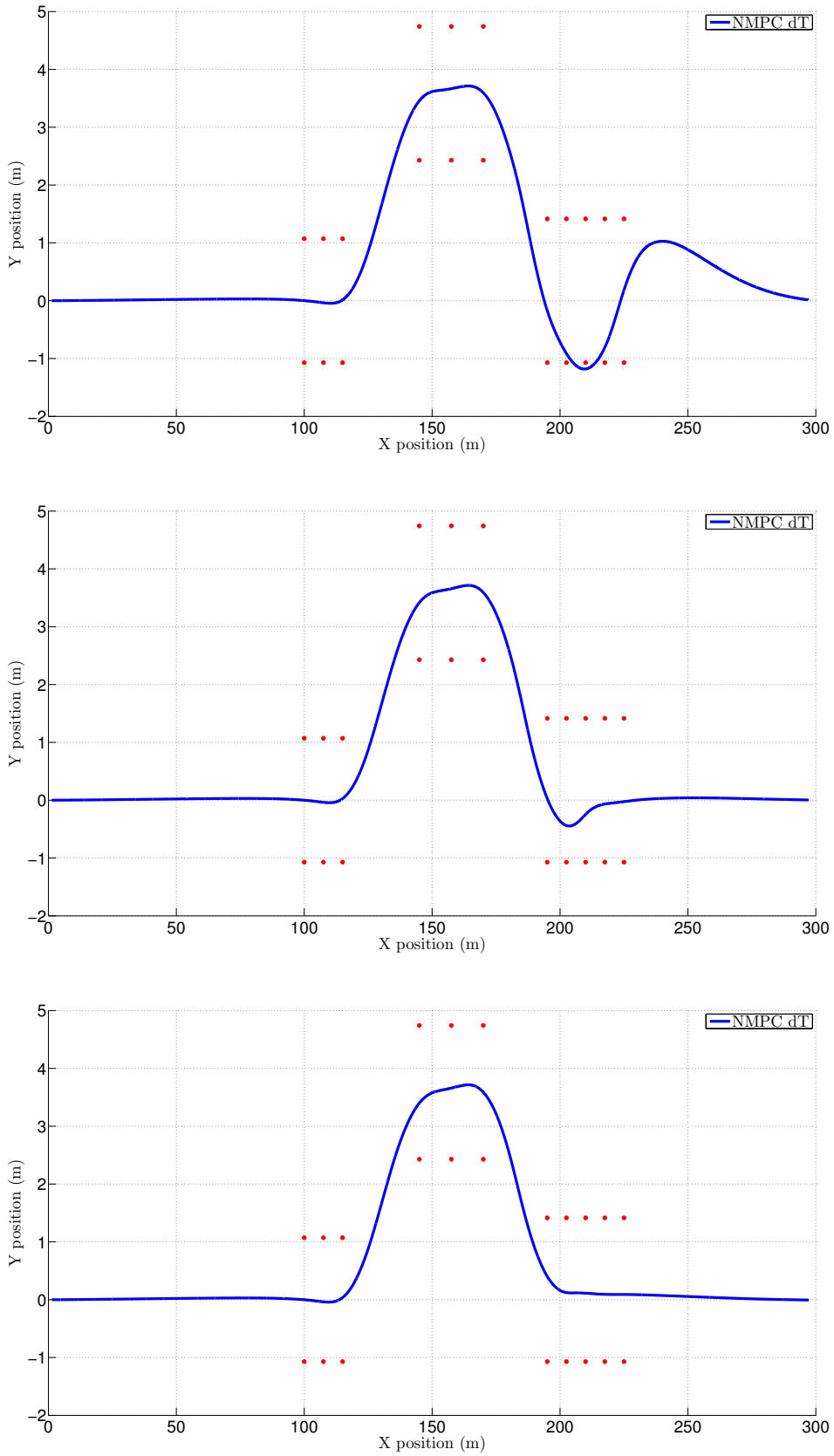

Figure 5.23: Position, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

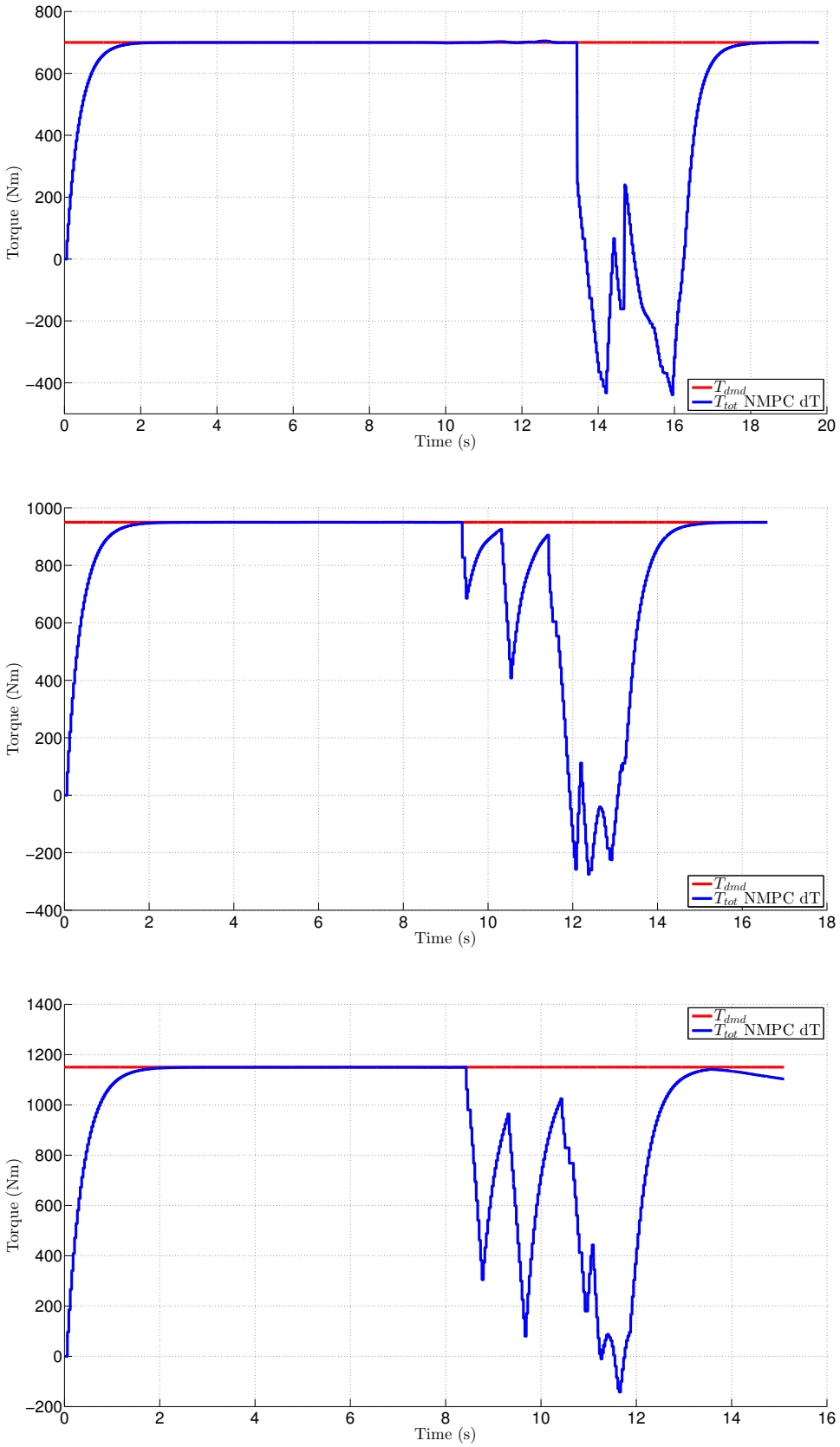

Figure 5.24: Total torque, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

5.4 Computational Performance

The computational time performance of both NMPC control strategies is larger than that of the linear strategy, although they all run in real-time on the laptop computer machine this research has been implemented. As seen in Fig.5.25, under the step-steer scenario the torque rate NMPC always solves optimal with an average time of 11ms compared to the wheel torque control input formulation which has a smaller average solve time of 5ms but it's more solve sensitive as the exitflag can reach a value of -6 in such low demanding cases.

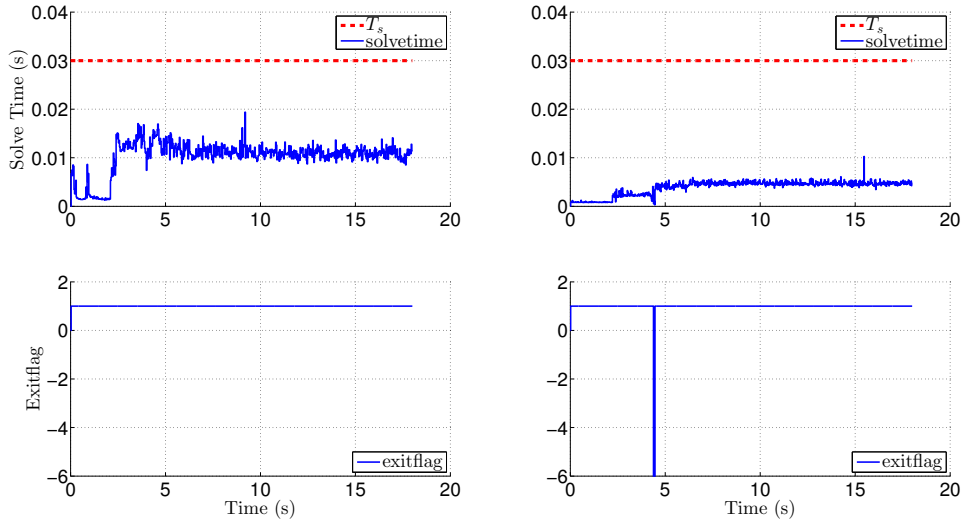

Figure 5.25: NMPC performance, step-steer at $T_{dmd} = 1000Nm$ and $V_{in} = 40kph$; Left images - torque rates NMPC; Right images - wheel torques NMPC.

In the Double Lane Change ISO, there some points of interest emphasized. The NMPC in Fig.5.26 shows a low average solve time but occasionally hits high values close to the sampling time. This usually occurs when the exitflag equals to zero, although in the tested scenarios we have safely chosen a higher safe sampling time so that the controller can solve in real time. Furthermore, it is easily observed that, in all road friction surfaces, the solver every now and then has trouble finding an optimal solution, where the exitflag

is equal to -6. In that case we take the previous optimal solution and apply it to the current state.

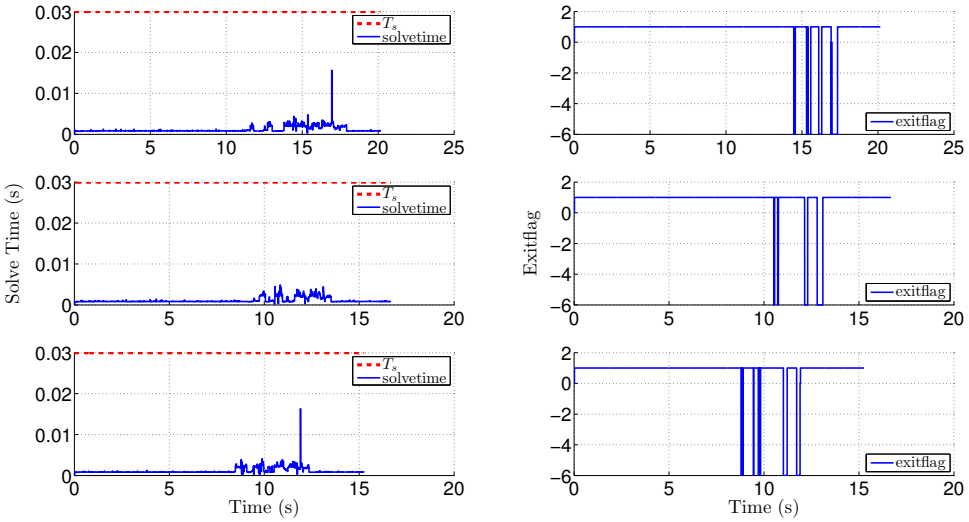

Figure 5.26: NMPC performance, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

The computational performance of the torque rate nonlinear formulation in Fig.5.27 has a greater solve time than the previous NMPC but finds an optimal solution more often especially while the road friction coefficient increases. An important note, also mentioned before, is that this controller has a different problem to solve thus its cost function formulation but also the weight parameters are different than the previous controllers tested and compared. Along these lines, it is hard to make a good comparison between the two NMPC strategies.

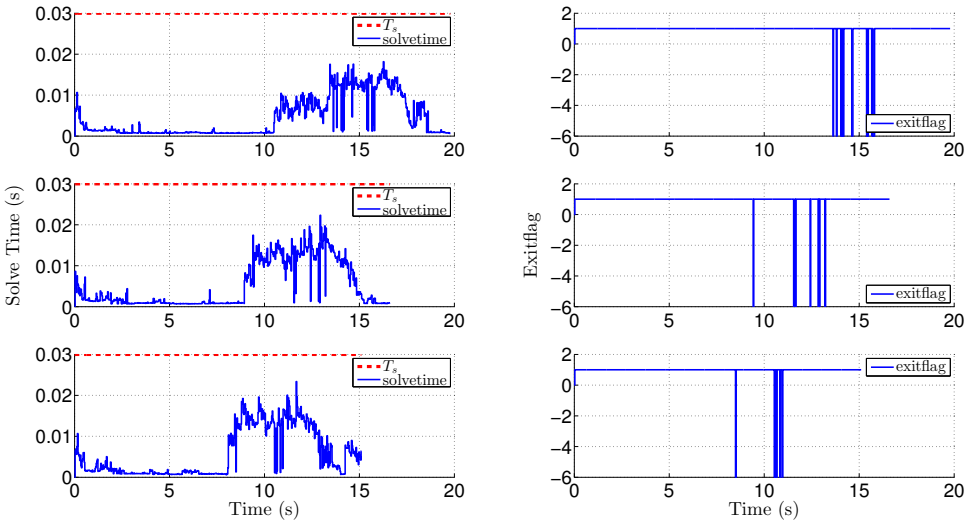

Figure 5.27: NMPC dT performance, double lane change; top - $\mu = 0.5$, $T_{dmd} = 700Nm$; middle - $\mu = 0.7$, $T_{dmd} = 950Nm$; bottom - $\mu = 0.9$, $T_{dmd} = 1150Nm$.

5.5 Discussion

The nonlinear strategy implemented in this section is tuned under the same weight parameter values of the linear MPC for comparison reasons. Under the step-steer maneuver both linear and nonlinear behave alike compared on the velocity regulation approach. However, in the Double Lane change scenario, the linear MPC shows better results on lower tyre-road friction than the nonlinear. That happens because the nonlinear controller lacks the ability to solve optimal given these weight parameters. The results of the NMPC improve when we increase the surface friction not only on a vehicle's trajectory perspective but also on the yaw rate response, velocity regulation and torque demand approach.

Finally taking the nonlinear control scheme a step further, we change the formulation to torque rate control inputs instead of the wheel torques used previously making the controller to take into account the torque change capabilities of the electric motor. The

results presented were also promising emphasizing the smooth total torque change of the vehicle compared to all the other control allocations. Both nonlinear controllers have similar response on yaw rate and trajectory but the torque rate NMPC takes much more computational time due to the increased amount of constraints. The wheel torque NMPC computational times are bigger than the linear version but they still remain under the sampling time making it also feasible for real-time implementation.

Chapter 6

Conclusions and Further Research

6.1 Concluding Remarks

In this research the fundamental aim was to investigate different predictive control approaches combining the classic torque vectoring technique with a velocity regulation approach during both limit and sublimit handling cases. All the developed optimal controllers are implementable in real-time on a laptop computer machine (CPU i7-4710HQ at 2.50GHz and 16GB of RAM). To solve our problem we used both linear and nonlinear control strategies and in order to compare both strategies we have made tuning performance compromises on both sides.

The linear torque vectoring formulation shows significant improvement over the equal torque split vehicle not only under constant steering angle but also using a high fidelity driver model through CarMaker. Furthermore, the importance of a velocity regulation

is assessed by adding a velocity constraint in the control strategy, showing additional improvement to the drivability of the vehicle. The driver's torque demand is not being neglected, on the contrary the controller meets that torque demand whenever possible while at the same time not affecting its ability to respect both the yaw rate reference generated and the velocity constraint. Finally, the linear controller shows very promising real-time computational results on the current hardware making it feasible for real-time vehicle implementation research.

The nonlinear strategy implemented shares some great results too, tuned under the same weight parameter values of the linear MPC for comparison reasons. Although the linear MPC has better results on lower tyre-road friction than the nonlinear, it is mainly because the nonlinear controller lacks the ability to solve optimal given these weight parameters. The results of the NMPC improve when we increase the surface friction not only on a vehicle's trajectory perspective but also on the yaw rate response, velocity regulation and torque demand approach. Its computational times are bigger than the linear version but they still remain under the sampling time making it also feasible for further real-time implementation research. Finally taking the nonlinear control scheme a step further, we change the formulation to torque rate control inputs instead of the wheel torques used previously making the controller to take into account the torque change capabilities of the electric motor. The results presented were also promising emphasising on the smooth total torque change of the vehicle compared to all the other control allocations. Both nonlinear controllers have similar response on yaw rate and trajectory but the torque rate NMPC takes much more computational time due to the increased amount of constraints. A further tuning investigation on both controllers will improve their ability to solve optimal and focus on the real-time deployment research.

6.2 Future Work

This research studied the integration of a torque vectoring technique with velocity regulation using both linear and nonlinear Model Predictive Control, in real-time through simulations computed in Simulink and CarMaker environments. Some of the most important future works are:

- better tuning of both linear and nonlinear strategies to expand the increase the number of optimal solutions under different testing maneuvers
- implementation of the current linear/nonlinear controller on a real-time dSPACE deployment machine which has less computational power than the current computer
- change the current linear formulation to torque rate control inputs and compare the results
- testing and validation of the controller using hardware-in-the-loop test rig and further implementation on a real electric test vehicle
- implementation of a tyre-road friction coefficient estimation which was assumed constant for the MPC strategies and in real vehicle implementation, there is no measurement available for this parameter
- include electric motor dynamics in the internal vehicle model and make a systematic sensitivity analysis of the controller's weight matrices

Appendix A

Tuning Parameters

All the values used for the weights where chosen after numerous simulations. The tuning of all our controllers is not constant but instead the parameters change value according to the vehicle's velocity. There are two main cases, the underspeeding case where the vehicle enters a turn with less speed than the V_{lim} as calculated in Chapter 3, and the overspeeding case where the vehicle has excessive speed and the controller needs to take action. The steady-state case is considered inside the underspeeding case.

The main body of the tuning algorithm for both LMPC and NMPC with wheel torque control inputs is:

```
if V_lim >= V %underspeeding case
  qr = 1/(1)^2;
  qr_e = 0.1;
  qV_e = 0.1;
  qTdmd = 8/(4*Tm_max)^2;
  qT = 0.08/(Tm_max)^2;
  if ((abs(r_ref - r) <= 5*pi/180.0) && (V_lim - V < 1))
     || hold == 1 % vehicle in steady-state
```

```
        qr = (-560*Kund+1.033)/(1)^2;%address the variety
            of understeer gradient
        qr_e = 4000;
        qV_e = 400000;
        qTdmd = 0.28/(4*Tm_max)^2;
        qT = 0.08/(Tm_max)^2;
        hold = 1;
        if (V_lim - V > 5) %unhold steady-state
                hold = 0;
        end
  end
else %overspeeding case
  qr = 0.08;
  qr_e = 0.1;
  qV_e = 0.1;
  qTdmd = 0.1/(4*Tm_max)^2;
  qT = 0.04/(Tm_max)^2;
  hold = 0;
end
```

The parameters have changed for the NMPC with torque rate control inputs into:

```
if V_lim >= V
  qr = 2/(1)^2;
  qr_e = 1000;
  qV_e = 100000;
  qTdmd = 2/(4*Tm_max)^2;
```

```
  qdT = 1/(dT_lim)^2;
  if ((abs(r_ref - r) <= 3*pi/180.0) && (V_lim - V < 1))
     || hold == 1
    qr = 4*(-560*Kund+1.033)/(1)^2;
    qr_e = 4000;
    qV_e = 400000;
    qTdmd = 0.6/(4*Tm_max)^2;
    qdT = 0.28/(dT_lim)^2;
    hold = 1;
    if (V_lim - V > 3)
      hold = 0;
    end
  end
else
  qr = 1/(1)^2;
  qr_e = 3;
  qV_e = 1;
  qTdmd = 2/(4*Tm_max)^2;
  qdT = 4/(dT_lim)^2;
  hold = 0;
end
```

References

[1] Delta Motorsport. (http://www.delta-motorsport.com/).

[2] IPG CarMaker simulation software. (https://ipg-automotive.com/products-services/simulation-software/carmaker/).

[3] E-VECTOORC project. (http://www.e-vectoorc.eu/), 2011-2014.

[4] Frank Allgöwer and Alex Zheng. *Nonlinear model predictive control*, volume 26. Birkhäuser, 2012.

[5] Osvaldo Barbarisi, Giovanni Palmieri, Stefano Scala, and Luigi Glielmo. LTV-MPC for Yaw Rate Control and Side Slip Control with Dynamically Constrained Differential Braking. *European Journal of Control*, 15(3-4):468–479, 2009.

[6] S. C. Baslamisli, I. Polat, and I. E. Kose. Gain Scheduled Active Steering Control Based on a Parametric Bicycle Model. *2007 IEEE Intelligent Vehicles Symposium*, 2(1):1168–1173, 2007.

[7] M. Sofian Basrah, Efstathios Siampis, Efstathios Velenis, Dongpu Cao, and Stefano Longo. Wheel slip control with torque blending using linear and nonlinear model predictive control. *Vehicle System Dynamics*, 55(11):1665–1685, 2017.

[8] D. Bianchi, A. Borri, M. D. Di Benedetto, S. Di Gennaro, and G. Burgio. Adaptive integrated vehicle control using active front steering and rear torque vectoring. *International Journal of Vehicle Autonomous Systems*, 8(2-4):85–105, 2010.

[9] Francesco Borrelli, Paolo Falcone, Tamas Keviczky, Jahan Asgari, and Davor Hrovat. Mpc-based approach to active steering for autonomous vehicle systems. *International Journal of Vehicle Autonomous Systems*, 3(2-4):265–291, 2005.

[10] Massimo Canale, Mario Milanese, and Carlo Novara. Semi-active suspension control using fast model-predictive techniques. *IEEE Transactions on control systems technology*, 14(6):1034–1046, 2006.

[11] C. C. Chan, Y. S. Wong, Alain Bouscayrol, and Keyu Chen. Powering Sustainable Mobility: Roadmaps of Electric, Hybrid, and Fuel Cell Vehicles. *Proceedings of the IEEE*, 97(4):603–607, 2009.

[12] D. Crolla and D. Cao. The impact of hybrid and electric powertrains on vehicle dynamics, control systems and energy regeneration. *Vehicle System Dynamics*, 50(Supplement):95–109, 2012.

[13] L. De Novellis, A. Sorniotti, and P. Gruber. Optimal Wheel Torque Distribution for a Four-Wheel-Drive Fully Electric Vehicle. *SAE Int. J. Passeng. Cars Mech. Syst.*, 6:128–136, 2013.

[14] L. De Novellis, A. Sorniotti, P. Gruber, J. Orus, J. M. Rodriguez Fortun, J. Theunissen, and J. De Smet. Direct yaw moment control actuated through electric drivetrains and friction brakes: Theoretical design and experimental assessment. *Mechatronics*, 26:1–15, 2015.

[15] Stefano Di Cairano, Hongtei Eric Tseng, Daniele Bernardini, and Alberto 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*, 21(4):1236–1248, 2013.

[16] Alexander Domahidi and Juan Jerez. FORCES Professional. embotech GmbH (`http://embotech.com/FORCES-Pro`), July 2014.

[17] P. Falcone and F. Borrelli. A model predictive control approach for combined braking and steering in autonomous vehicles. *Proceedings of the IEEE 2007 Mediterranean Conference on Control & Automation (MED'07)*, pages 1–6, 2007.

[18] P Falcone, F Borrelli, J Asgari, H E Tseng, and D Hrovat. Predictive Active Steering Control for Autonomous Vehicle Systems. *Control Systems Technology, IEEE Transactions on*, 15(3):566–580, 2007.

[19] P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H.E. Tseng. A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems. *2007 46th IEEE Conference on Decision and Control*, pages 2980–2985, 2007.

[20] Paolo Falcone, H. Eric Tseng, Francesco Borrelli, Jahan Asgari, and Davor Hrovat. MPC-based yaw and lateral stabilisation via active front steering and braking. *Vehicle System Dynamics*, 46(March 2015):611–628, 2008.

[21] Yoshimi Furukawa and Masato Abe. Advanced Chassis Control Systems for Vehicle Handling and Active Safety. *Vehicle System Dynamics*, 28(2-3):59–86, 1997.

[22] Cong Geng, Lotti Mostefai, Mouloud Denai, and Yoichi Hori. Direct yaw-moment control of an in-wheel-motored electric vehicle based on body slip angle fuzzy observer. *IEEE Transactions on Industrial Electronics*, 56(5):1411–1419, 2009.

[23] Thomas D Gillespie. Fundamentals of vehicle dynamics. warrendale, pa: Society of automotive engineers, 1992.

[24] Timothy Gordon, Matthijs Klomp, and Mathias Lidberg. Control mitigation for over-speeding in curves: Strategies to minimize off-tracking. In *Proceedings of the 11th International Symposium on Advanced Vehicle Control (AVEC'12).*, 2012.

[25] Bernd-Robert Hoehn, Karsten Stahl, Philipp Gwinner, and Ferdinand Wiesbeck. Torque-vectoring driveline for electric vehicles. In *Proceedings of the FISITA 2012 World Automotive Congress*, pages 585–593. Springer, 2013.

[26] Yoichi Hori, Yasushi Toyoda, and Yoshimasa Tsuruoka. Traction Control of Electric Vehicle: Basic Experimental Results Using the Test EV UOT Electric March. *IEEE Transactions on Industry Applications*, 34(5):1131–1138, 1998.

[27] Valentin Ivanov, Klaus Augsburg, Dzmitry Savitski, Jiri Plihal, Pavel Nedoma, and Jaroslav Machan. Advanced cost functions for evaluation of lateral vehicle dynamics. *Lecture Notes in Electrical Engineering*, 198 LNEE(VOL. 10):425–440, 2013.

[28] B Jacobsen. Potential of electric wheel motors as new chassis. 216:631–640.

[29] Milad Jalali, Ehsan Hashemi, Amir Khajepour, Shih-ken Chen, and Bakhtiar Litkouhi. Integrated model predictive control and velocity estimation of electric vehicles. *Mechatronics*, 46:84–100, 2017.

[30] Milad Jalali, Amir Khajepour, Shih-ken Chen, and Bakhtiar Litkouhi. Integrated stability and traction control for electric vehicles using model predictive control. *Control Engineering Practice*, 54:256–266, 2016.

[31] L Kakalis and a Zorzutti. Brake based torque vectoring for sport vehicle performance improvement. *SAE Paper*, 1(1):514–525, 2009.

[32] T. Keviczky, P. Falcone, F. Borrelli, J. Asgari, and D. Hrovat. Predictive control approach to autonomous vehicle steering. *2006 American Control Conference*, pages 4670–4675, 2006.

[33] Donghyun Kim Donghyun Kim, Sungho Hwang Sungho Hwang, and Hyunsoo Kim Hyunsoo Kim. Vehicle Stability Enhancement of Four-Wheel-Drive Hybrid Electric Vehicle Using Rear Motor Control. *IEEE Transactions on Vehicular Technology*, 57(2):727–735, 2008.

[34] R Kim, H Harada, H Minabe, and R Kizu. Electronic Control Of Car Chassis Present Status And Future Perspective. pages 173–188, 1988.

[35] Ken Koibuchi, Masaki Yamamoto, Yoshiki Fukada, and Shoji Inagaki. Vehicle stability control in limit cornering by active brake. Technical report, SAE Technical Paper, 1996.

[36] R Kunii, A Iwazaki, Y Atsumi, and A Mori. Development of sh-awd (super handling-all wheel drive) system. *HONDA R AND D TECHNICAL REVIEW*, 16(2):9–16, 2004.

[37] Nikolay Kyurkchiev and Svetoslav Markov. Sigmoid functions: some approximation and modelling aspects. *Some Moduli in Programming Environment Mathematica,(LAP Lambert Acad. Publ., Saarbrucken, 2015) ISBN*, pages 978–3, 2015.

[38] Heinz Leffler, Reinhard Auffhammer, Reent Heyken, and Harald Röth. New driving stability control system with reduced technical effort for compact and medium class passenger cars. Technical report, SAE Technical Paper, 1998.

[39] E K Liebemann and T Fuehrer. More safety with vehicle stability control. *SAE Technical Papers*, 2007.

[40] E K Liebemann, K Meder, J Schuh, and G Nenninger. Safety and performance enhancement: the Bosch electronic stability control (ESP). *SAE Paper*, 20004:21–60, 2004.

[41] Jan Marian Maciejowski. *Predictive control: with constraints*. Pearson education, 2002.

[42] Hans-Georg Matuttis and Jian Chen. *Understanding the discrete element method: simulation of non-spherical particles for granular and multi-body systems*. John Wiley & Sons, 2014.

[43] Takami Miura, Yuichi Ushiroda, and Kaoru Sawase. Development of Integrated Vehicle Dynamics Control System 'S-AWC'. *Mitsubishi Motors Technical Review*, pages 21–25, 2008.

[44] Kanghyun Nam, Hiroshi Fujimoto, and Yoichi Hori. Lateral stability control of in-wheel-motor-driven electric vehicles based on sideslip angle estimation using lateral tire force sensors. *IEEE Transactions on Vehicular Technology*, 61(5):1972–1985, 2012.

[45] Lorenzo Pinto, Simon Aldworth, Martin Watkinson, Paul Jeary, and Maria Franco-Jorge. Advanced yaw motion control of a hybrid vehicle using twin rear electric motors. *Advanced Vehicle Control*, pages 640–645, 2010.

[46] Damrongrit Piyabongkarn, Jae Y. Lew, Rajesh Rajamani, and John A. Grogg. Active driveline torque-management systems. *IEEE Control Systems Magazine*, 30(4):86–102, 2010.

[47] Damrongrit Piyabongkarn, Jae Y. Lew, Rajesh Rajamani, John A. Grogg, and Qinghui Yuan. On the use of torque-biasing systems for electronic stability control: Limitations and possibilities. *IEEE Transactions on Control Systems Technology*, 15(3):581–589, 2007.

[48] Wang Qiu, QU Ting, Yu Shuyou, GUO Hongyan, and Chen Hong. Autonomous vehicle longitudinal following control based on model predictive control. In *Control Conference (CCC), 2015 34th Chinese*, pages 8126–8131. IEEE, 2015.

[49] Rajesh Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[50] Rajesh Rajamani and Damrongrit Neng Piyabongkarn. New paradigms for the integration of yaw stability and rollover prevention functions in vehicle stability control. *IEEE Transactions on Intelligent Transportation Systems*, 14(1):249–261, 2013.

[51] Robert J Rieveley and Bruce P Minaker. Control for Hybrid Powertrains. 2007(724), 2007.

[52] Andrés Eduardo Rojas Rojas, Haymo Niederkofler, and Wolfgang Hirschberg. Mechanical design of in-wheel motor driven vehicles with torque-vectoring. Technical report, SAE Technical Paper, 2011.

[53] K Sawase and K Inoue. Classification and analysis of lateral torque-vectoring differentials using velocity diagrams. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 222:1527–1541, 2008.

[54] Kaoru Sawase and Yoshiaki Sano. Application of active yaw control to vehicle dynamics by utilizing driving / breaking force. 20:289–295, 1999.

[55] Kaoru Sawase, Yuichi Ushiroda, and Takami Miura. Left-right torque vectoring technology as the core of super all wheel control (s-awc). *Mitsubishi Motors Technical Review*, 18:16–23, 2006.

[56] Robert W Schallock, Kenneth R Muske, and James C Peyton Jones. Model predictive functional control for an automotive three-way catalyst. *SAE International Journal of Fuels and Lubricants*, 2(2009-01-0728):242–249, 2009.

[57] Y Shibahata, N Kuriki, K Kitamura, K Honda, K Wada, H Kajiwara, A Nori, K Kuwahara, and S Okuma. Development of left-right torque distribution system. *HONDA R&D Tech Rev*, 9:166–180, 1997.

[58] Yasuji Shibahata. Progress and future direction of Chassis control technology. *Annual Reviews in Control*, 29(1):151–158, 2005.

[59] Yasuji Shibahata, K Shimada, and T Tomari. The improvement of vehicle maneuverability by direct yaw moment control. In *International Symposium on Advanced Vehicle Control, 1992, Yokohama, Japan*, 1992.

[60] Efstathios Siampis. *OPTIMAL TORQUE VECTORING CONTROL STRATEGIES FOR STABILISATION OF ELECTRIC VEHICLES AT THE LIMITS OF HANDLING*. PhD thesis, 2017.

[61] Efstathios Siampis, Matteo Massaro, and Efstathios Velenis. Electric rear axle torque vectoring for combined yaw stability and velocity control near the limit of handling. *Proceedings of the IEEE Conference on Decision and Control*, pages 1552–1557, 2013.

[62] Efstathios Siampis, Efstathios Velenis, Salvatore Gariuolo, and Stefano 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.

[63] Efstathios Siampis, Efstathios Velenis, and Stefano Longo. Rear wheel torque vectoring model predictive control with velocity regulation for electric vehicles. *Vehicle System Dynamics*, 53(11):1555–1579, 2015.

[64] E. N. Smith, E. Velenis, D. Tavernini, and D. Cao. Effect of handling characteristics on minimum time cornering with torque vectoring. *Vehicle System Dynamics*, 00(00):1–28, 2017.

[65] William Squire and George Trapp. Using complex variables to estimate derivatives of real functions. *Siam Review*, 40(1):110–112, 1998.

[66] R. Tchamna and I. Youn. Yaw rate and side-slip control considering vehicle longitudinal dynamics. *International Journal of Automotive Technology*, 14(1):53–60, 2013.

[67] Hongtei Eric Tseng, Behrouz Ashrafi, Dinu Madau, Todd Alien Brown, and Darrel Recker. The development of vehicle stability control at Ford. *IEEE/ASME Transactions on Mechatronics*, 4(3):223–234, 1999.

[68] a T Van Zanten. Bosch ESP System: 5 Years of Experience. *SAE Technical Paper 2000-01-1633*, 109(724):428–436, 2000.

[69] Efstathios Velenis, Diomidis Katzourakis, Emilio Frazzoli, Panagiotis Tsiotras, and Riender Happee. Steady-state drifting stabilization of RWD vehicles. *Control Engineering Practice*, 19(11):1363–1376, 2011.

[70] Efstathios Velenis, Panagiotis Tsiotras, and Jianbo Lu. Optimality properties and driver input parameterization for trail-braking cornering. *European Journal of Control*, 14(4):308–320, 2008.

[71] J C Wheals, M Deane, S Drury, G Griffith, P Harman, R Parkinson, S Shepherd, a Turner, Ricardo Driveline, Transmission Systems, and Ricardo Plc. Design and Simulation of a Torque Vectoring Rear Axle. *Engineering*, 2006(724), 2006.

[72] Jonathan C Wheals, Ricardo Driveline, Transmission Systems, Reprinted From Transmissions, and Drivelines Symposium Awd. Torque Vectoring Driveline : SUV-based Demonstrator and Practical Actuation Technologies. *Engineering*, (724), 2005.

[73] Stephen J Wright. *Primal-dual interior-point methods*. Siam, 1997.

[74] N. Takahashi K. Suzuki Y. Ushiroda, K. Sawase and K. Manabe. Development of super ayc. Technical Report 15:7376, Mitsubishi Motors Technical Review, 2003.

[75] Hai Yu, Wei Liang, Ming Kuang, and Ryan McGee. Vehicle handling assistant control system via independent rear axle torque biasing. *Proceedings of the American Control Conference*, (3):695–700, 2009.

[76] A. Zanelli, A. Domahidi, J.L. Jerez, and M. Morari. FORCES NLP: an efficient implementation of interior-point methods for multistage nonlinear nonconvex programs. *International Journal of Control*, 2017.