UNIVERSITÀ
DEGLI STUDI
FIRENZE

Doctoral Programme in Industrial Engineering

Dottorato di Ricerca in Ingegneria Industriale

XXXI

A control system framework for autonomous vehicles at the limits of handling

ING/IND-14

Doctoral Candidate

Tommaso Novi

Dean of the Doctoral Programme

Ph.D. Prof. Maurizio De Lucia

Supervisors

Ph.D. Prof. Renzo Capitani

Ph.D. Eng. Claudio Annicchiarico

External referees

Prof. Mauro Velardocchia

Prof. Antonio Gugliotta

Years 2015/2018

To Sandra, Linda and Irene.

Acknowledgement

This research project has been performed at the Department of Industrial Engineering of the University of Florence.
I would like to start by thanking my supervisors Prof. Renzo Capitani and Ph.D. Eng. Claudio Annicchiarico for giving me the possibility to complete my doctorate and for helping me grow from a technical and personal point of view. It was a pleasure to be mentored by them and I will always be thankful for their guidance during this journey.
I am grateful to Prof. Mauro Velardocchia and Prof. Antonio Gugliotta for having accepted to review my thesis and Prof. Mikael Nybacka who has accepted to be part of the committee.
I am indebted to Prof. John Lygeros for hosting me for six very productive months at the Automatic Control Laboratory (IfA) at ETH Zurich where I had the possibility to meet extraordinary people.
A special thank you goes to Meccanica 42 and Danisi Engineering for supporting my research activity and providing me with very useful ideas and data. Thank you also to Embotech for providing me with their software and support, especially during my time in Zurich.
Next I want to thank my fantastic collegues from V2D laboratory. Francesco, an incredible friend who has helped me with many of the decisions I have made in the last year and Cesare, with whom I enjoyed having very interesting conversations which often led to hours of discussions.
I would also like to thank the great people that I met in Zurich. Alex, for teaching me about optimisation and really helping me improve my research. Giampaolo, for making me feel at home even in a different country. Nicolò, for sharing great adventures and also helping me out during my period abroad. Basilio, for including me in many interesting and curious events and becoming a very good friend.
A thank you goes also to all the people who I have crossed during this three year course, the members of Firenze Race Team who always reminded me of why I chose this career and the students I tutored with their theses.
A very special thank you goes to my mother Linda, my sister Irene and all the rest of my family who have supported me through thick and thin in this adventure and have spent countless hours giving me suggestions and keeping me pushing in the right direction. Last but not least I would like to thank my dear Sandra, the person who every day has shown me love and support and has been my biggest shoulder to lean on in hard moments, I will never forget all that you have done for me and I can only try to give you back all the support that I can.

Summary

Autonomous Driving (AD) at the limits of handling represents a big technical challenge of which vehicle dynamics and control are at the core. The main complication is the thin line separating the vehicle from operating at its peak performance and underperforming. To evaluate the effect that the decision-making process has on vehicle performance, one can utilise path-following algorithms. In this research, a mathematical framework is developed which studies all areas of the vehicle dynamics and control of a car driving on a given trajectory in non-linear conditions. The framework comprises three main areas.

First, the vehicle state estimation is studied, concentrating on the states which are not and cannot be measured on a vehicle with low-cost sensors, namely the sideslip angle. This can be observed with model-based methods or machine learning techniques. However, with model-based methods it is difficult to obtain accurate results given the large non-linearities and uncertainties in the dynamics and with machine learning techniques large datasets are required to avoid extrapolation. Therefore, an integrated Artificial Neural Network (ANN) and Unscented Kalman Filter (UKF) observer is developed which uses only Inertial Measurement Unit (IMU) measurements and can work as a standalone sensor. The goal is to use only numerical tools to develop the virtual sensor, therefore, the ANN is trained solely with numerical data obtained using a Vi-Grade model. The ANN outputs a pseudo-sideslip angle which is used as an input for the UKF. Since this is based on a kinematic model, the UKF is not affected by model mismatch and is capable of correcting the estimate despite model uncertainty. The UKF requires longitudinal velocity as an input, however, this is not measured directly by the IMU. Therefore, longitudinal velocity is estimated by means of direct integration with integral damping and integral reset value. The pseudo-sideslip angle is also corrected to improve the convergence of the UKF.

Second, a robotic control scheme is developed for AD path-following at the limits of handling. This generally requires a compromise between computation time and model complexity. To tackle this a hierarchical controller made of two Non-linear Model Predictive Controls (NMPCs) is developed. The advantage of this type of scheme is that the two levels of the controller can interact to guarantee the desired outcome. While the higher level NMPC operates on a long prediction horizon, the lower level NMPC operates on a short horizon. The difference is that the high-level NMPC is based on a simple point-mass model and tyre-dependant "gg-diagram" constraint and is used solely to calculate velocity profiles. The output is then used as a terminal constraint and terminal cost by the lower level which is based on

a seven degrees of freedom vehicle model with full Pacejka Magic Formula (MF) tyre formulation on all tyres, load transfers and Limited Slip Differential (LSD). Because of the precomputed terminal set, the low-level NMPC only requires a short horizon and focuses on exploiting the vehicle performance in real-time. For both controllers, the full Non-linear Optimisation Problem (NLP) is solved at each step.

Third, the stability of the vehicle from a classic vehicle dynamics point of view is analysed with the aim of discussing how certain design paradigms can be changed when the human driver is replaced with a machine. Specifically, the vehicle stability of traditional vehicles is designed to be inherently stable leading to an understeering attitude. These design goals are required since it is safer for a human driver, however, this causes a decrease in peak lateral grip due to oversized rear tyres. With advanced autonomous driving features, the passive vehicle dynamics design goals could be set in a different way thereby gaining in both peak grip and transient response. Thus, several vehicle models with differing dynamic behaviours are developed and tested on a Driver-in-Motion (DiM) dynamic driving simulator driven by a number of expert human drivers. The same tests are then run again in a Model-in-the-Loop (MiL) simulation where the vehicle is controlled by means of a NMPC. The results pose some interesting questions on how commercial vehicles can be designed if driven by a robotic controller.

The outcome of this work is a step further into the research of autonomous vehicles at the limits of handling from both an estimation and control perspective. This represents an advancement in the state of the art. The results of the framework developed show the effectiveness of the algorithms presented in this research at controlling a driverless vehicle and estimating its states by focusing on the vehicle dynamics.

Contents

List of Figures

List of Tables

Research question

Autonomous driving is changing the way that we conceive vehicles and study vehicle dynamics, but what are the steps required to deliver a robotic controller that can outperform the human being as a driver?

Moreover, how can we design methods and algorithms so that self driving vehicles can maximise the theoretical performance achievable with a car and increase the level of safety on roads?

Chapter 1

Introduction

1.1 Motivation

The automotive world is facing an authentic revolution of which autonomous driving and automation in vehicles are the main cause. Most Universities, research centres, Original Equipment Manufacturers (OEMs) and high-tech companies are pushing technology to produce self driving cars and autonomous driving features. The reason behind this "industrial revolution" is the need for an increase in safety and the will to reduce the number of road fatalities for which human error holds major responsibility. Despite the efforts in recent years to increase safety there have been approximately fifty road deaths per million inhabitants in the European Union (EU) in 2017. This has shown a 2% decrease in the number of road deaths between 2016-2017 in the EU which follows a 2% reduction in 2016, a 1% increase in 2015 and stagnation in 2014. As a result, the number of road deaths has declined by just 3% since 2013. This widens the gap between the actual and desired progress in terms of fatalities towards the EU 2020 target.[1] These trends are shown in Figure 1.1.
Self driving vehicles can potentially eliminate the human factor, thus, drastically reducing the number of fatalities. However, to do this there are a number of challenges that have to be faced from a technological, juridical and ethical point of view. The former is the one of interest from an engineering standpoint and is itself a multidisciplinary problem. Companies which have been the main players in the growth of software and electronic hardware over the recent years have become a necessity for the success of autonomous driving technologies. It has become a custom that giants in the automotive field such as BMW, FCA and JLR pair up with companies like Google, Intel or NVIDIA to share know-how in order to deliver a full autonomous vehicle.
AD dates back to many years ago. The first attempts are from the 1920s with the radio-controller "American Wonder" and the 1930s Futurama exhibit sponsored by General Motors. The real development started in more recent years, mostly due to the increased performance of electronic hardware and intelligent computers which allow the performance of advanced calculations in a limited amount of time. A big step forward was taken in 2014 when the Society of Automotive Engineers (SAE) [2] came up with a classification for AD. This classification divides autonomous driving into six different levels which go from no automation where the human being is in

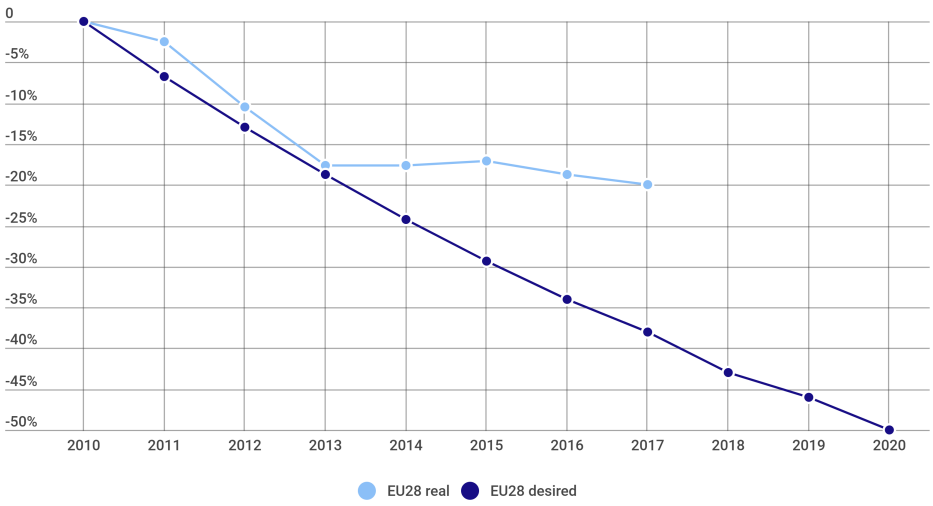

Figure 1.1: EU road fatalities trend.

total control of the vehicle, via partial automation with the aid of Advanced Driver Assistance Systems (ADAS), to full automation where an intelligent machine is in complete control of the vehicle.
Considering the following definitions:

- **Dynamic driving task** include the operation (steering, braking, accelerating, monitoring the vehicle and roadway) and tactical (responding to events, detemining when to change lanes, turn, use signals, etc.) aspects of the driving task, but not the strategic (determining destinations and waypoints) aspect of the driving task.
- **Driving mode** is a type of driving scenario with characteristic *dynamic driving task* requirements (e.g. expressway merging, high speed cruising, low speed traffic jam, closed-campus operations, etc.)
- **Request to intervene** is notification by the *automated driving system* to a *human driver* that he/she should promptly begin or resume of the *dynamic driving task*.

The six levels of automation can be defined as in Figure 1.2. One of the major distinctions is between level two and three. Whilst in the former the human driver performs part of the dynamic driving task, in the latter the self driving vehicle performs the dynamic driving task. Note that the SAE excludes from this classication warning and momentary intervention systems which do not automate any part of the dynamic driving task on a sustained basis and therefore do not change the human driver's role in performing the dynamic driving task.
Level two automated systems are nowadays implemented as standard features by most OEMs. Systems such as emergency braking have become mandatory for all cars by 2022 and other more advanced automated driving features such as Adaptive Cruise Control (ACC) and Automated Lanekeeping (AL) are almost a standard in all production vehicles. However, most of these systems are developed and delivered to OEMs by tier one suppliers like Bosch, Continental and others and are,

SAE level	Name	Narrative Definition	Execution of Steering and Acceleration/ Deceleration	*Monitoring* of Driving Environment	Fallback Performance of *Dynamic Driving Task*	System Capability *(Driving Modes)*
***Human driver* monitors the driving environment**						
0	No Automation	the full-time performance by the *human driver* of all aspects of the *dynamic driving task*, even when enhanced by warning or intervention systems	Human driver	Human driver	Human driver	n/a
1	Driver Assistance	the *driving mode*-specific execution by a driver assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	Human driver and system	Human driver	Human driver	Some driving modes
2	Partial Automation	the *driving mode*-specific execution by one or more driver assistance systems of both steering and acceleration/ deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	**System**	Human driver	Human driver	Some driving modes
***Automated driving system* ("system") monitors the driving environment**						
3	Conditional Automation	the *driving mode*-specific performance by an *automated driving system* of all aspects of the dynamic driving task with the expectation that the *human driver* will respond appropriately to a *request to intervene*	System	**System**	Human driver	Some driving modes
4	High Automation	the *driving mode*-specific performance by an automated driving system of all aspects of the *dynamic driving task*, even if a *human driver* does not respond appropriately to a *request to intervene*	System	System	**System**	Some driving modes
5	Full Automation	the full-time performance by an *automated driving system* of all aspects of the *dynamic driving task* under all roadway and environmental conditions that can be managed by a *human driver*	System	System	System	**All driving modes**

Figure 1.2: SAE levels of driving automation.

therefore, not optimised for any vehicle. On the other hand, these systems have been developed to be robust and easily tunable by each car manufacturer.
There are at least five key concepts to take into consideration when developing an Autonomous Vehicle (AV), namely: vehicle dynamics and commands execution, understanding and perceeving the environment, decision-making, driver transition from human to machine and vice versa and, finally, the use of time and human machine interface (HMI).

Vehicle dynamics and command execution

This key concept can be divided itself into three main categories: vehicle control, estimation of the vehicle dynamics and quantification of handling.

- **Vehicle Control -** Since a human being is able to control only a few inputs contemporarily (i.e. steering, pedals and gear shift), it is very limited in terms of control of the vehicle. This aspect is even more important when considering the vehicle at at-limit handling conditions (i.e. non-linear tyre range). An autonomous vehicle, instead, could be able to operate contemporarily and independently the various controls of the vehicle, such as wheel toe angles and wheel torques (and possibly active suspension). Also, a machine is able to exert much higher forces and actuate the various controls with higher accuracy, speed and repeatability. All these facts allow an autonomous vehicle to have (theoretically) a much better control of the vehicle when compared to a human driver.

- **Estimation of the Vehicle Dynamics -** A human being, when driving, bases its decisions in terms of vehicle control on the feedback that he/she gets from the car and his/her experience. Professional drivers develop a much

higher sensitivity to this feedback. This allows them to react in a better way in terms of decision-making and time to make that decision. However, the maximum performance of a vehicle (e.g. in an obstacle avoidance manoeuvre at high speed) occurs when a tyre is in its non-linear range. In these conditions, the force generated by tyres vary very rapidly and the loss of control of the vehicle can happen very quickly without giving any perceivable warning to the driver. This is because the force generated by the tyres varies so quickly that it is impossible for a human being to process what is happening to the vehicle dynamics. This can be overcome by an autonomous vehicle which uses computers that process data much quicker than the human brain, algorithms based on analytical rules and high precision sensors which are not based on experience (unlike the human senses).

- **Quantification of Handling -** An autonomous vehicle also has a greater capacity of quantifying which are the manoeuvres that the vehicle can perform considering its physical limits. This is mainly due to the fact that an autonomous vehicle has greater knowledge of how much margin the vehicle has in terms of tyre adherence. Also, an autonomous vehicle can be certain of how it can act on the various controls of the vehicle in terms of frequency, force and delays of actuation. A human being, instead is not able to quantify its limits in terms of actuation of the various controls.

Environment understanding

Most of the decisions made by a human during a journey are based on: infrastructures, what other vehicles on the road are doing and anything else that can cause an accident (e.g. a pedestrian or a cat crossing the road). During a certain route, there is a continuous change of trajectories and speed profiles because of the unpredictability of other vehicles. A lot of this is because cars are being driven by human beings. Therefore, an autonomous vehicle surrounded by vehicles driven by a human being would have to continuously monitor everything that is going on around it to be able to dynamically change its decisions. This includes events that happen unexpectedly (e.g. a cat crossing the road). Considering the case where 100% of the vehicles are autonomous, the single autonomous vehicle would still have to monitor what is happening, since, unpredictable events (such as the cat crossing the road) can still happen. Also, an autonomous vehicle will still have to interact with other autonomous vehicles present on the road. However, it would not have to deal with the unpredictability of human driving. Technologies such as V2V, V2I and many others are necessary.

Decision-making

One of the main features that a human driver has is that it is able to make decisions when driving. These decisions can become very important when unpredictable events occur (e.g. a kid suddenly crossing the road). In such situations, the time which is available for the driver to think is not much. Also, when these events occur the manoeuvres that the vehicle has to perform are often in the non-linear range of the tyres. Therefore, the decisions made by the driver on how to react, where to go and how to perform a certain manoeuvre do not take into consideration the vehicle's limits and the humans limits (in terms of actuation). If an autonomous vehicle could perform a real-time Failure Mode and Effect Analysis (FMEA), considering

the combination of severity of the incident and probability of the incident for all the possible manoeuvres that the vehicle could actually perform starting from a certain dynamic state, it could then decide which would be the best manoeuvre to perform. Unlike when a human being is driving, the decided manoeuvres would be at that point performable.

Driver transition

Autonomous vehicles could revolutionize the world of transportation. However, many people will always enjoy driving and will want to have the possibility to do so. Considering, for instance, a journey from the city to a ski resort, it is possible that the driver might want to sit back and let the vehicle drive along the motorway. However, on a mountain pass, the driver might want to take over and enjoy driving the vehicle himself/herself. For situations such as these, it is fundamental that an an autonomous vehicle has the possibility to easily switch from autonomous driving mode to human driving and vice versa.

Time-use

One of the largest problems in peoples' life nowadays is the lack of time. A lot of the time available by a human is lost in traffic or during its journeys. If road vehicles were to be autonomous, the user would be able to use the time spent to get to work (e.g. in the morning on the drive to work) as he/she best prefers. To do this the vehicle has to accomodate the user in the best way possible. For instance, if the user wants to work on a computer on his/her way to work, it has to be able to do it as if he/she was in the office.

1.2 Research purpose

This research focuses on autonomous driving at at-limit handling and falls between level three and four of the SAE categorization and into the vehicle control and command execution key concept. Specifically, the main purpose of this work is to develop and implement algorithms which can fully exploit vehicle performance, even better than what a human can do.
This goal can be achieved by correctly understanding the physics of the vehicle and applying advanced control techniques which can easily adapt to the situation, be robust enough to be used on a road vehicle and, most importantly, achieve maximum vehicle performance. Despite there being a lot of research in the literature on such topics, most of the research is focused on developing advanced control algorithms without putting too much importance on the vehicle's actual performance when related to its physical capabilities and to real-time feasibility.
In this research the vehicle dynamics is put at the centre of the project and the control system and state estimation are developed around understanding the system dynamics. This is to be obtained with low-cost equipment and high-fidelity vehicle models. Additionally, the final objective is to evaluate how a robotic controller can change the way of designing vehicles from a passive vehicle dynamics point of view. The techniques used are multidisciplinary and involve algorithms such as neural networks, unscented Kalman filter and model predictive control. Great importance is given to the result analysis and to developing tools which allow us to correctly evaluate the performance of the automated driving system.

1.3 Research process

The work of this thesis started after having gained knowledge on what the industrial and research worlds have studied and developed and having characterised the areas of interest in autonomous driving. An in-depth analysis of the literature of the specific fields to be tackled was done and the methods and tools to be used to develop the desired algorithms were chosen. Partnerships with companies in the industry such as Meccanica 42, Danisi Engineering and Embotech were started to gain know-how from different engineering sectors. The project continued with the layout of the control system and the decision regarding the main areas of focus.

The first problem faced was the state estimation of the various quantities, considering that the system would be operating in a small range where the vehicle's behaviour is very unpredictable and can quickly change. The literature review revealed important areas which could be studied and tools which could be used to optimise results. An algorithm was developed to estimate non-measurable quantities and using measurements coming only from affordable sensors. This was also tested in an experimental setup and proved to give good results. This part of the project was published in a journal paper.[3]

The second problem which was tackled was the development of a vehicle controller. Again, the literature review opened up many possibilities and roads to pursue. The need for real-time feasibility and vehicle performance maximisation led to a hierarchical controller scheme which was developed by means of numerical tools. Thanks to industrial partnerships it was possible to optimise the results and have access to advanced solvers and vehicle models which could guarantee real life feasibility. To properly solve this problem, a collaboration with ETH Zürich started with whom the controller was developed during a six months stay at Institut für Automatik. The results from this study were published in a journal paper.[4]

Finally, using the results obtained in the rest of the research an analysis on how vehicle design can be changed with autonomous driving was investigated. To do this a dynamic driving simulator was used and results between a human driver and an autonomous driver were compared. This part of the research is only a preliminary analysis and further studies can be done to complete the results. The results of this study were presented at a conference and published in a journal.[5]

1.4 Thesis outline

Chapter 2

This chapter describes the literature analysis of the various areas studied in this research. The state of the art is analysed in detail as it represents the starting point of development and allows us to properly place the contribution of this research with respect to the already existing frameworks. Finally, the trajectory generation method used in this paper is discussed along with the minimum curvature algorithm and with its results.

Chapter 3

The sideslip angle observer is here discussed in detail. The chapter begins with an analysis of the steady-state, transient and frequency response of a vehicle to the control inputs. Particular attention is paid to the coupling between sideslip angle and the vehicle states which can be measured by means of an IMU.
The novel integrated ANN and UKF observer is then described together with an algorithm to define the ANN structure and the model that couples the ANN with a kinematic UKF observer. A longitudinal observer based on direct integration is presented and an additional strategy to correct the estimation and improve convergence is proposed. Experimental results are then shown and discussed.

Chapter 4

A novel hierarchical control scheme to control the vehicle at the limits of handling is presented. This controller is made of two NMPCs that interact with each other. First the vehicle models and constraints used in both levels are described, then the optimisation problems formulations are shown. At the end of the chapter, an in depth analysis of the results is done which compares the outcome of the controller with those of a commercial software. Comments are made on both the performance of the vehicle and of the control scheme.

Chapter 5

With the controller developed in the previous chapter, the question of whether vehicles can be designed with lower stability if driven by machines rather than humans is discussed. A number of vehicles with different characteristics are developed and tested on a driving simulator by expert drivers. The same tests are then repeated with a simplified version of the previous controller. The results are finally commented on and interesting outcomes are found which pose further questions for the future.

Chapter 6

In this chapter the conclusions are discussed. These summarize the contributions presented in the previous chapters and highlight how the research places itself in respect to the state of the art.

Chapter 2

Autonomous Driver - State of the Art

In this chapter, a state of the art analysis of the various areas of research covered in this work is described. The three main areas which are studied are the state estimation, the vehicle control and the influence of autonomous driving on passive vehicle dynamics. Specifically, for each of these topics a literature review of the field of interest is done and the contribution which has been brought is highlighted. The technical matters are described in detail in future chapters.
This chapter concludes with an in depth description of the trajectory and path used in this work. In fact, this work focuses on path-following algorithms and, despite being robust on any path generated, it is necessary to define a trajectory. In this work a minimum curvature trajectory is used. Note that this trajectory is theoretically time optimal as it minimises the required acceleration (thus forces) for a given speed. In reality this trajectory tends to make the problem harder to solve from a numerical standpoint. This is because the minimum curvature trajectory makes the vehicle very fast during curve entry. Hence, a greater amount of combined lateral and longitudinal acceleration is required to stay on the path during this phase of a curve. However, especially during braking manoeuvres, this complicates the integration since wheel dynamics become stiff.

2.1 State Estimation

In the vehicle dynamics and command execution key concept, the estimation of the vehicle states represents one of the aspects to consider. Having precise knowledge of the current state of the vehicle together with a proper mathematical model of it allows the quantification of how much margin a vehicle has in terms of adherence, and thus how the various commands can be executed to ensure maximum performance. However, most states can be easily measured directly with sensors. Quantities such as wheel speed, vehicle position, steering wheel angle, yaw rate, lateral and longitudinal accelerations etc. are normally measured on standard vehicles. To increase the usability of these measurements simple linear filters can be used. On the other hand, lateral velocity or sideslip angle are very difficult to measure with affordable sensors. These are however very important for what concerns the tyre adherence knowledge, particularly when the vehicle is near the

grip limit.[6] In such conditions, the vehicle is operating in the non-linear range of the tyre curve, i.e. large slip angles and, therefore, a large vehicle sideslip angle. However, these are highly unstable conditions since even a small variation in the contact patch between tyres and road can induce a rapid variation in available grip.[7] These unstable conditions can be analysed by tracing phase portraits and studying the stability of the equilibrium points. Some results of these analyses are shown in 2.1.

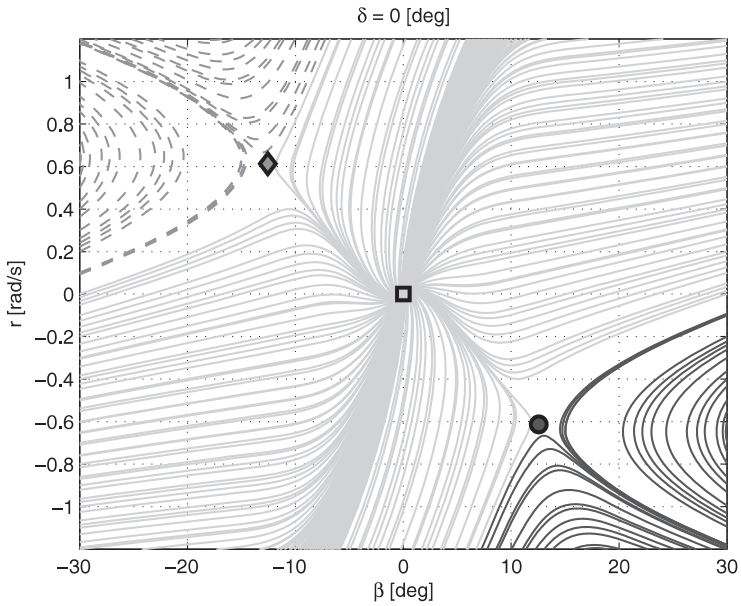

Figure 2.1: Typical phase portrait with no steering angle.[8]

Some researchers such as Edelmann et al. [9] and Voser et al. [8], have shown the existence of a bifurcation and demonstrated that above certain values of steering angle and velocity, no state trajectories converge to the equilibrium point since a saddle-node bifurcation is present, making the system highly unstable for any value of the sideslip angle. However, there are many other conditions for which the vehicle is stable within a certain range of sideslip angle and yaw rate. Knowing the vehicle's states and specifically the sideslip angle is, therefore, particularly important for stability control systems which act on independent wheel braking and torque to limit the sideslip angle amplitude. This concept can be extended to path-following autonomous vehicles in at-limit handling conditions for which both maintaining stability and guaranteeing that the vehicle is operating within the tyre saturation region are important. However, the sideslip angle is also important for tracking errors. In fact, as for Hu et al. [10], the velocity vector at the centre of gravity should be along the tangent direction of the desired path to prevent conflict between the sideslip angle and the yaw rate controls.
As previously mentioned, sideslip angle can be measured directly only by means of very expensive sensors such as optical sensors based on the Doppler effect, high frequency Differential Global Positioning System (DGPS) which require a fixed base and are therefore not applicable to road vehicles or tyres which allow the

measurement of wheel loads such as Pirelli's Cyber tyre.[11] Alternatively, sideslip angle can be estimated with observers. In this work the focus is on the estimation of the sideslip angle. However, to be able to create a standalone smart-sensor, only sensors which do not require a specific position in the vehicle (i.e. wheel speed sensors or steering wheel angle sensors) are used.

2.1.1 Literature review

Many types of model-based observers have been developed over the years to estimate the sideslip angle. Wang et al. [12] use a closed-loop state feedback observer based on IMU measurements, longitudinal velocity and an inverted Dugoff tyre model.[13] Grip et al. [14] use a non-linear observer based on asymptotic stabilisation of estimation errors guaranteed by means of Lyapunov functions. Other approaches are those of Shraim et al. [15] and Cadiou et al. [16] who use sliding mode observers and Zhao et al. [17] who use moving horizon strategies. With all these methods, estimation is strongly influenced by the vehicle and tyre models and system uncertainties. Although modelling errors cannot be completely eliminated with model-based observers, estimation can still be improved with adaptive methods. Zhang et al. [18] use a gain-scheduling observer based on a linear-parameter-varying system whilst You et al. [19] use online adaptive laws based on yaw rate dynamics and lateral acceleration measurements.
The most common method to estimate the sideslip angle, based on vehicle dynamics models, is the Kalman Filter (KF) and its derivations. Many different ways of applying this filter can be found in the literature. Ryu et al. [20] apply linear KFs to a lateral dynamics model, based on measurements of both GPS and IMU. Doumiati et al. [21] applied an Extended Kalman Filter (EKF) directly to a simplified vehicle model in which the road friction coefficient is considered to be known, using a Dugoff tyre model. The best description of tyre dynamics is by Huang et al. [22] and Li et al. [23] who use the standard Pacejka tyre model.[24] To reduce the error arising from model uncertainty, the latter use a sideslip angle rate feedback and sideslip damping for error accumulation. In this case, a separate estimator is used to evaluate the road friction coefficient. Hodgson et al. [25] also use the Pacejka tyre model, however, they adapt the tyre force curve in real-time by using a simplified vehicle model. Other approaches do not use already existing tyre models but rather a separate observer to estimate tyre force. Baffet et al. [26] use a Sliding Mode Observer (SMO) whilst Lian et al. [27] calculate cornering stiffness by means of a Recursive Least Squares (RLS) regression model. These forces are then used in a simplified vehicle model and an EKF is applied to estimate the sideslip angle. Many other methods to estimate tyre forces exist; Acosta et al. [28] have done a survey of these methods. For Julier et al. [29] the EKF is only reliable for systems which are almost linear within the operating frequency range. This would make it necessary to operate at very high frequency when tyres are in the saturation region to assure linearity, which is, however, not feasible with a normal production Electronic Control Unit (ECU). The EKF can also become unstable due to the need to calculate the Jacobians at every time step. Using the UKF and Unscented Transform (UT) can solve these issues. The UT allows calculation of the statistics of a random variable which is subject to a non-linear transformation. A graphic outline of the differences between the UKF and EKF can be seen in Figure 2.2.

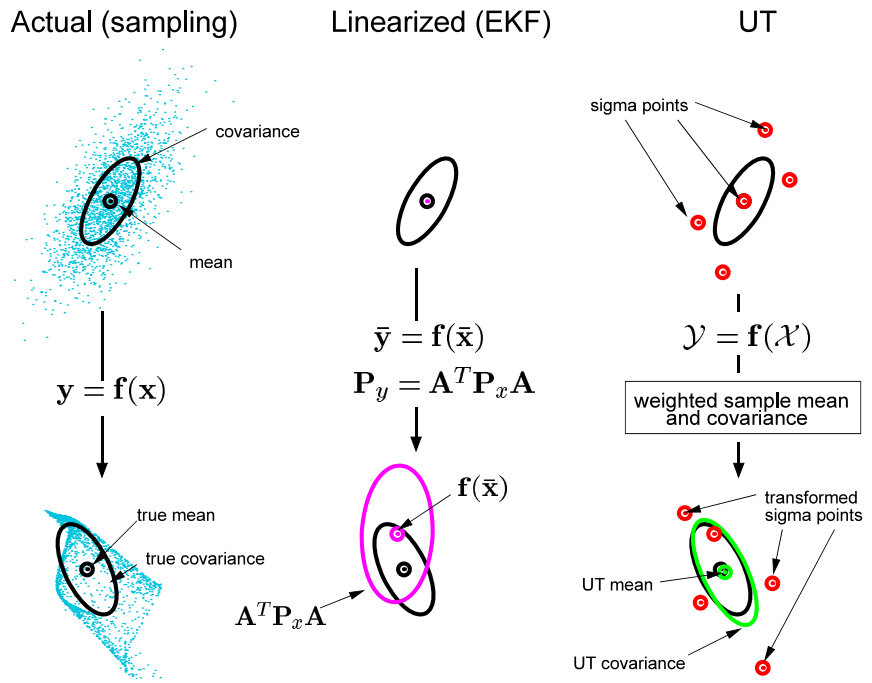

Figure 2.2: Unscented transform.[30]

Morrison et al. [31], Antonov et al. [32] and Cui et al. [33] use the UKF to estimate the sideslip angle on heavy duty trucks and passenger car vehicles using high fidelity vehicle models. Chen et al. [34] add an integral correction, with damping and integral value reset, to the model-based UKF to improve model uncertainty.
A completely different approach is the one based on machine learning techniques and, specifically, supervised learning. The main algorithm used in the literature is ANN which has been demostrated to be capable of approximating any function.[35] An application of ANN to sideslip angle estimation was done by Du et al. [36] using IMU measurements, wheel velocity and steering wheel angle as network inputs. In this case, only pure lateral behaviour was tested and only on a small number of manoeuvres based on numerical data. Melzi et al. [37] tested the algorithm adding combined slip behaviour and a larger number of manoeuvre types, still limited, however, to ISO standards. Melzi et al. used a hidden layer with hyperbolic tangent transfer functions and trained the network using the same experimental data on which the performance of the estimation is evaluated. The problem with using wheel velocity is that at at-limit handling conditions it requires preconditioning in many situations, e.g. locking of a LSD. Also, wheel velocity sensors are generally attached to a vehicle's CAN line, and, therefore, are not standalone. Additionally, using multiple sensors with different frequency acquisition requires some sort of signal synchronization. Sasaki et al. [38] and Wei et al. [39] use only lateral acceleration and yaw rate which can be measured directly with an IMU. The former train and test ANN on experimental data only, the latter apply a General Regression Neural Network (GNRR) trained on numerical data and tested on experimental data. These papers show that it is possible to estimate the sideslip angle by means of ANN using only IMU measurements. The results obtained by these researchers

are promising. However, they lack generality and show good results only in very specific situations since they do not address the effects of vehicle speed variation and combined slip (no longitudinal acceleration is considered). Additionally, a heuristic approach is used to define the ANN structure in these papers.
Adaptive Neuro-Fuzzy Inference Systems (ANFIS) is another common machine learning technique used, an example of which is given by Boada et al. [40] who use IMU measurements, steering wheel angle and longitudinal velocity. By means of numerical results, the authors show that ANFIS outperforms both ANN and model-based estimators such as Kalman Filters (KFs).
Finally, there are many examples in the literature which integrate the two approaches. Acosta et al. [41] apply a stochastic EKF based on a single track model and use ANFIS for road friction estimation. Boada et al. [40, 42] use ANFIS to estimate a pseudo-sideslip angle which is fed into a two-degrees-of-freedom dynamic model UKF which filters the signal. GPS, IMU and steering wheel angle measurements are used and the virtual sensor is tested by means of numerical data.
The two main issues of all these approaches are the following:

1. Model-based observers are subject to model uncertainty and do not work properly when external conditions are varied or unknown;

2. Machine learning techniques depend on the training dataset and consequently cannot generalise the problem. The papers in the literature show that it is feasible to estimate sideslip angle using only IMU measurements but that estimation is only correct when the tested dataset is very similar to the training dataset.

2.1.2 Contribution

This study proposes a novel sideslip angle estimator based on an integrated ANN and UKF which solves the aforementioned issues. The first minor contribution is that only IMU measurements are used for the estimation. The scheme of this observer can be seen in Figure 2.3.
The ANN is trained by means of numerical data only. The second contribution is an algorithm that can be used to find the best network architecture in a numerical environment. As in the work of Boada et al. [42], the ANN outputs a pseudo-sideslip angle which is fed into a UKF to correct the estimation. It will be shown how the sideslip angle time response given by the ANN is accurate, unlike the magnitude which is, however, corrected by the UKF. The third and main contribution of this model is that the UKF is based on a kinematic model of the sensor rather than on a vehicle dynamics model. The use of a kinematic model allows complete freedom from vehicle model uncertainty, specifically the tyres, and still corrects estimation given by the ANN. The longitudinal velocity of the kinematic model is estimated by direct integration with integral damping [43] and integral reset value correction. Due to model uncertainty and the different closed-loop behaviour (driver) of the numerical and experimental environment, the pseudo-sidelslip angle output is saturated within values seen during training. To improve the convergence of the algorithm, a correction strategy for the pseudo-sideslip angle is adopted, which represents the final contribution. The proposed estimator shows very good results in both numerical and experimental environments, also in conditions previously unknown to the ANN.

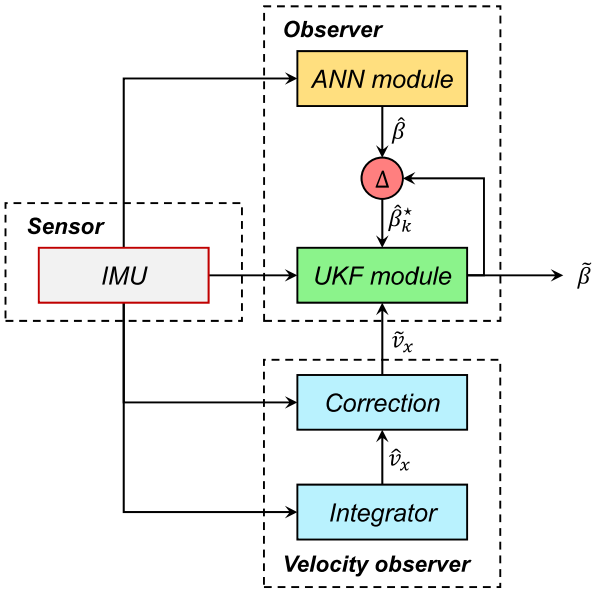

Figure 2.3: Proposed observer structure.

2.2 Robotic Controller

Another aspect to consider in the vehicle dynamics and command execution concept is vehicle control. Driving at at-limit handling and exploiting vehicle performance has always represented a big challenge, particularly in motorsport.[44–47] As already mentioned in the previous section, at at-limit handling the tyres are operating in their non-linear range and the vehicle is generally unstable, as even a small variation in the contact between tyres and road can induce a rapid variation of grip. Additionally, the feedback that a driver gets from the vehicle is limited due the human's inability to process such feedback in the short amount of available time. At-limit handling is also important in every day driving where critical manoeuvres are often necessary: i.e. sudden obstacle avoidance manoeuvres.
In this work a new controller for a real-time feasible autonomous driver which can exploit the vehicle performance is developed.

2.2.1 Literature review

There are two different types of algorithms which have been analysed to develop the controller. First, lap time optimisation methods are analysed. These algorithms find the optimal control sequence which minimises time. The inputs are computed offline and thus cannot be applied for AD. However, they represent the basics for controlling an autonomous vehicle at at-limit handling. Second, algorithms which control autonomous cars are analysed, with a focus on those which exploit vehicle performance.

Lap time optimisation

The first attempts to find the sequence of inputs that maximise vehicle performance were done with offline simulations for motorsport applications. The goal was to obtain the minimum lap time control input sequence and the optimal trajectory. The first lap time simulation was done by Mercedes Benz in the 1930s.[7] In the past decades, many different researchers have studied this topic in various ways. One approach is to simulate the Quasi-Steady-State (QSS) conditions. Brayshaw et al. [48] used the QSS method on a predefined path. The results from this method are shown in Figure 2.4. The algorithm starts by calculating the vehicle's acceleration limits from a series of constrained non-linear optimisation problems. Then, peaks in the curvature data are identified as an apex of a corner and the maximum possible steady-state speed for each curve is found. The maximum acceleration and deceleration between apex pairs is then calculated and, finally, speed profiles are obtained together with the input sequence. Using the same method, Tremlett et al. [49] studied the influence of LSD on lap time simulation.

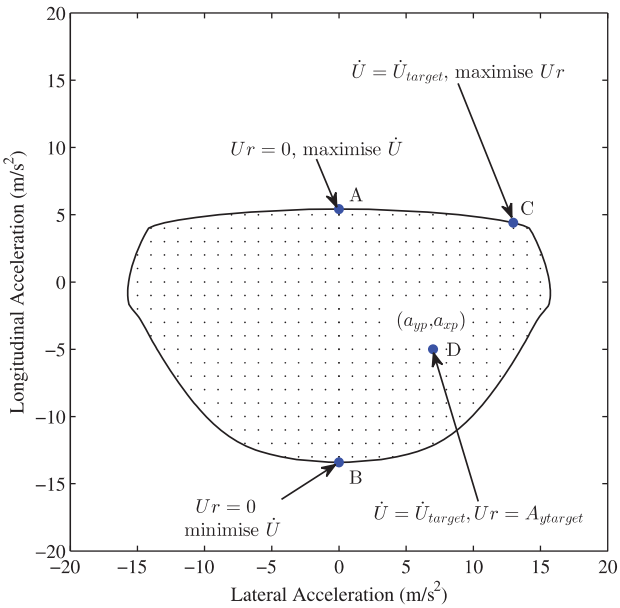

Figure 2.4: QSS method.[49]

A more realistic approach is the one based on transient vehicle behaviour and optimal control techniques. This approach can be divided in two main categories: direct and indirect methods. The former have the advantage that they do not require the adjoint equations which become more complicated to derive as the vehicle model and boundary constraints increase in complexity. However, the latter are more accurate and there is not a risk of incurring into a local minimum. Direct methods were first applied by Casanova et al. [50] who used direct multiple shooting applied to a seven degrees of freedom non-linear vehicle model solved with Sequential Quadratic Programming (SQP). A very similar approach was used by

Kelly et al. [51, 52] who additionally implemented a thermodynamic tyre model. Tremlett et al. [53] also apply a study on tyre usage in terms of temperature and wear with these algorithms. Perantoni et al. [54] apply direct lap time optimisation to study a set of vehicle parameters on a formula type vehicle. One of the issues when trying to minimise lap time with optimal control is the objective function formulation. In fact, it is non-trivial if the best strategy is to minimise time or maximise velocity. This problem is addressed by means of direct optimisation, by Velenis et al. [55]
Indirect methods were first applied by Tavernini et al. to analyse optimal handbrake cornering [56] and the effect of road surface and car transmission layout.[57] Tremlett et al. also use this method to study optimal control of motorsport differentials [58] while Rucco et al. [59, 60] improve the optimisation algorithm by including a derivation of suitable transverse coordinates that allow to split the dynamics into longitudinal and lateral and propose a continuation method for the intermediate trajectories. Lot et al. [61] use indirect methods to study lap time optimisation of a go-kart and use an independent variable transformation from time to space. Dal Bianco et al. [62] also consider a suspension model to a formula type vehicle to analyse how suspensions influence lap time.
All of these papers require a long time to solve the optimisation problem; to improve the computation time and move towards real-time implementation, Velenis and Tsiotras developed a semi-analytical model which allows generating the optimal velocity profile by applying Pontryagin's Maximum Principle. This was done first on a point-mass material model [63] and then extended to a half-car model.[64] To apply this in real-time, a receding horizon strategy was also applied.[65, 66]

Towards autonomous driving

For AD it is crucial for the controller to be executed in real-time, which normally means between one and one-hundred milliseconds, depending on the exact task. This is linked to the dynamics which are controlled. Generally, the lower the lever of the controller the higher the speed should be. Hence, the update rate of one milliseconds can be referred to controllers such as stability controllers where the dynamics are very fast whereas the update rate of one hundred milliseconds can be referred to navigation tasks where the dynamics are much slower since the decision-making process changes relatively slowly, based on the estimated traffic en-route. This requirement renders most of the previously discussed approaches not applicable. However, over the last decades many different control techniques have been developed for AD. In the here presented methods, controllers that focus on at-limit handling are discussed. These controllers can be divided into two groups: first the ones that separate longitudinal and lateral dynamics; and second the ones that integrate all inputs into one controller. Within the first category is the controller of Zhang et al. [67] who developed a lateral controller only by means of sliding mode using a first order sliding surface to control lateral velocity and yaw rate. More recently, Klomp et al. [68] worked on a pure lateral controller for driving at the limits of handling using a preview path curvature and using a feedforward and feedback steering control calculated with a linear bicycle model. The most interesting approach that falls into this category is the one of Kritayakirana et al. [69, 70] who developed a controller considering clothoid curves as trajectory and a quasi-static linearised bicycle model. A feedforward steering and longitudinal input are calculated based on the understeering gradient [71] and the ellipse adherence of

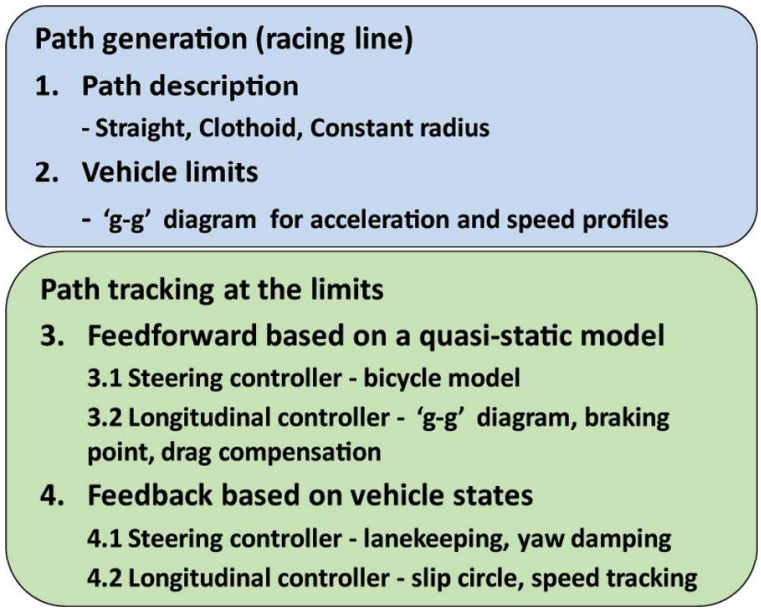

Figure 2.5: Autonomous controller based on clothoid curves.[70]

the tyres. Feedback controllers are then added to assure at-limit handling driving and vehicle stability. Specifically: a lanekeeping feedback with lookahead for tracking purposes, a yaw damping feedback to minimise yaw oscillation and a slip circle feedback which is used to control rear tyre slip so that rear axle saturation is avoided. The same authors tried using the centre of percussion [72] for the feedforward and feedback steering control as to increase controller robustness to external disturbances coming from the rear axle. A summary of this controller is shown in Figure 2.5.
Into the other category fall the controllers that combine lateral and longitudinal dynamics in one algorithm. This is mostly done with MPC which is an established method for controlling a dynamical system by using predictions of the state evolution into the future. The control input is obtained as the solution to a constrained optimisation problem, solved in real-time. The optimisation is solved with a receding horizon strategy whereby only the first input of the optimisation solution is used after which the horizon is shifted by one time (or space) step and the optimisation is repeated. The concept of MPC is shown in Figure 2.6.
Prokop et al. [74] are within the first to have explored this technique and showed how MPC is very similar to the way that human drivers plan their trajectories when driving. However, the interest of this work is in tracking rather than path planning. Despite being used for lateral control only, some of the first papers on MPC in AD for tracking purposes are the ones of Borrelli et al. [75] and Keviczky et al..[76] Kraus et al. [77] and Cui et al. [33] also use a similar approach and at the same time estimate the vehicle states, however, while operating far from tyre saturation. The approach of Borrelli et al. was extended by Falcone et al. [78] to control all inputs of the vehicle. The MPC was solved with a Linear Time Varying (LTV) method to decrease the computational complexity given by the full NLP, although a very simplified vehicle model is used in the formulation presented.

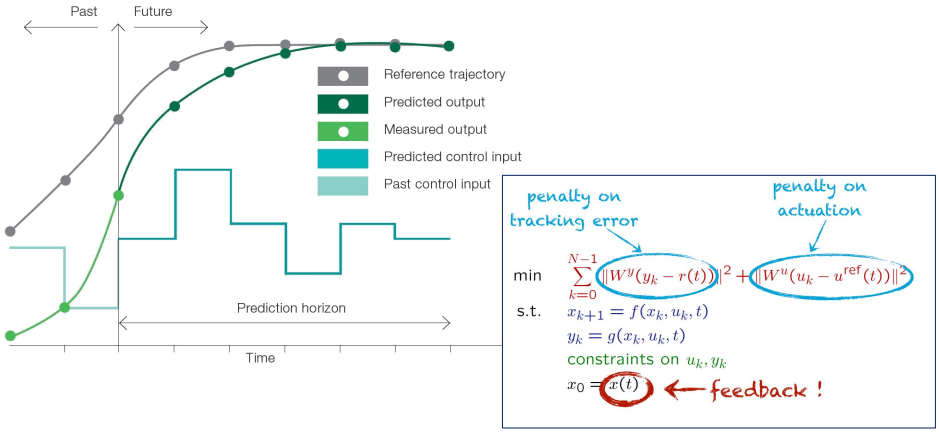

Figure 2.6: MPC structure.[73]

Focusing on the control of the vehicle at its limits, the LTV-MPC was also used by Timings et al. [79, 80] only for lateral control. However, unlike in the aforementioned papers, the objective function of the optimisation was to maximise progression along the track.
To obtain the real optimal control input sequence and exploit a vehicle's performance, it is necessary to consider high fidelity vehicle models, properly define an objective function with which time can be minimised and solve the NLP without linearising while maintaining real-time feasibility. Also, one of the critical aspects is the horizon length of the MPC. Since a full scale vehicle driving around a track requires long braking distance, especially at high speed, to complete an entire lap, long horizons are generally required for the robotic controller.
Four different approaches have been developed in recent years, all of which address some of the presented issues.
The first method proposed by Liniger et al. [81] uses a hierarchical control structure, where in the upper level the progress optimal trajectory is computed using a tree search algorithm and a simplified vehicle model. In the lower level the previously computed trajectory is tracked using a MPC, which uses a more accurate vehicle model with nonlinear tyre forces. To deal with the issue of the horizon length, viability theory is used to compute a terminal constraint that guarantees safety of all trajectories remaining in that set. This terminal constraint allows us to run shorter horizons and speed up the tree search. The same authors extended the method in [82] to allow for game theoretic decision-making of two cars racing head-to-head. However, the viability theory approach used in these papers requires for the terminal constraint to be pre-computed and this operation requires a few hours, thus it is not possible to use it in real-time.
The second, is based on an iterative MPC in which the autonomous driver learns lap after lap as for Rosolia et al..[83, 84] This approach is however computationally very expensive and requires the car to drive around the circuit many times before performing at its limits.
Another approach is to use the Model Predictive Contouring Control (MPCC) formulation as suggested by Liniger et al..[85] With this method the controllers ob-

jective is a trade-off between the progress along the track and the contouring error. In this work, the Quadratic Programming (QP) approximation is solved by means of an interior point method which allows fast sampling times and long horizons. The final approach which can be found in the literature is to transform the state-space into space dependant, this method has the great advantage of having an explicit formulation of time which can be used as an objective function. On the other hand, one of the disadvantages is that the horizon is space dependant, thus, for higher velocities, the same lookahead as for low velocities is used. This approach was first used by Gao et al..[86] An extension of this but with a more complete vehicle model was done by Frasch et al. [87] who also include load transfer and wheel dynamics. These controllers maintain real-time feasibility. However, a manoeuvre with a very short horizon is simulated, e.g an obstacle avoidance manoeuvre with approximately twenty-five meters lookahead. Also, these authors solve the QP approximation obtained by linearising the NLP. This method is also known as Real Time Iteration (RTI) and allows the solving of the optimisation very quickly but most of the time resulting in a suboptimal solution due to only solving one QP, instead of solving the SQP approach to convergence. Verschueren et al. [88, 89] use the same approach, however, focusing on an autonomous racing application and using simpler models to facilitate real-time implementation. Finally, Anderson et al. [90] also transform the problem into space dependant and integrate a genetic algorithm to adjust the costs of the optimisation objective function between minimum time and maximum exit velocity, a summary of this controller is shown in Figure 2.7. The main disadvantage of this approach is the high computational burden to solve the genetic algorithm.

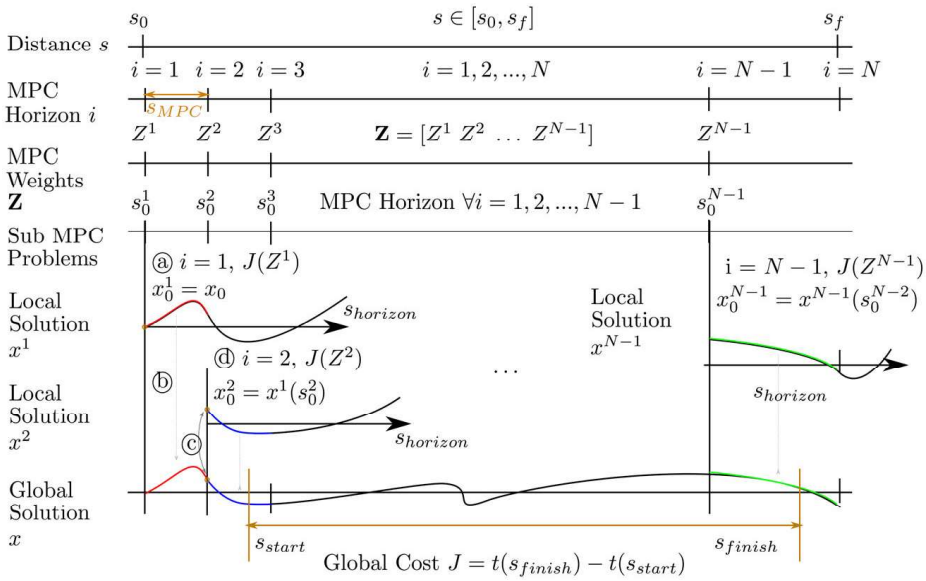

Figure 2.7: Genetic algorithm integration in the path-following problem.[90]

The main issues with all the discussed papers are that none of them are using a high fidelity model and that the prediction horizons used are not long enough for

autonomous racing on a race track. This leads mainly to two shortcomings, first, the controller cannot achieve the best possible performance the vehicle could deliver, since, in this range of a vehicle's performance, even little changes can make a big difference. Second, to accurately predict critical decision points, long horizons are needed. This is very important, for example, for braking points.

2.2.2 Contribution

In this research, a robotic controller for AD that can tackle the previously discussed shortcomings of existing methods is proposed. A novel hierarchical control scheme composed by two NMPCs is introduced.
The contributions are threefold; first, a novel hierarchical structure is proposed, wherein the upper level computes a velocity profile given by a simple point-mass model with acceleration constraints representing the tyre limits. Since the problem is very simple it can be solved for long prediction horizons, even over several hundred meters. The velocity profile is then used as a terminal constraint in the lower level, which can run on short prediction horizons and thereby retain real-time feasibility. By solving both problems in a receding horizon fashion, this method allows for maximum flexibility against disturbances, and it would even allow updating the path online. This is the main advantage compared to computing the velocity profile with a more sophisticated model offline. Additionally, it is possible with the proposed algorithm to run the controller in less controlled environments.
The second contribution is the low-level controller. Even though the controller is similar to the one found in [87] the proposed approach uses a higher fidelity model using the full Pacejka Magic Formula 6.1 [91, 92] with all combined effects and a LSD formulation is included in the dynamics, thus the torque limits of the engine-gearbox drivetrain are realistically approximated. Finally, the approach used does solve the full NLP and not only approximates it using a RTI method.
The performance of the controller is verified using the commercial software Vi-Grade and a validated vehicle model in a MiL simulation. The simulations show that the proposed method is indeed able to control a car at-limit handling and achieves expert human driver-like performance.

2.3 Passive Vehicle Dynamics

The final topic which was studied in this research is the influence of AD on passive vehicle dynamics. In fact commercial vehicles have typically been designed considering the human in control, hence to be stable. Consequently, these tend to have an understeering characteristic (under steady-state and linear tyre range assumptions). These features are desired since they tend to make the vehicle safer and less prone to loss of control when external perturbations are applied. Additionally, understeering vehicles are more intuitive to drive since counter steering manoeuvers are generally not required. It is a well-known fact that understeering vehicles tend to saturate the front tyres before the rear ones, thus decreasing the overall attainable peak lateral acceleration. Also, most of these vehicles tend to become unstable at higher accelerations as shown in the papers of Bundorf et al. [93] and more recently Allen et al. [44] and Hac et al..[94]

2.3.1 Literature review

Over the last decades, many researchers have worked on active systems which allow the stabilisation of the vehicle, a survey of which is done by Mousavinejad et al..[95] Some of the first studies were by Bosch and Van Zanten; their systems aim to stabilise a car in critical conditions driven by a human. The most important ones are the Vehicle Dynamics Control (VDC) system of Bosch [96] and the Electronic Stability Program (ESP).[97] More recently many more types of control systems have been used to improve vehicle stability considering also the influence of the driver as an external disturbance, as in the case of Carvalho et al. [98] who use robust control techniques. Other types of vehicle stability controllers, which do not include the driver as an uncertainty, are the controllers of Le et al. [99] who base their systems on adaptive sliding-mode, Jalali et al. [100] who use MPC, Lu et al. [101] who use proportional-derivative control (PD) and Morgando et al. [102] who use wheel forces measurements. Additionally, researchers such as Huijun et al. [103] have worked on stability controllers considering uncertain or unknown vehicle parameters which require prior identification. Recently, stability has been studied for critical manoeuvres such as drifting by Velenis et al..[104] On the other hand, in the past few years, some researchers have studied how to optimise vehicle parameters to obtain the best performance from a vehicle considering a completely passive vehicle, as in the case of the already cited work of Perantoni et al..[54] Also for active vehicles such as electric ones with four wheel torque vectoring, the passive vehicle dynamics has been analysed to find the best configuration for time optimality by Smith et al..[105] These two works both use optimal control to compute the minimum time inputs. However, both of these consider offline optimisation and simulation. Until now nobody has studied how the passive vehicle dynamics can be modified with an autonomously controlled vehicle. With the recent development in AD, it could be possible to gain peak lateral grip despite the loss of open-loop stability. Some studies by Mastinu et al. [106] on pure vehicle stability for autonomous cars have been carried out recently. By showing that an autonomous car can better handle the vehicle at its limits, the aim is to demonstrate that new horizons are opening up in the design of cars.

2.3.2 Contribution

Specifically, in this research the effect of passive stability of a vehicle (intended as open-loop stability) driven by a robotic controller rather than a human driver is analysed. The goal is to show how passive stability can be drastically decreased for certain driving conditions. Thereby, the contributions can be summarized as follows: first, the limits of some reference drivers due to instability are determined. Several vehicle models were implemented in a numerical environment with different values of static margin and peak grip. These vehicles were then tested on several double lane change manoeuvres on a dynamic driving simulator. Second, it is shown that a nonlinear model predictive controller is able to go beyond the performance of the human drivers and successfully achieve the double lane change for all the tested models. Even though NMPC is an often proposed control technique for autonomous cars (see Section 2.2.1), in this work it is shown that NMPC and unstable passive vehicle dynamics can be combined to achieve a more dynamically capable closed-loop system. Both a pure lateral controller and a combined lateral-longitudinal controller were implemented. All simulations are run without active

stability systems such as ESP or ABS. Note that a manoeuvre where low passive stability is an advantage is investigated, however, for other situations high passive stability is still needed. Thus, additional active controls should be addressed to actively vary the passive stability depending on the driving situation, e.g. active roll stiffness.

2.4 Trajectory Generation

The control framework developed in this work is based on a known reference path. In the literature review several methods to generate an ideal path to minimise lap time were revealed. In this work a rather simple approach to find a good reference path by applying the minimum curvature trajectory approach of Braghin, Cheli et al. [107] is used. The circuit used in this work as a reference is Hockenheimring in its short configuration and in Figure 2.8 the results of the minimum curvature algorithm on this circuit are shown.

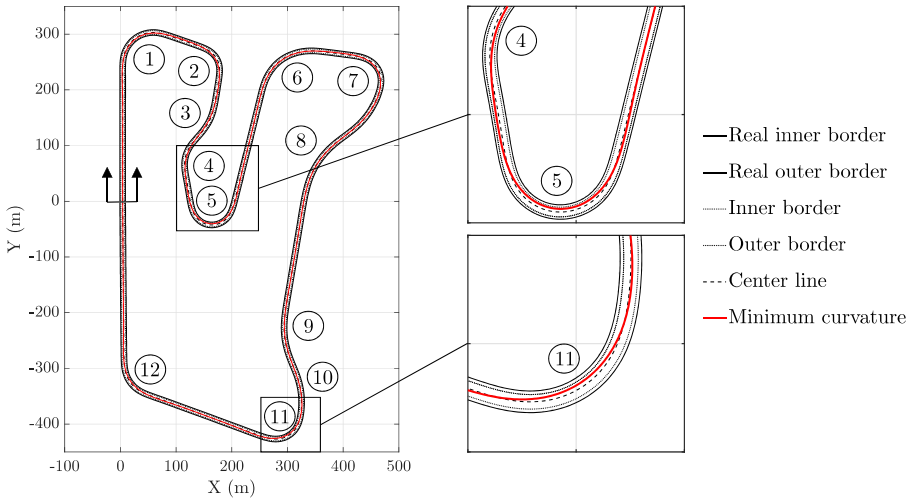

Figure 2.8: Minimum curvature trajectory.

First the track is discretised in various segments as shown Figure 2.9 and the position of a given point for every i^{th} segment is identified,

$$\begin{aligned}\vec{\mathbf{P}}_i = x_i\vec{\mathbf{i}} + y_i\vec{\mathbf{j}} &= [x_{r,i} + \epsilon_i(x_{l,i} - x_{r,i})]\vec{\mathbf{i}} + [y_{r,i} + \epsilon_i(y_{l,i} - y_{r,i})]\vec{\mathbf{j}} \\ &= [x_{r,i} + \epsilon_i \Delta x_i]\vec{\mathbf{i}} + [y_{r,i} + \epsilon_i \Delta y_i]\vec{\mathbf{j}}.\end{aligned} \tag{2.1}$$

The parameter ϵ_i determines the position of each point along the track width t_w relatively to the inner boarder of the track. Each point of $\vec{\mathbf{P}}_i$ is linked to the adjacent ones through natural cubic splines [108] which identify the trajectory in

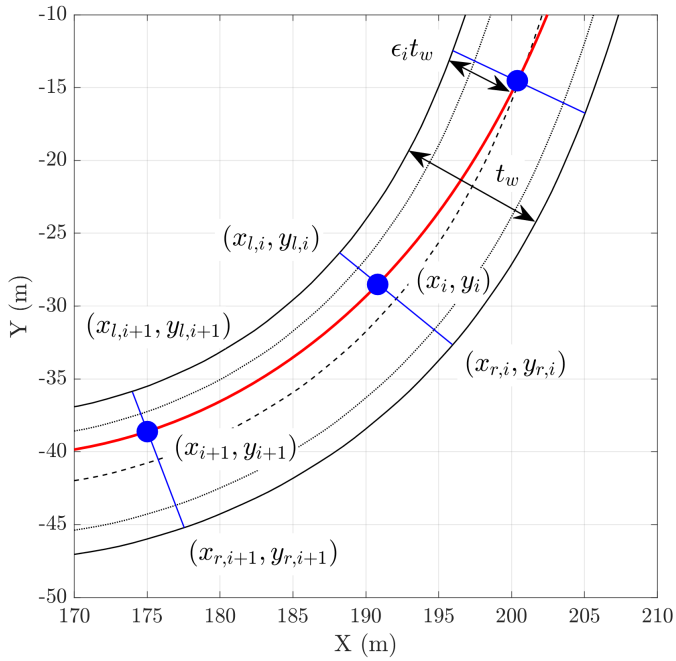

Figure 2.9: Trajectory discretisation.

every segment. To do this the reference curve must be expressed with a curvilinear abscissa s approach. The advantage of using natural cubic splines is that these are smooth and continuous in first and second derivatives and, therefore, continuous in curvature. The cubic spline formulation can be written with the following set of equations,

$$\begin{cases} x_i(\tau) &= a_{i,x} + b_{i,x}\tau + c_{i,x}\tau^2 + d_{i,x}\tau^3, \\ y_i(\tau) &= a_{i,y} + b_{i,y}\tau + c_{i,y}\tau^2 + d_{i,y}\tau^3, \\ \tau(s) &= \frac{s - s_{i,0}}{ds_i}. \end{cases} \tag{2.2}$$

Where a third order polynomial function of τ describes the trajectory in each segment. The curvilinear abscissa has been normalised to the length of the i^{th} track segment. Note that the discretisation takes place in τ, i.e. the parametrisation variable space, as opposed to time or trajectory arc length. The track curvature $\hat{k}$ of each element is determined according to the following expression,

$$\begin{aligned} \hat{k}^2 = \left(\frac{d^2x(s)}{ds^2}\right)^2 + \left(\frac{d^2y(s)}{ds^2}\right)^2 &= \left(\frac{d^2x(\tau)}{d\tau^2}\left(\frac{d\tau(s)}{ds}\right)\right)^2 + \left(\frac{d^2y(\tau)}{d\tau^2}\left(\frac{d\tau(s)}{ds}\right)\right)^2 \\ &= \left(\frac{d\tau(s)}{ds}\right)^4 \left[\left(\frac{d^2x(\tau)}{d\tau^2}\right)^2 + \left(\frac{d^2y(\tau)}{d\tau^2}\right)^2\right]. \end{aligned} \tag{2.3}$$

If every segment of the track has the same length $\tilde{s}$, the curvature expression can be simplified to the following equation,

$$\hat{k}^2 = \Big(\frac{1}{d\tilde{s}}\Big)^4 \left[\Big(\frac{d^2x(\tau)}{d\tau^2}\Big)^2 + \Big(\frac{d^2y(\tau)}{d\tau^2}\Big)^2 \right]. \tag{2.4}$$

The track curvature $k(s)$ can thus be minimised considering the sum of the all the curvatures in each segment and found as follows,

$$k^2 = \sum_{i=0}^{n-2} \left[\Big(\frac{d^2x(\tau)}{d\tau^2}\Big)^2 + \Big(\frac{d^2y(\tau)}{d\tau^2}\Big)^2 \right]. \tag{2.5}$$

The identification of the minimum curvature trajectory can be reduced to an optimisation problem the goal of which is to identify all ϵ_i parameters, which become the independent variables. A new vector $\bar{\epsilon}$ can be defined as $\bar{\epsilon} = [\epsilon_0, \epsilon_2, ..., \epsilon_{n-1}]$ with n being the number of points and with $\epsilon_i \in [0, 1]$. The optimisation problem can therefore be formulated as follows,

$$\begin{aligned} \min_{\epsilon} \quad & \sum_{i=0}^{n-2} \left[\Big(\frac{d^2x(\tau)}{d\tau^2}\Big)^2 + \Big(\frac{d^2y(\tau)}{d\tau^2}\Big)^2 \right] \\ \text{s.t.} \quad & 0 \le \epsilon_i \le 1\,, \quad \forall i \in \{0, 1, ..., n-1\} \\ & x_{r,i} + \epsilon_i \Delta x_i \\ & y_{r,i} + \epsilon_i \Delta y_i\,. \end{aligned} \tag{2.6}$$

Note that the parametrisation of the trajectory is fixed to coincide with the track centreline parametrisation before solving the optimisation problem and is kept constant so that the optimisation problem is convex. Indeed, whenever any feasible trajectory's arc length parametrisation differs from the track centreline parametrisation, its aggregate curvature will not be correctly measured. In fact, the unique trajectory where this assumption holds in a strict sense and where the curvature is precisely captured is the one whose arc length parametrisation is the track centreline parametrisation itself. Consequently, this approach compares tracks based on their curvature but fails to correctly measure curvature in all but one (trivial) case. Suboptimal solutions are thus expected from a curvature minimisation point of view. Additionally, instead of minimising a finite sum of k samples, the sum of squared samples are minimised, which will lead to a distorted solution.
One of the advantages of using cubic splines for interpolating is the ease of sampling second derivatives at the spline knots. Considering the following notations:

- Second order derivatives in x and y: $z_{x,i} \coloneqq \Big(\frac{d^2x(\tau)}{d\tau^2}\Big)$ and $z_{y,i} \coloneqq \Big(\frac{d^2y(\tau)}{d\tau^2}\Big)$,
- Distance between two sample points: $h_i \coloneqq \tau_{i+1} - \tau_i = \Delta\tau$,
- $b_{x,i} \coloneqq \frac{1}{h_i}(x_{i+1} - x_i)$ and $b_{y,i} \coloneqq \frac{1}{h_i}(y_{i+1} - y_i)$.

The samples of $z_{x,i}$ and $z_{y,i}$ can be found by solving simple tridiagonal systems. The exact form of these systems depends on the boundary conditions imposed on the splines. With natural cubic splines, the boundary conditions are $\tilde{z}_{x,0} = 0$, $\tilde{z}_{y,0} = 0$ and $\tilde{z}_{x,n-1} = 0$, $\tilde{z}_{y,n-1} = 0$. The spline problem can be reformulated,

$$\begin{cases} \tilde{H}_s \tilde{z}_x &= \tilde{b}_x, \\ \tilde{H}_s \tilde{z}_y &= \tilde{b}_y. \end{cases} \tag{2.7}$$

Where $\tilde{H}_s$ can be defined as follows,

$$\tilde{H}_s = \begin{bmatrix} 2(h_0+h_1) & h_1 & & & & \\ h_1 & 2(h_1+h_2) & h_2 & & & \\ & \ddots & \ddots & \ddots & & \\ & & h_{n-4} & 2(h_{n-4}+h_{n-3}) & h_{n-3} & \\ & & & h_{n-3} & 2(h_{n-3}+h_{n-2}) \end{bmatrix}. \tag{2.8}$$

The second derivatives $\tilde{z}_x$, $\tilde{z}_y$ can be defined as,

$$\tilde{z}_x = \begin{bmatrix} z_{x,1} \\ \vdots \\ z_{x,n-2} \end{bmatrix}, \qquad \tilde{z}_y = \begin{bmatrix} z_{y,1} \\ \vdots \\ z_{y,n-2} \end{bmatrix}. \tag{2.9}$$

And finally, $\tilde{b}_x$ and $\tilde{b}_y$ can be defined as,

$$\tilde{b}_x = 6 \begin{bmatrix} b_{x,1} - b_{x,0} \\ \vdots \\ b_{x,n-2} - b_{x,n-3} \end{bmatrix}, \qquad \tilde{b}_y = \begin{bmatrix} b_{y,1} - b_{y,0} \\ \vdots \\ b_{y,n-2} - b_{y,n-3} \end{bmatrix}. \tag{2.10}$$

The linear system of equation in this form is under-determined. However, the use of natural cubic splines provides two additional conditions. The first and last second derivatives have to be equal to zero. Thus, the system converts into a determined linear system of equations.

$$z_{x,i} = z_{y,i} = 0, \quad i = \{0, n-1\}. \tag{2.11}$$

Matrix $\tilde{H}_s$ is extended to H_s which has dimension $n \times n$, as are $\tilde{z}_x$ and $\tilde{z}_y$ to z_x and z_y which have dimension $n \times 1$ and finally $\tilde{b}_x$ and $\tilde{b}_y$ to b_x and b_y which also have dimension $n \times 1$.

$$H_s = \begin{bmatrix} 1 & 0_{1\times n-2} & 0 \\ h_0 & \tilde{H}_s & h_2 \\ 0_{n-3\times 1} & & 2h_{n-2} \\ 0 & 0_{1\times n-2} & 1 \end{bmatrix}, \tag{2.12}$$

$$z_x = \begin{bmatrix} z_{x,0} \\ \tilde{z}_x \\ z_{x,n-1} \end{bmatrix}, \; z_x = \begin{bmatrix} z_{y,0} \\ \tilde{z}_x \\ z_{y,n-1} \end{bmatrix}, \tag{2.13}$$

$$b_x = \begin{bmatrix} 0 \\ \tilde{b}_y \\ 0 \end{bmatrix}, \; b_y = \begin{bmatrix} 0 \\ \tilde{b}_y \\ 0 \end{bmatrix}. \tag{2.14}$$

H_s is strictly diagonal dominant and thus invertible,

$$z_x = H_s^{-1} b_x, \quad z_y = H_s^{-1} b_y. \tag{2.15}$$

Since b_x depends on h_i and x, it can be written as follows,

$$b_x = B_s x, \quad b_y = B_s y. \tag{2.16}$$

Where the matrix B_s can be written as follows,

$$B_s \coloneqq 6 \begin{bmatrix} 0 & 0 & 0 & & & \\ \frac{1}{h_0} & -\left(\frac{1}{h_0}+\frac{1}{h_1}\right) & \frac{1}{h_1} & & & \\ & \frac{1}{h_1} & -\left(\frac{1}{h_1}+\frac{1}{h_2}\right) & \frac{1}{h_2} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \frac{1}{h_{n-3}} & -\left(\frac{1}{h_{n-3}}+\frac{1}{h_{n-2}}\right) & \frac{1}{h_{n-2}} \\ & & & 0 & 0 & 0 \end{bmatrix}. \tag{2.17}$$

This finally turns the entire spline problem into the following system of equations,

$$z_x = H_s^{-1} B_s x, \tag{2.18}$$

$$z_y = H_s^{-1} B_s y. \tag{2.19}$$

Where $H_s^{-1}B_s$ is constant since the parametrisation is constant and not a part of the optimisation. As already mentioned, the spline parametrisation is chosen to be constant since introduction of the parametrisation as a variable in the optimisation problem would cause it to be non-convex.
The objective function in the optimisation problem in (2.6) can be rewritten by substituting (2.18) and obtaining,

$$z_x^T z_x + z_y^T z_y = x^T (H_s^{-1}B_s)^T (H_s^{-1}B_s) x + y^T (H_s^{-1}B_s)^T (H_s^{-1}B_s) y. \tag{2.20}$$

Defining $G_s \coloneqq H_s^{-1}B_s$, $\tilde{C}$ can be defined as follows,

$$\tilde{C} = x^T G_s^T G_s x + y^T G_s^T G_s y. \tag{2.21}$$

Considering the definition of $x = x_{r,i} + \epsilon_i \Delta x_i$ and $y = y_{r,i} + \epsilon_i \Delta y_i$ as in (2.1) the following is obtained,

$$\tilde{C} = \epsilon^T H_{\tilde{C}} \epsilon + B_{\tilde{C}} \epsilon + C_{\tilde{C}}. \tag{2.22}$$

Where,

$$H_{\tilde{C}} \coloneqq \Delta_x^T G_s^T G_s^T \Delta_x + \Delta_y^T G_s^T G_s^T \Delta_y, \tag{2.23}$$

$$H_{\tilde{C}} \coloneqq 2(x_r^T G_s^T G_s^T \Delta_x + y_r^T G_s^T G_s^T \Delta_y), \tag{2.24}$$

$$H_{\tilde{C}} \coloneqq x_r^T G_s^T G_s^T x_r + y_r^T G_s^T G_s^T y_r. \tag{2.25}$$

Finally, the least curvature problem can be formulated as a Quadratic Program (QP).

$$\begin{aligned} \min_{\epsilon} \quad & \tilde{C} = \epsilon^T H_{\tilde{C}} \epsilon + B_{\tilde{C}} \epsilon + \text{const.} \\ \text{s.t.} \quad & 0 \le \epsilon_i \le 1\,, \quad \forall i \in \{0, 1, ..., n-1\}. \end{aligned} \tag{2.26}$$

One more variation will be considered to the formulation, specifically, instead of minimising the sum of squared curvature samples $||k^2||_1$, the maximum squared curvature sample is minimised, i.e. $||k^2||_\infty$.
Since for some trajectories the mismatch in parametrisations highly penalises the results and the curvature is overestimated to the point of not being able to compete with solutions that are effectively worse but more similar to the centreline,

an iterative approach is used. Each iteration consists of solving the least curvature problem and finding a new arc length parametrisation (approximated by a chord length parametrisation), to be assumed as given in the next iteration. The end condition of the process is given by a threshold on $||\epsilon - \epsilon^{+}||$.
The results of the iterative procedure are shown in Figure 2.10.

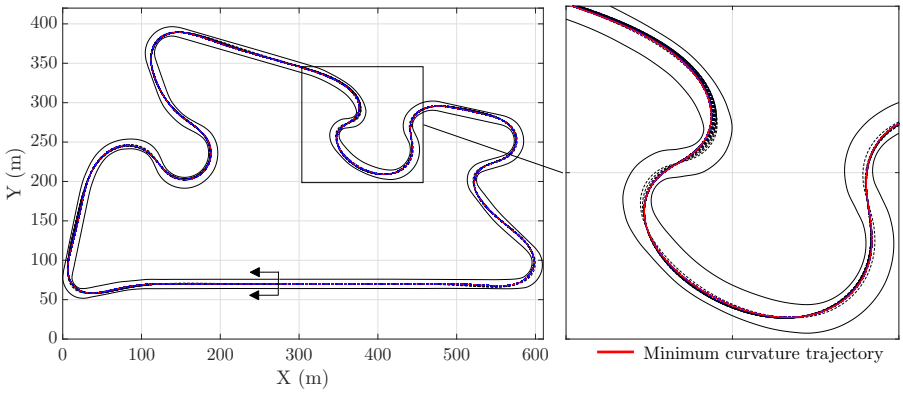

Figure 2.10: Minimum curvature iterative procedure.

Note that the sum of squared curvature samples can be shown to be non-increasing in the number of iterations.

Chapter 3

Sideslip Angle Estimation

This chapter describes the sideslip angle estimator algorithm in detail. The results of the proposed observer on both numerical and experimental tests are discussed. The chapter starts with a quantitative and qualitative analysis of the sideslip angle, and this analysis highlights the need for an estimator. The proposed observer composed of an ANN and UKF is then described. Both modules of the observer are discussed separately. Finally, the results of the algorithm and its effectiveness are shown.

3.1 Sideslip angle analysis

Before describing the observer design, the coupling and relationships of the various vehicle states are analysed under different working conditions. Particular emphasis is put on the sideslip angle and how it is coupled with other vehicle quantities such as accelerations and yaw rate. For neural networks it is particularly important to analyse how the output depends on the inputs since these depend on each other in the ANN with an algebraic formulation. This relationship has to be non-linear to be able to correctly predict the non-linear dynamics of the system under all working conditions. It is particularly important to analyse every situation, thus, the manoeuvres which are analysed are ramp steer manoeuvres, which characterise the steady-state behaviour, step steer manoeuvres, which characterise the transient behaviour and sine steer manoeuvres, which characterise the frequency behaviour of the vehicle. All the following analyses will be done for a fixed longitudinal velocity. These analyses are very important to choose the correct dataset for the ANN.

3.1.1 Steady-state behaviour

The steady-state behaviour can be studied considering the stability derivative notation used in [7] to describe the lateral and yaw vehicle dynamics. Note that for this formulation the well-known bicycle linear model as in Figure 3.1 is considered. Considering longitudinal velocity v_x, lateral velocity v_y and yaw rate $\dot{\psi}$ with respect to a vehicle reference system with origin in its centre of gravity (Cg) and vertical axis pointing upwards; and approximating the longitudinal velocity with the absolute velocity of the centre of gravity V, the equations of motion of the

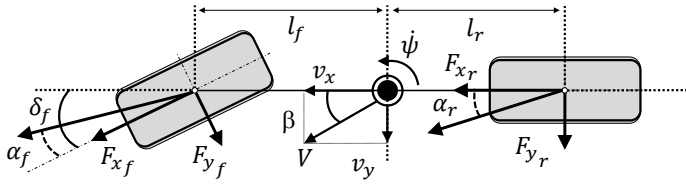

Figure 3.1: Bicycle model.

vehicle can be written as follows,

$$mV\dot{\psi} = Y_\beta \beta + Y_{\dot{\psi}} \dot{\psi} + Y_\delta \delta, \tag{3.1}$$

$$0 = N_\beta \beta + N_{\dot{\psi}} \dot{\psi} + N_\delta \delta. \tag{3.2}$$

With β being the sideslip angle, m vehicle mass and δ being the wheel toe angle. The definition of the various stability derivatives are shown in Table 3.1 considering the front wheelbase l_f and rear wheelbase l_r as respectively the distance between the front axle and Cg and rear axle and Cg; and the front tyre cornering stiffness C_f and rear tyre cornering stiffness C_r. Cornering stiffness is defined as the tangent line to the lateral tyre force curve to slip angle in its linear range and is dependant exclusively on tyre characteristics. Only yaw damping and lateral

Table 3.1: Stability derivates formulation and definition.

Stability derivate	Formulation	Definition
Damping-in-sideslip	$Y_\beta = C_f + C_r$	$Y_\beta = {}^{\partial Y}/_{\partial \beta}$
Lateral force/yaw coupling	$Y_{\dot{\psi}} = \frac{1}{V}\left(l_f C_f - l_r C_r\right)$	$Y_{\dot{\psi}} = {}^{\partial Y}/_{\partial \dot{\psi}}$
Control force	$Y_\delta = -C_f$	$Y_\delta = {}^{\partial Y}/_{\partial \delta}$
Direction stability	$N_\beta = l_f C_f - l_r C_r$	$N_\beta = {}^{\partial N}/_{\partial \beta}$
Yaw damping	$N_{\dot{\psi}} = \frac{1}{V}\left(l_f^2 C_f + l_r^2 C_r\right)$	$N_{\dot{\psi}} = {}^{\partial N}/_{\partial \dot{\psi}}$
Control moment	$N_\delta = -l_f C_f$	$N_\delta = {}^{\partial N}/_{\partial \delta}$

force/yaw coupling depend on vehicle velocity, the others depend only on the cornering stiffness' and other constant parameters. Yawing velocity response, lateral acceleration response and sideslip angle response to control (steering angle) are

expressed as follows,

$$\frac{\dot{\psi}}{\delta} = \frac{Y_\beta N_\delta - N_\beta Y_\delta}{N_\beta Y_{\dot{\psi}} - N_\beta mV - Y_\beta N_{\dot{\psi}}} = \frac{Y_\beta N_\delta - N_\beta Y_\delta}{Q}, \tag{3.3}$$

$$\frac{a_y}{\delta} = \frac{V\big(Y_\beta N_\delta - N_\beta Y_\delta\big)}{N_\beta Y_{\dot{\psi}} - N_\beta mV - Y_\beta N_{\dot{\psi}}} = \frac{V\big(Y_\beta N_\delta - N_\beta Y_\delta\big)}{Q}, \tag{3.4}$$

$$\frac{\beta}{\delta} = \frac{Y_\delta N_{\dot{\psi}} - N_\delta\big(Y_{\dot{\psi}} - mV\big)}{N_\beta Y_{\dot{\psi}} - N_\beta mV - Y_\beta N_{\dot{\psi}}} = \frac{Y_\delta N_{\dot{\psi}} - N_\delta\big(Y_{\dot{\psi}} - mV\big)}{Q}. \tag{3.5}$$

Comparing the control responses, it can be seen that yaw rate, lateral acceleration and sideslip angle have a behaviour that depends on the vehicle velocity in a different way. Thus, even in a simplified vehicle model acting within the linear range of the tyre force curve and in stationary conditions, the vehicle states are coupled and the system is non-linear. Moreover, once again, the various states highly depend on vehicle velocity which needs to be estimated itself and on the tyre force curve.
The steady-state behaviour can be analysed by considering a ramp steer manoeuvre. Because of the slow variation in steering wheel angle, every value of steering angle corresponds to a separate steady-state condition. The variation of sideslip angle, lateral acceleration and yaw rate to steering angle is represented by the control response equations (3.3), (3.4) and (3.5). These depend on the steady-state characteristics of the vehicle which can be summed up with the understeering gradient (UG) expressed as follows,

$$UG = \delta - \delta_{\text{Ack}} = -\alpha_f + \alpha_r = \frac{mg}{l_f + l_r}\left(\frac{l_f}{C_f} - \frac{l_r}{C_r}\right) = mg\left(\frac{N_\beta}{Y_\delta N_\beta - N_\delta Y_\beta}\right). \tag{3.6}$$

Where α_f and α_r are the front and rear slip angles of the here considered bicycle model and δ_{Ack} corresponds to the kinematic steering angle.
The vehicle considered in this work has a positive UG, i.e. understeering vehicle and its steady-state behaviour can be seen in Figure 3.2, where the states have been normalised. Note that the following results have been obtained by means of the commercial software Vi-Grade, the vehicle model implemented in this software has been previously extensively validated. The results prove the aforementioned analytical results.
Hence, since steady-state manoeuvres do not depend on time, the states depend only on the current instant and no dynamics of the various states is of interest for the sideslip angle estimation. What is important is the non-linear behaviour which is mainly due to the tyres' saturation, hence, the amplitude of the various states.

3.1.2 Transient behaviour

The transient behaviour of the vehicle can be studied by means of step steer manoeuvres. It is important to excite the system with step inputs of different duration to analyse its full behaviour. In a vehicle with positive stability margin (i.e. understeering), it can be demonstrated that the vehicle is stable and the system has a critical damping factor smaller than one. For this reason the oscillations about the steady-state value caused by the continuous variations of slip angles will always converge to the steady-state value. Slip angle variations are due to the front and rear tyres generating an opposite yaw moment. Looking at sideslip angle, the

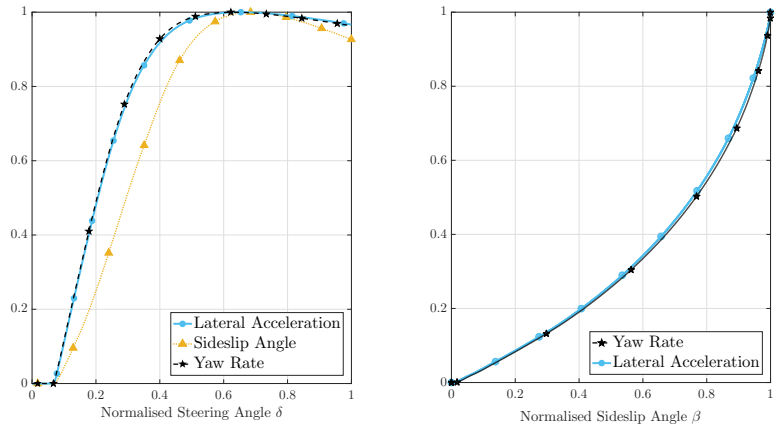

Figure 3.2: Ramp steer manoeuvre output.

variations in tyre slip angles continuously move the instant centre of curvature of the vehicle causing it to oscillate as well. The slip angle vibrations (given by the continuous variations in the kinematics) also cause the yawing acceleration and lateral acceleration to oscillate. A plot of the sideslip angle during different step impulses can be seen in Figure 3.3.

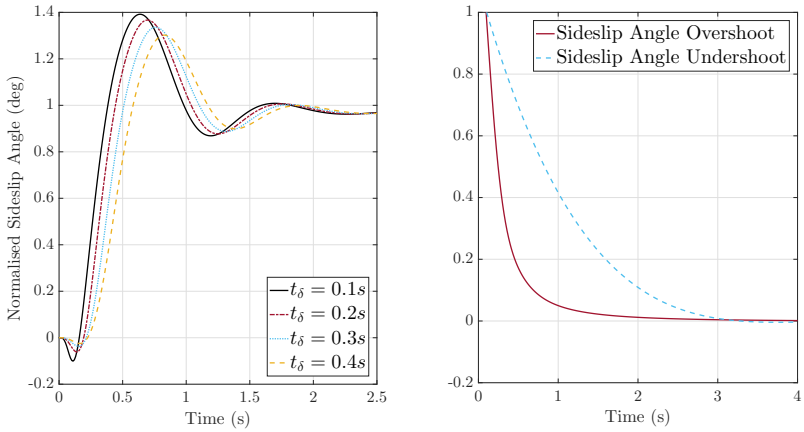

Figure 3.3: Step steer inputs at different duration and undershoot-overshoot dependency on input duration.

Sideslip angle also presents another characteristic, differently from lateral acceleration and yaw rate. In fact, whilst the latter have a monotone increasing trend in the first part of the manoeuvre, the sideslip angle decreases to negative values (i.e. the absolute velocity vector points to the inside of the curve) and only after a certain amount of time, depending on the excitation of the system, does it grow to

positive values. For the vehicle under examination the time period during which the sideslip angle undershoots goes from approximately 0,1s to 0,5s when varying the excitation time. When instead the excitation of the step input has a longer duration, the undershoot tends to evanish. The normalised trends with respect to their steady-state value have been plotted in Figure 3.3. The undershoot in the sideslip angle highly depends on the ratio between yaw acceleration and lateral acceleration (here named inertia ratio). The higher the ratio, the harder it will be for the vehicle to rotate, therefore, the lateral displacement will be higher with respect to the rotation, meaning the velocity vector will point towards the inside of the curve. The opposite happens when the ratio is very small, the vehicle's yaw rate will grow quickly whilst lateral velocity will not. Therefore, the velocity vector will immediately point towards the outside of the curve. Since the variation in this ratio influences the way the slip angles vary over time, its variation also varies the natural frequencies of the system. Generally, the higher the inertia ratio, the lower the natural frequency. The results of the sideslip angle step response when varying the inertia ratio of ±50% while keeping every other parameter constant can be viewed in Figure 3.4. The vehicle sideslip angle step response resembles

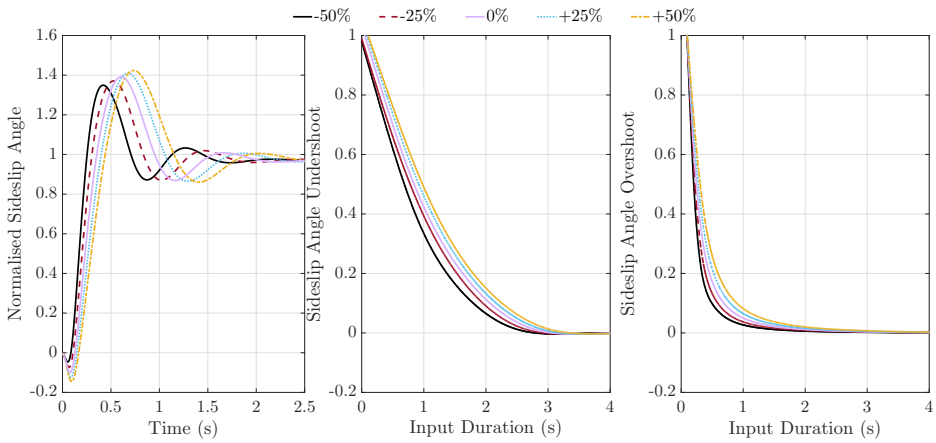

Figure 3.4: Sensitivity analysis of the ratio between yawing acceleration and lateral acceleration.

a second order system with a zero in the right half-plane, which causes the undershoot, and two complex conjugate poles in the left half-plane which cause the oscillations around the steady-state values and the attenuation of these vibrations. Considering a linear system the transfer function can be expressed as follows,

$$G(s) = \mu \frac{(1+\tau s)}{\left(\frac{s^2}{\omega_n^2} + 2\frac{\zeta_n}{\omega_n}s + 1\right)}. \tag{3.7}$$

Table 3.2 reports the percentage variations compared to the baseline vehicle of natural frequency, critical damping and the time constant of the second order system fitted models by varying the values of inertia ratio. As expected, the critical damping is subject to very little change since the tyres, wheelbase and centre of gravity position of the vehicle examined have not been changed and neither has

Table 3.2: Second order system parameter variation.

Inertia ratio	Natural frequency ω_n	Critical damping ζ_n	Time constant τ
-50%	$-18,33\%$	$-3,87\%$	$-49,41\%$
-25%	$-11,67\%$	$-3,23\%$	$-24,71\%$
0%	0%	0%	0%
$+25\%$	$+16,67\%$	$+3,23\%$	$+27,06\%$
$+50\%$	$+33,33\%$	$+6,45\%$	$+52,94\%$

the velocity of the manoeuvre. These are in fact the main parameters on which the damping of the vehicle depends. This can be partly seen in the damping terms (i.e. yaw damping and damping-in-sideslip) of the simplified stability derivative formulation. However, this formulation does not consider all parameters since it represents steady-state. Unlike damping, natural frequency and time constant vary in an almost linear way and with very high gradients. By means of the results found it is possible to trace the zeros and poles as in Figure 3.5. As visible by the Pole-Zero map, the higher the percentage of yawing torque to lateral acceleration, the smaller the right half plane zero, hence the undershoot, but also the closer the stable complex conjugated poles get to the right half plane, hence decreasing the stability margin. Thus, for neural network purpose, the relationship between the monotone curve of the measurable inputs and desired output will vary according to the ratio between yaw acceleration and lateral acceleration. It can be concluded that a new training will be required for a vehicle with different inertial characteristics. Comparing the steady-state and step-response behaviours, it can be seen how the input-output relationship between measurable states and sideslip angle is not uniquely defined but depends on the dynamics of the manoeuvre as one would expect.

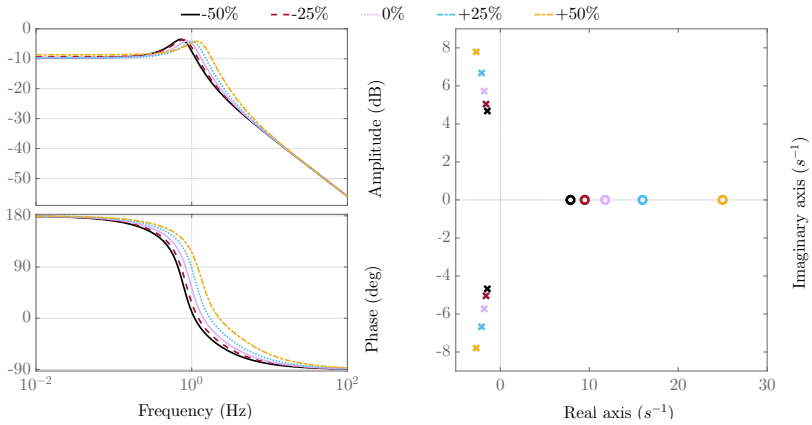

Figure 3.5: Fitted second order systems sideslip angle response.

3.1.3 Frequency behaviour

Finally, the frequency response of the system is analysed. Once again, since the interest is analysing input-output relationship for neural network purpose, measurable quantities of the system which strongly influence the vehicle sideslip angle are analysed, such as lateral acceleration and yaw rate. To do this the Bode plots of these normalised quantities mapped out against the maximum value obtained in these simulations are plotted. To trace the Bode plots and consider only the stationary component of the Laplace transform, several sine steer manoeuvres with equal vehicle velocity, steering amplitude and different frequency have to be simulated. The last period out of fifty of the input sine wave, i.e. steering angle, was taken as a time window to analyse gain and phase plots. The phase and gain were calculated respectively as the difference between the abscissa and ordinate of the peaks of the input-output sinusoids. The phase plots are very important since the variation in phase shift between the various measurements and sideslip angle complicate the estimation, especially since neural networks are a static algorithm. The yaw rate Bode plot shows once again a typical second order system behaviour. The peak value is at a frequency which corresponds to the optimal compromise between two factors. The first is the phase shift between front and rear tyre slip angles. The second factor is that as the frequency increases, there is less time for a velocity or rotational speed to build up given a certain acceleration. Therefore, yaw rate will tend to decrease. This phenomenon is degenerative, because as yaw rate decreases, tyre slip angles vary according to the axle to which they belong. Considering the most simple tyre slip angle formulation and the same bicycle model, these can be expressed as follows,

$$\alpha_f = \frac{v + rl_1}{V} - \delta, \tag{3.8a}$$

$$\alpha_r = \frac{v - rl_2}{V}. \tag{3.8b}$$

The front slip angle increases since yaw rate contribution is opposite to the steering angle contribution, therefore, yawing torque given by the front tyres will increase. On the contrary, the rear slip angle decreases and so does anti-yawing torque given by the rear tyres. The slip angle variation due to yaw rate decrease is not very noticable in the Bode gain plot. In the phase plot, however, this effect is quite important. In fact, the trend is monotonically decreasing, as could be expected due do the inertia of the system. For very low frequencies, yaw rate has a phase lead mapped out against steering angle when considering the maximum values of the sinusoids for phase calculation. Note that this is true only because of the way that the phase was calculated (time shift between peaks). When the transient is almost neglectable, so for low frequency inputs, the growth of total anti-yawing torque due to the increase in yaw rate in the single sinusoidal manoeuvre is greater than the increase of yawing torque given by the front tyres due to the increase in steering angle. Hence, yaw rate tends to have a phase lead measured against steering angle. As the frequency grows, transient effects gain greater importance and yaw rate tends to have a phase lag compared to steering angle.
A minor factor can also be found in the transient behaviour of the tyre due to its elasticity. The classic parameter used to represent the tyre transient behaviour is the relaxation length.[24] As the frequency grows, the force produced by the tyre tends to be lower respect to the steady-state value.

It can be concluded that yaw rate Bode gain plot, increases until a certain value after which it acts as a low-pass filter as in Figure 3.6. After having analysed yaw

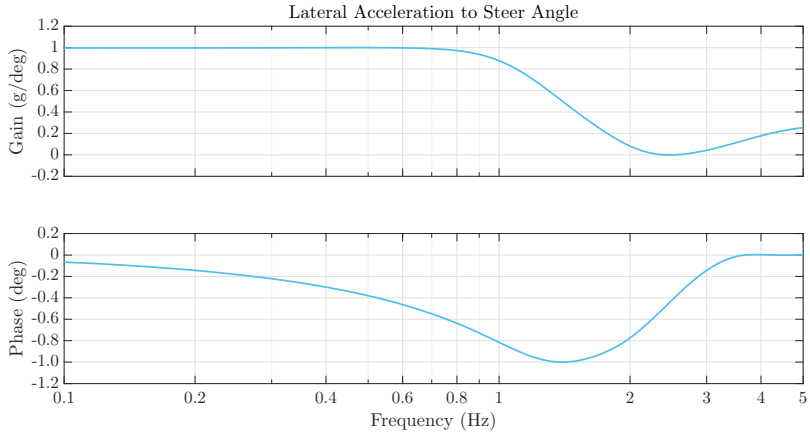

Figure 3.6: Yaw rate to steering wheel angle Bode plot.

rate frequency response of the vehicle, lateral acceleration response can be analysed. The gain diagram looks like a low-pass filter as well. The cutoff frequency depends mainly on mass and inertial properties of the vehicle. However, it also corresponds to the frequency at which yaw rate starts increasing. As previously explained, when yaw rate grows, the rear tyres' slip angle decreases whilst the front tyre slip angle increases. This reduction makes the total lateral force generated by the tyres decrease, despite the growth in front slip angle. This is true even when the tyres are in their linear range as in Figure 3.7 and tyre forces can be written as the product between cornering stiffness and slip angle. The reason is that the slip angle generated by the front tyre mainly depends on steering angle, and yaw rate adds only a very small contribution to it, contrarily to the rear tyres where slip angle depends only on lateral velocity and yaw rate. It is interesting to notice that the gain plot presents a minimum. This is because at higher frequencies lateral acceleration (and lateral velocity) and yaw rate are filtered so the rear slip angle is very small. Contrarily to the front tyres due to steering angle. Also, for high frequencies front and rear slip angles are almost in counter phase: hence, lateral forces generated by the front and rear wheels are opposite so a reduction in rear slip angle makes the total lateral force grow. The same thing can be said for the phase plot. At higher frequencies, when the slip angles depend mostly on steering angle, lateral acceleration has very little phase lag compared to steering angle. The sideslip angle frequency behaviour depends mainly on the frequency response of the states analysed so far. Since it is the angle between the velocity vector at the centre of gravity and the longitudinal axis of the vehicle, the variation of these two vectors can be analysed. Considering a constant longitudinal velocity, lateral velocity can be considered instead of absolute velocity. The vehicle heading angle (or yaw angle) which gives information on the orientation of the longitudinal axis in a fixed reference system, will look like yaw rate gain plot. However, since yaw angle is the integral over time of yaw rate, an increase in frequency will not guarantee a growth in yaw angle even considering positive yaw rate gain due to integration.

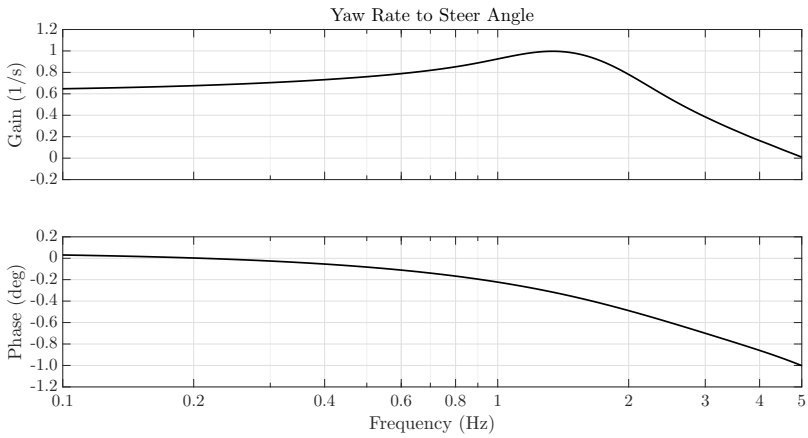

Figure 3.7: Lateral acceleration to steering angle Bode plot.

The same thing happens to lateral velocity which is the integral over time of $^{dv_y}/_{dt}$ which is kinematically equal to the difference between lateral acceleration and the product between longitudinal velocity and yaw rate.
Therefore, sideslip angle depends on the relationship between these integrals. Finally, the resulting Bode plot of sideslip angle to steering wheel angle can be seen in Figure 3.8. Because of its high dependence on lateral acceleration and yaw rate,

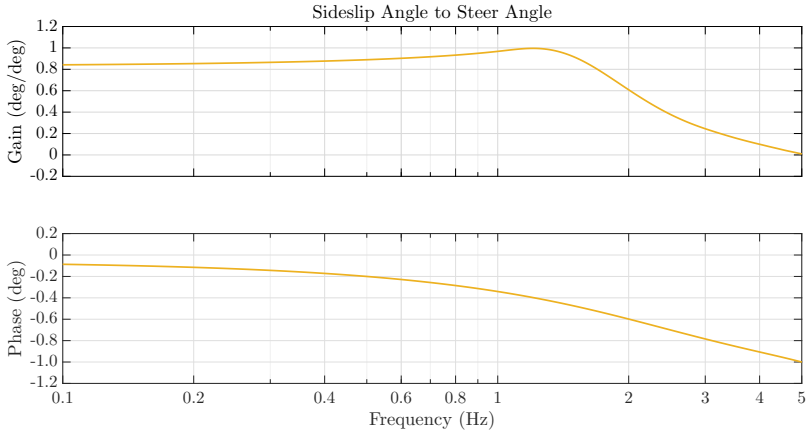

Figure 3.8: Sideslip angle to steering angle Bode plot.

vehicle sideslip angle Bode plot respect to lateral acceleration a_y and yaw rate can be traced as in Figure 3.9. The results found are significant in the design of the ANN. Because of the phase lags between sideslip angle and the other states, the neural network has to consider the history of the manoeuvre to estimate correctly the desired output, confirming the results of the step-response analysis. Another important fact is that very little variations of the vehicle's geometrical features, tyre force curves or inertial features will make the front and rear slip angles vary

in a different way thus changing frequency behaviour.

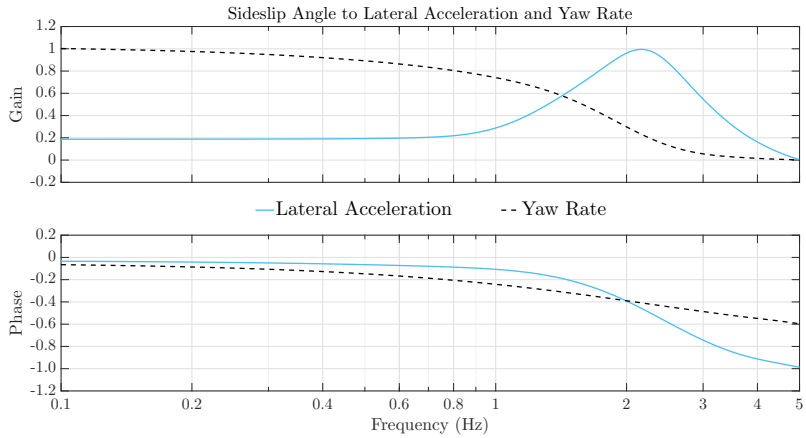

Figure 3.9: Sideslip angle to lateral acceleration and yaw rate Bode plots.

3.2 ANN module

This section describes the ANN module. This module outputs a pseudo-sideslip angle which is used as a measurement for an UKF. The neural network performance is strongly influenced by its architecture which is normally heuristically defined. In this work, an algorithm to define the structure of the ANN is developed considering the sideslip angle analysis of Section 3.1.

3.2.1 Architecture definition procedure

When designing a neural network, there are three main issues:[109]

1. the network is not "powerful" enough to fit the data,

2. the network overfits the training results,

3. the network extrapolates when deployed.

To avoid these problems, an algorithm was developed to define the network's structure. This requires first dividing the dataset into three sets: a training set over which the network is trained, a validation set used to evaluate overfitting and a test set used to evaluate the network. Then, some architecture features which are not to be varied have to be defined. Finally, a set of parameters of the network's architecture which are to be varied have to be selected (e.g. number of neurons) together with their initial value p^i and percentage variation $k^i \in (0, 1)$ for $i = 1, ..., m$. The matrix of architecture parameters $\mathbf{P}$ and variation vector $\mathbf{K} = [k^1, k^2, ..., k^m]'$ can

at this point be defined in the following way,

$$\mathbf{P} = \begin{bmatrix} p^1 - k^1 p^1 & p^1 & p^1 + k^1 p^1 \\ p^2 - k^2 p^2 & p^2 & p^2 + k^2 p^2 \\ \vdots & \vdots & \vdots \\ p^m - k^m p^m & p^m & p^m + k^m p^m \end{bmatrix} = \begin{bmatrix} p_l^1 & p_n^1 & p_p^1 \\ p_l^2 & p_n^2 & p_p^2 \\ \vdots & \vdots & \vdots \\ p_l^m & p_n^m & p_p^m \end{bmatrix}$$

Given a loss function $R(\rho, \hat{\rho})$ on any estimated $\hat{\rho}$ and real ρ data; the estimated values of the training $\hat{\gamma}$, validation $\hat{\eta}$, test $\hat{\zeta}$ and overall $\hat{\theta}$ datasets; the correspondent real values γ, η, ζ and θ; and an assigned maximum value for the training dataset loss function l_γ, validation dataset loss function l_η and test dataset loss function l_ζ; Algorithm 1 can be defined.

Algorithm 1: Architecture definition

```
Data: Datasets, General architecture features
Result: ANN estimator
i = 1;
while (1) do
    for j ← 1 to 3 do
        q = [p_{i,j}, p_{i+1,2}, ..., p_{i+m−1,2}]
        train ANN with parameters → q
        if R(γ, γ̂) < l_γ then
            if R(η, η̂) < l_η then
                if R(ζ, ζ̂) < l_ζ then
                    print Deploy network;
                    return ANN
                else
                    print Network extrapolates;
            else
                print Network overfits;
        else
            print Network failed;
        ξ{j} = R(θ, θ̂)
    s = index[min(ξ)]
    p_{i+m,2} = p_{i,s}
    p_{i+m,1} = p_{i,s} − k_{i} p_{i,s}
    p_{i+m,3} = p_{i,s} + k_{i} p_{i,s}
    i = i + 1
```

3.2.2 General features

Before running the algorithm, some general architecture features have to be defined. To avoid reaching a local minimum during the optimisation process, five different initial conditions were considered for each network architecture. These conditions were chosen so that the scaled input data applied to the network would have a

Gaussian distribution within a range of the transfer function which would guarantee a high initial gradient descent, i.e. speed up the training. To train the network, early stopping was used and due to the large amount of data given to the network, the scaled conjugate gradient method was used as an optimisation algorithm. The loss function $R(\rho, \hat{\rho})$ used to evaluate and train the network is the root mean square error,

$$R(\rho, \hat{\rho}) = \sqrt{\frac{\sum_{t=1}^{n}(\hat{\rho} - \rho)^2}{n}}. \tag{3.9}$$

The experimentally validated Vi-Grade vehicle model was used to create datasets composed only of numerical data to reduce cost and time in development. To keep computation time low and be able to run the network on standalone embedded automotive platforms with low computing power and memory, only one hidden fully connected layer was considered, keeping the number of neurons under one hundred. For the same reasons, more advanced architectures such as Long Short Term Memory (LSTM) [110] or Gated Recurrent Unit (GRU) [111] were not taken into consideration.

The ANN developed in this work is based on the work of Melzi et al..[37] The hidden layer uses hyperbolic tangent transfer functions while the output layer uses linear transfer functions. The hyperbolic tangent was chosen since its shape is very similar to typical tyre force curves, however, a comparison with radial basis transfer functions which are local transfer functions was done and is discussed in Appendix B. Since hyperbolic tangent transfer functions are global transfer function, they activate everywhere except close to the origin meaning that the output will be the sum of all activation functions.

Table 3.3: Input manoeuvres for training and validation sets.

Manoeuvre	Parameter	Unit	Range
Sine steer	Steer amplitude	°	5,0 - 75
(Freq. response)	Steer frequency	Hz	0,1 - 5,0
Step steer	Final steer angle	°	5,0 - 75
(Step response)	Step duration	s	0,1 - 1,0
Circuit	Max lat. acc.	g	0,1 - 1,0
(Combined slip)	Max long. acc.	g	0,1 - 1,0

The network inputs, coming solely from an IMU, are the accelerations along the three axes and rotational velocities about the three axes. The dataset used comprises different manoeuvre types; different manoeuvre configurations were used for each manoeuvre type (e.g. a step steer manoeuvre type is defined by a step duration and amplitude configuration). The dataset manoeuvre types and parameter range of the different configurations can be seen in Table 3.3. The manoeuvres were chosen so that the network would be trained over the entire working range of the vehicle. For each input manoeuvre type, a multiple of three different values for each manoeuvre configuration parameter was chosen (within the predefined range). This allowed the data to be divided between training and validation so that the latter was always a subset of the former with regards to signal amplitude and frequency. With this procedure, the validation set properly evaluated overfitting rather than extrapolation. Finally, a test set was suitably chosen to evaluate

network performance. The various manoeuvre configurations in the test set were not present in the training and validation sets. Also for the circuit, a different test circuit was used to the ones used for training and validation. The test set manoeuvre types and configuration parameter range can be seen in Table 3.4.

Table 3.4: Input manoeuvres for test set.

Manoeuvre	Parameter	Unit	Range
Sine steer	Steer amplitude	°	10 - 55
(Low freq. response)	Steer frequency	Hz	1,0
Sine steer	Steer amplitude	°	10 - 55
(High freq. response)	Steer frequency	Hz	3,0
Step steer	Final steer angle	°	10 - 55
(Step response)	Step duration	s	0,2
Circuit	Max lat. acc.	g	0,1 - 1,0
(Combined slip)	Max long. acc.	g	0,1 - 1,0

3.2.3 Structure definition

The general structure of the network is shown in Figure 3.10.
After having defined the general architecture, constraints and datasets of the network, Algorithm 1 was run to obtain the exact architecture by building matrix **P** with the following parameters in the following order:

- **Par. 1** → Number of input manoeuvres (open-loop),
- **Par. 2** → Ratio of sine steer to step steer manoeuvres,
- **Par. 3** → Number of hidden layer neurons,
- **Par. 4** → Number of delays.

The first parameter varied was the number of total manoeuvres. Only the number of open-loop manoeuvres was varied since they are dependant on the vehicle and are therefore reproducible, unlike closed-loop manoeuvres which are dependant on the driver and were thus kept constant. Generally, training improves with a larger amount of data. However, with too many manoeuvres compared to the number of neurons, the network might underfit the data as it would not be powerful enough to approximate the function. This also depends on the number of tapped delays as these also increase the amount of data and hence the dimension of the input space. The second parameter varied was the ratio between number of sine steer configurations and step steer configurations within the predefined range (higher discretisation) whilst keeping the total number of manoeuvres approximately the same. Note that it was not possible to maintain the exact same number of manoeuvres since multiples of three for each parameter variation were necessary. This ratio was chosen as a parameter since the frequency response of a non-linear system is harder to identify compared to the step response.[71]
The next parameter which was varied was the number of neurons. The higher the

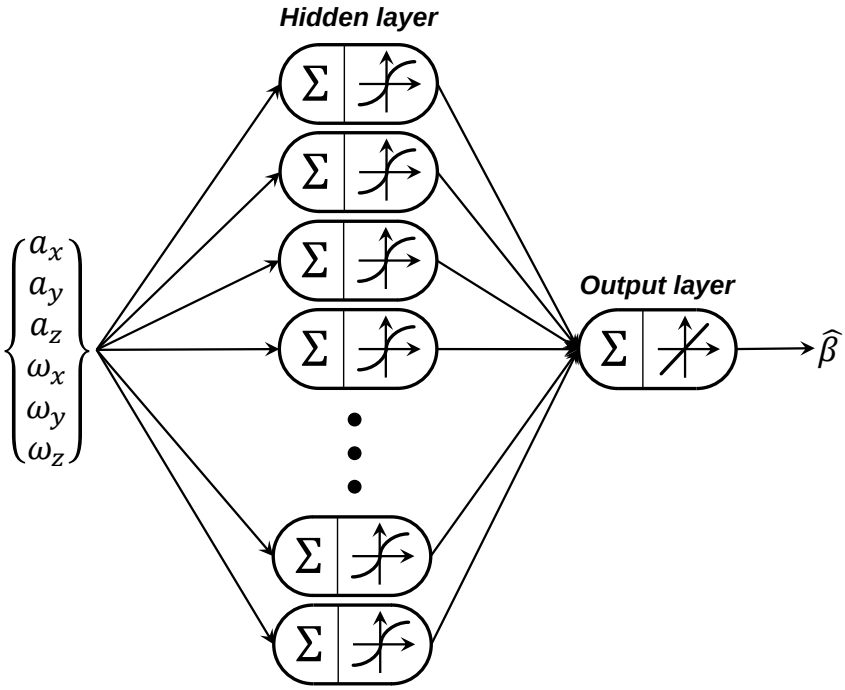

Figure 3.10: General structure of ANN module.

number, the more powerful the network, meaning it can approximate more complicated functions. However, it is also easier for the system to overfit. The lower the number of neurons, the harder it is to overfit but the more probable that the network will underfit and not be powerful enough. Since the main factor which causes an ANN to overfit is the ratio between input data and neurons, the number of neurons was varied after choosing the total number of manoeuvres.
Finally, to give some time memory to the ANN, the number of tapped delays was varied, considering a frequency acquisition of 100Hz, 10 delays correspond to 0,1s.

3.2.4 Results

Running the algorithm, the system converged after one sweep of every parameter with $l_\gamma = l_\eta = l_\zeta = 0,2°$. The final structure of the ANN obtained with the proposed algorithm is the following:

- **Par. 1** $\rightarrow$ 450 (1 hour and 25 minutes driving)
- **Par. 2** $\rightarrow$ 1 (225 sine steers and 225 step steers)
- **Par. 3** $\rightarrow$ 90 (hidden layer)
- **Par. 4** $\rightarrow$ 10 (equal to 0,1s in time)

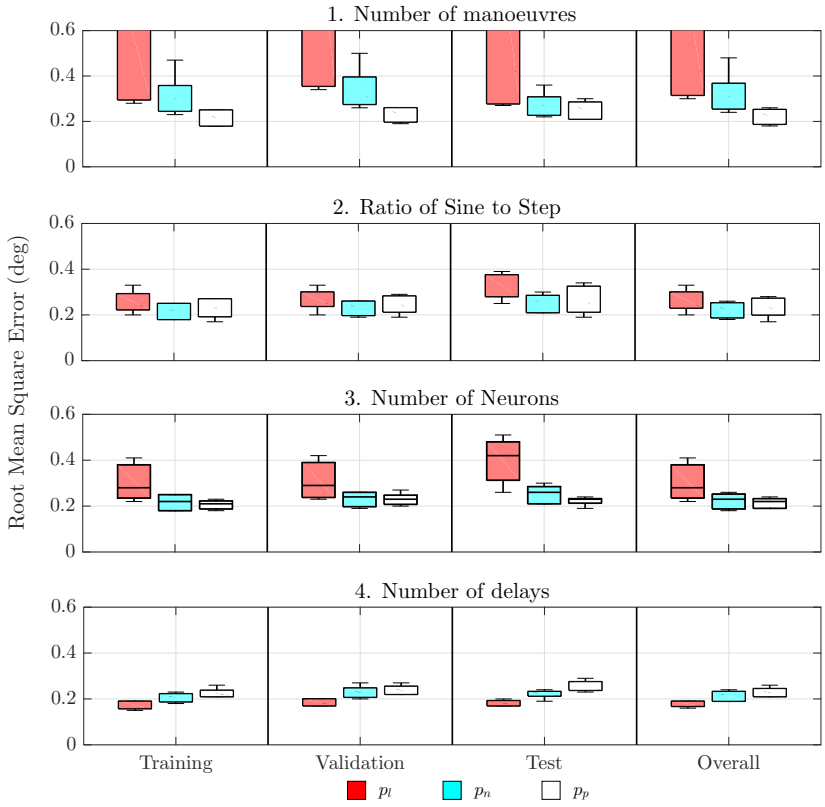

Figure 3.11: Algorithm 1 results.

The results of the algorithm are shown in Figure 3.11. For each step, the loss function of each architecture, represented by a different colour were plotted using five different initial conditions for the training set, validation set, test set and overall set. The loss functions are represented as boxplots to show error distribution, in order to evaluate if a network with a certain architecture and initial conditions reaches a local minimum instead of a global minimum, thus guaranteeing great performance for some manoeuvre types but terrible performance for others. The average of the loss function of the five initial conditions was used as the evaluation criteria in the algorithm. As expected, performance increased with the increase in the number of input manoeuvres. The variation of the ratio of sine steer to step steer configurations in step two did not much influence network performance, although the smallest error arose when the number of sine steer and step steer configurations was equal. In step three, the higher the number of neurons, the better the performance. No overfitting occured as can be seen in the validation boxplot, due also to the higher number of manoeuvres. Finally, the lowest number of tapped delays resulted in the best performance.
The results of the deployed ANN on the test set can be seen in Figure 3.12. Specifically, for each manoeuvre type in Table 3.4 (low frequency sine steer, high frequency

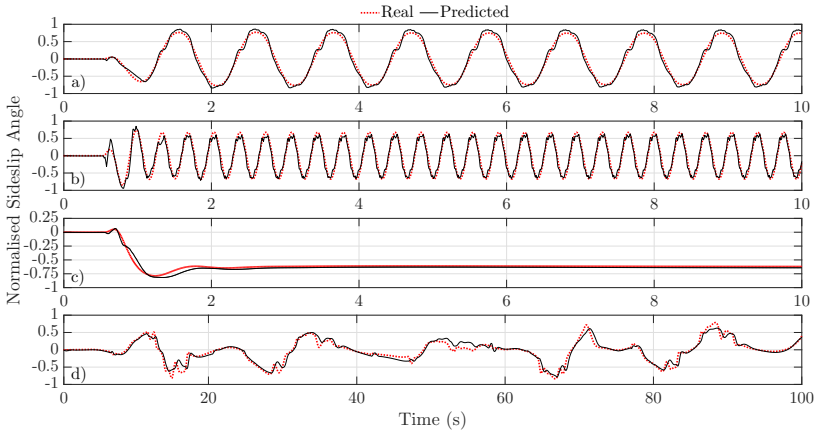

Figure 3.12: Sideslip angle ANN estimation on test set. a) Low frequency sine steer. b) High frequency sine steer. c) Step steer. d) Circuit.

sine steer, step steer and circuit), only the manoeuvre configuration with the highest values of I/O signals is shown. In these conditions, the vehicle is handling at the limits and is operating in the non-linear region.
For confidentiality reasons, the sideslip angle values have been normalised to the maximum value obtained in all datasets. The ANN obtained with the proposed method shows very good results for all types of manoeuvres in a numerical environment.
Due to different external conditions (road friction), model uncertainty (tyre model) and closed-loop behaviour (driver), when applying this ANN to the experimental data, the sideslip angle estimated by the ANN saturates to the maximum value seen in the numerical environment as will be shown in the following section. Thus, the value estimated by this ANN will serve as a correction measurement for the UKF.

3.3 UKF module

In this section the remaining part of the proposed observer will be discussed. This is a UKF based on a kinematic model whose inputs are IMU measurements, the pseudo-sideslip angle estimated by the ANN and the longitudinal velocity given by direct integration with integral damping and integral value reset. For better observer convergence, UKF output is used to correct the pseudo-sideslip angle so that the measurement entering the UKF and its output asymptotically tend to the same value.

3.3.1 UKF algorithm

The UKF was first proposed by Julier et al..[112] Unlike the EKF where the time update is conducted by a linearised system through a Jacobi matrix, the UKF uses a non-linear function model. The main idea is to use the UT to propagate the so-called “sigma points” a group of symmetrically distributed points around the previous estimated system states which contain the information for the expected

mean value and variance of the system. Consider a non-linear system in discrete time form with additive noise,

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, t_k) + \mathbf{v}_k, \tag{3.10a}$$
$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, t_k) + \mathbf{w}_k. \tag{3.10b}$$

Where $\mathbf{x}_k \in \mathbb{R}^n$ represents the state vector, $\mathbf{u}_k \in \mathbb{R}^m$ is the input vector, $\mathbf{y}_k \in \mathbb{R}^q$ the measurement vector. The process noise $\mathbf{v}_k$ and measurement noise $\mathbf{w}_k$ are white Gaussian noises, i.e. zero mean and uncorrelated and defined as follows,

$$\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k^v), \tag{3.11a}$$
$$\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}_k^w). \tag{3.11b}$$

Where $\mathbf{R}_k^v$ and $\mathbf{R}_k^w$ are, respectively, the process noise and measurement noise covariance matrices. The UKF algorithm can be presented considering the state vector $\mathbf{x}_k$ (random variable) having mean value $\tilde{\mathbf{x}}_k$ and covariance $\mathbf{Q}_k$.
1. Initialise mean value and variance matrix as follows,

$$\tilde{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0], \tag{3.12a}$$
$$\mathbf{Q}_0 = \mathbb{E}[(\mathbf{x}_0 - \tilde{\mathbf{x}}_0)(\mathbf{x}_0 - \tilde{\mathbf{x}}_0)^T]. \tag{3.12b}$$

Where $\tilde{\mathbf{x}}_0$ is the *a posteriori* estimation of the expected mean value for $k = 0$ and $\mathbf{Q}_0$ is the *a posteriori* estimation of the variance matrix for $k = 0$.
2. Calculate sigma points $\boldsymbol{\mathcal{X}}_{k-1} \in \mathbb{R}^{n \times (2n+1)}$ for $k \in \{1, ..., \infty\}$ according to the following,

$$\boldsymbol{\mathcal{X}}_{k-1} = \begin{bmatrix} \tilde{\mathbf{x}}_{k-1} & \tilde{\mathbf{x}}_{k-1} + \mathbf{A}_{k-1} & \tilde{\mathbf{x}}_{k-1} - \mathbf{A}_{k-1} \end{bmatrix}, \tag{3.13a}$$
$$\mathbf{A}_{k-1} = \sqrt{(n+\lambda)\mathbf{Q}_{k-1}}, \tag{3.13b}$$
$$\lambda = \alpha^2(n + \kappa) - n. \tag{3.13c}$$

Where λ is a scaling parameter, the constant α determines the spread of the sigma points around $\mathbf{x}_{k-1}$ and is usually set to a small positive value. The constant κ is a secondary scaling parameter. Many studies have been carried out to calculate the square root of covariance matrix $\mathbf{Q}_{k-1}$. In this work Cholesky factorization was used for which a Hermitian positive-definite matrix B can be decomposed as $B = LL^+$, with L being a lower triangular matrix with real and positive diagonal terms.
3. Time update by transforming the sigma points with the non-linear functions as follows,

$$\boldsymbol{\mathcal{X}}_{k|k-1} = \mathbf{f}(\boldsymbol{\mathcal{X}}_{k-1}, \mathbf{u}_{k-1}, t_k), \tag{3.14a}$$
$$\boldsymbol{\mathcal{Y}}_{k|k-1} = \mathbf{g}(\boldsymbol{\mathcal{X}}_{k-1}, t_k). \tag{3.14b}$$

And computing the *a priori* estimation of the expected mean value $\tilde{\mathbf{x}}_k^-$, variance matrix $\mathbf{Q}_k^-$ and measurement estimation $\tilde{\mathbf{y}}_k^-$,

$$\tilde{\mathbf{x}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}_{i,k|k-1}, \tag{3.15a}$$

$$\mathbf{Q}_k^- = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,k|k-1} - \tilde{\mathbf{x}}_k^-][\mathcal{X}_{i,k|k-1} - \tilde{\mathbf{x}}_k^-]^T + \mathbf{R}_k^v, \tag{3.15b}$$

$$\tilde{\mathbf{y}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,k|k-1}. \tag{3.15c}$$

Where the W_i weights are represented by,

$$W_0^{(m)} = \lambda/(n+\lambda), \tag{3.16a}$$

$$W_0^{(c)} = \lambda/(n+\lambda) + (1 - \alpha^2 + \gamma), \tag{3.16b}$$

$$W_i^{(m)} = W_i^{(c)} = 1/\{2(n+\lambda)\} \qquad i = 1, ..., 2n. \tag{3.16c}$$

Where γ is used to incorporate prior knowledge of the distribution of the state vector (for Gaussian distributions, $\gamma = 2$ is optimal).
4. Measurement update by computing the measurement estimation variance $\mathbf{Q}_{y_k y_k}$ and covariance matrix $\mathbf{Q}_{x_k y_k}$ between $\tilde{\mathbf{x}}_k^-$ and $\tilde{\mathbf{y}}_k^-$,

$$\mathbf{Q}_{y_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T + \mathbf{R}_k^w, \tag{3.17a}$$

$$\mathbf{Q}_{x_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} [\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T. \tag{3.17b}$$

And finally, calculating the Kalman gain $\mathcal{K}_k$, *a posteriori* estimation of expected mean value $\tilde{\mathbf{x}}_k$ and variance matrix $\mathbf{Q}_k$ as follows,

$$\mathcal{K}_k = \mathbf{Q}_{x_k y_k} \mathbf{Q}_{y_k y_k}^{-1}, \tag{3.18a}$$

$$\tilde{\mathbf{x}}_k = \tilde{\mathbf{x}}_k^- + \mathcal{K}_k(\mathbf{y}_k - \tilde{\mathbf{y}}_k^-), \tag{3.18b}$$

$$\mathbf{Q}_k = \mathbf{Q}_k^- - \mathcal{K}_k \mathbf{Q}_{y_k y_k} \mathcal{K}_k^T. \tag{3.18c}$$

3.3.2 Kinematic model

Since the model used for the UKF is a pure kinematic model, it does not depend on any tyre model. The model describes the kinematics of the vehicle's planar motion. Given longitudinal acceleration a_x, lateral acceleration a_y, yaw rate $\dot{\psi}$, longitudinal velocity v_x, lateral velocity v_y and sideslip angle β, a set of differential equations can be written in the vehicle's reference system with origin in its centre of gravity as follows,

$$\dot{v}_x = a_x + v_y \dot{\psi}, \tag{3.19a}$$

$$\dot{v}_y = a_y - v_x \dot{\psi}. \tag{3.19b}$$

From the definition of β (neglecting roll and pitch motions), v_y can be expressed as,

$$v_y = v_x \tan\beta. \tag{3.20}$$

And consequently its derivative,

$$\dot{v}_y = \dot{v}_x \tan\beta + v_x(1+\tan^2\beta)\dot{\beta}. \tag{3.21}$$

Substituting (3.20) and (3.21) in (3.19a) and (3.19b) a new set of differential equations is obtained,

$$\dot{v}_x = a_x + v_x\dot{\psi}\tan\beta, \tag{3.22a}$$

$$\dot{\beta} = \frac{1}{v_x(1+\tan^2\beta)}(a_y - v_x\dot{\psi} - v_x\dot{\psi}\tan^2\beta - a_x\tan\beta). \tag{3.22b}$$

Applying some trigonometric identities, the sideslip angle kinematics can be expressed as,

$$\dot{\beta} = \frac{a_y}{v_x}\cos^2\beta - \frac{a_x}{2v_x}\sin 2\beta - \dot{\psi}. \tag{3.23}$$

Thus the state vector becomes $\mathbf{x}(t) = [\beta]$, the input vector $\mathbf{u}(t) = [a_x, a_y, \dot{\psi}, v_x]$, the measurement vector $\mathbf{y}(t) = [\hat{\beta}]$ with $\hat{\beta}$ being the pseudo-sideslip angle, v_x is found by direct integration while a_x, a_y and $\dot{\psi}$ are measured with an IMU. Applying the UKF equations to this system, the estimated sideslip angle $\tilde{\beta}$ is obtained.

3.3.3 Longitudinal velocity observer

The longitudinal velocity is kept external to the UKF to avoid influencing the covariance matrices and because it can be found with good precision by direct integration using integral damping and integral reset value (DI-IDIRV) to eliminate integration drift. First, moving average filters with a five tenths of a second window are applied to the signals coming from the IMU. Then, integral damping is added to obtain,

$$\dot{v}_x = \int \left[a_x + v_x\dot{\psi}\tan\beta - \frac{v_x}{\tau}sgn(\dot{v}_x)\right]dt. \tag{3.24}$$

Where τ ($\tau \geq 1$) represents a damping coefficient. A fourth order explicit Runge-Kutta numerical integration is used to solve the equation. Finally, the integral value is reset with a steady-state value. The condition for steady-state [7] is that yaw moment must be zero. When this occurs, the steady-state longitudinal velocity v_x^s can be found,

$$v_x^s = \frac{a_y}{\dot{\psi}}. \tag{3.25}$$

A quasi steady-state condition is considered and "activates" at time step k in the following way. Given the time discrete system and $\dot{\psi}(t) = [\dot{\psi}^{(k-l)}, \dot{\psi}^{(k-l+1)}, ..., \dot{\psi}^{(k)}]$ comprising yaw rate at step k and yaw rate at the l previous time steps, the following conditions must be fulfilled:

$$|\dot{\psi}^i| \leq \epsilon \quad i = k-l, k-l+1, ...k, \tag{3.26a}$$

$$\sum_{i=k-l}^{k} |\dot{\psi}^i| \leq d. \tag{3.26b}$$

with $\epsilon, d \in \mathbb{R}$ being small positive numbers. When this happens, the integral value is reset to the steady-state value as for (3.25). This integral reset value method is valid under the assumption that:

- the vehicle is not driving in a straight line on a flat road, in which case the sidelip angle is in any case zero,
- the vehicle is a non-neutral steering vehicle, in which case even with external disturbance or a banked road the sideslip angle is different to zero but no yaw rate is generated.

Experimental results of this method and the effectiveness of the algorithm will be shown in the next section.

Figure 3.13: Experimental setup.

3.3.4 Pseudo-sideslip angle correction

The pseudo-sideslip angle is based on supervised learning. The hyperbolic tangent function used as transfer function in the ANN has two horizontal asymptotes. Thus, the output of the network is saturated to the maximum and minimum value seen during training. To increase performance of the UKF, at time step k the pseudo-sideslip angle $\hat{\beta}_k$ is corrected and a corrected pseudo-sideslip angle $\hat{\beta}_k^\star$ is obtained. This is done with proportional feedback correction as follows,

$$\hat{\beta}_k^\star = C_k \hat{\beta}_k. \tag{3.27}$$

Table 3.5: InvenSense MPU-6050 datasheet.

Parameter	Value	Unit	Sensor
Power spectral density	400	$\mu g/\sqrt{Hz}$	Acceler.
Total RMS noise	0,05	°/s-rms	Gyrosc.
Low-frequency RMS noise	0,033	°/s-rms	Gyrosc.
Rate noise spectral density	0,005	$°/s/\sqrt{Hz}$	Gyrosc.

where the proportional gain C_k at time step k is found with the following system of equations,

$$C_k = \begin{cases} 1, & \text{if } |\tilde{\beta}_{k-1} - \hat{\beta}_{k-1}| \geq L_k, \\ U_k, & \text{if } C_k \geq U_k, \\ \frac{\tilde{\beta}_{k-1}}{\hat{\beta}_{k-1}}, & \text{otherwise.} \end{cases} \tag{3.28}$$

Since the correction is recursive, the lower saturation L_k and upper saturation U_k are used to make the algorithm stable.

3.4 Results

In this final section, firstly the experimental setup used to obtain the real data will be described and then the results of the algorithm will be shown for both the longitudinal velocity and sideslip angle estimation. The benefits of the proposed algorithm against standalone solutions will be described.

3.4.1 Experimental setup

The experimental data was obtained on a sport saloon which can be seen in Figure 3.14.
The vehicle was equipped with a two axis non-contact optical sensor with a halogen lamp for sideslip angle measurement, specifically the Kistler Correvit S-Motion which has a measurement accuracy angle of ±0,2° and angle resolution of ±0,1°. The measurement is guaranteed with a minimum speed of ±0,1km/h and a maximum speed of ±250km/h, accelerations up to ±18g and angle speeds up to ±300°/s, with a frequency of 500Hz. Since the sensor was mounted in front of the vehicle's front axis, the velocity measurements were shifted to the centre of gravity by means of rigid body laws. The vehicle was also equipped with Kistler HF Sensors, an optical laser height-sensor, on both sides of the vehicle, to validate the Vi-Grade model used to train the ANN. Additionally, a Kistler MSW Sensor, a non-contact optical steering angle sensor, was mounted behind the steering wheel to measure both steering wheel torque and angle for numerical model validation. To measure accelerations and rotational velocities, the MPU-6050 three axis IMU of InvenSense was used whose noise characteristics are shown in Table 3.5. This IMU was installed on a Meccanica 42 M42A2C10 ECU comprising an AVR 32bit CPU, 512kB flash memory, 64kB EEPROM and 64kB RAM with a maximum clock signal of 66MHz. This ECU is also provided with a 4.1 bluetooth, a 30A and 40V H-bridge and two CAN interfaces.
The acquisition system used to convert the signals is shown in Figure 3.14. It includes a Vector VN1640A CAN/LIN interface with a time stamp accuracy of 1μs

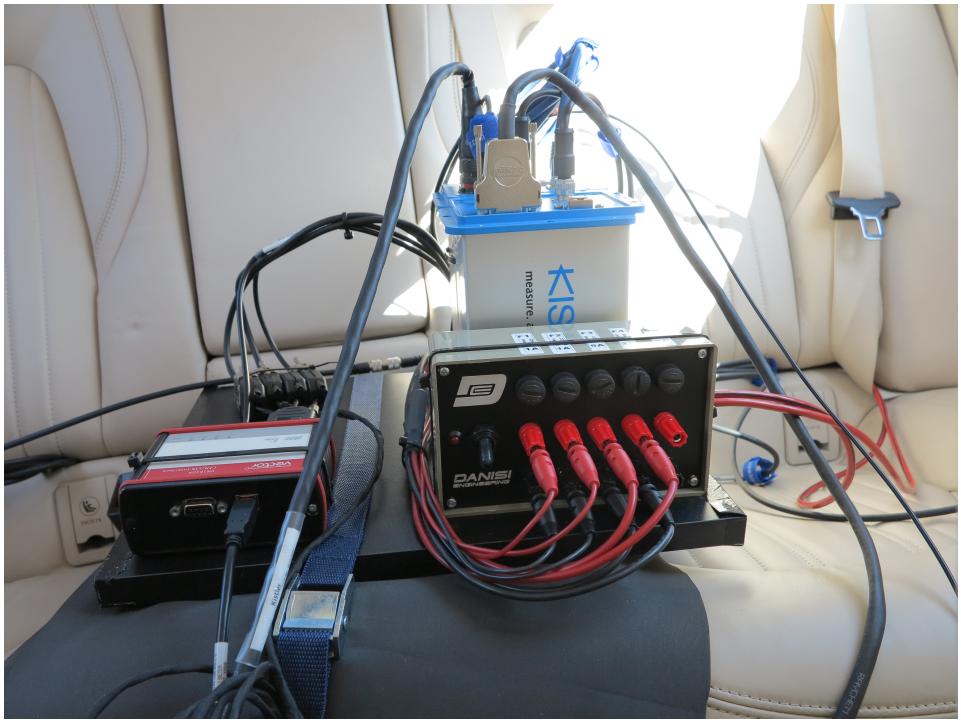

Figure 3.14: Acquisition system.

that can acquire CAN signals at a rate up to 2Mbit/s. The testing was done at Marzaglia (MO), Italy with professional test drivers.
The proposed algorithm was run at a frequency of 100Hz on the described experimental setup. The bottleneck proved to be the CAN system which limited the system's performance to such frequency.

3.4.2 IMU measurements

A sample of the acquisitions obtained during testing with the IMU described above are shown in Figure 3.15, specifically, yaw rate, longitudinal acceleration and lateral acceleration. A moving average filter with a five tenths of a second window was used to smoothen the inputs fed to the estimator. This filter was chosen since it does not require much memory and is capable of smoothening high frequency oscillations while introducing very little phase lag given the size of the time window. The results obtained from the Meccanica 42 IMU were compared with the IMU installed in the Correvit sensor to ensure there was no drift in the gyroscope and accelerometer measurements. Since the Correvit sensor outputs are the longitudinal and lateral velocities, the sideslip angle was easily calculated, while the longitudinal velocity was used as a comparison value for the DI-IDIRV estimator. The other IMU measurements, i.e. vertical acceleration, roll rate and pitch rate were also filtered with the same moving average filter. These signals were used only in the ANN and neglected for the UKF where planar motion was considered.

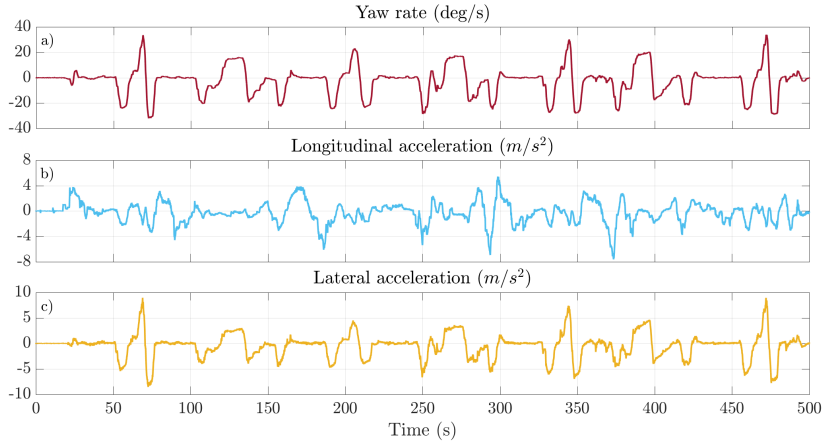

Figure 3.15: IMU measurements. a) Yaw rate. b) Longitudinal acceleration. c) Lateral acceleration.

3.4.3 Longitudinal velocity estimation

The longitudinal velocity of the vehicle was estimated based on the DI-IDIRV observer. The same moving average filter used for IMU measurements was used to filter the estimator's feedback.
Note that since for the UKF dynamics the longitudinal velocity appears in two terms in the denominator, it is important for the estimated longitudinal velocity to not diverge into very small values for correct sideslip angle estimation and numerical integration stability. If the estimated longitudinal velocity is larger than the real value, the sideslip angle derivative will be a lower value than the real one. The term with longitudinal acceleration generally contributes little to the dynamics for normal values of sideslip angle due to the sine multiplication. The results of the estimation can be seen in Figure 3.16 where the integral resets are also illustrated.

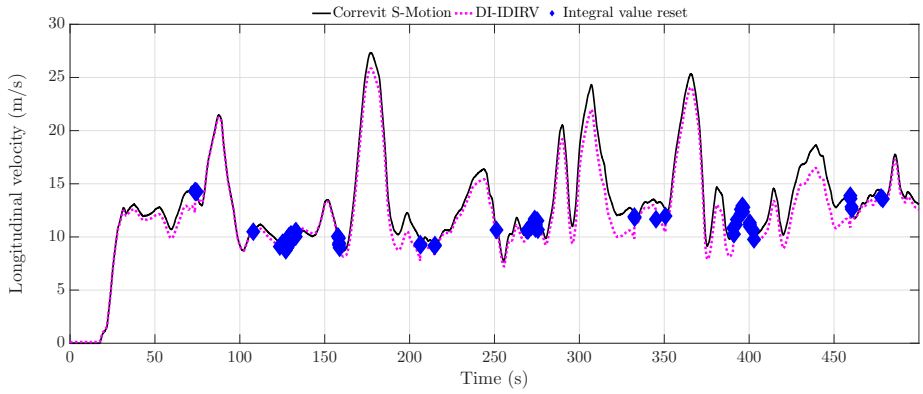

Figure 3.16: Longitudinal velocity estimation.

Table 3.6: Longitudinal velocity estimation error.

Observer	MAE (m/s)	RMSE (m/s)	R
DI-IDIRV	0,88	1,13	0,98

With proper tuning, the integral resets correspond to the real longitudinal velocity. Without the integral reset, the observer is highly unstable and estimated longitudinal velocity quickly diverges, making sideslip angle estimation error very large. The moving average filter applied to the inputs and the integral damping improve numerical integration stability. The estimated velocity profile is very similar to the measured one. Both the velocity variations and quasi steady-state behaviour are well observed. A quantitative analysis by means of different error estimates, shown in Table 3.6, illustrates the high efficiency. Specifically, the Mean Absolute Error (MAE), the Root Mean Square Error (RMSE), and the correlation factor R have been calculated. Due to the small relative magnitude of the error estimates, it can be concluded that the proposed observer efficiently estimates longitudinal velocity by using only IMU measurements. This is also confirmed by the results shown in Figure 3.17. Here, the probability density function (PDF) of the velocity estima-

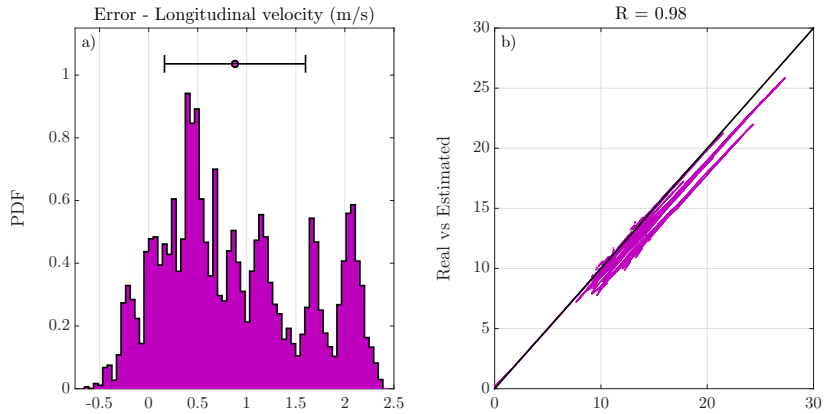

Figure 3.17: Longitudinal velocity estimation error. a) Probability density function of the estimation error. b) Real value vs estimated value.

tion error is represented. It can be noted that the error is random, not having a normal distribution, and is between approximately -0,5m/s and 2,5m/s.
The lower boundary, equal to -0,5m/s, is very important since, as already mentioned, longitudinal velocity appears only as a denominator in the sideslip angle dynamics of the UKF. Finally, it should be noted that estimated velocity is independent of velocity magnitude.

3.4.4 Sideslip angle estimation

Having considered longitudinal velocity estimation and IMU data filtering, sideslip angle estimation can now be discussed.
The results of the pseudo-sideslip angle, the corrected pseudo-sideslip angle and

UKF estimated sideslip angle will be shown and compared.

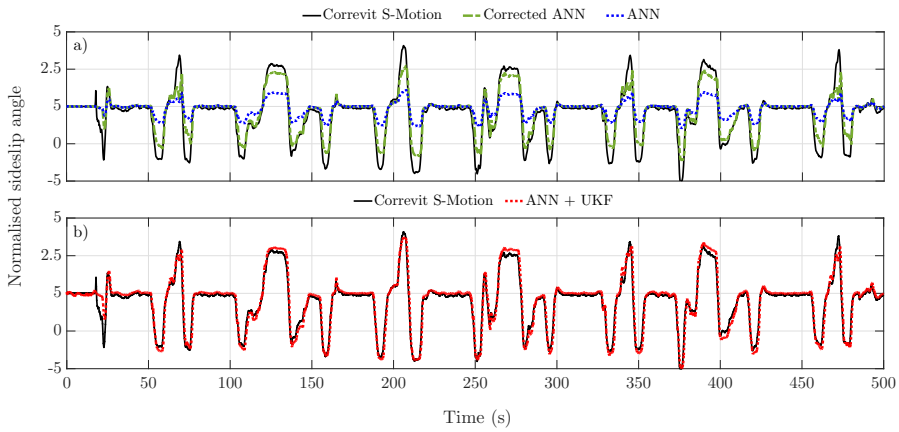

Figure 3.18: Sideslip angle estimation. a) ANN and Corrected ANN estimation. b) ANN + UKF estimation.

Sideslip angle estimation results are shown in Figure 3.18, where for confidentiality reasons the normalised sideslip angle has been plotted. Normalisation was done using the maximum value of the network's training dataset, shown in Figure 3.12. Due to external disturbances, model uncertainty and, especially, different closed-loop behaviours, the magnitude of the sideslip angle is often a higher value than those found in the numerical environment. For this reason, despite great effort to avoid ANN extrapolation, the ANN cannot correctly estimate the experimental data in terms of magnitude. However, the sideslip angle time response given by the ANN is still very accurate and sideslip angle variations are actually estimated correctly. As expected, the pseudo-sideslip angle is saturated to the maximum and minimum network training dataset value while time response is very good. Thus, the proposed strategy for the neural network structure definition can be considered efficient despite needing magnitude correction.
The result of the integrated ANN and UKF, shown in Figure 3.18, shows how sideslip angle estimation is properly corrected with the proposed method. Both time response and magnitude of the estimated sideslip angle are similar to the measured one. Note that in the first sideslip angle variation, the sideslip angle measurement given by the Correvit sensor is not coherent with the IMU measurements. This large discontinuity would seem to indicate that the Correvit measurement is wrong for the 20s to 23s time window. Excluding this, the proposed observer is very efficient in its estimation. This is confirmed by an absolute error time analysis of the observer, shown in Figure 3.19, and a quantitative analysis, shown in Table 3.7. In Figure 3.19, the real value versus the estimated value is shown on the left while the sideslip angle error PDF is shown on the right. The table are reports the MAE, RMSE and R values.
In addition to the quantitative analysis of the longitudinal velocity observer, the mean value of the error μ and standard deviation σ were calculated as the error distribution resembles a normal distribution. The corrected ANN, which depends on the output of the UKF, also improves the estimation given by the ANN. The

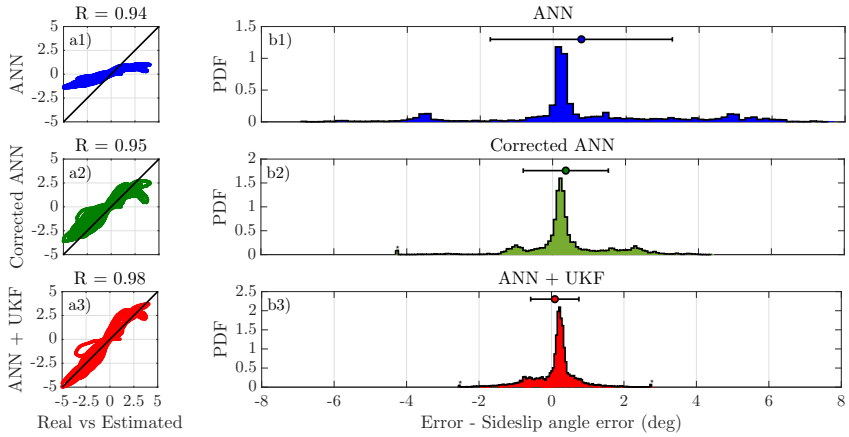

Figure 3.19: Sideslip angle estimation error. a1-3) Real value vs estimated value. b1-3) Probability density function of the estimation error.

Table 3.7: Sideslip angle estimation error.

Observer	MAE (deg)	RMSE (deg)	R	μ (deg)	σ (deg)
ANN	0,77	2,63	0,94	0,77	2,50
Corrected ANN	0,36	1,34	0,95	0,36	1,17
ANN + UKF	0,09	0,70	0,98	0,09	0,67

pseudo-sideslip angle correction saturation ensures that the observer's output remains stable. Since $\hat{\beta}^{\star}$ and $\tilde{\beta}$ recursively depend on each other, the correction helps increase speed and accuracy of the convergence of the algorithm.
For small sideslip angle values the ANN estimates very well, while for larger values estimation is incorrect. The proposed algorithm properly corrects the ANN estimate. Since for many seconds the vehicle is in a straight line and the sideslip angle is close to zero, the correlation factor and mean absolute error have very good values for all observers. However, the root mean square error of the ANN is very high. The proposed ANN + UKF shows instead very good values for all metrics with an improvement of 73,3% on the RMSE with respect to the ANN and of 47,7% with respect to the corrected ANN. Also the low mean error and standard deviation show how the proposed observer estimates the sideslip angle with very high precision.

Chapter 4

Hierarchical Robotic Controller

In this chapter the robotic controller developed is discussed in detail. The hierarchical control scheme developed is described and particularly the vehicle models used for the two levels and the constraint used in the NMPCs are defined as well as the optimisation problems.
Finally, the results of the proposed controller are shown and compared with the ones given by a commercial software. The results show some clear improvement with respect to the current state of the art and guarantee real-time feasibility for AD.

4.1 Modelling

In this section, all aspects concerning the vehicle models used in the control scheme will be discussed. Specifically, the high fidelity vehicle model used for the low-level control is described, which consists of planar dynamics, vertical dynamics, wheel dynamics and tyre forces. Furthermore, the point-mass dynamics and acceleration limits used in the high-level controller are discussed.

4.1.1 Vehicle model

The vehicle model used for the low-level controller is a seven degrees of freedom model which describes the planar dynamics of the vehicle, neglecting heave, pitch and roll motions. To properly consider vertical force tyre sensitivities and eliminate algebraic loops, a dynamic formulation of both longitudinal and lateral load transfers are used. Additionally, the variation of vertical load due to aerodynamic effects is present in the model.
Wheel dynamics are properly modelled, including the effect of limited slip differential since at at-limit handling, the presence of a LSD heavily influences the vehicle dynamics [113] especially during combined accelerations.
The tyres are modelled with a MF6.1 formulation comprising of all combined effects. However, aligning and overturning moments generated by the tyres are neglected.
The engine torque is modelled considering maximum and minimum torque at the wheel for each gear and using the envelope curve as a torque limit over the entire range of wheel speeds. This is possible since an additional gear controller which shifts gear at pre-determined wheel speeds is implemented. Note that this type of

controller is based on a discrete state machine which also uses the time from the previous gear change as a control input, this assures no oscillation between gears. The input vector $\mathbf{u}$ is composed of engine torque T_e, braking pedal percentage B_p and steering wheel angle δ_w,

$$\mathbf{u}(t) = [T_e, B_p, \delta_w]\,. \tag{4.1}$$

The twelve-dimensional state vector is composed of positions, yaw angle, lateral and longitudinal velocities, yaw rate, wheel speeds and vertical load transfer. Since each state is discussed in detail further on, here only the state vector $\mathbf{x}$ is defined,

$$\mathbf{x}(t) = [X, Y, \psi, v_x, v_y, \dot{\psi}, \omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}, \Delta F_{z_x}, \Delta F_{z_y}]\,. \tag{4.2}$$

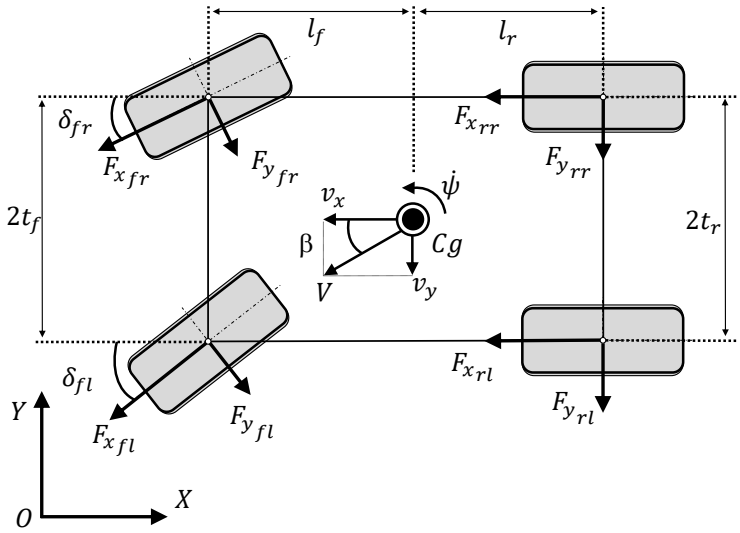

Figure 4.1: Vehicle model scheme.

Planar dynamics

The four-wheel classic seven degrees of freedom model describes longitudinal position X, lateral position Y, and yaw angle ψ of a vehicle with respect to a fixed reference system as well as the already defined longitudinal velocity v_x, lateral velocity v_y and yaw rate $\dot{\psi}$ with respect to a vehicle reference system with origin in Cg and vertical axis pointing upwards. The vehicle model described is illustrated in Figure 4.1. The equations of motion of positions and yaw angle with respect to the fixed reference system are described as follows,

$$\dot{X} = v_x \cos(\psi) - v_y \sin(\psi)\,, \tag{4.3}$$

$$\dot{Y} = v_y \cos(\psi) + v_x \sin(\psi)\,, \tag{4.4}$$

$$\dot{\psi} = d\psi/dt\,. \tag{4.5}$$

To formulate the equation of motion for the remaining states, the following subscripts to refer to individual wheels are introduced: $(*)_{as}$ where $a \in \{f, r\}$ denotes front and rear axis and $s \in \{l, r\}$ denotes the left and right side respectively. The front-left and front-right steering angles, respectively δ_{fl} and δ_{fr} are calculated with Ackermann geometry [7] as follows:

$$\delta_{fl} = \frac{2\delta(l_f + l_r)}{2(l_f + l_r) - \delta t_f}, \tag{4.6}$$

$$\delta_{fl} = \frac{2\delta(l_f + l_r)}{2(l_f + l_r) + \delta t_f}. \tag{4.7}$$

The steering angle δ is obtained by multiplying the steering ratio given by the pinion-rack engagement with δ_w. Even though Ackermann geometry is considered in the NMPC model, to simplify the notation in the following equations Ackermann steering is neglected and the left and right toe angles are considered to be equal $\delta_{fr} = \delta_{fl}$. Considering the front and rear track t_f and t_r, yaw inertia J_{zz}, longitudinal tyre forces $F_{x_{as}}$ and lateral tyre forces $F_{y_{as}}$, air density ρ, front vehicle surface S, vehicle drag coefficient C_x, the already defined front and rear wheelbase l_f and l_r and vehicle mass m, and neglecting static toe, the planar dynamics can be written as,

$$m\dot{v}_x = F_x - mv_y\dot{\psi}\,, \tag{4.8}$$

$$m\dot{v}_y = F_y + mv_x\dot{\psi}\,, \tag{4.9}$$

$$J_{zz}\ddot{\psi} = M_z\,. \tag{4.10}$$

Where the total forces F_x and F_y and moment M_z acting on the vehicle are described as,

$$F_x = \cos(\delta)(F_{x_{fl}} + F_{x_{fr}}) - \sin(\delta)(F_{y_{fl}} + F_{y_{fr}}) + F_{x_{rl}} + F_{x_{rr}} - \rho S C_x v_x^2/2\,, \tag{4.11}$$

$$F_y = \cos(\delta)(F_{y_{fl}} + F_{y_{fr}}) + \sin(\delta)(F_{x_{fl}} + F_{x_{fr}}) + F_{y_{rl}} + F_{y_{rr}}\,, \tag{4.12}$$

$$\begin{aligned} M_z = {} & \cos(\delta)(F_{x_{fr}} - F_{x_{fl}})t_f + \sin(\delta)(F_{x_{fl}} + F_{x_{fr}})l_f + \\ & \cos(\delta)(F_{y_{fl}} + F_{y_{fr}})l_f + \sin(\delta)(F_{y_{fl}} - F_{y_{rl}})t_f + \\ & (F_{x_{rr}} - F_{x_{rl}})t_r - (F_{y_{rl}} + F_{y_{rr}})l_r\,. \end{aligned} \tag{4.13}$$

Wheel dynamics

The forces acting on the vehicle are strongly influenced by the wheel's dynamics. Specifically: engine, gearbox, differential and wheels are considered in this model comprising inertia, transmission ratio and torque distribution. A visual scheme of the layout is shown in Figure 4.2. In the here presented model, the following effects are neglected: the clutch engagement (the clutch is considered to be one with the gearbox) and the losses in the transmission (unitary efficiency). This allows to model the input torque to the differential T_d as a function of engine torque, $T_d = T_e/(\tau_g\tau_d)$, where τ_g and τ_d are the transmission ratios of the gearbox, and differential respectively.

Since the controller will be operating mainly at at-limit handling, it is important for the controller to predict combined adherence for each tyre to maximise performance. For this reason, the model should correctly predict how slip ratio and slip angle evolve as a consequence of the control input. Therefore, wheel speed

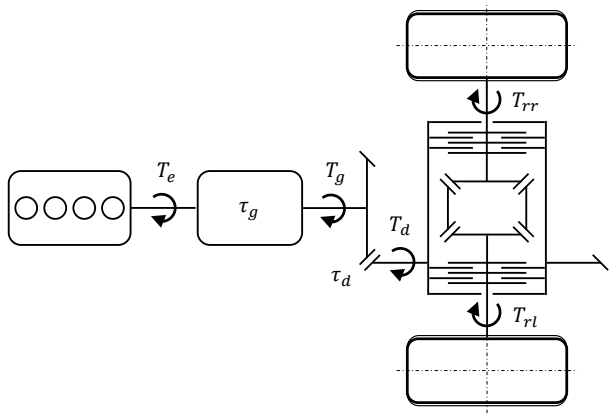

Figure 4.2: Transmission layout.

ω_{as} of each wheel needs to be correctly modelled. In this work a Rear Wheel Drive (RWD) vehicle is used, thus, the front wheels are only affected by braking torques, whereas the rear wheels are also influenced by engine torque. Braking torques of each wheel are computed based on front and rear brake transfer functions from brake percentage to brake torque k_f^b and k_r^b (i.e., depending on master cylinder, brake disc radius, friction coefficient and number of brake pistons), front to rear brake biases b_b, differential locking percentage k_l^d, and wheel rolling resistance $M_{y_{as}}$. To correctly model the torque of the drivetrain at each rear wheel it is also important to consider the LSD effect, since at at-limit handling locking of the differential occurs and torque is transferred from the faster wheel to the slower one. Since a LSD model causes the wheel dynamics to be discontinuous, which in turn would prohibit the use of continuous optimisation solvers, two simplifications are necessary. First, preload is neglected and second, the hyperbolic tangent function is used to approximate the sign function normally present in LSD models. Based on the above-described torques, the wheel dynamics can be written as,

$$J_{\mathrm{rot}_{fl}}\dot{\omega}_{fl} = -(B_p k_f^b b_b) - F_{x_{fl}} R_{fl} - M_{y_{fl}}\,, \tag{4.14}$$

$$J_{\mathrm{rot}_{fr}}\dot{\omega}_{fr} = -(B_p k_f^b b_b) - F_{x_{fr}} R_{fr} - M_{y_{fr}}\,, \tag{4.15}$$

$$J_{\mathrm{rot}_{rl}}\dot{\omega}_{rl} = \frac{1}{2} T_d[1 - k_l^d \tanh(\omega_{rl} - \omega_{rr})] - B_p(1 - b_b)k_r^b - F_{x_{rl}} R_{rl} - M_{y_{rl}}\,, \tag{4.16}$$

$$J_{\mathrm{rot}_{rr}}\dot{\omega}_{rr} = \frac{1}{2} T_d[1 + k_l^d \tanh(\omega_{rl} - \omega_{rr})] - B_p(1 - b_b)k_r^b - F_{x_{rr}} R_{rr} - M_{y_{rr}}\,. \tag{4.17}$$

Where $J_{\mathrm{rot}_{as}}$ is the inertia of the respective wheel including the brake rotor and drive shaft in case of the rear wheels.

Vertical dynamics

Vertical dynamics ought to be considered in the model since tyres have a non-linear characteristic with respect to vertical forces which vary due to planar dynamics, i.e. load transfer. However, load transfers again depend on tyre forces and thus,

vertical forces; this causes an algebraic loop. One possible way to resolve this issue is to introduce a dynamical system describing vertical loads. More precisely, following the framework in [114] a dynamical system for the longitudinal and lateral weight transfers is formulated. Given these weight transfers, vertical load acting on each tyre $F_{z_{as}}$ can be expressed as the sum between static vertical load $F_{z0_{as}}$, longitudinal and lateral weight transfers ΔF_{z_x} and ΔF_{z_y} and downforce $F_{l_{as}}$. Hence, the total vertical force acting on each tyre is expressed as,

$$F_{z_{as}} = F_{z0_{as}} \pm \Delta F_{z_x} \pm \Delta F_{z_y} {}^{t_a}/_{t_{\text{avg}}} + F_{l_{as}} \,. \tag{4.18}$$

Where the sign in front of ΔF_{z_x} is negative for the front wheels and positive for the rear wheels and the sign in front of ΔF_{z_y} is positive for the right wheels and negative for the left wheels. To account for front to rear lateral load transfer distribution, the lateral load transfer is multiplied by t_a, which is the front track for the front wheels' vertical load calculation and rear track for the rear wheels' vertical load calculation, and divided by t_{avg}, which is the average between front and rear track. The static vertical load for each tyre can be found with simple geometrical considerations and is found as follows,

$$F_{z0_{fl}} = F_{z0_{fr}} = \frac{mgl_r}{2(l_f + l_r)}, \qquad F_{z0_{rl}} = F_{z0_{rr}} = \frac{mgl_f}{2(l_f + l_r)} \,. \tag{4.19}$$

The aerodynamic load acting on each tyre can be expressed considering the vehicle's lift coefficient C_z and is given by $F_{l_{as}} = \frac{\rho S C_z}{8} v_x^2$.
Finally, dynamic load transfers are expressed as a first-order delay system. These equations are characterised by a longitudinal time constant τ_x and lateral time constant τ_y, which implicitly represent equivalent pitch and roll suspension stiffness, and centre of gravity's height from the ground h.

$$\Delta \dot{F}_{z_x} = \frac{1}{\tau_x} \left(\frac{h}{2(l_f + l_r)} F_x - \Delta F_{z_x} \right), \tag{4.20}$$

$$\Delta \dot{F}_{z_y} = \frac{1}{\tau_y} \left(\frac{h}{2(t_f + t_r)} F_y - \Delta F_{z_y} \right). \tag{4.21}$$

Actuator dynamics

A simple actuator dynamics model is considered, whereby the inputs cannot be changed directly but the rate of change can be controlled. This corresponds to simple integrator dynamics. This also allows the limitation of the rate of change of the real inputs and optimise for smooth input trajectories. However, this results in a larger state vector $\hat{\mathbf{x}}$ which also contains the (real) inputs, and the input $\hat{\mathbf{u}}$ to the system become the rate of the inputs,

$$\hat{\mathbf{u}}(t) = [\dot{T}_e, \dot{B}_p, \dot{\delta}_w] \,, \tag{4.22}$$

$$\hat{\mathbf{x}}(t) = [X, Y, \psi, v_x, v_y, \dot{\psi}, \omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}, \Delta F_{z_x}, \Delta F_{z_y}, T_e, B_p, \delta_w] \,. \tag{4.23}$$

Note that similarly, it would also be possible to include more complex actuator dynamics. This could be especially interesting for the steering and the engine torque, but for computational reasons this simple integrator model is used.

Tyre forces

Tyre forces are modelled by means of the Pacejka MF6.1 tyre model. In this formulation, large camber angles and pressure changes are considered in addition to a better rolling resistance $M_{y_{as}}$ description. Overturning $M_{x_{as}}$ and aligning torques $M_{z_{as}}$ are neglected due to the computational burden. Particularly the latter are important for vehicle handling purposes. However, the contribution which these give to the model is not enough to justify the increase in computation time (approximately three times higher in this control scheme due to the need to calculate the Jacobian and the Hessian for the optimiser). Concerning forces, the entire combined force formulation was implemented for both longitudinal and lateral behaviour. This is especially important since tyres represent the only component which exchanges forces with the ground. A classic slip ratio σ_{as} and slip angle α_{as} formulation is used [115] which considers velocity at the wheel centre of each wheel $v_{c_{as}}$ calculated as follows,

$$v_{c_{fl}} = \sqrt{(v_x - \dot{\psi} t_f/2)^2 + (v_y + \dot{\psi} l_f)^2} = \sqrt{v_{x_{c_{fl}}}^2 + v_{y_{c_{fl}}}^2}, \tag{4.24}$$

$$v_{c_{fr}} = \sqrt{(v_x + \dot{\psi} t_f/2)^2 + (v_y + \dot{\psi} l_f)^2} = \sqrt{v_{x_{c_{fr}}}^2 + v_{y_{c_{fr}}}^2}, \tag{4.25}$$

$$v_{c_{rl}} = \sqrt{(v_x - \dot{\psi} t_r/2)^2 + (v_y - \dot{\psi} l_r)^2} = \sqrt{v_{x_{c_{rl}}}^2 + v_{y_{c_{rl}}}^2}, \tag{4.26}$$

$$v_{c_{rr}} = \sqrt{(v_x + \dot{\psi} t_r/2)^2 + (v_y - \dot{\psi} l_r)^2} = \sqrt{v_{x_{c_{rr}}}^2 + v_{y_{c_{rr}}}^2}. \tag{4.27}$$

This results in the following slip formulation,

$$\sigma_{as} = \frac{\omega_{as} R_{as} - v_{x_{c_{as}}} \cos(\delta_{as}) + v_{y_{c_{as}}} \sin(\delta_{as})}{v_{x_{c_{as}}} \cos(\delta_{as}) + v_{y_{c_{as}}} \sin(\delta_{as})}, \tag{4.28}$$

$$\alpha_{as} = \tan^{-1}\left(\frac{v_{y_{c_{as}}}}{v_{x_{c_{as}}}}\right) - \delta_{as}. \tag{4.29}$$

The resulting adherence ellipse for the front and rear tyres for a single vertical load is shown in Figure 4.3. The exact tyre force can be seen in the aforementioned

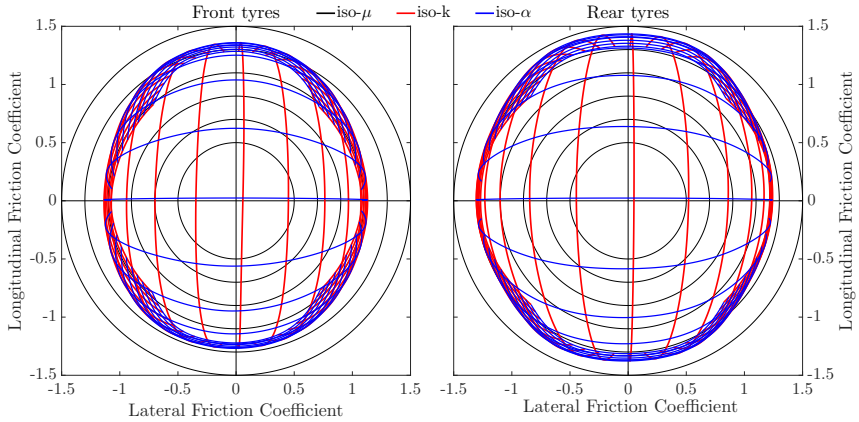

Figure 4.3: Tyre combined forces.

reference, however, tyre force results as a function of the following quantities:

$$F_{x_{as}} = f(\sigma_{as}, \alpha_{as}, F_{z_{as}}, v_x, v_y, \dot{\psi}), \qquad F_{y_{as}} = g(\sigma_{as}, \alpha_{as}, F_{z_{as}}, v_x, v_y, \dot{\psi})\,. \quad (4.30)$$

4.1.2 State-space transformation

The model described above can be written in state space form $\dot{\hat{\mathbf{u}}}(t) = f(\hat{\mathbf{x}}(t), \hat{\mathbf{u}}(t))$. With the model written in this form, time is the independent variable. However, since the goal of the control system is to minimise how much time the vehicle takes to complete a lap, one popular approach is to transform the model from a time-dependent to a track-dependant (spatial) formulation. Additionally, this transformation allows for a natural formulation of obstacles and general road bounds under varying vehicle speed.
To obtain this transformation, the reference path C must be transformed by describing it with a curvilinear abscissa approach. Thus, it can be expressed as a function of its curvature k and the parametrisation of the curve by its arc-length s as in Section 2.4. With this approach, the cartesian position and angular coordinates X, Y, and ψ can be replaced by the longitudinal position on the reference line s, the lateral error n and the heading angle error α. A visual scheme of this transformation can be seen in Figure 4.4. The equation of motion of the vehicle's

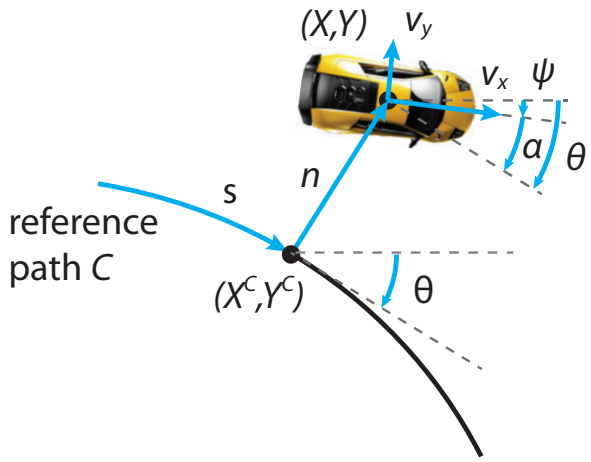

Figure 4.4: State space transformation.

global position in the curvilinear abscissa formulation can be written as,

$$\dot{s} = \frac{v_x \cos(\alpha) - v_y \sin(\alpha)}{1 - nk(s)}\,, \quad (4.31a)$$

$$\dot{n} = v_x \sin(\alpha) + v_y \cos(\alpha)\,, \quad (4.31b)$$

$$\dot{\alpha} = \dot{\psi} - \frac{v_x \cos(\alpha) - v_y \sin(\alpha)}{1 - nk(s)} k(s)\,. \quad (4.31c)$$

The system of equations is valid under the assumption that the vehicle always stays at a lateral distance n that is smaller than the distance of the local centre of curvature of the road, i.e. $n < 1/k(s)$ and that the vehicle's total velocity is

greater than zero $||v_x + v_y|| > 0$. Based on this state transformation the system can now be transformed into the spatial form, where the independent variable is no longer time but s the position along the reference path,

$$\dot{\tilde{\mathbf{x}}}(t) = \frac{d\tilde{\mathbf{x}}}{dt} = \frac{d\tilde{\mathbf{x}}}{ds}\frac{ds}{dt} = \dot{s}\tilde{\mathbf{x}}' \Rightarrow \tilde{\mathbf{x}}'(s) = \frac{f(\tilde{\mathbf{x}}(t), \hat{\mathbf{u}}(t))}{\dot{s}} = \tilde{f}(\tilde{\mathbf{x}}(s), \hat{\mathbf{u}}(s))\,. \tag{4.32}$$

Where $\tilde{\mathbf{x}}'(s)$ denotes the derivative of $\tilde{\mathbf{x}}(s)$ with respect to s, and for the transformation to be well defined it needs to hold that $\dot{s} > 0$, which is the case if the absolute velocity is positive. Note that this transformation renders the s state redundant, thus the state $\tilde{\mathbf{x}}(s)$ is given by,

$$\tilde{\mathbf{x}}(s) = [n, \alpha, v_x, v_y, \dot{\psi}, \omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}, \Delta F_{z_x}, \Delta F_{z_y}, T_e, B_p, \delta_w]\,. \tag{4.33}$$

4.1.3 Point-mass dynamics

The high-level controller is based on a simple point-mass model as shown in Figure 4.5.

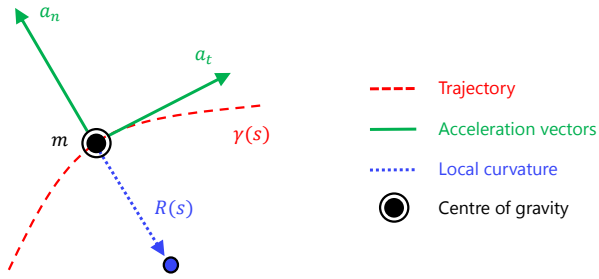

Figure 4.5: Point-mass dynamics model.

The model represents a mass of which the position is always on the reference path, thus, the curvature is always known. With the curvature at each point of the trajectory and the acceleration limits of the mass, it is possible to calculate a velocity profile.
More precisely, the velocity of a point-mass along the reference path can be described by the following model,

$$\frac{dv_t}{dt} = a_t = \frac{f_t}{m} - \frac{\rho S C_x}{2m} v_t^2\,. \tag{4.34}$$

Where m, ρ, S, and C_x are as defined in Section 4.1.1, a_t is the tangential acceleration and f_t is the tangential force applied to the point-mass and, therefore, the input to the point-mass model. Furthermore, since the tangential velocity of the point-mass is known, the centripetal acceleration a_n is given by $a_n = v_t^2 k(s)$. Thus, both tangential force and centripetal acceleration are known which allows constraining the accelerations within the limits of the vehicle, mainly the tires and engine limits which are discussed in the next section.
However, since curvature is needed to compute centripetal acceleration, it is beneficial to again transfer the system into spatial form, where s is the independent

variable and thus $k(s)$ is fixed. Moreover, for the given point-mass dynamics, where lateral velocity v_y and deviation from the path n are zero, the curvilinear abscissa dynamic formulation expressed in (4.31a) simplifies in $\dot{s} = v_t$. Thus, the state transformation is significantly simpler than in the general case described in the previous section. Additionally, similar to Section 4.1.1, an integrator actuator dynamics is introduced, mainly to be able to minimise (indirectly) longitudinal jerk. Thus, the dynamics of the point-mass model in the spatial form are,

$$\begin{aligned} v_t(s)' &= \frac{1}{v_t}\left(\frac{f_t}{m} - \frac{\rho S C_x}{2m} v_t^2\right), \\ f_t(s)' &= \frac{j_t}{v_t}. \end{aligned} \tag{4.35}$$

The dynamics can be written in the following state space model, $\mathbf{q}(s)' = g(\mathbf{q}(s), \mathbf{w}(s))$, where $\mathbf{q}(s) = [v_t, f_t]$ is the state vector and $\mathbf{w}(s) = j_t$ is the input.

Acceleration limits

The acceleration limits of the vehicle can be represented with the gg-diagram. This represents the vehicle's performance limits measured at the centre of gravity under steady-state behaviour, hence it can be used as a constraint for the point-mass model. Since it is important for the high-level controller to not underestimate the potential performance of the vehicle, a slightly overoptimistic approach to computing the gg-limits based on tyre models is proposed. This is achieved by neglecting load transfer and thereby vertical force tyre sensitivity. Note that the acceleration limits here described are used as a constraint only for the point-mass model and high-level controller. Because of the structure of the hierarchical scheme, it is intended for the point-mass model to overestimate vehicle performance when lateral dynamics are most important and short prediction horizons are sufficient so that the low-level controller can be free to make decisions in these situations. On the other hand, when the dynamics are dominated by longitudinal dynamics (e.g. braking), the high-level controller should correctly predict vehicle behaviour since the prediction horizon required is too long for the low-level controller, thus additional acceleration limits which include longitudinal weight transfer are included. The method runs a sweep of slip angles and slip ratios at a fixed longitudinal velocity and calculates the correspondent longitudinal and lateral tyre forces. A matrix of tyre loads for every combined slip condition and each wheel is then obtained. From each tyre force matrix, alpha-shapes [116] are used to calculate the convex hull of tyre adherence ellipse, as shown on the left in Figure 4.6. The various tyre force boundaries are then summed in the $\sigma - \alpha$ space and divided by the total vehicle mass to find the acceleration limits of the vehicle. Note that the sum in the $\sigma_{as} - \alpha_{as}$ space slightly underestimates vehicle's performance, this sum should be done in the $F_{x_{as}} - F_{y_{as}}$ space, however, the computational burden would be much higher. At this point, another convex hull, which can be expected to be similar to an ellipse, is obtained, as shown on the right in Figure 4.6. This represents the acceleration limits of the vehicle. With this method, the gg-diagram is only dependant on tyre performance, which is the theoretical maximum performance that a vehicle can obtain. Finally due to the need for an analytical formulation, a least squares approach is used to calculate the best fit ellipse. This ellipse equation can be used in the MPC scheme.

The best fit ellipse method used is based on the conic equation of the ellipse,

$$ax^2 + bxy + cy^2 + dx + ey + f = 0. \tag{4.36}$$

The model is fitted by means of a least squares approach. The ellipse equation is obtained from the following equations,

$$g(x, y, \mathbf{A}) \coloneqq x^2 + bxy + cy^2 + dx + ey = f. \tag{4.37}$$

Where x and y are the single measurements and $\mathbf{A}$ is the vector of parameters to be estimated, hence $\mathbf{A} = [a, b, c, d, e]$. It is now possible to define a cost function as follows,

$$\begin{aligned} J_{\mathbf{A}} \coloneqq & (g_c(x_c, y_c, \mathbf{A}) - f_c)^T (g_c(x_c, y_c, \mathbf{A}) - f_c) \\ = & (X\mathbf{A} + f_c)^T (X\mathbf{A}) = \mathbf{A}^T X^T X A + 2 f_c^T X \mathbf{A} + n f^2. \end{aligned} \tag{4.38}$$

Where $g_c(x_c, y_c; \mathbf{A})$ is a vector which is function of all measurements, in fact, each element is $g(x, y, \mathbf{A})$,

$$g_c(x_c, y_c; \mathbf{A}) = \begin{bmatrix} g(x_1, y_1, \mathbf{A}_1) \\ g(x_2, y_2, \mathbf{A}_2) \\ \vdots \\ g(x_n, y_n, \mathbf{A}_n) \end{bmatrix}. \tag{4.39}$$

X is a matrix,

$$X = \begin{bmatrix} x_c^T x_c & x_c^T y_c & y_c^T y_c & x_c & y_c \end{bmatrix} = \begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 \\ x_2^2 & x_1 y_1 & y_1^2 & x_2 & y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_n y_n & y_n^2 & x_n & y_n \end{bmatrix}. \tag{4.40}$$

Finally, f_c is defined as $f_c = \vec{1} f$ which has dimension $n \times 1$.
The derivation of the cost function with respect to the vector of parameters $\mathbf{A}$ yields in the following,

$$\mathbf{A}^T X^T X = -f_c^T X = -f\Big(\sum_{i=1}^{n} X_{ij}\Big). \tag{4.41}$$

Normalising by $-f$, which is unknown, yields the least squares solution $\hat{\mathbf{A}}$,

$$\hat{\mathbf{A}} = \Big(\sum_{i=1}^{n} X_{ij}\Big)(X^T X)^{-1}. \tag{4.42}$$

Now, the parameters from the conic equation need to be extracted. The first thing to do is check if the ellipse has has a tilt. The orientation is present if the coefficient of the term xy is not zero. If so, the tilt (ϕ) has to be first removed substituting x and y with $\tilde{x} = (x \cos\phi + y \sin\phi)$ and $\tilde{y} = (-\sin\phi x + y \cos\phi)$.

$$a\tilde{x}^2 + b\tilde{x}\tilde{y} + c\tilde{y}^2 + d\tilde{x} + e\tilde{y} + f = 0. \tag{4.43}$$

Calling $c_\phi = \cos\phi$ and $s_\phi = \sin\phi$ the above equation can be simplified.

$$\begin{aligned} &x^2(ac_\phi^2 - bc_\phi s_\phi + cs_\phi^2) + xy(2ac_\phi s_\phi + b(c_\phi^2 - s_\phi^2) - 2cc_\phi s_\phi) + \\ &y^2(ac_\phi^2 + bc_\phi s_\phi + cs_\phi^2) + x(dc_\phi - es_\phi) + y(ds_\phi - ec_\phi) + f = 0. \end{aligned} \tag{4.44}$$

The orientation is easily found by setting the new b coefficient equal to zero.

$$2ac_\phi s_\phi + b(c_\phi^2 - s_\phi^2) - 2cc_\phi s_\phi = 0 \implies \phi = \frac{1}{2}\tan^{-1}\Big(\frac{b}{c-a}\Big). \tag{4.45}$$

Now the constants c_ϕ and s_ϕ can be found and so can the ellipse with no tilt, which equation is as follows,

$$a'x^2 + c'y^2 + d'x + e'y + f' = 0. \tag{4.46}$$

The various coefficient (a', c', d', e') are expressed below.

$$a' = ac_\phi^2 - bc_\phi^2 s_\phi^2 + cs_\phi^2, \tag{4.47}$$
$$b' = 2ac_\phi s_\phi + b(c_\phi^2 - s_\phi^2) - 2cc_\phi s_\phi, \tag{4.48}$$
$$c' = as_\phi^2 + b(c_\phi s_\phi + cc_\phi^2, \tag{4.49}$$
$$d' = dc_\phi - es_\phi, \tag{4.50}$$
$$e' = ds_\phi + ec_\phi. \tag{4.51}$$

Next, the non-tilted ellipse equation has to be transformed into canonical form.

$$\Big(\frac{x - x_0}{a}\Big)^2 + \Big(\frac{y - y_0}{b}\Big)^2 = 1. \tag{4.52}$$

With (x_0, y_0) begin the ellipse centre and a and b being the ellipse sub-axis.
Using a square completion method the following can be defined,

$$f'' = -f' + \frac{d'^2}{4a'} + \frac{e'^2}{4c'}. \tag{4.53}$$

Such that,

$$a(x - x_0)^2 = a'\Big(x^2 + x\frac{d'}{a'} + \frac{d'}{2a'}^2\Big), \tag{4.54}$$
$$c(y - y_0)^2 = c'\Big(y^2 + y\frac{e'}{c'} + \frac{e'}{2c'}^2\Big). \tag{4.55}$$

Which yields the transformations,

$$x_0 = -\frac{d'}{2a'}, \tag{4.56}$$
$$y_0 = -\frac{e'}{2c'}, \tag{4.57}$$
$$a = \sqrt{|f''/a'|}, \tag{4.58}$$
$$b = \sqrt{|f''/c'|}. \tag{4.59}$$

However, the vertical load tyre sensitivity cannot be neglected for all effects, mainly when considering downforce and hard braking. Therefore, the vertical load caused

by aerodynamic forces are considered since it can have a great effect on vehicles with high downforce. Furthermore, the change in downforce is slow and can be approximated as purely dependant on the longitudinal velocity of the car. Thus, the gg-diagram, including downforce effects, can be computed by solving the method described above for different longitudinal velocities and let the gg-diagram also depend on longitudinal velocity as described in Algorithm 2.

Algorithm 2: High-level acceleration limits

Data: Tyre model, Aerodynamic coefficients
Result: Acceleration ellipse
for $i \leftarrow \underline{v}_x$ **to** $\overline{v}_x$ **do**
  calculate $F_{z_{as}} = F_{z0_{as}} + F_{l_{as}}$ (See (4.18))
  for $j \leftarrow \underline{\alpha}_{as}$ **to** $\overline{\alpha}_{as}$ **do**
    for $k \leftarrow \underline{\sigma}_{as}$ **to** $\overline{\sigma}_{as}$ **do**
      calculate $F_{x_{as}}(k, j, F_{z_{as}}, i, 0, 0)$, $F_{y_{as}}(k, j, F_{z_{as}}, i, 0, 0)$ (See (4.30))
find convex hull of $F_{x_{as}}$, $F_{y_{as}}$ matrix with α-shape (Figure 4.6)
calculate sum of external polygons in $\sigma_{as} - \alpha_{as}$ space and divide by mass
fit best ellipse: $\left(\frac{a_x - a_{x_0}}{a}\right)^2 + \left(\frac{a_y - a_{y_0}}{b}\right)^2 = 1$

In the case of braking, considering load tyre sensitivity is especially important due to the great load transfer obtained during braking manoeuvres and to the criticality of such manoeuvres; these effects have to be considered to obtain a feasible velocity profile for the lower level. However, compared to downforce it is not necessary to change the complete gg-diagram, but the limit can be considered by bounding the maximum longitudinal deceleration. This constraint is calculated with an iterative process. This consists of initialising the maximum longitudinal deceleration as the one calculated from the vehicle's adherence ellipse. From this value, longitudinal stationary weight transfer is calculated and, therefore, the new vertical tyre loads. With the new loads, it is then possible to calculate the new vehicle's maximum deceleration. This process is repeated until the difference between the starting vertical load and the one given by the load transfer caused by the braking effort is smaller than a threshold λ_{thresh}. In this work, 500N is used as a threshold. This iterative procedure is further described in Algorithm 3. Note that due to the low downforce of the car considered these two effects can be considered separately, in case of a high-downforce car, the deceleration limit would also depend on longitudinal velocity.

Algorithm 3: Maximum deceleration

Data: Aerodynamic coefficients, Longitudinal velocity, Tyre model
Result: Maximum longitudinal deceleration
initialise $\underline{a}_x$ from $\left(\frac{a_x - a_{x_0}}{a}\right)^2 + \left(\frac{a_y - a_{y_0}}{b}\right)^2 = 1$
calculate $F_{z_{as}} = F_{z0_{as}} - \Delta F_{z_x} + F_{l_{as}}$ (See (4.18))
while $F_{z_{fs}} - F'_{z_{fs}} \geq \lambda_{thresh}$ **do**
  calculate $\underline{a}'_x$ and $\Delta F'_{z_x}$ given vertical loads
  calculate $F'_{z_{as}} = F_{z0_{as}} - \Delta F'_{z_x} + F_{l_{as}}$ with $\underline{a}'_x$

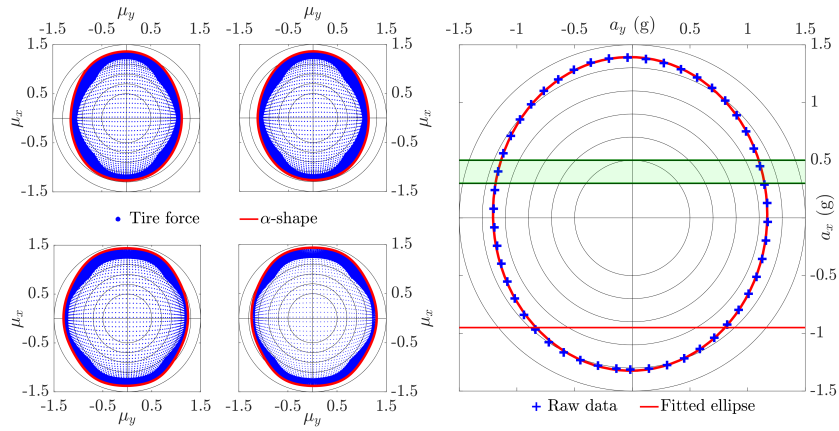

Figure 4.6: Acceleration limits. On the left tyre forces and α-shapes are shown, on the right the resulting gg-diagram consisting of the tyre limits (raw and fitted shape).

Finally, longitudinal accelerations are limited by engine power, which does depend on longitudinal velocity and is considered in the power curve of the engine-gearbox unit. The resulting gg-diagram is mainly dominated by the ellipsoidal shaped tyre gg-limits as shown in Figure 4.6. This shape is cut with two bounds on the longitudinal acceleration, in negative direction caused by the tyre limits under braking (see red horizontal line in Figure 4.6) and positive direction by the engine power, which is additionally dependent on velocity (see green shaded area in Figure 4.6). Thus at low velocities the car can accelerate more than at higher velocities. The resulting constraint is summarized in the set $\mathcal{A}(v_t)$ where v_t highlights that the constraint does depend on longitudinal velocity. Note that other methods that compute the gg-diagram exist, some even include load transfer. But this work explicitly aims for a slightly overestimated gg-diagram. With the procedure described in this section the acceleration limits are dependant on vehicle velocity. Specifically, the gg-diagram is calculated on a sweep of longitudinal velocities as is the longitudinal acceleration constraint since it is dependant on engine power. However, the LSD model is not included in this constraint as the acceleration limits constraint here presented are not used in the low-level control and, as already mentioned, the acceleration limits here described purposely overestimate the vehicle performance especially when the dynamics are dominated by lateral dynamics.

4.2 Control scheme

In this section the proposed hierarchical control scheme will be described. In particular, first an overview of the control scheme will be discussed with details on how the high-level and low-level controllers interact with each other. Then, the high-level controller will be described in detail and its mathematical framework will be shown. Finally, the low-level controller will be described and details about its formulation will be given.

4.2.1 Hierarchical control scheme

The proposed hierarchical control scheme is designed to overcome one of the major issues with MPC controllers for autonomous driving applications. If the goal is to follow a reference path, as in the proposed controller, one has to use a long prediction horizon to adapt longitudinal velocity before curves to stay within the limits of handling in a racing setting and to guarantee passenger comfort in an AD setting. In the setting studied in this paper, where at-limit handling on a racing circuit is of interest, the horizon needs to be long since the car can drive over 200km/h on straights, thus the horizon needs to be several hundred meters to correctly predict the braking point. On the other hand, one would like to use detailed vehicle models with fast actuation times to precisely predict the vehicle's motion, thus controlling the car also at at-limit handling. However, when combining a detailed vehicle model as discussed in Section 4.1.1 with a prediction horizon of several hundred meters, the NMPC problem takes at least several minutes to solve. Therefore, the problem is split into a two level hierarchical controller, where the high-level NMPC controller uses a long horizon N_h and a simple model (point-mass model discussed in Section 4.1.3) to compute a velocity profile for the upcoming two hundred and fifty meters. The low-level NMPC on the other hand uses a detailed vehicle model (vehicle model discussed in Section 4.1.1), and a short prediction horizon N_l to render the problem real-time feasible. The velocity profile from the higher level controller is then used as a terminal velocity constraint in the lower level. This couples the long term velocity profile form the higher level controller to the short term precise decision-making of the lower level NMPC. The control scheme is described in further detail in Algorithm 4.

Algorithm 4: Hierarchical control scheme

Data: Track limits, Vehicle model, Point-mass model, Constraints
Result: Optimal input sequence
initialise q_0 and $\tilde{x}_0$
for $i \leftarrow 1$ **to** N_{sim} **do**
  solve problem (4.61) and obtain $\mathbf{v_x} = [v_{x,1}, ..., v_{N_h}]$
  calculate $v_{x,T}$ by interpolating $\mathbf{v_x}$ and $\mathbf{s} = [s_1, ..., s_{N_l}]$
  solve problem (4.62) and obtain $\hat{\mathbf{u}}(t) = [\dot{T_e}, \dot{B_p}, \dot{\delta_w}]$
  compute $\mathbf{u}(s) = [T_e, B_p, \delta_w]$
  apply $\mathbf{u}(s)$ to vehicle and calculate q_0 and $\tilde{x}_0$

For example, the terminal velocity allows the low-level NMPC to start braking at the correct braking point without even "seeing" the next curve. The concept of the hierarchical controller is illustrated in Figure 4.7 together with a block diagram in Figure 4.8. Where the velocity profile found by the high-level controller is plotted on a typical slow curve after a straight line and the point along the trajectory at which this velocity is used as a terminal constraint for the low-level control is shown.
The advantage of this hierarchical approach is that the high-level problem, even with a prediction horizon of two hundred and fifty meters, can be solved within less than a millisecond on a thread of an Intel i7-6700HQ CPU at 2.60GHz. At the same time the horizon of the low-level problem can be reduced to below thirty meters which allows solving the problem in approximately one hundred millisec-

onds, thus rendering the whole hierarchical controller real-time feasible. Finally, the higher level only determines a terminal constraint, thus the low-level NMPC is free to optimise its own trajectory except of the terminal velocity. This allows the vehicle really to act at at-limit handling and as a consequence, in certain areas of the circuit, the low-level controller achieves a higher speed than the generally optimistic high-level controller.

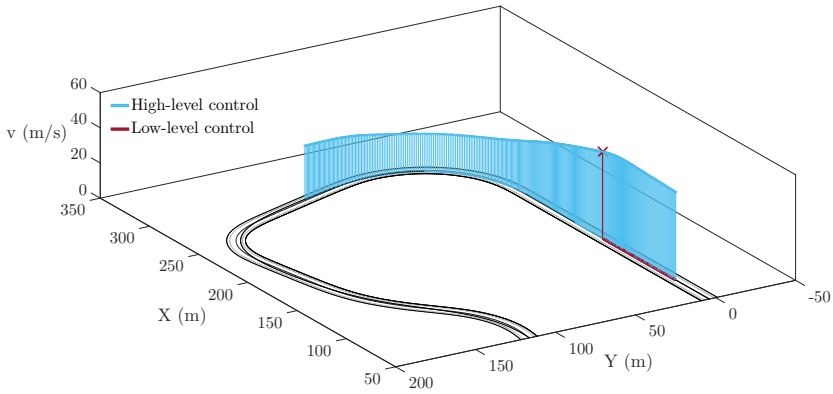

Figure 4.7: Hierarchical control scheme.

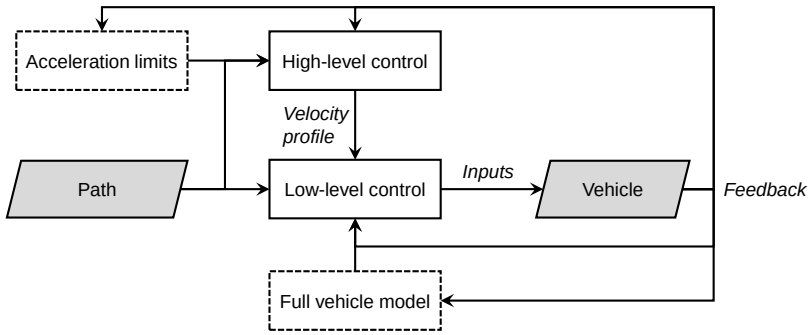

Figure 4.8: Hierarchical control scheme - block diagram.

4.2.2 High-level controller

The high-level NMPC is designed to find the time optimal trajectory of a point-mass particle following a reference path. The model of the point-mass particle $g(\mathbf{q}, \mathbf{w})$ is given by (4.35), where the state $\mathbf{q}$ comprises tangential velocity v_t and tangential force f_t and the input $\mathbf{w}$ is the rate of change of tangential force j_t. Note that centripetal acceleration is given by $a_n = v_t^2 k(s)$. The accelerations of the point-mass particle are constrained to lay within $\mathcal{A}(v_t)$, which is described in Section 4.1.3. Where the acceleration limits come from tyre adherence limits, braking limits and limited engine power. The minimum time objective is straight forward to achieve since the model is formulated in the spatial form with distance as the independent variable. Thus the minimum time objective has the following transformation,

$$\min \int_{s_0}^{s_f} t \, d\xi = \min \int_{s_0}^{s_f} \frac{1}{\dot{s}(\xi)} \, d\xi \,, \tag{4.60}$$

where s_0 and s_f are the start and finish position, $\dot{s}$ is defined in (4.31a) and ξ is the curvilinear abscissa integration variable. Since n, α, and v_y are zero for the point-mass model, $\dot{s} = v_t$. Thus $1/\dot{s}(\xi)$ further simplifies to $1/v_t(\xi)$. Finally, the dynamics and the cost function are discretised to transform the problem into a format such that a NLP solver can solve the problem. The dynamics are discretised using a fourth order Runge-Kutta method, whereas the cost integral is approximated with a Riemann integral. Note that this approach is often referred as a multiple shooting.

$$\begin{aligned}
\min_{\mathbf{q},\mathbf{w}} \quad & \sum_{0}^{N_h} \frac{1}{v_{t,k}} + w_k^T R w_k \\
\text{s.t.} \quad & q_0 = \mathbf{q}(0) \\
& q_{k+1} = g_d(q_k, w_k) \\
& \left[\frac{f_{t,k}}{m}, v_{t,k}^2 k(s_k) \right] \in \mathcal{A}(v_t) \\
& v_{t,k} > 0 \\
& w_k \in \mathcal{W} \,.
\end{aligned} \tag{4.61}$$

Where $\mathbf{q} = [q_0, ..., q_{N_h}]$ is the state trajectory and $\mathbf{w} = [w_0, ..., w_{N_h - 1}]$ the input trajectory. Furthermore, is $\mathbf{q}(0)$ the measured state, $g_d(q_k, w_k)$ the discretised version of the dynamics, $v_{t,k} > 0$ is imposed to ensure that the spatial transformation is well defined, and $\mathcal{W}$ represents upper and lower bounds on the input $\mathbf{w}$.

4.2.3 Low-level controller

Similar to the high-level controller the goal of the low-level controller is also to follow a path in a time optimal way. However, compared to the high-level controller, the model consists of a seven degrees of freedom model, a full MF6.1 tyre model, a LSD and tyre vertical load variations as described in Section 4.1.1. Furthermore, the car can deviate from the reference path, thus constraints on lateral distance to the path are introduced to guarantee that the car stays inside the track limits. Finally, the terminal velocity constraint is introduced based on the optimal solution of the

high-level controller. Thus the discrete time MPC problem can be formulated as follows,

$$\begin{aligned}
\min_{\tilde{\mathbf{x}},\hat{\mathbf{u}}} \quad & \sum_{k=0}^{N_l} j_{\text{time},k} + j_{\text{dyn},k} + j_{u,k} + j_{\hat{u},k} + j_{\text{slack},k} \\
\text{s.t.} \quad & \tilde{x}_0 = \tilde{\mathbf{x}}(0) \\
& \tilde{x}_{k+1} = \tilde{f}_d(\tilde{x}_k, \hat{u}_k) \\
& n_k \geq \underline{n}_k - \rho_k \,, \quad n_k \leq \overline{n}_k + \rho_k \,, \quad \rho \geq 0 \\
& \tilde{x}_k \in \tilde{\mathcal{X}} \\
& \hat{u}_k \in \hat{\mathcal{U}} \\
& v_{x,N_l} \leq v_{x,T} \,.
\end{aligned} \tag{4.62}$$

Where $\tilde{\mathbf{x}} = [\tilde{x}_0, ..., \tilde{x}_{N_l}]$ is the state trajectory and $\hat{\mathbf{u}} = [\hat{u}_0, ..., \hat{u}_{N_l-1}]$ the input trajectory. The cost is built up of several components, first the time minimisation $j_{\text{time},k} = 1/\dot{s}_k$, where $\dot{s}$ is defined in (4.31). Second, regularisation costs related to the dynamics of the car, penalising deviations from the path, angle deviations from the path and sideslip angle, $j_{\text{dyn},k} = d_n n_k^2 + d_\alpha \alpha_k^2 + d_\beta [\tan^{-1}(v_{y,k}/v_{x,k})]^2$, where d_n, d_α, and d_β are weights. Third, $j_{u,k} = u_k^T R u_k$ and $j_{\hat{u},k} = \hat{u}_k^T R_\Delta \hat{u}_k$ penalise deviations from zero in the real inputs as well as the input rates, with R and R_Δ being positive definite weight matrices. Finally, $j_{\text{slack},k} = \gamma \rho^2$ is the penalisation of the slack multipliers related to the soft constraint implementation of the track boundary constraints. Furthermore, $\tilde{\mathbf{x}}(0)$ is the measured state, $\tilde{f}_d(\tilde{x}_k, \hat{u}_k)$ the discretised vehicle dynamics described in Section 4.1.1, where an implicit Runge-Kutta method is used to integrate the dynamics. The boarder constraints are implemented as bounds $\underline{n}$ and $\overline{n}$ on the orthogonal deviation from the path n, to avoid feasibility issues the constraints are implemented as soft constraints. $\tilde{\mathcal{X}}$ and $\hat{\mathcal{U}}$ represent bounds on all the states and inputs, explicitly on the real inputs but also on the derivatives of the inputs. Note that the bounds on the drivetrain torque do change depending on wheel speed of the rear axle, which approximates the changing torque bounds due to the engine speed and gear. A plot of the wheel torque envelopes considering each gear is shown in Figure 4.9 with the feasible region considered by the controller highlighted in grey. Note that also the maximum and minimum torque for each gear are illustrated. This approach assumes that the

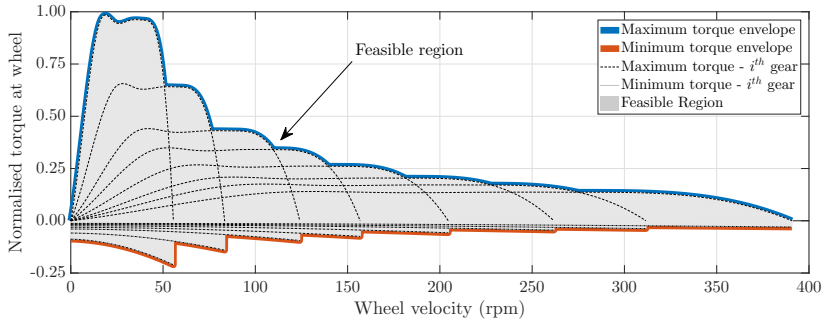

Figure 4.9: Torque constraints.

gear which achieves the highest torques is always selected, which is a reasonable

assumption for automatic racing and helps to avoid integer variables due to the gear. Finally, the longitudinal velocity at the end of the horizon is limited by $v_{x,T}$, the terminal constraint from the higher level. Note that the low-level controller is not allowed to drive faster than the terminal velocity constraint only at the end of the horizon while the minimum time objective is concerned about achieving the best possible velocity.
In the above description an implicit integrator is used. This is necessary since the wheel dynamic equations are stiff, especially under braking for low speed. To further increase the robustness with respect to integration errors, the discretisation distance is chosen to be adaptive. In the spatial formulation, the discretisation distance is fixed, thus when the car is driving slower, the "discretisation time" gets longer. Hence, integration errors are getting larger exactly when the dynamics are most challenging. To counteract this effect, the discretisation distance is adapted based on the minimal velocity of the last optimal solution. This results in a behaviour similar to time discrete systems where the horizon length in distance depends on velocity. In addition to adding numerical robustness, with the proposed method the horizon grows when the velocity is higher and is shorter when the velocity is lower. This is closer to what a human does since for low velocities a very long horizon is not required. With this method, the advantages of both space and time formulations are maintained and the algorithm is proved to be very robust to critical situations such as wheel locking.

4.3 Results

In this section the results of the proposed algorithm will be shown and discussed. The simulations are run on a previously validated Vi-Grade model. The controller has been implemented in MATLAB on an off-the-shelf laptop with an Intel i7-6700HQ CPU at 2.60GHz. To solve the NMPC problems, FORCES Pro NLP solver is used [117], which is a NLP solver tailored for MPC problems exploiting the given sparsity pattern, more information about the solver can be found in the paper of Zanelli et al..[118] The low-level controller solves the optimisation at a mean frequency of 10Hz and feeds a new input (input rate) at the same frequency. The real input is applied at a frequency of 100Hz considering a constant (input) rate. The high-level control runs at a mean frequency of 200Hz, however, the input is fed to the model at the same frequency as the lower level (10Hz for the rate and 100Hz for the real input). The vehicle model runs at 1kHz. The horizon length of the high-level controller is $N_h = 250$ with a discretisation distance of 1m which results in a prediction horizon of two hundred and fifty meters while the low-level has a horizon length of $N_l = 30$ with a discretisation distance of 1m at 30m/s which resuts in a prediction horizon of thirty meters.
The simulations that will be shown in this section are those of a sport saloon driving around Hockenheim circuit. First, the outputs of the high-level controller will be shown. Then, the results of the low-level controller of the hierarchical scheme will be discussed. Note that the two controllers are tied together due to the feedback that the high-level control gets from the low-level control and the terminal constraint that the high-level feeds to the low-level. However, for clarity of exposure, the results will be initially analysed separately. The results of the high-level control consider only the values corresponding to the terminal curvilinear abscissa of the low-level control.

4.3.1 High-level controller results

The goal of the high-level controller is to supply the terminal velocity constraint to the low-level. It is important not to underestimate the vehicle's performance. As mentioned in the previous section, vehicle's lateral behaviour can easily be predicted by the low-level controller even with a short horizon since the track's curvature is large and smooth. On a straight line instead, it is the high-level controller which has to dictate the terminal velocity and, more importantly, the braking point. In fact, to correctly predict when to start braking it is important to have very long horizons. The velocity profile which the high-level controller calculates along the horizon should have abrupt changes in acceleration (e.g. hard braking) since this way the low-level controller will have a realistic braking profile as a reference.
Figure 4.10 shows the velocity profile calculated by the high-level controller by running the high-level controller in a receding horizon fashion with point-mass model as a simulation model. The top figure shows the velocity profile while the lower figure shows both the tangential and normal acceleration profiles. The velocity profile is very smooth and shows the typical high gradients given by a braking manoeuvre executed by a human. Additionally, longitudinal velocity has a quadratic profile due to aerodynamic forces. With regards to the acceleration profiles, both normal and tangential accelerations tend to the gg-diagram constraints implemented in the controller. Since positive longitudinal acceleration is bounded with velocity, maximum longitudinal acceleration varies along the circuit. The oscillatory profiles of the accelerations are due to the states being in a small region near the bounds, leading the solver to oscillate around a feasible region. Since acceleration constraints are modelled as hard constraints, when the point-mass finds itself near to the bounds, the solver tends to push the vehicle to the limit while maintaining the results feasible, resulting in a oscillatory profile. However, as shown in the velocity profile, these high frequency oscillations are damped in the velocity domain. Therefore it can be concluded, that these high frequency fluctuations do not influence the purpose of the controller which is to give a terminal constraint to the lower level.

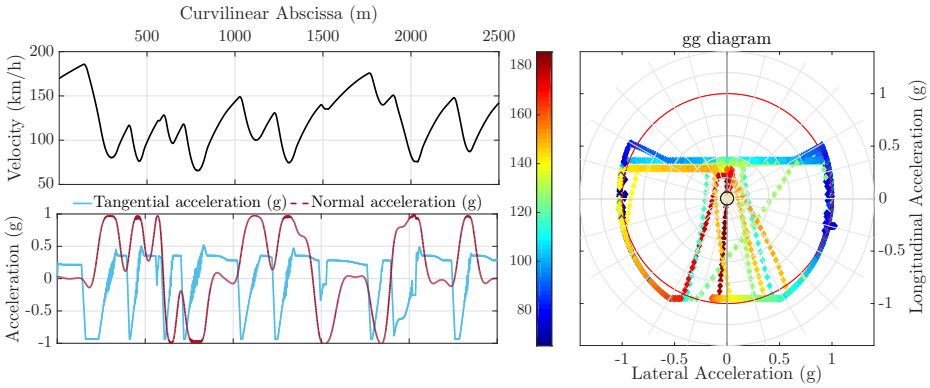

Figure 4.10: High level controller velocity profile.

The resulting gg-diagram obtained by the high-level controller is also shown in

Figure 4.10.
Note that the high-level controller acts at the limits of the theoretical gg-diagram (acceleration constraints). In fact, in the acceleration space, the resulting acceleration profile generates a shape which is similar to the adherence ellipse shown in Figure 4.6. Additionally, the maximum negative deceleration (braking) is saturated to the value used as a constraint and found with the previously described algorithm (Algorithm 3). On the other hand, the maximum positive acceleration tends to follow different horizontal lines at different speeds. This is a consequence of a velocity dependant torque constraint in the controller. For low speeds and high lateral accelerations, which correspond to low speed corners exits, the point-mass' acceleration limits are given by the tyre constraints rather than torque constraints. For this reason the maximum longitudinal acceleration at high lateral acceleration and low velocity is higher than in other areas of the state space.

4.3.2 Low-level controller results

The lower level controller is the one responsible for the real performance of the vehicle. Assuming that the higher level did not underestimate the acceleration limits, the terminal constraint feeds the braking point to the low-level. Thus, it is interesting to analyse the difference in the velocity profiles obtained by the low-level controller and the high-level controller. Note that since the high-level receives its initial condition as a feedback from the low-level, the velocity in the prediction horizon of the low-level is illustrated. The low-level has a short horizon and solves the optimisation problem slowly while the high-level has a long horizon and solves the problem quickly. The comparison of the velocity profiles can be seen in Figure 4.11. Note that the high-level controller results are based on a point-mass

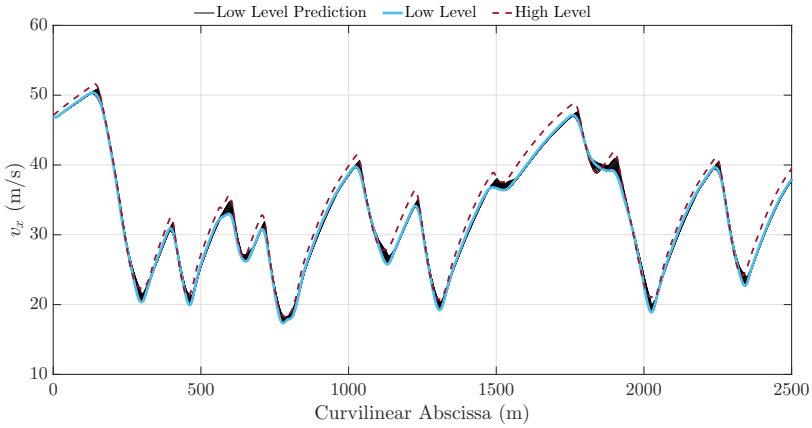

Figure 4.11: Velocity profiles comparison.

model on a given trajectory. Thus, absolute velocity is equal to tangential velocity. On the other hand, the low-level controller is based on rigid body dynamics: thus, absolute velocity comprises also lateral velocity. However, in this plot only longitudinal velocity is shown since it is the main contribution in terms of velocity for low sideslip angles.
It can be seen how terminal velocity and real velocity are very similar but, as expected, the real velocity is lower than the terminal constraint almost everywhere.

However, for certain values of curvilinear abscissa, the real velocity is higher than the terminal one. This result shows one of the advantages of using only a terminal set for the controller instead of the profile along the entire horizon. Additionally, the velocity profile generated by the high-level controller is far from realistic given the model's simplifications. Note that in the abscissa plots the proposed algorithm proves to be effective since the real velocity follows the terminal velocity where abrupt braking is required. However, when the vehicle is performing a curve, real velocity is much lower than the terminal constraint. In this condition, the low-level control, as expected, performs less than the high-level control due to lateral load transfer and vertical tyre sensitivity. The need of a weight transfer formulation in the low-level NMPC model is highlighted here. Longitudinal positive accelerations are very similar for both controllers, as can be seen from the gradients of the velocity profiles. To fully evaluate the performance of the controller, the gg-diagram is shown in Figure 4.12. In this figure both the acceleration diagram and the el-

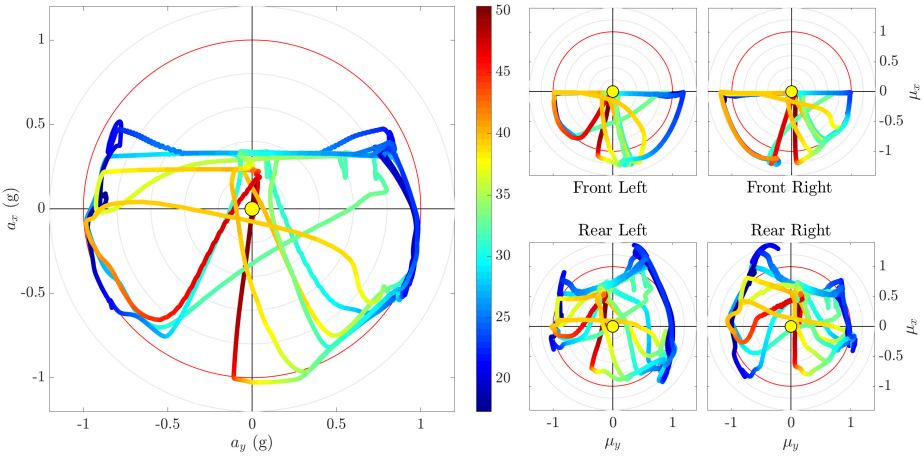

Figure 4.12: Low level controller gg-diagram.

lipse adherence of the four tyres are shown. Note that these results come from a commercial software vehicle model which is controlled by the proposed scheme. Analysing the acceleration diagram, it can be noted how the low-level controller has a very similar profile to the acceleration diagram of the point-mass model. The acceleration diagram shows how the controller is able to exploit the combined behaviour of the vehicle where normally a lot of performance is lost. In these conditions the tyres are operating in their non-linear range and every little variation can make the vehicle loose control or performance. On the other hand, a robotic controller based on a predictive control with a high fidelity model can maximise vehicle's performance. In the positive longitudinal behaviour, it can be observed how for low speed and high lateral acceleration, once again, the peak acceleration is given by the engine torque rather than the tyres. Finally, since the analysed vehicle has negative downforce (as most commercial vehicles), the peak lateral acceleration decreases with speed. Positive lateral accelerations in this circuit are more frequent due to the circuit orientation.
In this figure tyre forces are also shown. The acceleration diagram is a direct result

of these forces as previously explained. The advantages of considering a combined tyre formulation, weight transfer and, more importantly, LSD formulation can be seen in the rear tyre force plots where the maximum force is obtained in combined slip conditions. Contrarily, the front tyres show a more symmetric behaviour.
The results of the proposed controller, in terms of acceleration diagrams, can be compared to Vi-Driver, the standard driver used in commercial software Vi-Grade. This driver is not a real-time driver and relies on backstepping when a simulation does not converge. This is very advantageous for Vi-Driver since it does not have real-time constraints in the solver. However, it is interesting to see how the results compare. The comparison is showed in Figure 4.13. Both the high-level controller

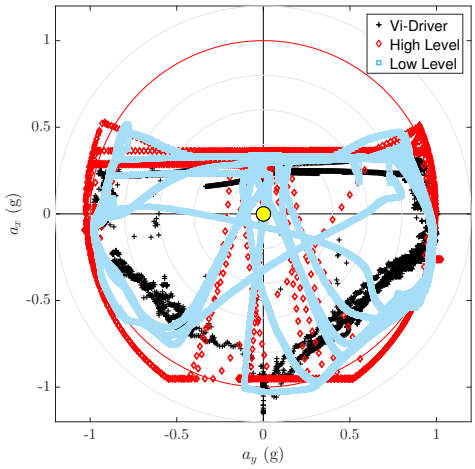

Figure 4.13: Comparison of gg-diagram.

and the low-level controller outperform the commercial driver. While for the high-level this was expected, the same cannot be said for the low-level. Note that the same vehicle, trajectory and solver for the vehicle model are used in all simulations. The combined behaviour of the proposed algorithm is where performance is most improved, showing the importance of considering the complete MF model, comprehensive of combined formulation. Also, the low-level controller performs very similarly to the higher level in terms of combined behaviour except for hard braking. This is due do the fact that under hard braking, wheel dynamics become critical for the solver since wheel under-rotation can easily occur. Overall, it can be concluded that the proposed method shows promising results when compared to state of the art controllers.
Besides longitudinal velocity and accelerations, it is important to analyse other significant states of the vehicle, such as sideslip angle and yaw rate. In Figure 4.14 both of these quantities are plotted together with longitudinal velocity. These are plotted versus time and with a colormap along the circuit. Sideslip angle gives an idea of the lateral slipping of the vehicle. Since tyre slip angle depends on yaw rate and sideslip angle, smaller sideslip angle implicates a greater possibility to yaw. However, due to non-linear behaviour of the dynamics and to the non neutral behaviour of the vehicle, the optimiser could converge to a solution where drifting around a curve is optimal for time. This would make the vehicle unstable and would

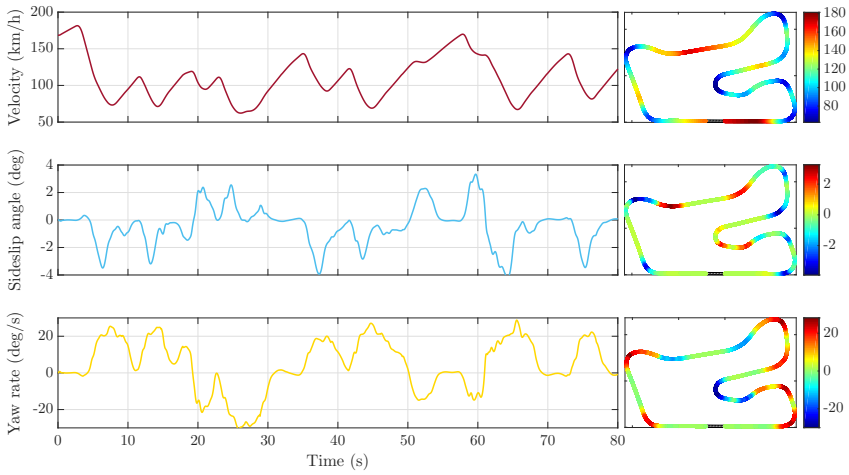

Figure 4.14: Velocity, sideslip angle and yaw rate of the proposed algorithm.

not be a realistic driving scenario. With the proposed controller, a cost term in the objective function of the low-level controller ensures a bounded sideslip angle, enhancing stability, while at the same time allowing the vehicle to drift slightly when required. The maximum absolute value of sideslip angle obtained around the circuit corresponds in this case to the maximum open-loop sideslip angle of the vehicle. The highest values of sideslip angle occurs when an inversion of yaw rate sign is required since, in such conditions, a pendulum effect is generated and the front and rear slip angles are in counter phase so a great yaw moment is produced. Note that the vehicle analysed is characterised by a positive stability margin. Considering yaw rate, the absolute values obtained are not excessive since the vehicle is stable. The important thing to note is that the controller is capable of inverting yaw rate sign without loosing control and performing agile manoeuvres.
Finally, the vehicle's inputs are analysed, namely steering angle, drive torque and braking percentage. As previously described, the controller calculates the rates of these quantities. Since the solver calculates a solution every 10Hz, this guarantees a smooth input sequence. In fact, an input rate is calculated every 10Hz, however, the inputs are fed to the Vi-Grade model at 1kHz. Thus, during each optimisation time window, one hundred different inputs are fed to the model. This is done with an interpolation in the optimal input sequence which the NLP solvers returns (input rate). Furthermore, due to the input dynamics used, the real inputs have first order hold characteristic, with the gradient of the inputs defined in (4.22). The results of the optimisation can be seen in Figure 4.15 where, for the reasons explained, the input sequence is very smooth. Note that in the drive torque curve, there are various peaks given by gear changes. The cost in the objective function on steering angle was kept very high to guarantee a smooth steering action, however, as shown in the figure, high frequency corrections still occur when necessary. The most critical input to calculate for the optimiser is brake percentage. This is due to the already mentioned stiff wheel dynamics. For this reason, only seldomly does brake percentage reach its maximum value.
Since the proposed controller is a path-following controller, it is interesting to also

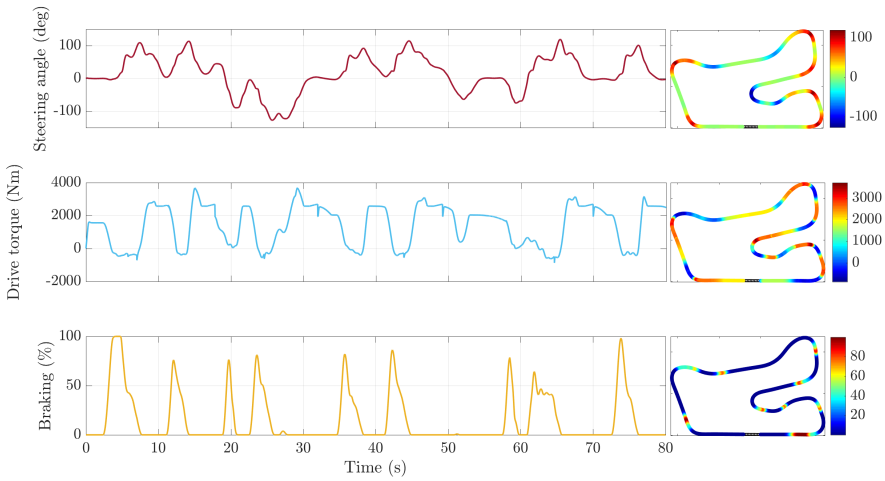

Figure 4.15: Inputs the proposed algorithm.

see how well the path is followed. Considering the coordinate transformation described in Section 4.1.2, this can be done by analysing the lateral error and heading angle. The high-level control cost function seen in (4.61) has a cost term $j_{\text{dyn},k}$ which comprises two path-following terms $d_n n_k^2$ and $d_\alpha \alpha_k^2$ and one stability term $d_\beta[\tan^{-1}(v_{y,k}/v_{x,k})]^2$. The cost on the path-following terms together with the time optimality one can be tuned to obtain better or worse vehicle performance or trajectory tracking. In this work these terms have been tuned to maximise vehicle performance rather than path tracking. Despite this, as shown in Figure 4.16 the lateral error is always under one meter and only rarely it is above half a meter.

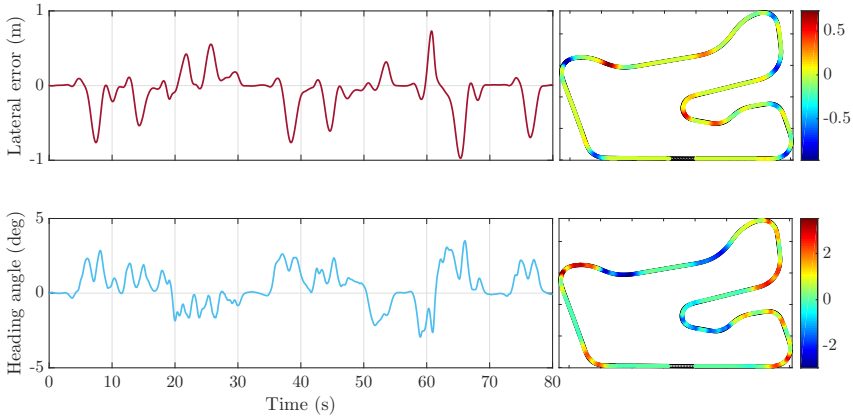

Figure 4.16: Lateral error and heading angle.

It is important to note how the areas on the track where the error is higher are when lateral acceleration is at its peak, meaning that the vehicle in these areas of the circuit is slipping and very little control authority is available since the tyres are at the limit of their performance. This is confirmed by the plots shown in

Figure 4.14 where yaw rate and sideslip angle (especially the latter) are highest in the same areas. This is particularly true when an inversion of yaw rate and sideslip angle vectors are present, meaning that the derivatives of these quantities are higher. In these parts of the lap, slip angles tend be in counter phase and the vehicle is hardest to control especially due to its understeering characteristic and high inertia. Similar comments can be made for the heading angle, however, in this case the state presents a higher frequency content in respect to the lateral error, mainly due to the greater capacity of the vehicle to yaw compared to its capacity to move laterally.
Since the controller solves an optimisation problem it is important to analyse also the time required for each optimisation. The solving time (of the low-level controller) is the critical one for real-time feasibility at 10Hz. Note that the inputs are applied at a rate of 100Hz. However this is a linear interpolation between the solution of the optimisation problem since actuator dynamics are considered. Figure 4.17 shows the optimisation time and number of iterations for each step of the solution.

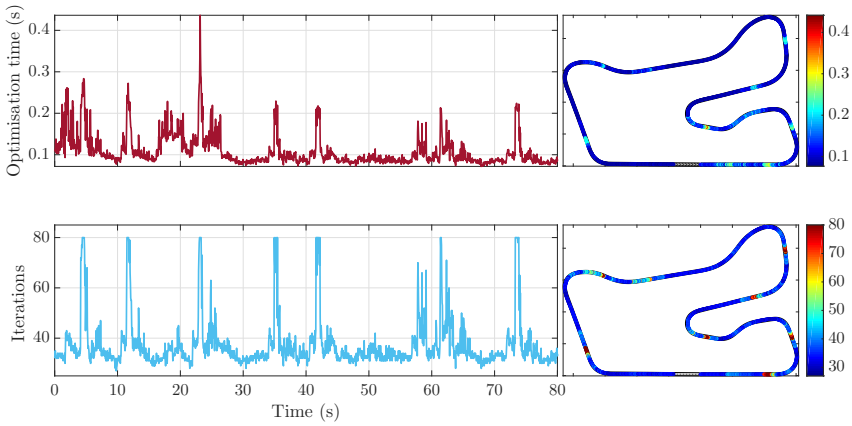

Figure 4.17: Optimisation time and number of iteration.

For most part of the circuit, the 10Hz target is achieved. Note that when the solver does not converge or if the time to solve the optimisation is larger than a given threshold, the controller applies the second step of the previous solution. It is possible to generalise this and increase the actuation frequency if necessary by using two inputs of each optimisation instead of one. This would half the solving time of each step and would still guarantee good results given the high fidelity model used. This is confirmed by the fact that the mismatch between the prediction and the model are very little and the prediction between two subsequent optimisations is similar as shown in Figure 4.18.
To decrease the maximum solving time, the number of iterations is limited to eighty. Note that this is reached only in corner entry during sharp braking phase. This is mostly because of the stiff wheel dynamics previously discussed.
Finally, the torque constraints and horizon length along the track are shown in Figure 4.19. Specifically, it is interesting to see how both the upper torque constraint and the lower torque constraint vary along the track. This is the result of the torque constraint definition shown in Figure 4.9. As previously explained, these

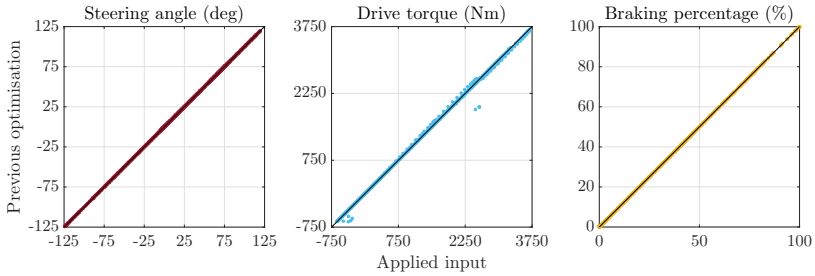

Figure 4.18: Difference between the applied input and the second input of previous optimisation.

constrains are calculated based on the velocity prediction of the previous solution of the optimisation. Once again this is possible thanks to the use of a high fidelity model.

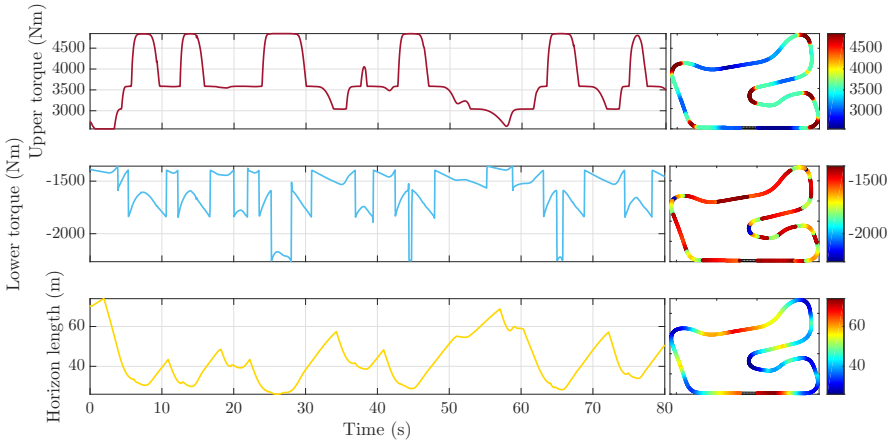

Figure 4.19: Torque constraints and horizon length.

The horizon length is also shown since, as previously discussed, a variable length prediction horizon is used based on the minimum velocity of the previous optimisation. Note how at low speeds the horizon is short while at high speeds the horizon is long. This corresponds to how a human would behave and guarantees an increase in both vehicle performance and numerical stability (stiff wheel dynamics).

Chapter 5

Autonomous Driving and Passive Vehicle Dynamics

A study to evaluate the influence of autonomous driving on some classic vehicle dynamics parameters is discussed in this chapter. The controller is still a NMPC but a simplified version of the hierarchical controller described in the previous chapter. Note that only the lower level is used.
Specifically, vehicle stability is put under discussion. The goal is to evaluate how much stability is needed with AD when compared to a human driver. For this reason, a driving simulator is used to compare various vehicle configurations and compare the results with the ones of a robotic controller. The outcome of this study shows interesting results in terms of how car manufacturers could design vehicles with AD.

5.1 Vehicle stability

There are many ways to analyse the stability of a vehicle and different methods and metrics can be found in the literature.[119] Since the goal of this work is to see the influence of stability on autonomous vehicles, the well-known static margin SM is used to characterize stability (under the assumptions of steady-state and linear tyre range).[7] Despite the SM being valid only under restricted assumptions, this simple parameter is chosen to evaluate vehicle stability since it allows the easy categorisation of different vehicle configurations and is often the metric used by car manufacturers. Considering the bicycle model shown in Section 3.1.1, the SM can be defined as:

$$SM = \frac{1}{(l_f + l_r)} \frac{(l_r C_r - l_f C_f)}{(C_f + C_r)}. \tag{5.1}$$

Hence, considering the UG defined in (3.21):

- UG=0 $\rightarrow$ SM=0 $\rightarrow$ Neutral vehicle

- UG>0 $\rightarrow$ SM>0 $\rightarrow$ Understeering vehicle

- $UG<0 \rightarrow SM<0 \rightarrow$ Oversteering vehicle

Considering linearised lateral equations of motion of a vehicle in state space form,

$$\begin{cases} mV\dot{\beta}(mV + C_f + C_r)\beta + \left(mV + C_f\frac{l_f}{V} + C_r\frac{l_r}{V}\right)\dot{\psi} = C_f\delta \\ J_z\dot{\psi} + (C_f l_f + C_r l_r)\beta + \left(C_f\frac{l_f^2}{V} + C_r\frac{l_2^2}{V}\right)\dot{\psi} = C_f l_f \delta. \end{cases} \tag{5.2}$$

Under the assumption of steady-state ($V = 0$) and constant inputs ($\delta = \text{cost.}$) and with $\mathbf{x} = [\beta, \dot{\psi}]^T$ being the state vector, the following linear system is obtained,

$$A\dot{\mathbf{x}} + B\mathbf{x} = C\delta. \tag{5.3}$$

Where,

$$A = \begin{bmatrix} mV & 0 \\ 0 & J_z \end{bmatrix}, \tag{5.4}$$

$$B = \begin{bmatrix} C_f + C_r & mV + C_f\frac{l_f}{V} + C_r\frac{l_r}{V} \\ C_f l_f - C_r l_r & J_z \end{bmatrix}, \tag{5.5}$$

$$C = \begin{bmatrix} C_f \\ C_r l_f \end{bmatrix}. \tag{5.6}$$

Given constant inputs, the solution of the equation si given by,

$$\mathbf{x} = \mathbf{x}_0 e^{\lambda t}. \tag{5.7}$$

Hence, the non-trivial solution can be found as,

$$det(\lambda A + B) = 0 \implies P^T P\lambda^2 + Q\lambda + R = 0, \tag{5.8}$$

where,

$$P = mJ_z V, \tag{5.9}$$

$$Q = m(C_f l_f^2 + C_r l_r^2) + J_z(C_f + C_r), \tag{5.10}$$

$$R = {}^1\!/_V[C_f C_r(l_f + l_r)^2 - mV^2(C_f l_f - C_r l_r)]. \tag{5.11}$$

The solutions that yield from (5.8) are the following,

$$\lambda_{1,2} = -\frac{Q}{2P} \pm \sqrt{\left(\frac{Q}{2P}\right)^2 - \frac{R}{P}}. \tag{5.12}$$

Given $\Delta = (\frac{Q}{2P})^2 - \frac{R}{P}$, three types of solutions can be found,

1. $R > 0$, $\Delta > 0 \implies Q > 2\sqrt{PR} = Q_{\text{crit.}}$
   $\lambda_1 = -c_1$, $\lambda_2 = -c_2 \in \mathbb{R} \implies$ asymptotically stable,
2. $R > 0$, $\Delta < 0 \implies Q < 2\sqrt{PR} = Q_{\text{crit.}}$
   $\lambda_1, \lambda_2 = -c + j\omega \implies$ oscillating asymptotically stable,
3. $R < 0$
   $\lambda_1 = -c_1\,, \lambda_2 = c_2 \in \mathbb{R} \implies$ statically unstable.

Therefore, if a vehicle has an understeering characteristic ($UG > 0$), the coefficient R is always positive and the vehicle is always stable, for all velocities. On the contrary, if a vehicle has an oversteering characteristic ($UG < 0$), the coefficient R is positive only below the critical velocity $V_{\text{crit.}} = \sqrt{1/UG}$ and the vehicle is stable only for the same conditions.
An understeering vehicle is stable for all conditions whilst an oversteering vehicle is unstable for velocities larger than the so called critical velocity.[120] To analyse the influence of vehicle stability on vehicle performance, various configurations of the same vehicle with different static margins are implemented in Vi-Grade. The only variation made between the various models are tyres.

5.1.1 Tyre merging

The nominal vehicle considered is a RWD and front steering car with differentiated front and rear tyres. The rear tyres have higher grip compared to the front ones due to a larger width. As in the previous chapter, the Pacejka MF is used to model tyres. Note that this model is used for both the simulator model and the one controlled by the NMPC model. Like in the hierarchical controller, due to the need to improve computation time, the NMPC Pacejka model neglects aligning torques and overturning moments. As described in Section 4.1.1, despite this being a model simplification, the increase in performance which these effects provide to the controller are smaller with respect to the loss given by the increase in computation time. The simplification is therefore the best compromise since: combined force effects which are the main cause of tyre performance are still kept into consideration and at the same time the controller is capable of performing in real-time. Starting from the original front and rear tyres, six different tyre models can be created in the following way. Every i^{th} MF parameter p_i^k (for a total of N parameters) of the k^{th} tyre model is obtained by interpolating linearly the correspondent i^{th} MF parameter of the original front p_i^f and rear p_i^r tyres. For every k^{th} tyre model, the merging factor of the interpolant line m^k (merge) is varied from 0 to 1 with a variation of 0,2 between model k and $k-1$ resulting in the following formulation:

$$p_i^k = (1 - m^k)p_i^r + m^k p_i^f,\ 0 \leq m^k \leq 1,\ m^k - m^{k-1} = 0,2,\ i = 1, ..., N, \tag{5.13}$$

$$\text{if } m^k = 1 \Rightarrow p_i^k = p_i^f \Rightarrow \text{Original front tyre}, \tag{5.14}$$

$$\text{if } m^k = 0 \Rightarrow p_i^k = p_i^r \Rightarrow \text{Original rear tyre}. \tag{5.15}$$

The Pacejka parameters of the original model are found by fitting real data and are then used to validate the vehicle model with experimental results. The goal of the scaling process is to simulate tyres with similar compound and construction but different size. The results of the tyre merging show that the pure longitudinal and lateral behaviour changes both in terms of peak friction coefficient μ_{peak} and cornering stiffness as shown in Figure 5.1. The latter is particularly true for the lateral behaviour where also the decay after μ_{peak} is very low. The combined force behaviour shown in Figure 5.2 also shows that lateral forces are more affected by the merging operation. The lateral $\mu_{y_{\text{peak}}}$ between the tyre with highest grip and the one with lowest is approximately 9% whilst the longitudinal one $\mu_{x_{\text{peak}}}$ is approximately only 5%.

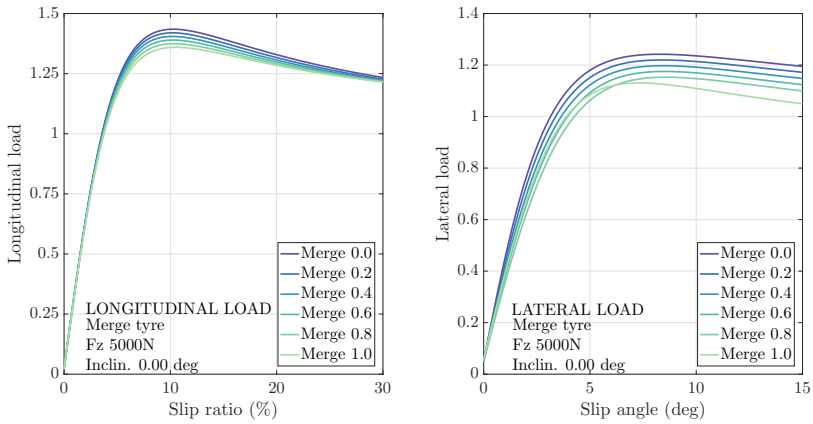

Figure 5.1: Tyre merging results – pure longitudinal and lateral behaviour.

5.2 Vehicle configurations

With the six tyre configurations, five different vehicle models can be created. As previously mentioned, these vehicles have the same inertial properties, K&C characteristics and geometrical features. The goal is to obtain vehicles with different SM values and, consequently, peak performance. The various tyre models used for the different vehicle models are shown in Table 5.1. Where the vehicle at Step00

Table 5.1: Vehicle configurations – tyre models.

	Step 00	Step 02	Step 04	Step 06	Step 08
Front tyre m^k	1,0	0,8	0,6	0,4	0,2
Rear tyre m^k	0,2	0,4	0,6	0,8	1,0

corresponds to the vehicle with the original tyres. The open-loop behaviour of these vehicles is analysed in both steady-state and transient manoeuvres to obtain values of SM, peak lateral acceleration (for steady-state manoeuvres) and time delay.

5.2.1 Steady-state

As in Section 3.1.1, the steady-state behaviour is analysed by means of a ramp steer manoeuvre and both peak lateral acceleration and static margin are evaluated and are shown in Figure 5.3. The reference value of the SM for the various configurations is taken in the linear range of the tyres, at a lateral acceleration of 0,2g. Static margin varies from a value of +12% of the original vehicle to a value of -8% for Step08. Hence, the vehicle goes from being stable and understeering to being unstable and oversteering. This is the result of the merging tyre process. For Step00, the front tyres saturate before the rear ones. The latter have a higher peak friction coefficient and, therefore, generate a large anti-yaw moment. The result is a vehicle with higher yaw damping (see Table 3.1), given also the higher rear cornering stiffness. The opposite is valid for Step08. The front to rear grip

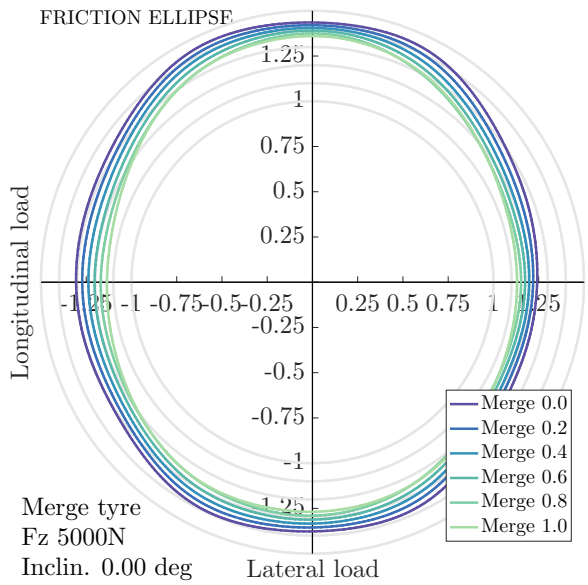

Figure 5.2: Tyre merging results – combined force behaviour.

distribution is in favour of the front tyres and the result is a vehicle with low yaw damping. However, the peak lateral acceleration also grows since both tyres are close to the limit of adhesion at the same time. Consequently, the lateral acceleration which the vehicle can sustain is higher. The values of static margin and peak lateral acceleration ($a_{y_{\text{peak}}}$) of the various models are shown in Table 5.2. The variation of these quantities for the different models with respect to the original vehicle have also been indicated.
Finally, since in the various vehicle configurations the lateral acceleration to steering angle relationship changes, due to the variation of understeering gradient, the steering ratio of the different models is varied. This is done in such a way that

Table 5.2: Steady-state results.

	Step 00	Step 02	Step 04	Step 06	Step 08
$a_{y_{\text{peak}}}$ **(g)**	1,0	0,8	0,6	0,4	0,2
$\Delta a_{y_{\text{peak}}}$	-	+1,1%	+3,5%	+6,1%	+9,0%
SM **at 0,2g**	12%	9%	3%	-3%	-8%
ΔSM **at 0,2g**	-	-25%	-75%	-125%	-167%

the perceived behaviour of the vehicle to a human driver would be the same for all vehicle models in steady-state. Specifically, the steering ratio is adapted to obtain the same lateral acceleration (0,5g) for a given steering wheel angle as in Figure 5.4. The modified steering ratio for each configuration k_{st}^{i} is calculated as the ratio between the steering wheel angle of the original vehicle configuration and

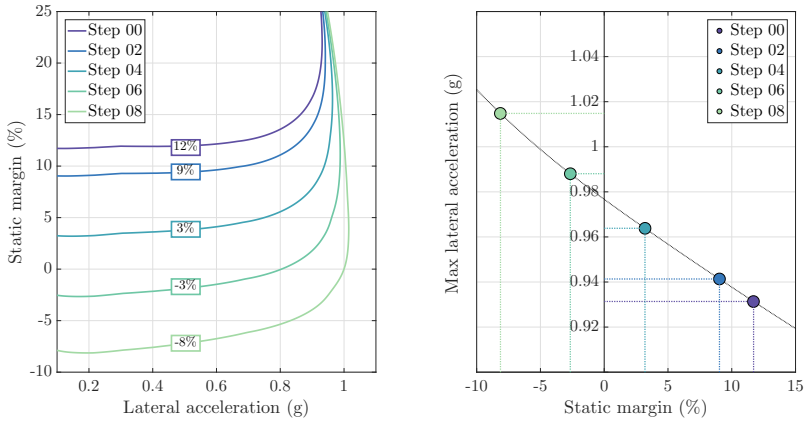

Figure 5.3: Steady-state open-loop response – static margin and peak lateral acceleration.

the steering rack displacement d_r at 0,5g of each i^{th} vehicle configuration as follows,

$$k_{st}^{i} = \frac{\delta_w^{STEP00}|_{a_y=0.5g}}{d_r^i|_{a_y=0.5g}}. \tag{5.16}$$

5.2.2 Step response

The transient behaviour is analysed by means of a step steer manoeuvre. This is done with the corrected steering ratio applied. With this manoeuvre the time delay of both lateral acceleration response and yaw rate response are evaluated, results of which can be seen in Figure 5.5. The metric to evaluate this is chosen as the time difference Δt between the peak value and 90% of that peak. As the vehicle's SM decreases, the time delay increases. The reason is that the stable vehicle's larger front slip angles (lower cornering stiffness) are obtained mainly with larger toe angles, particularly in the first phase of the manoeuvre. Thus, since toe angle

Table 5.3: Step response results.

	Step 00	Step 02	Step 04	Step 06	Step 08
Δt_{a_y} **(s)**	0,51	0,52	0,56	0,61	0,70
Δs_{a_y} **(s)**	10,2	10,4	11,2	12,2	14,0
Δ**lag**$_{a_y}$	-	+1,9%	+9,8%	+19,6%	+37,3%
Δt_r **(s)**	0,47	0,48	0,49	0,51	0,57
Δt_r **(s)**	9,4	9,6	9,8	10,2	11,4
Δ**lag**$_r$	-	+2,1%	+4,2%	+8,5%	+21,3%

variation has no delay (neglecting steering dynamics), the rate of front slip angles in the stable vehicle is greater than the one in the unstable vehicle. In the unstable vehicle, a larger contribution to slip angle generation is given by lateral velocity and yaw rate. Since these are the time integral of yaw moment and lateral force which are caused by slip angles themselves, these effects are slower than toe angle variation. The numeric values of this analysis are shown in Table 5.3. Since the

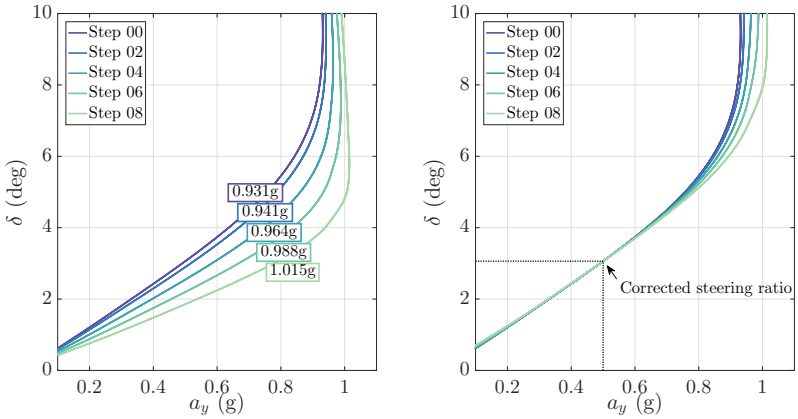

Figure 5.4: Steady-state open-loop response – understeering gradient at 20m/s. On the left the original steering ratio. On the right the corrected steering ratio.

manoeuvre is performed with a forward velocity of 20m/s, the delays in terms of space Δs have been calculated and so have lag variations Δlag with respect to the original configuration.

5.3 Test manoeuvre

A closed-loop avoidance manoeuvre is defined which requires both large amounts of grip and stability in order to evaluate the role of a human driver in comparison to a robotic controller. The selected manoeuvre is visible in Figure 5.6 and is a variation of the classic ISO double lane change manoeuvre (ISO 3888 [121]). However, some of the parameters are varied in order to make it more similar to a real world avoidance manoeuvre. The manoeuvre is first tested by human drivers on the DiM installed at the Advanced Vehicle Dynamics centre at Danisi Engineering. With the human tests it is possible to define one configuration of the manoeuvre which exploits the limits of the reference human drivers on a highly unstable vehicle (configuration Step08). This same scenario is then used for the autonomous vehicle. The lane offsets and widths chosen are similar to a typical lane found on country roads (approximately 3,75m) and are kept constant throughout the study. The manoeuvre execution is to arrive at 100km/h (approximately 28m/s) through the first gate and lift off the throttle after passing the last cone of the entry gate. From then, the only input for the driver is the steering wheel until the end of the manoeuvre. The velocity is maintained by adapting the gear ratios so that the driver could keep full throttle in second gear and maintain the rpm limiter and speed. The lift off adds additional instability due to the vertical load variation. To replicate an unplanned avoidance action, the steering wheel is kept to zero until passing the last cone, thus eliminating the anticipation of the steering manoeuvre. The same constraints are maintained for the robotic controller which is also only allowed to steer after the last cone. To guarantee no anticipation in the steering action, a graphical "blocker" of approximately 2m of height which covers the entire width of the lane is added. Additionally, the direction of the obstacle (i.e. right or left) is completely random. The blocker makes a big difference in the manoeuvre

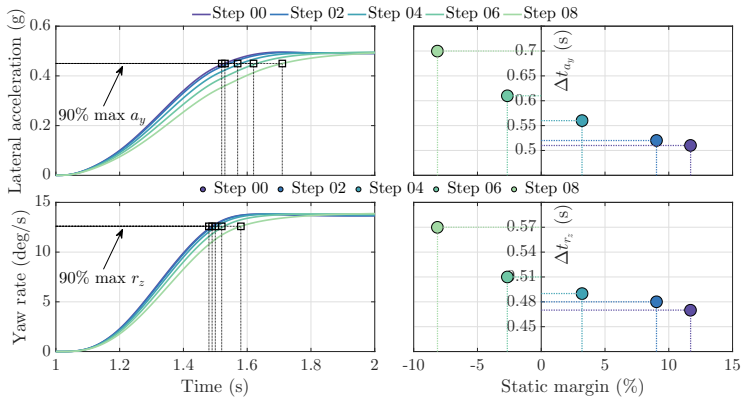

Figure 5.5: Step response – time delay of lateral acceleration and yaw rate variation with static margin.

success due to the impossibility to anticipate. In fact, with the introduction of the blocker a large delay is introduced (approximately 0,3 – 0,5s and 8,4 – 14,0m at 28m/s). The entry width a is set to 2,6m while the avoidance length e is set to 43m. The latter is kept constant since the performance of the vehicle through the first lane change is largely a function of the reaction time of the driver, while the focus of this study is on the controllability rather than reaction times. The obstacle offset b is set to 3,75m which is a typical lane width for a variety of roads including freeways, highways and main country roads. The idea of the setup is to model an avoidance manoeuvre whereby the entire right lane is blocked by an obstacle. The avoidance lane width c is set to the same distance. This is different to the ISO standard which uses the vehicle width in the lane width calculations. However, this allows for a more road realistic situation. The second lane change represents a secondary avoidance manoeuvre (e.g. from oncoming traffic). The recovery length f is found to have a large impact on the results of the manoeuvre since it requires an inversion of yaw rate vector direction which induces critical sideslip velocities, particularly in unstable vehicles. This distance is the only variable distance within the simulation and is swept from 35m to 20m with variations of 3m between a test and another. Finally, the recovery lane width d is also set to 3,75m and left constant. A summary of the geometrical features of the manoeuvre is shown in Table 5.4.

Table 5.4: Manoeuvre geometry.

	a	b	c	d	e	f
Distance (m)	2,6	3,75	3,75	3,75	43	35-20
Variable	no	no	no	no	no	no

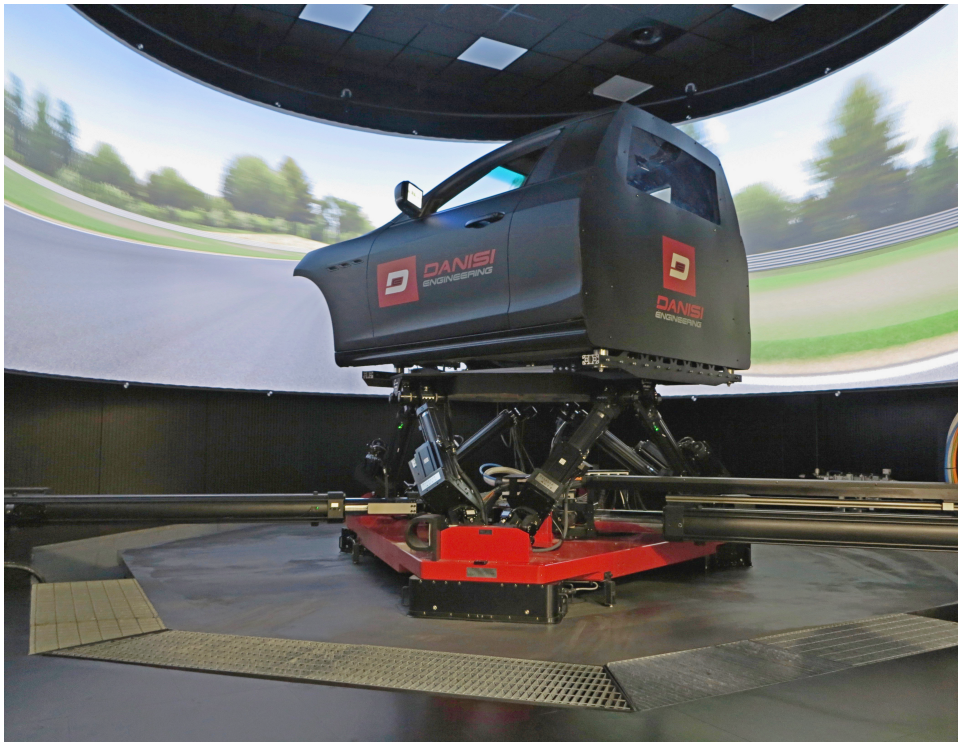

Figure 5.6: DiM driver simulator.

5.4 Human driver results

Three different human drivers are used as a sample, all with significant simulator experience. The results shown in Figure 5.8 represent the average result of the three drivers. The tests for the human drivers are performed in the DiM always using the same motion cueing. Manoeuvre success is defined as a passage of the entire obstacle without touching any cones. Every combination of vehicle static margin and recovery length is repeated ten times for each driver. This adds up to a total of three hundred data points. A correlation coefficient of approximately -0,7 is found between the recovery length and static margin (expressed as percentage points). Hence, every 1% decrease of static margin increases the required recovery distance by 0,7m in the double lane change manoeuvre for this particular test configuration with these reference drivers.
The vehicle with the highest peak grip (Step08) has very peculiar handling although it has more lateral grip capacity, its reaction to steering inputs is extremely slow (as expected from the results of the previous sections) to the point that any steering manoeuvre requires anticipation. The vehicle results in being practically unmanageable in transient changes of direction for a human driver due to the lack of rear cornering stiffness. Human drivers have many limits in terms of controlling a vehicle, mainly due to a low reaction time and low actuation power. Additionally, humans are highly affected by instinct in critical situations. Finally, since human drivers do not have an explicit formulation of the vehicle dynamics but only one based upon experience, the vehicle dynamics cannot be exploited.

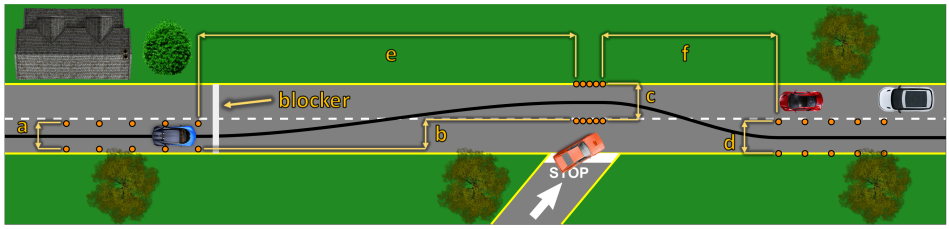

Figure 5.7: Test manoeuvre.

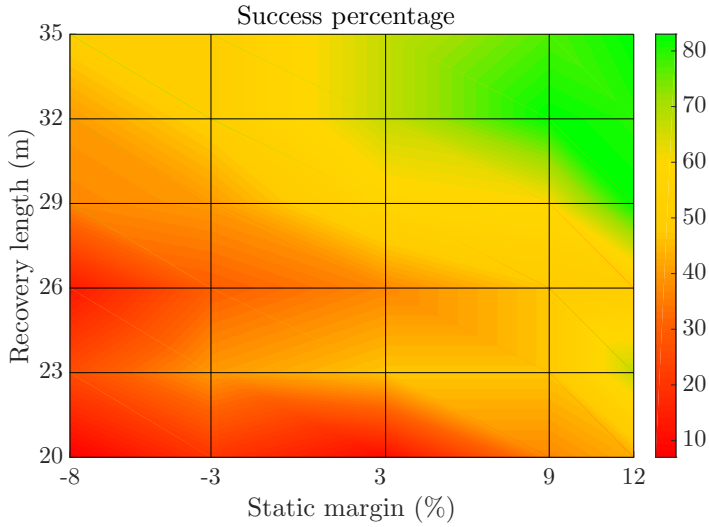

Figure 5.8: Heat map of the human results – all configurations. Green represents success and red represents fail.

5.5 Autonomous driver results

An autonomous driver is developed to see the benefits of the "more unstable" vehicle. Unlike the human driver, a model predictive controller with a long enough prediction horizon and an accurate vehicle model takes into account the lag effects. For this reason a NMPC is developed based on a validated seven degrees of freedom model. The controller sends the inputs and gets the feedback from the same Vi-Grade vehicle model used in the DiM. Another advantage of the NMPC is that it is possible to use constraints, which allows the use of road boundaries as constraints and also limits the inputs. Two types of controllers are developed: a pure lateral one, where only the steering is actuated, and a combined lateral-longitudinal controller, where also the throttle and brake pedals can be actuated. The former controller is the one used to compare the benefits of different vehicle configurations when driven autonomously. The latter controller is used only to evaluate the benefits of combined slip and inputs but does not represent a comparison with the human driver. In the MPC setup this is a straight forward extension of the lateral controller.

5.5.1 Controller overview

The controller developed is a NMPC based on previously developed frameworks [87, 114] and the hierarchical controller described in the previous chapter. The vehicle model considered is the one described in Section 4.1.1 where once again the slip ratios and slip angles are calculated with a classic kinematic formulation and the tyre forces are evaluated with a full MF formulation, neglecting aligning and overturning moments.
This time, the reference path of the NMPC is the centreline between the cones interpolated with a piecewise cubic Hermite interpolating polynomial.[122] However, only after the vehicle passes the blocker are the obstacles (constraints) available and the path generated. This way the robotic controller gets the obstacle information at the same position as the human. Once the vehicle passes the blocker and the reference path is available, the error input signal to the controller is different to zero and the online optimisation of the NMPC begins. Until this point an open-loop full throttle and no steering policy is used just like for the human.
Considering again the state-space transformation as in Section 4.1.2 and actuator dynamics as in Section 4.1.1, the vector of states $\tilde{\mathbf{z}}(s)$ given by (4.33) neglecting longitudinal controls (B_p and T_e) becomes the following,

$$\tilde{\mathbf{z}}(s) = [n, \alpha, v_x, v_y, \dot{\psi}, \omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}, \Delta F_{z_x}, \Delta F_{z_y}, \delta_w]\,. \tag{5.17}$$

Once again the system of equation is valid under the assumption that the vehicle never stops (only necessary when transforming all states), i.e. $||v_x + v_y|| > 0$, and the vehicle always stays at a lateral distance n that is smaller than the distance of the local centre of curvature of the road, i.e. $n < 1/k(s)$.
This time the input vector $\hat{\mathbf{y}}(t)$ is given by the following,

$$\hat{\mathbf{y}}(t) = [\dot{\delta}_w]\,. \tag{5.18}$$

The goal of the controller is to stay within the bounds of the road (given by the cones) at all time. The discrete time MPC problem is very similar to the low-level control in (4.62) but without the time optimality cost and the terminal velocity constraint and cost. Thus it can be formulated as follows,

$$\begin{aligned}
\min_{\tilde{\mathbf{z}},\hat{\mathbf{y}}} \quad & \sum_{k=0}^{N_g} j_{\text{dyn},k} + j_{y,k} + j_{\hat{y},k} + j_{\text{slack},k} \\
\text{s.t.} \quad & \tilde{z}_0 = \tilde{\mathbf{z}}(0) \\
& \tilde{z}_{k+1} = \tilde{f}_d(\tilde{z}_k, \hat{y}_k) \\
& n_k \geq \underline{n}_k - \rho_k\,, \quad n_k \leq \overline{n}_k + \rho_k\,, \quad \rho \geq 0 \\
& \tilde{z}_k \in \tilde{\mathcal{Z}} \\
& \hat{y}_k \in \hat{\mathcal{Y}}\,.
\end{aligned} \tag{5.19}$$

Where $\tilde{\mathbf{z}} = [\tilde{z}_0, ..., \tilde{z}_{N_g}]$ is the state trajectory and $\hat{\mathbf{y}} = [\hat{y}_0, ..., \hat{y}_{N_{g-1}}]$ the input trajectory, N_g is the horizon length. The cost is again built up of regularization costs related to the dynamics of the car, penalising deviations from the path, angle deviations from the path and sideslip angles, $j_{\text{dyn},k} = \eta_n n_k^2 + \eta_\alpha \alpha_k^2 + \eta_\beta [\tan^{-1}(v_y/v_x)]^2$, where η_n, η_α, and η_β are weights. The real inputs, composed of steering angle only $\hat{\mathbf{y}}(t) = [\dot{\delta}_w]$, have a penalisation on the deviations from zero $j_{y,k} = y_k^T R y_k$ as

does the input rates $j_{\hat{y},k} = \hat{y}_k^T R_\Delta \hat{y}_k$, with R and R_Δ being again positive definite weight matrices. Once again soft constraints are implemented for the track boundaries and the slack multipliers are penalised $j_{\text{slack},k} = \gamma\rho^2$. Moreover, $\tilde{\mathbf{z}}(0)$ is the measured state, $\tilde{f}_d(\tilde{z}_k, \hat{y}_k)$ the discretised vehicle dynamics. This time an explicit Runge-Kutta method is sufficient to integrate the dynamics since no braking action is required. The boarder constraints are implemented as bounds $\underline{n}$ and $\overline{n}$ on the orthogonal deviation from the path n. As in the low-level of the hierarchical control, to avoid feasibility issues the constraints are implemented as soft constraints and $\tilde{\mathcal{Z}}$ and $\hat{\mathcal{Y}}$ represent bounds on all the states and inputs, explicitly on the real inputs but also on the derivatives of the inputs.
Note that in this configuration of the controller, given different goals and the increased complexity of the manoeuvre in terms of feasibility due to the extreme instability, different tuning of the controller is required. Specifically the cost on the reference path is smaller since it is not as important while the cost on the road bounds are larger. As for the NMPC described in Section 4.2, Forces Pro NLP solver is used for the implementation of the NMPC.
For the controller, only the most critical configurations are tested, thus, the manoeuvre with recovery length equal to 20m and the vehicle model Step08. However, to be able to compare the results with the human ones, also the original vehicle, configuration Step00 is tested. One of the 7% of the success manoeuvres of the human drivers for Step08 is selected as a reference for the comparison with the robotic controller. Unlike the human driver, the results of the NMPC controller are repeatable under ideal conditions. Note that, to be able to compare the passive vehicle dynamics for autonomous driving and not the performance of the controller itself, the controller is tuned for configuration Step08. The same tuning is then used for Step00. However, it could be possible to improve the performance of the original vehicle by properly tuning the controller. Note that the vehicle models are modified in the NMPC depending on which configuration is tested since this type of controller is highly dependent on vehicle model. The results shown in this section have been obtained without considering any external disturbances or parameter uncertainty, a separate study would be necessary to investigate the controller's performance in presence of such uncertainties. However, the goal of this research is to analyse how the passive stability can be varied comparing a robotic controller with a human driver and not to design the perfect autonomous driver. Therefore, the only model uncertainty is the simplification in tyre and vehicle model.

5.5.2 Lateral controller results

The lateral controller has the steering angle δ_w as the only input command, the driving torque and braking torque are kept constant and equal to zero. The steering wheel angle velocity is constrained to a value of 1000deg/s. Note that the longitudinal equations are maintained in the controller's equality constraints since for this manoeuvre, particularly with the unstable vehicle, it is expected to drive at high sideslip angles, thus combined slip effects and induced drag become relevant. This controller is used to compare the human driver to the robotic controller. The goal of the study is to show that with autonomous vehicles, it is possible to rethink how commercial vehicles are designed from a passive vehicle dynamics perspective. It is possible to increase the instability gaining performance in terms of peak grip despite the increase of time delay. The latter is very well dealt with the NMPC

as will be shown now. Figure 5.9 shows the outcome of the NMPC described on a typical manoeuvre like the one previously described with the more unstable vehicle. Specifically the predicted trajectory for each optimisation is showed. It is clear how the predicted trajectory is smooth and always within the bounds even with such an unstable vehicle.

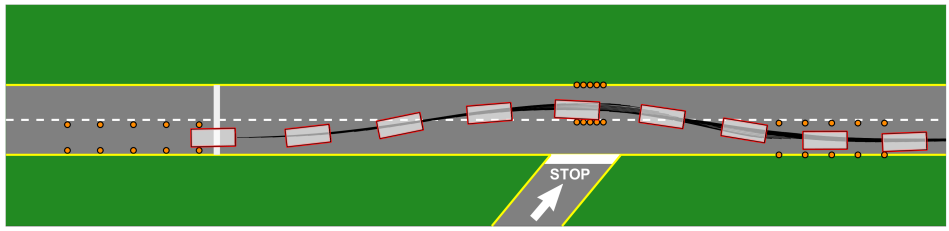

Figure 5.9: Prediction horizon.

The steering profiles and positions in the X-Y plane are shown in Figure 5.10; as expected, the reaction time of the controller is much faster than the human driver. Note that the human driver and the robotic controller both start steering after

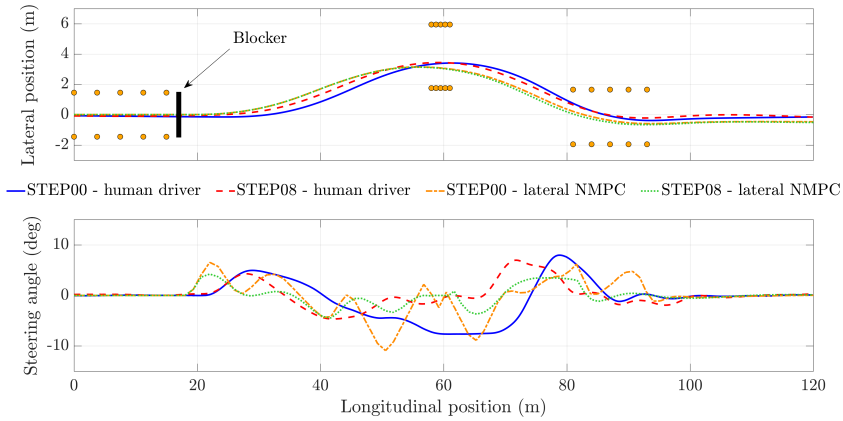

Figure 5.10: Lateral controller results – position and steering profiles.

the graphical blocker. However, the robotic controller has a faster reaction time. In fact, the blocker does not influence the delay in the response, unlike for the human. This greatly simplifies the manoeuvre for the controller as it is not required to increase the steering angle and incur into high sideslip angles, especially for the unstable vehicle. While for the human driver the steering profile for the stable vehicle is much smoother than the unstable one, for the NMPC the unstable vehicle has a much smoother profile. This is due to the fact that the controller is predictive, hence future time delays are modeled. Thus, since the unstable model has very little yaw damping due to the low rear cornering stiffness, its turning capacity is very high, this allows the controller to actually anticipate the manoeuvre and steer less. On the other hand, the stable vehicle is harder to drive for the controller since the controllable front wheels are near the saturation and have lower

grip, additionally to high yaw damping given by the rear axle. As a result, the unstable configuration maintains a greater distance from the cones compared to the stable one. Also note that the controller tuning is obtained using the unstable vehicle, thus proper tuning could improve the performance of the stable vehicle. The peaks and pits of the steering profiles of the controller are more numerous compared to the human driver. This is the big advantage of having a motor with high power controlling the steering wheel instead of a human. However, this also results in a very unnatural way of driving. Each steering manoeuvre has in fact a counter steering input to stabilise the vehicle. One of the key aspects is that the human driver necessitates high lateral acceleration due to the delays, this is visible in the plots in Figure 5.11. Contrarily, the lateral accelerations of the controller

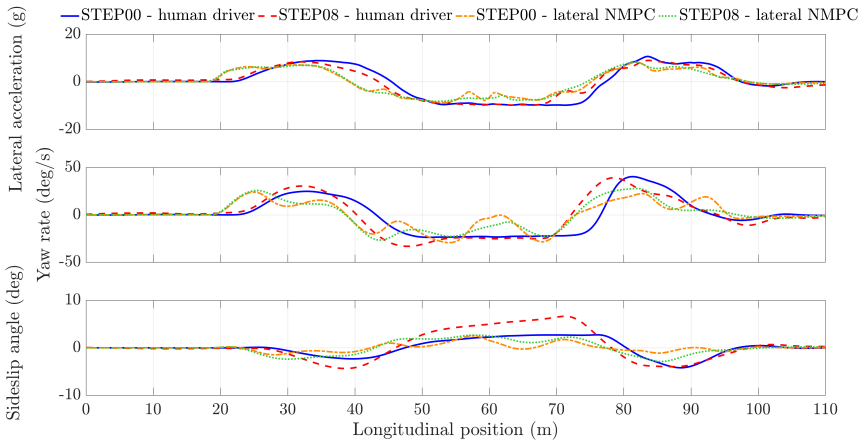

Figure 5.11: Lateral controller results – lateral acceleration, yaw rate and sideslip angle profiles.

are far from the vehicle's full potential because of the prediction horizon. Also, although yaw rate for the robotic controller oscillates a lot resulting in a jerkier manoeuvre, the maximum and average values are much lower. As expected, even with a smoother steering angle profile, the unstable configuration for the controller reaches higher yaw rates and sideslip angles in the second avoidance manoeuvre where the yaw rate vector sign inversion and high sideslip velocities are required. The human driver and robotic controller for the two configurations are compared in Figure 5.12 using different metrics and plotting them in a spider plot. In particular, it is interesting to notice the different steering requests for the various cases $|\delta|_{\text{mean}}$ and $|\delta|_{\text{max}}$, how much lateral acceleration is needed to complete the manoeuvre and how much grip is consequently still available $|a_y|_{\text{mean}}$ and $|a_y|_{\text{max}}$ and, finally, how much the vehicle is actually turning and slipping with the given steering profile $|\dot{\psi}|_{\text{mean}}$, $|\dot{\psi}|_{\text{max}}$, $|\beta|_{\text{mean}}$ and $|\beta|_{\text{max}}$. The results show that the unstable vehicle controlled by the NMPC has better values compared to both to the stable vehicle driven by the robotic controller and the human driver in both configurations.

5.5.3 Combined controller results

Since the maximum lateral acceleration of the lateral controller is far from the peak, in this section the additional control authority of a combined controller is tested to

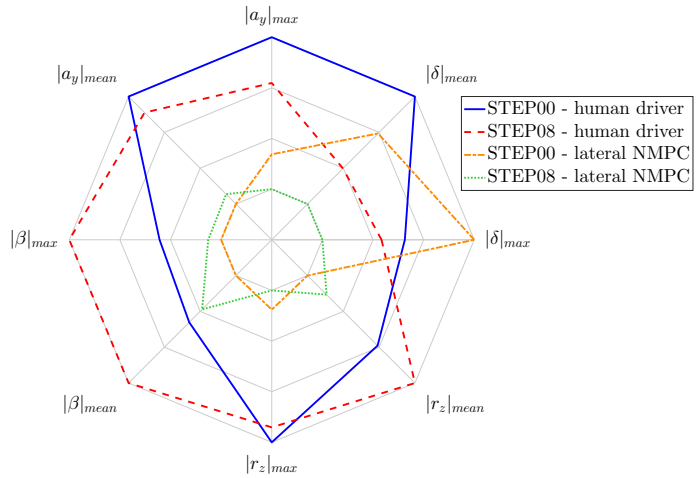

Figure 5.12: Lateral controller results – spider plot.

evaluate an improvement in performance. The execution of the manoeuvre is still the same, with the lift off at the last cone and start of the inputs only after the blocker. However, this time the controller can also brake or accelerate. Thus, the input vector is again $\hat{\mathbf{u}}(t)$ and the state vector is $\tilde{\mathbf{x}}(s)$ as in Section 4.2.3.
To assure the vehicle completes the manoeuvre in the minimum time possible (to exploit the vehicle's performance and avoiding the vehicle from stopping), a term is added to the NMPC cost function, yielding in the following,

$$\begin{aligned}
\min_{\tilde{\mathbf{x}},\hat{\mathbf{u}}} \quad & \sum_{k=0}^{N_g} j_{\mathrm{dyn},k} + j_{u,k} + j_{\hat{u},k} + j_{\mathrm{slack},k} + j_{\mathrm{vel},k} \\
\text{s.t.} \quad & \tilde{x}_0 = \tilde{\mathbf{x}}(0) \\
& \tilde{x}_{k+1} = \tilde{f}_d(\tilde{x}_k, \hat{u}_k) \\
& n_k \geq \underline{n}_k - \rho_k\,, \quad n_k \leq \overline{n}_k + \rho_k\,, \quad \rho \geq 0 \\
& \tilde{x}_k \in \tilde{\mathcal{X}} \\
& \hat{u}_k \in \hat{\mathcal{U}}\,.
\end{aligned} \tag{5.20}$$

The additional cost $j_{\mathrm{vel},k} = \zeta(v_{x,k} - v_{x,\mathrm{init}})^2$ penalises deviations in the vehicle's forward velocity $v_{x,k}$ from the speed at the beginning of the manoeuvre $v_{x,\mathrm{init}}$, with ζ being a weight. Once again, the tuning of the controller is done on configuration Step08.
During the first steering manoeuvre, the combined controller reaches higher steering angles, lateral accelerations, sideslip angles and yaw rates. This is due to the controller braking as it enters the other lane. This induces an even larger weight transfer, reducing yaw damping. However, because the front tyres are working under combined slip and the cornering stiffness is lower, the steering angle requested is higher. The steering angle profiles of the combined controllers can be seen in Figure 5.13 and are smoother than the pure lateral ones due to the combined effects (which gives induced yaw moments) and the possibility of varying speed with longitudinal controls. During the second avoidance, the sideslip angle and

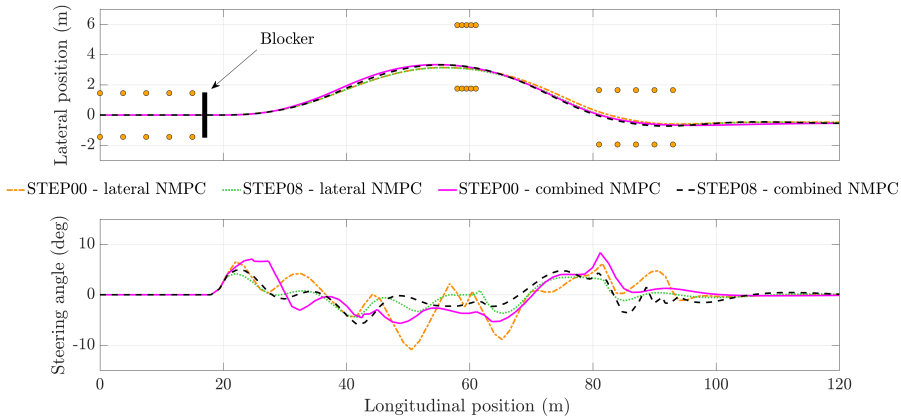

Figure 5.13: Combined controller results – position and steering profiles.

yaw rate of the combined controller on the unstable vehicle, visible in Figure 5.14, reaches much higher values than the lateral controller. This is because after the

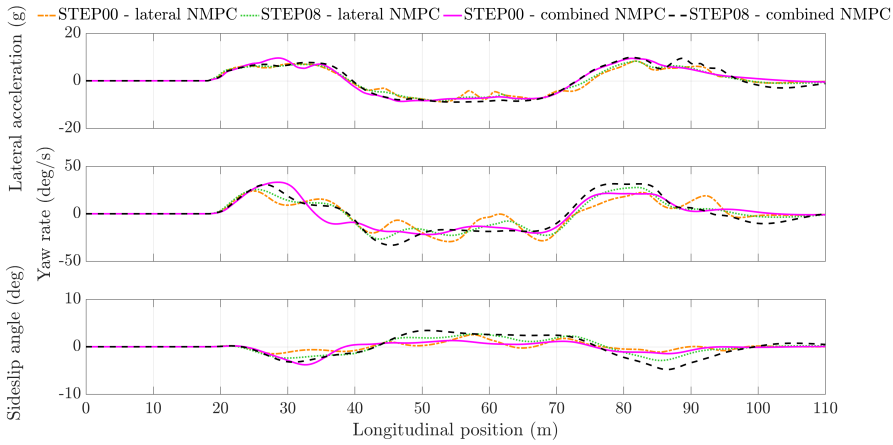

Figure 5.14: Combined controller results – lateral acceleration, yaw rate and sideslip angle profiles.

first braking manoeuvre, the controller forces the vehicle to accelerate reducing the lateral force produced by the rear tyres. The combined controller spider plot is shown in Figure 5.15 and looks very similar to the one of the human driver shown in Figure 5.12 but with a similar steering values to the lateral controller spider plot. However, even if the accelerations are very similar, the sideslip angle and yaw rate are much lower, showing that even with an unstable vehicle, the robotic controller is capable of maintaining high controllability. The combined controller spider plots are larger than the lateral controller ones, since the manoeuvre execution time for the former is significantly lower than the latter and more grip is used. Furthermore, the combined controller on the unstable vehicle has a lower execution time and higher accelerations than the stable vehicle controlled by the NMPC. The velocity profiles together with the accelerating torque and brake percentage are represented

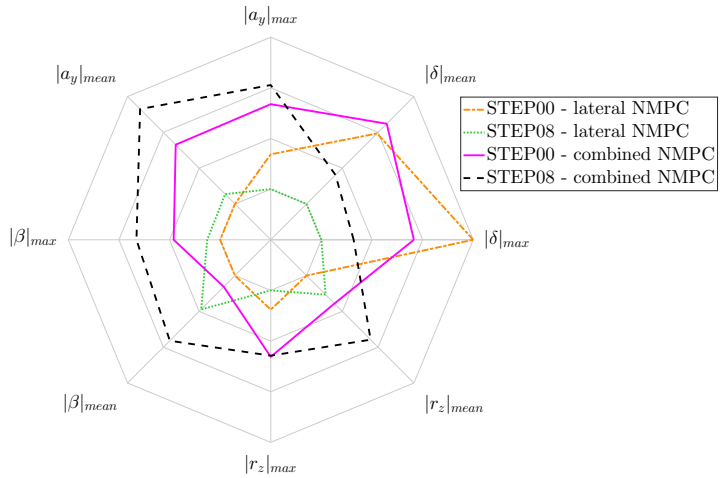

Figure 5.15: Combined controller results – spider plot comparing.

in Figure 5.16. Note that in the accelerating torque profile, only the closed-loop torque is represented. However, as shown in the longitudinal velocity profile, the throttle is kept fully open until the visual blocker is reached by the vehicle.

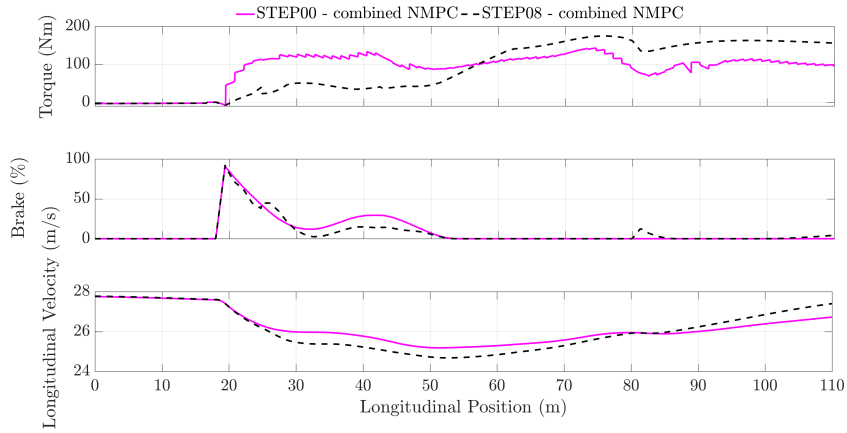

Figure 5.16: Combined controller – torque, brake percentage and longitudinal velocity profiles.

Chapter 6

Conclusion and Outlook

This research has presented a mathematical framework for autonomous driving at the limits of handling. The framework comprises of the vehicle's state estimation, a novel robotic controller and a study on the influence of autonomous driving on passive vehicle dynamics. All three topics are separate areas of development, but all are linked and necessary when studying self driving vehicles.
The observer is based on an integrated ANN and UKF method and uses only IMU measurements. Numerical data is used to train the ANN, drastically reducing the cost of development. An algorithm to define the best neural network structure was proposed and experimental results show that the algorithm is effective. In fact, the deployed ANN found with the described method predicted the sideslip angle time response of a real vehicle accurately despite underestimating its magnitude due to extrapolation. This is caused by different closed-loop behaviour of the real driver and the one used during training together with model uncertainty.
Therefore, a novel integrated ANN and UKF observer that uses a kinematic model is presented which corrects the sideslip angle estimation of the neural network. The kinematic model uses longitudinal velocity which is observed with direct integration with integral damping and integral value reset.
With the described method the UKF is independent of vehicle and tyre models. An additional strategy to improve the convergence and accuracy of the observer is presented which consists of a correction of the pseudo-sideslip angle predicted by the ANN and used as a measurement by the UKF.
This final virtual sensor shows very good results on experimental data.
Next the hierarchical control scheme was presented. This is made of two levels: first is the high-level NMPC which is based on a point-mass particle of which the state equations are space dependent and which travels on the reference path. The NMPCs constraints are given by a tyre based gg-diagram, a maximum torque deployed by the engine and a maximum deceleration which the point-mass can deliver. This controller can solve the optimisation problem in under ten milliseconds using horizons of several hundred meters. The outcome of is a velocity profile which main goal is to predict the braking point. The second is the low-level NMPC which, unlike the former, is based on a high fidelity model. Specifically a seven degrees freedom vehicle model transformed into space dependant equations with vertical dynamics, LSD model and with Pacejka MF6.1 tyre model including combined force effects. Since the model has very little mismatch with the real vehicle, the input sequence found by the controllers brings the vehicle to its limits while

solving the optimisation problem in approximately one hundred milliseconds with a horizon length of a few tenths of meters. Since actuator dynamics are included in the NMPC model, the inputs to the vehicle model are then applied every ten milliseconds. Despite the short horizon the feasibility is guaranteed by the use of a terminal velocity constraint which is extrapolated by the velocity profile calculated by the higher level controller. This interaction between the two levels is especially important in manoeuvres when hard braking is required. This also makes the controller robust and capable of completing a lap of a circuit in a time optimal manner. The proposed hierarchical controller is solved on a minimum curvature trajectory. However, given the receding horizon strategy of the terminal constraint it could be applied to any trajectory. With this scheme it is possible to correctly predict the vehicle dynamics thanks to the use of high-fidelity models while solving a time optimal autonomous driving problem in real-time thanks to the possibility of using short horizons.

Finally, this research presented a study on the influence of autonomous driving on passive vehicle dynamics stability. The dependancy of manoeuvre geometry and vehicle stability on the ability to perform a closed-loop avoidance manoeuvre was highlighted with a sweep of tests completed by both human drivers and NMPC controllers. The predictive control outperformed the human driver thanks to its knowledge on the vehicle dynamics and its evolution by means of a receding horizon strategy. This can guarantee a prediction, thus anticipation, of time delays and unstable behaviour. The outcome of the simulations and simulator runs were that the robotic controller completed all of the tests successfully unlike the (expert) human drivers. However, when driving an unstable vehicle, even the NMPC required controlling the inputs in a very unnatural way in order to keep the vehicle response sufficiently damped. When given the possibility of controlling both the longitudinal and the lateral dynamics to the robotic controller the results improved even further. The results obtained could open up new possibilities in the design of passive vehicle dynamics. It could be particularly interesting to vary the passive stability of a vehicle by means of active controls (e.g. active anti-rollbars) and adapting it the specific driving situation.

All the key areas of this research were based on an in depth analysis of the literature. The presented algorithms and methods represent an addition to the state of the art in terms vehicle performance and real-time control systems application.

Appendix A

ANN Transfer Function

In this appendix the hyperbolic tangent transfer function ANN described in Chapter 3 will be compared to a radial basis transfer function ANN. Particularly, the results of the ANN with structure found by means of Algorithm 1 will be shown. To compare and evaluate the results, the networks with the lowest overall RMSE are discussed. The results of the networks are analysed on the test set showed in Table 3.4. The specific manoeuvres are listed in Table A.1. The neural networks are analysed for

Table A.1: Test set manoeuvre description.

Manoeuver type	Manoeuver	Parameter 1	Value 1	Parameter 2	Value 2
Sine steer	Sine LF 01	Steer. freq.	$1,0$Hz	Steer. ampl.	$10\,$deg
Sine steer	Sine LF 02	Steer. freq.	$1,0$Hz	Steer. ampl.	$25\,$deg
Sine steer	Sine LF 03	Steer. freq.	$1,0$Hz	Steer. ampl.	$40\,$deg
Sine steer	Sine LF 04	Steer. freq.	$1,0$Hz	Steer. ampl.	$55\,$deg
Sine steer	Sine HF 01	Steer. freq.	$3,0$Hz	Steer. ampl.	$10\,$deg
Sine steer	Sine HF 02	Steer. freq.	$3,0$Hz	Steer. ampl.	$25\,$deg
Sine steer	Sine HF 03	Steer. freq.	$3,0$Hz	Steer. ampl.	$40\,$deg
Sine steer	Sine HF 04	Steer. freq.	$3,0$Hz	Steer. ampl.	$55\,$deg
Step steer	Step 01	Step dur.	$0,2$s	Steer. angle	$10\,$deg
Step steer	Step 02	Step dur.	$0,2$s	Steer. angle	$25\,$deg
Step steer	Step 03	Step dur.	$0,2$s	Steer. angle	$40\,$deg
Step steer	Step 04	Step dur.	$0,2$s	Steer. angle	$55\,$deg
Circuit	Circuit 01	Max lat. acc.	$0,1$g	Max long. acc.	$0,1$g
Circuit	Circuit 02	Max lat. acc.	$0,4$g	Max long. acc.	$0,4$g
Circuit	Circuit 03	Max lat. acc.	$0,7$g	Max long. acc.	$0,7$g
Circuit	Circuit 04	Max lat. acc.	$1,0$g	Max long. acc.	$1,0$g

each single test manoeuvre, hence, for steady-state behaviour, transient behaviour and frequency response of the vehicle. All the results shown in the rest of this appendix consider normalised values of sideslip angle respect to the maximum value in all manoeuvres. The values of RMSE are absolute.

Sine steer

The first manoeuvre which is analysed is the frequency response. As explained in Chapter 3, for different frequencies, a different response can be expected, both in

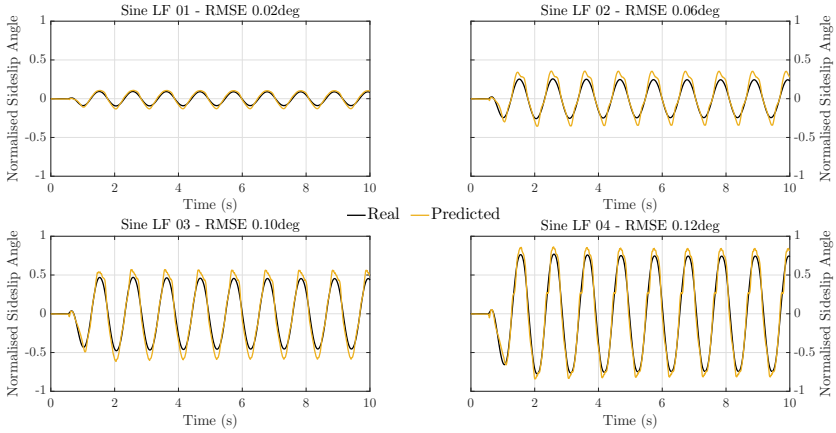

Figure A.1: Low frequency sine steer manoeuvre estimation – hyperbolic tangent.

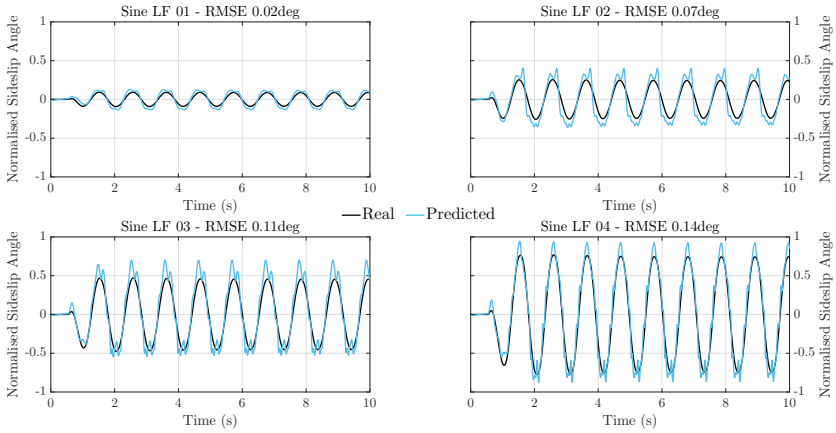

Figure A.2: Low frequency sine steer manoeuvre estimation – radial basis.

terms of amplitude and phase. For these reasons, different outputs will correspond to certain inputs, especially considering the phase lag variation between the inputs and the sideslip angle. Therefore, both a low frequency and high frequency sine steer manoeuvre are analysed. Also, a different steering angle is applied for the two frequencies. This allows to evaluate dynamics excited with different steering wheel velocity profiles and, mostly, exploit the tyre saturation region. The low frequency estimation of the hyperbolic tangent transfer function network and radial basis transfer function network can be seen respectively in Figure A.1 and Figure A.2. In both cases the estimated value follows the trend very well, for both phase and amplitude. The resulting loss functions are in fact very low. The radial basis transfer function network shows also a higher frequency mode other than the excited frequency, resulting in a greater oscillation, especially at the peak value. In this case, for higher steering angles, the performance decreases slightly for both networks. The same analysis can be done for high frequency sine steer manoeuvres, the results for the two networks are shown in Figure A.3 and Figure A.4.

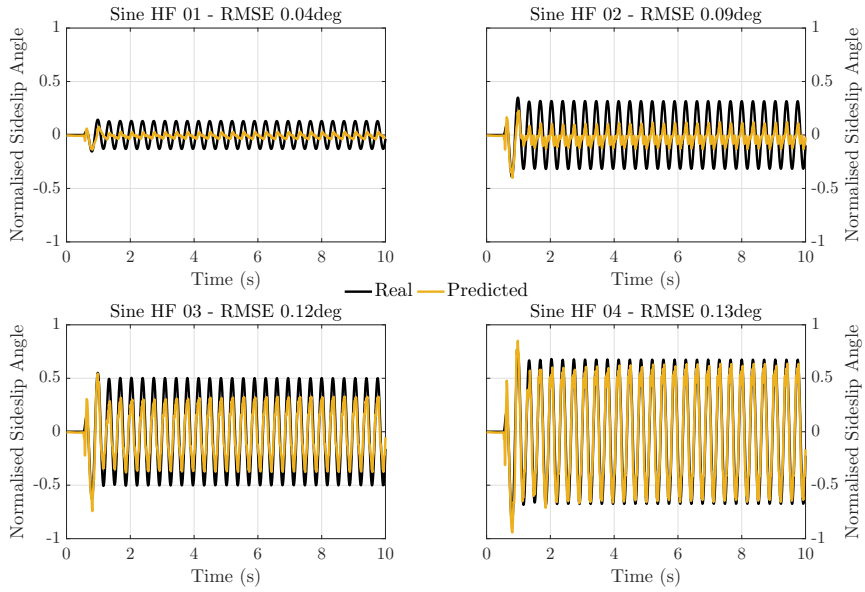

Figure A.3: High frequency sine steer manoeuvre estimation – hyperbolic tangent.

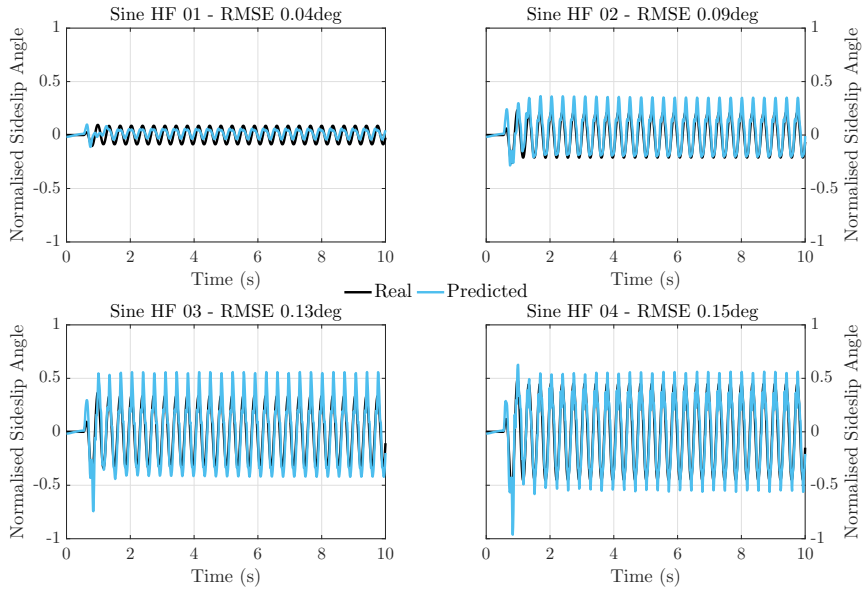

Figure A.4: High frequency sine steer manoeuvre estimation – radial basis.

In this case, the hyperbolic tangent transfer function network still shows better loss function values, mainly due to the peak values which are overestimated by the radial basis transfer function network. For low steering angles, however, the former network shows poor results. This is because the amplitude of the input signals

are very low, even more than the noise given by the sensor. Therefore, although the relative error might be large, the absolute value is still very low and so is the RMSE. For these low values, no kind of sideslip controller is required. Since the amplitude of the sideslip angle is so low, this error is more than acceptable as it is much lower than the loss function goal.

Step steer

A step steer manoeuvre can characterise both transient and steady-state behaviour of a vehicle.

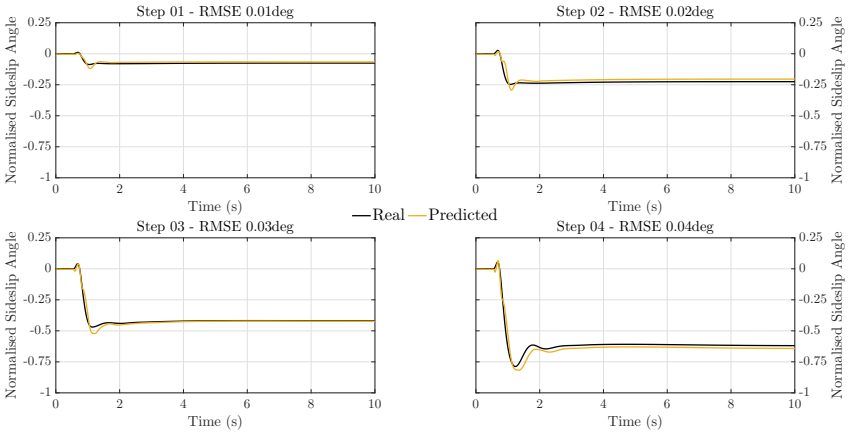

Figure A.5: Step steer manoeuvre estimation – hyperbolic tangent.

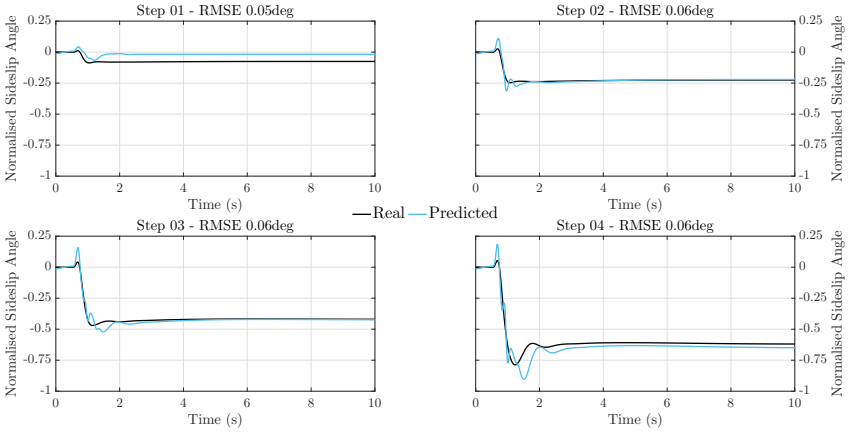

Figure A.6: Step steer manoeuvre estimation – radial basis.

It is interesting to see how the network approximates the undershoot, overshoot and steady-state value of sideslip angle at different levels of steering angle. The higher the steering angle, the larger the overshoot and undershoot (due to a faster steering wheel velocity) and the larger the non-linearity given by the tyre forces.

The networks estimation of the step steer can be seen in Figure A.5 and Figure A.6 respectively. Once again, the estimated value of the hyperbolic tangent transfer function is closer to the real value than the estimated value of the radial basis transfer function. In this case, even for very low steering angles, the estimation is very good. The values of RMSE are very low for all step durations. Also, as in the sine steer manoeuvres, the radial basis estimator has a higher frequency mode which sums up to the excited mode, particularly for higher amplitudes of sideslip angle. The hyperbolic tangent network, approximates very well the undershoot, overshoot and steady-state values. The trend of the estimated overshoot looks like the one of high steering angles even when the system is excited for low steering angles, meaning that the network slightly overfits the data in this condition. Viceversa, the undershoot is estimated very well for all steering angle amplitudes, except for a very small sign inversion in the beginning of the manoeuvre, which in absolute value is lower than $0,01$ degrees. For medium steering angle amplitudes, even the hyperbolic tangent network shows a higher frequency mode which disappears for lower and higher excitations. For these excitation amplitudes, the transient behaviour is very good even for a very a very aggressive step input in terms of time as the one analysed.

Circuit

Finally, the estimation is analysed on a circuit. Since the tyres are generating combined forces of high entity, the errors are expected to be the highest since the non-linearities and coupling between states will be enhanced. Different levels of maximum accelerations around the circuit are tested.

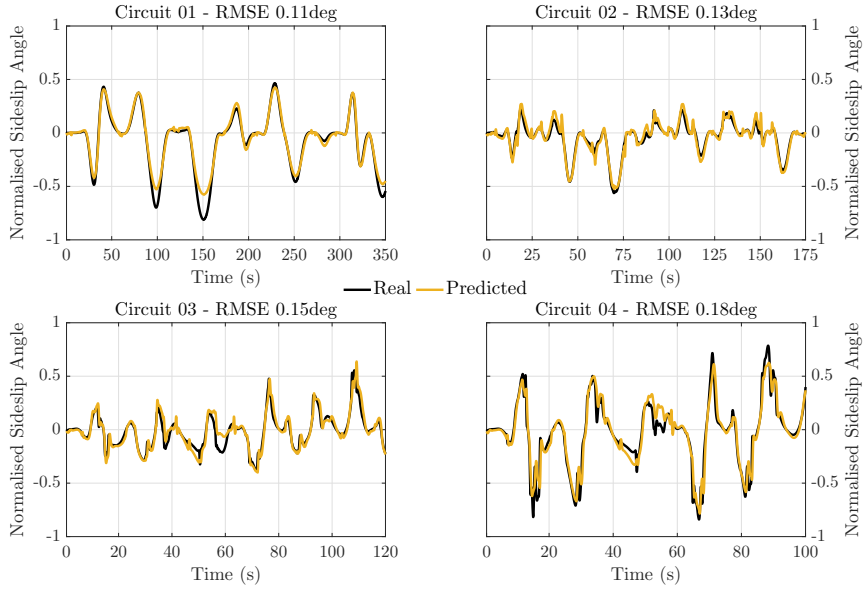

Figure A.7: Circuit manoeuvre estimation – hyperbolic tangent.

The results of the two networks can be seen in Figure A.7 and Figure A.8. In this case, the hyperbolic tangent network estimation is much better respect to the

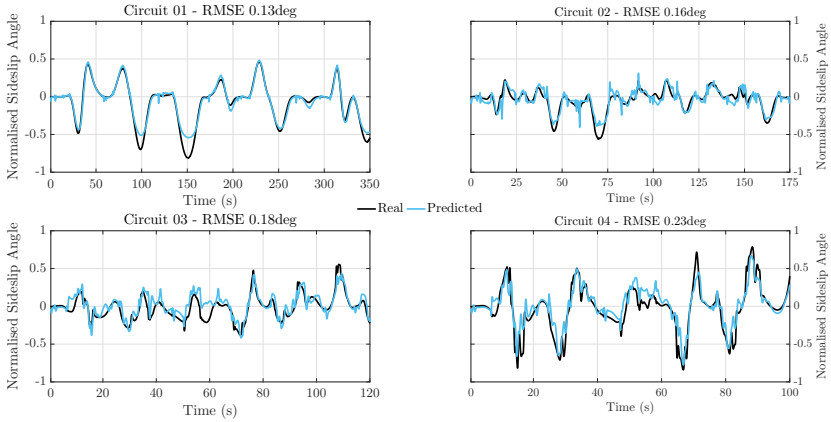

Figure A.8: Circuit manoeuvre estimation – radial basis.

radial basis one. At maximum acceleration level, the hyperbolic tangent RMSE is approximately 21% lower than the radial basis one. In the radial basis network case, the highest acceleration training set circuit estimation is what makes the radial basis network fail the break condition in Algorithm 1 at step four. Because of the presence of higher frequency content in the radial basis network, the results are far from the real value in many areas of the circuit. On the other hand, the hyperbolic tangent network follows very well the trend of the real value, for all acceleration conditions. In very few conditions the network does not approximate sideslip angle properly, however, because of the continuous transient manoeuvres during a circuit lap and considering all the information on frequency behaviour and step response previously analysed for the transient behaviour, the hyperbolic tangent neural network shows much better results in combined acceleration situations.

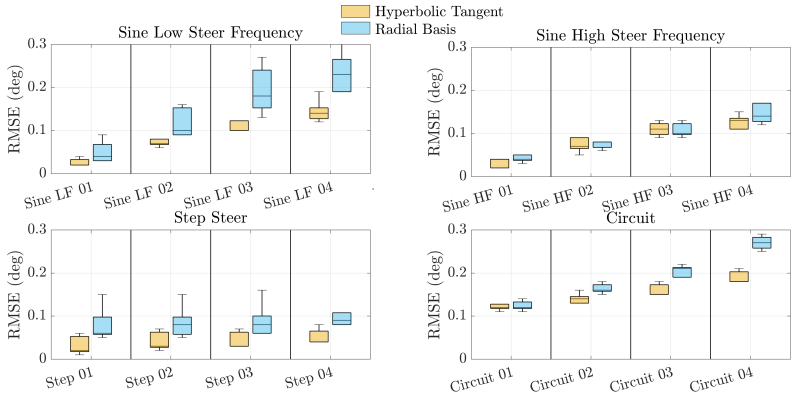

Figure A.9: Comparison between different training methods.

Comparison

A final comparison between the hyperbolic tangent and radial basis networks can be done. The results of the two networks are compared by evaluating the loss function on the manoeuvres of Table A.1. A boxplot of these results is shown in Figure A.9. The hyperbolic tangent network, besides having a better loss function average, also has a smaller deviation, except for low amplitude steering angles at high frequency steering manoeuvres. Concerning these manoeuvres, it is interesting to notice that although the time plot of Figure A.3 and Figure A.4 show a better visual result for the radial basis network, the overall boxplot is better for the hyperbolic tangent network. This is because the results shown in the aforementioned figures are the ones found with Algorithm 1 which have the lowest overall RMSE considering the same ANN structure but with five different initial conditions. However, the boxplots shows the error of all five initial conditions. In this case the radial basis network shown in the time plots, in addition to being the one with best loss function value, slightly overfits the results of that specific manoeuvre unlike the hyperbolic tangent network with best performance. Overall, the hyperbolic tangential transfer function is more appropriate for sideslip angle estimation.

Appendix B

Piecewise Cubic Hermite Spline

Let $[a, b]$ be an interval divided in N subintervals with $N + 1$ nodes and $N - 1$ internal nodes $(N + 1 - 2)$. A cubic polynomial can be found for each of these subintervals.

Piecewise Cubic Hermite polynomials

Considering $p_{3,N}(x)$ as the combination of these polynomials, a set of constraints can be defined such that $p_{3,N}(x)$ and its first derivatives $p'_{3,N}(x)$ are continuous at the interior nodes. With this set of constraints, for each of the N intervals there are four unknowns. However, there are $2(N-1)$ equation thanks to the $p_{3,N}(x)$ and $p'_{3,N}(x)$ continuity condition yielding in $4N - 2(N-1) = 2(N+1)$ degrees of freedom which means specifying the function value f and f' its derivative at all $N+1$ nodes. $4N - 2(N-1) = 2(N+1)$ prescribe $f(x_i)$ and $f'(x_i)$ at $i = 0, ..., N$, where f can be represented in terms of Hermite basis functions $h_i^{(0)}(x)$ and $h_i^{(1)}(x)$ as follows,

$$f(x) \approx h(x) = \sum_{i=0}^{N} f(x_i) h_i^{(0)}(x) + \sum_{i=0}^{N} f(x_i) h_i^{(1)}(x). \tag{B.1}$$

Where the superscript indicates the derivative order and the subscript the subinterval,

$$\begin{cases} h_i^0(x_j) = \delta_{ij} & \text{and} \quad \frac{d}{dx} h_i^{(1)}(x_j) = \delta_{ij}, \\ \frac{d}{dx} h_i^0(x_j) = 0 & \text{and} \quad h_i^{(0)}(x_j) = 0. \end{cases} \tag{B.2}$$

Considering the interval $[-1, 1]$, Lagrange basis functions can be constructed,

$$N_a^1(\zeta) = \frac{1}{2}(1 + \zeta_a \zeta) : N_a^1(\zeta_b) = \delta_{ab}, \quad \zeta_{a,b} = \pm 1. \tag{B.3}$$

Where the following can be defined,

$$h_1^{(0)}(\zeta) = (\alpha_1 \zeta + \beta_1)[N_1^1(\zeta)]^2; \quad h_2^{(0)}(\zeta) = (\alpha_2 \zeta + \beta_2)[N_2^1(\zeta)]^2, \tag{B.4}$$

$$h_1^{(1)}(\zeta) = (\gamma_1 \zeta + \delta_1)[N_1^1(\zeta)]^2; \quad h_2^{(1)}(\zeta) = (\gamma_2 \zeta + \delta_2)[N_2^1(\zeta)]^2. \tag{B.5}$$

The coefficients α_i, β_i and δ_i can be found using (B.2) as follows,

$$1 = h_1^{(0)}(-1) = \beta_1 - \alpha_1, \tag{B.6}$$

$$0 = h_1^{0'}(-1) = \alpha_1 1 + (\beta_1 - \alpha_1)\frac{2}{2}(1 - (-1))\frac{(-1)}{2} = 2\alpha_1 - \beta_1, \tag{B.7}$$

$$0 = h_1^{(1)}(-1) = \delta_1 - \gamma_1, \tag{B.8}$$

$$1 = h_1^{(1)}(-1) = 2\gamma_1 - \delta_1. \tag{B.9}$$

Hence $\alpha_1 = \delta_1 = \gamma_1 = 1$ and $\beta_1 = 2$, thus,

$$h_1^{(0)}(\zeta) = \frac{1}{4}(2 + \zeta)(1 - \zeta)^2, \tag{B.10}$$

$$h_2^{(0)}(\zeta) = \frac{1}{4}(2 - \zeta)(1 - \zeta)^2, \tag{B.11}$$

$$h_1^{(1)}(\zeta) = \frac{1}{4}(1 + \zeta)(1 - \zeta)^2, \tag{B.12}$$

$$h_2^{(1)}(\zeta) = \frac{1}{4}(\zeta - 1)(1 + \zeta)^2. \tag{B.13}$$

This concept can be extended considering a generic interval $[x_i, x_{i+1}]$ by using the following linear transformation,

$$\begin{aligned} x(\zeta) &= x_i N_1^{(1)}(\zeta) + x_{i+1} N_2^{(1)}(\zeta) \\ &= x_i \frac{1}{2}(1 - \zeta) + x_{i+1}\frac{1}{2}(1 + \zeta) \\ &= \Big(\frac{x_i + x_{i+1}}{2}\Big) + \Big(\frac{x_{i+1} - x_i}{2}\Big)\zeta. \end{aligned} \tag{B.14}$$

This transformation can be inverted yielding in the following,

$$\zeta(x) = \frac{2x - (x_i + x_{i+1})}{(x_{i+1} - x_i)} = \frac{2x - (x_i + x_{i+1})}{\Delta x_i}. \tag{B.15}$$

Thus some further simplification can be done,

$$\begin{array}{lll} 1 + \zeta = \frac{x_{i+1} - x_i + 2x - x_i - x_{i+1}}{x_{i+1} - x_i} = \frac{2(x - x_i)}{x_{i+1} - x_i} & \Longrightarrow & 2 + \zeta = \frac{\Delta x_i + 2(x - x_i)}{\Delta x_i}, \\ 1 - \zeta = \frac{2(x_{i+1} - x)}{x_{i+1} - x_i} & \Longrightarrow & 2 - \zeta = \frac{\Delta x_i + 2(x_{i+1} - x)}{\Delta x_i}. \end{array} \tag{B.16}$$

Thus the basis functions can be found,

$$h_i^{(0)}(x) = \frac{[\Delta x_i + 2(x - x_i)](x_{i+1} - x)^2}{\Delta x_i^3}, \tag{B.17}$$

$$h_{i+1}^{(0)}(x) = \frac{[\Delta x_i + 2(x_{i+1} - x)](x - x_i)^2}{\Delta x_i^3}, \tag{B.18}$$

$$h_i^{(1)}(x) = \frac{(x - x_i)(x_{i+1} - x)^2}{\Delta x_i^2}, \tag{B.19}$$

$$h_{i+1}^{(1)}(x) = \frac{(x_{i+1} - x)(x - x_i)^2}{\Delta x_i^2}. \tag{B.20}$$

$$\tag{B.21}$$

Piecewise Cubic Spline interpolation

Considering now cubic splines an additional condition is added and $p_{3,N}(x)$, $p'_{3,N}(x)$ and $p''_{3,N}(x)$ have to be continuous at the interior points. This yields in $4N - 3(N-1) = N + 3 = N + 1 + 2$. Which means specifying f at all the $N+1$ internal nodes and adding two conditions which are, for so-called natural splines, $p''_2(x_0) = 0 = p''_3(x_N)$.
The Hermite cubic basis expansion can be used as a starting point. However, in addition to the Piecewise Cubic Hermite polynomials described in the previous section, other continuity conditions are required at the $N-1$ internal nodes. Also, since $f'(x_i)$ are not known in the case of splines, other conditions are imposed. Let us consider the interval $[x_k, x_{k+1}]$, the following expression can be written,

$$\begin{aligned} s(x) =& f_k \frac{[\Delta x_k + 2(x - x_k)](x_{k+1} - x)^2]}{\Delta x_k^2} + f_{k+1} \frac{[\Delta x_k + 2(x_{k+1} - x)](x - x_k)^2]}{\Delta x_k^3} \\ &+ s'_k \frac{(x - x_k)(x_{k+1} - x)^2}{\Delta x_k^2} + s'_{k+1} \frac{(x - x_{k+1})(x - x_k)^2}{\Delta x_k^2}. \end{aligned} \tag{B.22}$$

The unknown quantities s'_k and s'_{k+1} replace f'_k and f'_{k+1} of the Hermite expansion and can be found with a system of equations defined by imposing the continuity at the $N-1$ internal nodes of $s''(x)$. Note that the condition of continuity on the first order derivatives is guaranteed by the choice of basis functions. Imposing the second order derivative continuity at x_k on the two intervals $[x_{k-1}, x_k]$ and $[x_k, x_{k+1}]$ the following can be obtained,

$$\Delta x_k s'_{k-1} + 2(\Delta x_k + \Delta x_{k-1}) s'_k + \Delta x_{k-1} s'_{k+1} = 3(f[x_k, x_{k+1}]\Delta x_{k-1} + f[x_{k-1}, x_k]\Delta x_k). \tag{B.23}$$

The described equations guarantee the $N-1$ conditions, however, as already mentioned, two more conditions are required which can given by imposing $f'_0 = s'_0$ and $s'_N = f'_N$ and imposing uniform intervals such that $\Delta x_k = \Delta x$, thus the following tridiagonal system of equations is obtained,

$$\begin{bmatrix} 4 & 1 & 0 & & \dots & & 0 \\ 1 & 4 & 1 & 0 & \dots & & 0 \\ 0 & 1 & 4 & 1 & & & \\ & 0 & & \ddots & & & \vdots \\ \vdots & & \ddots & & & & \\ & & & & & 0 & \\ & & & & & 1 & \\ 0 & & \dots & & 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} s'_1 \\ s'_2 \\ \\ \\ \vdots \\ \\ \\ s'_{N-1} \end{bmatrix}$$

$$= 3 \begin{bmatrix} f[x_1, x_2] + f[x_0, x_1] \\ f[x_2, x_3] + f[x_1, x_2] \\ \\ \\ \vdots \\ \\ \\ f[x_{N-1}, x_N] + f[x_{N-2}, x_{N-1}] \end{bmatrix} - \begin{bmatrix} s'_0 \\ \\ \\ \vdots \\ \\ \\ s'_N \end{bmatrix}. \tag{B.24}$$

Bibliography

[1] D. Adminaite, T. Calinescu, G. Jost, H. Stipdonk, and H. Ward, "Ranking EU progress on road safety," European Transport Safety Council, 2018. [Online]. Available: https://etsc.eu/12th-annual-road-safety-performance-index-pin-report/ (visited on 06/19/2018).

[2] "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," SAE International, 2018. [Online]. Available: https://www.sae.org/standards/content/j3016_201806/ (visited on 06/15/2018).

[3] T. Novi, R. Capitani, and C. Annicchiarico, "An integrated artificial neural network–unscented Kalman filter vehicle sideslip angle estimation based on inertial measurement unit measurements," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, p. 095 440 701 879 064, Aug. 2018, ISSN: 0954-4070, 2041-2991. DOI: 10.1177/0954407018790646.

[4] T. Novi, A. Liniger, R. Capitani, and C. Annicchiarico, "Real-time control for at-limit handling driving on a predefined path," *Vehicle System Dynamics*, vol. 0, no. 0, pp. 1–30, Apr. 2019, ISSN: 0042-3114. DOI: 10.1080/00423114.2019.1605081.

[5] T. Novi, A. Liniger, R. Capitani, M. Fainello, G. Danisi, and C. Annicchiarico, "The Influence of Autonomous Driving on Passive Vehicle Dynamics," English, *SAE International Journal of Vehicle Dynamics, Stability, and NVH*, vol. 2, no. 4, Apr. 2018. DOI: 10.4271/2018-01-0551.

[6] D. A. Crolla, "Vehicle Dynamics—Theory into Practice," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 210, no. 2, pp. 83–94, Apr. 1996, ISSN: 0954-4070, 2041-2991. DOI: 10.1243/PIME_PROC_1996_210_250_02.

[7] W. F. Milliken and D. L. Milliken, *Race Car Vehicle Dynamics*. Warrendale, PA, U.S.A: SAE International, 1995, ISBN: 978-1-56091-526-3.

[8] C. Voser, R. Y. Hindiyeh, and J. C. Gerdes, "Analysis and control of high sideslip manoeuvres," en, *Vehicle System Dynamics*, vol. 48, no. sup1, pp. 317–336, Dec. 2010, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423111003746140.

[9] J. Edelmann and M. Plöchl, "Handling characteristics and stability of the steady-state powerslide motion of an automobile," en, *Regular and Chaotic Dynamics*, vol. 14, no. 6, pp. 682–692, Dec. 2009, ISSN: 1560-3547, 1468-4845. DOI: 10.1134/S1560354709060069.

[10] C. Hu, R. Wang, F. Yan, and N. Chen, "Should the Desired Heading in Path Following of Autonomous Vehicles be the Tangent Direction of the Desired Path?" *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2015, ISSN: 1524-9050, 1558-0016. DOI: 10.1109/TITS.2015.2435016.

[11] G. Audisio, F. Cheli, S. Melzi, and M. Velardocchia, "CyberTM Tyre for Vehicle Active Safety," en, in *Conference Paper at AIMETA 2009 Congress*, p. 10.

[12] W. Wang, L. Yuan, S. Tao, W. Zhang, and T. Su, "Estimation of vehicle side slip angle in nonlinear condition based on the state feedback observer," in *Automation and Logistics (ICAL), 2010 IEEE International Conference On*, IEEE, 2010, pp. 632–636.

[13] N. Ding and S. Taheri, "A Modified Dugoff Tire Model for Combined-slip Forces," en, *Tire Science and Technology*, vol. 38, no. 3, pp. 228–244, Sep. 2010, ISSN: 0090-8657. DOI: 10.2346/1.3481696.

[14] H. a. F. er Grip, L. Imsland, T. A. Johansen, J. C. Kalkkuhl, and A. Suissa, "Vehicle sideslip estimation," *Control Systems, IEEE*, vol. 29, no. 5, pp. 36–52, 2009.

[15] H. Shraim, M. Ouladsine, L. Fridman, and M. Romero, "Vehicle parameter estimation and stability enhancement using sliding modes techniques," *International journal of vehicle design*, vol. 48, no. 3-4, pp. 230–254, 2008.

[16] J. C. Cadiou, A. El Hadri, and F. Chikhi, "Non-linear tyre forces estimation based on vehicle dynamics observation in a finite time," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 218, no. 12, pp. 1379–1392, 2004.

[17] Haiyan Zhao and Hong Chen, "Estimation of Vehicle Yaw Rate and Side Slip Angle using Moving Horizon Strategy," IEEE, 2006, pp. 1828–1832, ISBN: 978-1-4244-0332-5. DOI: 10.1109/WCICA.2006.1712670.

[18] H. Zhang, X. Huang, J. Wang, and H. R. Karimi, "Robust energy-to-peak sideslip angle estimation with applications to ground vehicles," en, *Mechatronics*, vol. 30, pp. 338–347, Sep. 2015, ISSN: 09574158. DOI: 10.1016/j.mechatronics.2014.08.003.

[19] S.-H. You, J.-O. Hahn, and H. Lee, "New adaptive approaches to real-time estimation of vehicle sideslip angle," en, *Control Engineering Practice*, vol. 17, no. 12, pp. 1367–1379, Dec. 2009, ISSN: 09670661. DOI: 10.1016/j.conengprac.2009.07.002.

[20] J. Ryu, E. J. Rossetter, and J. C. Gerdes, "Vehicle sideslip and roll parameter estimation using GPS," in *Proceedings of the AVEC International Symposium on Advanced Vehicle Control*, 2002.

[21] M. Doumiati, A. Victorino, A. Charara, and D. Lechner, "A method to estimate the lateral tire force and the sideslip angle of a vehicle: Experimental validation," in *American Control Conference (ACC), 2010*, IEEE, 2010, pp. 6936–6942.

[22] Y. Huang, C. Bao, J. Wu, and Y. Ma, "Estimation of Sideslip Angle Based on Extended Kalman Filter," en, *Journal of Electrical and Computer Engineering*, vol. 2017, pp. 1–9, 2017, ISSN: 2090-0147, 2090-0155. DOI: 10.1155/2017/5301602.

[23] L. Li, G. Jia, X. Ran, J. Song, and K. Wu, "A variable structure extended Kalman filter for vehicle sideslip angle estimation on a low friction road," en, *Vehicle System Dynamics*, vol. 52, no. 2, pp. 280–308, Feb. 2014, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2013.877148.

[24] H. B. Pacejka, *Tire and Vehicle Dynamics*, eng, ser. SAE-R 342. Warrendale, Pa: Society of Automotive Engineers, 2002, OCLC: 249480435, ISBN: 978-0-7680-1126-5.

[25] G. Hodgson and M. C. Best, "A Parameter Identifying a Kalman Filter Observer for Vehicle Handling Dynamics," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 220, no. 8, pp. 1063–1072, Aug. 2006, ISSN: 0954-4070, 2041-2991. DOI: 10.1243/09544070D18304.

[26] G. Baffet, A. Charara, and D. Lechner, "Estimation of vehicle sideslip, tire force and wheel cornering stiffness," en, *Control Engineering Practice*, vol. 17, no. 11, pp. 1255–1264, Nov. 2009, ISSN: 09670661. DOI: 10.1016/j.conengprac.2009.05.005.

[27] Y. F. Lian, Y. Zhao, L. L. Hu, and Y. T. Tian, "Cornering stiffness and sideslip angle estimation based on simplified lateral dynamic models for four-in-wheel-motor-driven electric vehicles with lateral tire force information," en, *International Journal of Automotive Technology*, vol. 16, no. 4, pp. 669–683, Aug. 2015, ISSN: 1229-9138, 1976-3832. DOI: 10.1007/s12239-015-0068-4.

[28] M. Acosta, S. Kanarachos, and M. Blundell, "Virtual tyre force sensors: An overview of tyre model-based and tyre model-less state estimation techniques," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, p. 095 440 701 772 819, Oct. 2017, ISSN: 0954-4070, 2041-2991. DOI: 10.1177/0954407017728198.

[29] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, vol. 3, Orlando, FL, 1997, pp. 182–193.

[30] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, Ieee, 2000, pp. 153–158.

[31] G. Morrison and D. Cebon, "Sideslip estimation for articulated heavy vehicles at the limits of adhesion *," en, *Vehicle System Dynamics*, pp. 1–28, Sep. 2016, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2016.1223326.

[32] S. Antonov, A. Fehn, and A. Kugi, "Unscented Kalman filter for vehicle state estimation," en, *Vehicle System Dynamics*, vol. 49, no. 9, pp. 1497–1520, Sep. 2011, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2010.527994.

[33] Q. Cui, R. Ding, B. Zhou, and X. Wu, "Path-tracking of an autonomous vehicle via model predictive control and nonlinear filtering," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, p. 095 440 701 772 819, Oct. 2017, ISSN: 0954-4070, 2041-2991. DOI: 10.1177/0954407017728199.

[34] G. Jia, C. Yang, L. Li, X. Ran, J. Song, and J. Chen, "UKF-based adaptive variable structure observer for vehicle sideslip with dynamic correction," en, *IET Control Theory & Applications*, vol. 10, no. 14, pp. 1641–1652, Sep. 2016, ISSN: 1751-8644, 1751-8652. DOI: 10.1049/iet-cta.2015.1030.

[35] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems (MCSS)*, vol. 2, no. 4, pp. 303–314, 1989.

[36] X. Du, H. Sun, K. Qian, Y. Li, and L. Lu, "A prediction model for vehicle sideslip angle based on neural network," in *Information and Financial Engineering (ICIFE), 2010 2nd IEEE International Conference On*, IEEE, 2010, pp. 451–455.

[37] S. Melzi and E. Sabbioni, "On the vehicle sideslip angle estimation through neural networks: Numerical and experimental results," en, *Mechanical Systems and Signal Processing*, vol. 25, no. 6, pp. 2005–2019, Aug. 2011, ISSN: 08883270. DOI: 10.1016/j.ymssp.2010.10.015.

[38] H. Sasaki and T. Nishimaki, "A side-slip angle estimation using neural network for a wheeled vehicle," SAE Technical Paper, Tech. Rep., 2000.

[39] W. Wei, B. Shaoyi, Z. Lanchun, Z. Kai, W. Yongzhi, and H. Weixing, "Vehicle Sideslip Angle Estimation Based on General Regression Neural Network," en, *Mathematical Problems in Engineering*, vol. 2016, pp. 1–7, 2016, ISSN: 1024-123X, 1563-5147. DOI: 10.1155/2016/3107910.

[40] B. L. Boada, M. J. L. Boada, A. Gauchía, E. Olmeda, and V. Díaz, "Sideslip angle estimator based on ANFIS for vehicle handling and stability," en, *Journal of Mechanical Science and Technology*, vol. 29, no. 4, pp. 1473–1481, Apr. 2015, ISSN: 1738-494X, 1976-3824. DOI: 10.1007/s12206-015-0320-x.

[41] M. Acosta, S. Kanarachos, and M. E. Fitzpatrick, "Robust Virtual Sensing for Vehicle Agile Manoeuvring: A Tyre-model-less Approach," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2017, ISSN: 0018-9545, 1939-9359. DOI: 10.1109/TVT.2017.2767942.

[42] B. Boada, M. Boada, and V. Diaz, "Vehicle sideslip angle measurement based on sensor data fusion using an integrated ANFIS and an Unscented Kalman Filter algorithm," en, *Mechanical Systems and Signal Processing*, vol. 72-73, pp. 832–845, May 2016, ISSN: 08883270. DOI: 10.1016/j.ymssp.2015.11.003.

[43] Y. Fukada, "Slip-Angle Estimation for Vehicle Stability Control," *Vehicle System Dynamics*, vol. 32, no. 4-5, pp. 375–388, Nov. 1999, ISSN: 0042-3114. DOI: 10.1076/vesd.32.4.375.2079.

[44] R. W. Allen, T. T. Myers, T. J. Rosenthal, and D. H. Klyde, "The effect of tire characteristics on vehicle handling and stability," SAE Technical Paper, Tech. Rep., 2000.

[45] T. Fujioka and T. Kimura, "Numerical simulation of minimum-time cornering behavior," *JSAE Review*, vol. 13, no. 1, pp. 44–51, 1992, ISSN: 0389-4304.

[46] R. W. Allen, T. J. Rosenthal, D. H. Klyde, and J. R. Hogue, "Computer Simulation Analysis of Light Vehicle Lateral/Directional Dynamic Stability," *SAE Transactions*, vol. 108, pp. 482–497, 1999, ISSN: 0096-736X.

[47] C. L. Clover and J. E. Bernard, "The Influence of Lateral Load Transfer Distribution on Directional Response," SAE International, Warrendale, PA, Tech. Rep. 930763, Mar. 1993. DOI: 10.4271/930763.

[48] D. L. Brayshaw and M. F. Harrison, "A quasi steady state approach to race car lap simulation in order to understand the effects of racing line and centre of gravity location," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 219, no. 6, pp. 725–739, Jun. 2005, ISSN: 0954-4070, 2041-2991. DOI: 10.1243/095440705X11211.

[49] A. Tremlett, F. Assadian, D. Purdy, N. Vaughan, A. Moore, and M. Halley, "Quasi-steady-state linearisation of the racing vehicle acceleration envelope: A limited slip differential example," en, *Vehicle System Dynamics*, vol. 52, no. 11, pp. 1416–1442, Nov. 2014, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2014.943927.

[50] D. Casanova, R. Sharp, and P. Symonds, "Minimum Time Manoeuvring: The Significance of Yaw Inertia," *Vehicle System Dynamics*, vol. 34, no. 2, pp. 77–115, Aug. 2000, ISSN: 0042-3114. DOI: 10.1076/0042-3114(200008)34:2;1-G;FT077.

[51] D. P. Kelly and R. S. Sharp, "Time-optimal control of the race car: A numerical method to emulate the ideal driver," en, *Vehicle System Dynamics*, vol. 48, no. 12, pp. 1461–1474, Dec. 2010, ISSN: 0042-3114. DOI: 10.1080/00423110903514236.

[52] D. P. Kelly and R. S. Sharp, "Time-optimal control of the race car: Influence of a thermodynamic tyre model," en, *Vehicle System Dynamics*, vol. 50, no. 4, pp. 641–662, Apr. 2012, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2011.622406.

[53] A. J. Tremlett and D. J. N. Limebeer, "Optimal tyre usage for a Formula One car," en, *Vehicle System Dynamics*, vol. 54, no. 10, pp. 1448–1473, Oct. 2016, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2016.1213861.

[54] G. Perantoni and D. J. Limebeer, "Optimal control for a Formula One car with variable parameters," en, *Vehicle System Dynamics*, vol. 52, no. 5, pp. 653–678, May 2014, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2014.889315.

[55] E. Velenis and P. Tsiotras, "Minimum time vs maximum exit velocity path optimization during cornering," in *2005 IEEE International Symposium on Industrial Electronics*, 2005, pp. 355–360.

[56] D. Tavernini, E. Velenis, R. Lot, and M. Massaro, "On the optimality of handbrake cornering," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference On*, IEEE, 2013, pp. 2233–2238.

[57] D. Tavernini, M. Massaro, E. Velenis, D. I. Katzourakis, and R. Lot, "Minimum time cornering: The effect of road surface and car transmission layout," en, *Vehicle System Dynamics*, vol. 51, no. 10, pp. 1533–1547, Oct. 2013, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2013.813557.

[58] A. Tremlett, M. Massaro, D. Purdy, E. Velenis, F. Assadian, A. Moore, and M. Halley, "Optimal control of motorsport differentials," en, *Vehicle System Dynamics*, vol. 53, no. 12, pp. 1772–1794, Dec. 2015, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2015.1093150.

[59] A. Rucco, G. Notarstefano, and J. Hauser, "Computing minimum lap-time trajectories for a single-track car with load transfer," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference On*, IEEE, 2012, pp. 6321–6326.

[60] A. Rucco, G. Notarstefano, and J. Hauser, "An Efficient Minimum-Time Trajectory Generation Strategy for Two-Track Car Vehicles," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1505–1519, Jul. 2015, ISSN: 1063-6536, 1558-0865. DOI: 10.1109/TCST.2014.2377777.

[61] R. Lot and N. Dal Bianco, "Lap time optimisation of a racing go-kart," en, *Vehicle System Dynamics*, vol. 54, no. 2, pp. 210–230, Feb. 2016, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2015.1125514.

[62] N. Dal Bianco, L. Roberto, and G. Marco, "Minimum time optimal control simulation of a GP2 race car," en, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, p. 095 440 701 772 815, Oct. 2017, ISSN: 0954-4070, 2041-2991. DOI: 10.1177/0954407017728158.

[63] E. Velenis and P. Tsiotras, "Optimal velocity profile generation for given acceleration limits: Theoretical analysis," *American Control Conference, 2005*, vol. 2, p. 5, 2005.

[64] E. Velenis and P. Tsiotras, "Optimal velocity profile generation for given acceleration limits; the half-car model case," in *2005 IEEE International Symposium on Industrial Electronics*, 2005, pp. 355–360.

[65] E. Velenis and P. Tsiotras, "Minimum-Time Travel for a Vehicle with Acceleration Limits: Theoretical Analysis and Receding-Horizon Implementation," en, *Journal of Optimization Theory and Applications*, vol. 138, no. 2, pp. 275–296, Aug. 2008, ISSN: 0022-3239, 1573-2878. DOI: 10.1007/s10957-008-9381-7.

[66] E. Velenis and P. Tsiotras, "Optimal velocity profile generation for given acceleration limits: Receding horizon implementation," in *American Control Conference, 2005. Proceedings of the 2005*, IEEE, 2005, pp. 2147–2152.

[67] J. Zhang, S. Xu, and A. Rachid, "Sliding mode lateral motion control for automatic steering of vehicles," in *Control Conference (ECC), 2001 European*, Sep. 2001, pp. 3636–3641.

[68] M. Klomp, K. Olsson, and C. Sandberg, "Non-linear steering control for limit handling conditions using preview path curvature," *International Journal of Vehicle Autonomous Systems*, vol. 12, no. 3, pp. 266–283, 2014.

[69] K. Kritayakirana and J. C. Gerdes, "Autonomous Cornering at the Limits: Maximizing a "g-g" Diagram by Using Feedforward Trail-Braking and Throttle-on-Exit," en, *IFAC Proceedings Volumes*, vol. 43, no. 7, pp. 548–553, Jul. 2010, ISSN: 14746670. DOI: 10.3182/20100712-3-DE-2013.00060.

[70] K. Kritayakirana and J. C. Gerdes, "Autonomous vehicle control at the limits of handling," *International Journal of Vehicle Autonomous Systems*, vol. 10, no. 4, pp. 271–296, 2012.

[71] M. Guiggiani, *The Science of Vehicle Dynamics*, en. Dordrecht: Springer Netherlands, 2014, ISBN: 978-94-017-7687-5. DOI: 10.1007/978-94-017-8533-4.

[72] K. Kritayakirana and J. C. Gerdes, "Using the centre of percussion to design a steering controller for an autonomous race car," en, *Vehicle System Dynamics*, vol. 50, no. sup1, pp. 33–51, Jan. 2012, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2012.672842.

[73] A. Bemporad, "Model predictive control," PhD course, IMT Lucca, 2017. [Online]. Available: http://cse.lab.imtlucca.it/~bemporad/mpc_course.html (visited on 05/01/2017).

[74] G. Prokop, "Modeling Human Vehicle Driving by Model Predictive Online Optimization," *Vehicle System Dynamics*, vol. 35, no. 1, pp. 19–53, Jan. 2001, ISSN: 0042-3114. DOI: 10.1076/vesd.35.1.19.5614.

[75] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and, "MPC-based approach to active steering for autonomous vehicle systems," *International Journal of Vehicle Autonomous Systems*, vol. 3, no. 2-4, pp. 265–291, Jan. 2005, ISSN: 1471-0226. DOI: 10.1504/IJVAS.2005.008237.

[76] T. Keviczky, P. Falcone, F. Borrelli, J. Asgari, and D. Hrovat, "Predictive control approach to autonomous vehicle steering," in *American Control Conference, 2006*, IEEE, 2006, 6–pp.

[77] T. Kraus, H. Ferreau, E. Kayacan, H. Ramon, J. De Baerdemaeker, M. Diehl, and W. Saeys, "Moving horizon estimation and nonlinear model predictive control for autonomous agricultural vehicles," en, *Computers and Electronics in Agriculture*, vol. 98, pp. 25–33, Oct. 2013, ISSN: 01681699. DOI: 10.1016/j.compag.2013.06.009.

[78] 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," in *Decision and Control, 2007 46th IEEE Conference On*, IEEE, 2007, pp. 2980–2985.

[79] J. P. Timings and D. J. Cole, "Minimum manoeuvre time of a nonlinear vehicle at constant forward speed using convex optimisation," in *Proceedings of The 10th International Symposium on Advanced Vehicle Control, Loughborough, UK*, 2010.

[80] J. P. Timings and D. J. Cole, "Efficient minimum manoeuvre time optimisation of an oversteering vehicle at constant forward speed," in *American Control Conference (ACC), 2011*, IEEE, 2011, pp. 5267–5272.

[81] A. Liniger and J. Lygeros, "Real-Time Control for Autonomous Racing Based on Viability Theory," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 2, pp. 464–478, Mar. 2019, ISSN: 1063-6536. DOI: 10.1109/TCST.2017.2772903.

[82] A. Liniger and J. Lygeros, "A non-cooperative game approach to autonomous racing," *arXiv preprint arXiv:1712.03913*, 2017.

[83] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks," *arXiv preprint arXiv:1609.01387*, 2016.

[84] U. Rosolia, A. Carvalho, and F. Borrelli, "Autonomous racing using learning model predictive control," in *American Control Conference (ACC), 2017*, IEEE, 2017, pp. 5115–5120.

[85] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1:43 scale RC cars: OPTIMIZATION-BASED AUTONOMOUS RACING," en, *Optimal Control Applications and Methods*, vol. 36, no. 5, pp. 628–647, Sep. 2015, ISSN: 01432087. DOI: 10.1002/oca.2123.

[86] Y. Gao, A. Gray, J. V. Frasch, T. Lin, E. Tseng, J. K. Hedrick, and F. Borrelli, "Spatial predictive control for agile semi-autonomous ground vehicles," in *Proceedings of the 11th International Symposium on Advanced Vehicle Control*, 2012.

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

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

[89] R. Verschueren, M. Zanon, R. Quirynen, and M. Diehl, "Time-optimal race car driving using an online exact hessian based nonlinear MPC algorithm," in *Control Conference (ECC), 2016 European*, IEEE, 2016, pp. 141–147.

[90] J. R. Anderson, B. Ayalew, and T. Weiskircher, "Modeling a professional driver in ultra-high performance maneuvers with a hybrid cost MPC," in *American Control Conference (ACC), 2016*, IEEE, 2016, pp. 1981–1986.

[91] H. B. Pacejka, *Tyre and Vehicle Dynamics*, eng, Reprint. Amsterdam: Elsevier Butterworth-Heinemann, 2004, OCLC: 249713869, ISBN: 978-0-7506-5141-7.

[92] I. J. Besselink, A. J. Schmeitz, and H. B. Pacejka, "An improved Magic Formula/Swift tyre model that can handle inflation pressure changes," en, *Vehicle System Dynamics*, vol. 48, no. sup1, pp. 337–352, Dec. 2010, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423111003748088.

[93] R. T. Bundorf, "The influence of vehicle design parameters on characteristic speed and understeer," SAE Technical Paper, Tech. Rep., 1967.

[94] A. Hac and M. O. Bodie, "Improvements in vehicle handling through integrated control of chassis systems," en, *International Journal of Vehicle Autonomous Systems*, vol. 1, no. 1, p. 83, 2002, ISSN: 1471-0226, 1741-5306. DOI: 10.1504/IJVAS.2002.001807.

[95] E. Mousavinejad, Q.-L. Han, F. Yang, Y. Zhu, and L. Vlacic, "Integrated control of ground vehicles dynamics via advanced terminal sliding mode control," en, *Vehicle System Dynamics*, vol. 55, no. 2, pp. 268–294, Feb. 2017, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2016.1256489.

[96] A. T. van Zanten, R. Erhardt, and G. Pfaff, "VDC, The Vehicle Dynamics Control System of Bosch," Feb. 1995. DOI: 10.4271/950759.

[97] A. T. Van Zanten, "Bosch ESP Systems: 5 Years of Experience," May 2000. DOI: 10.4271/2000-01-1633.

[98] A. Carvalho, G. Palmieri, H. E. Tseng, L. Glielmo, and F. Borrelli, "Robust vehicle stability control with an uncertain driver model," in *Control Conference (ECC), 2013 European*, IEEE, 2013, pp. 440–445.

[99] A.-T. Le and C.-K. Chen, "Vehicle stability control by using an adaptive sliding-mode algorithm," *International Journal of Vehicle Design*, vol. 72, no. 2, pp. 107–131, 2016.

[100] M. Jalali, E. Hashemi, A. Khajepour, S.-k. Chen, and B. Litkouhi, "A combined-slip predictive control of vehicle stability with experimental verification," en, *Vehicle System Dynamics*, vol. 56, no. 2, pp. 319–340, Feb. 2018, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2017.1376751.

[101] J. Lu, L. Xu, D. Eisele, S. Samuel, M. Rupp, and L. Tellis, "An Advanced Yaw Stability Control System," Mar. 2017. DOI: 10.4271/2017-01-1556.

[102] A. Morgando, M. Velardocchia, A. Vigliani, B. G. van Leeuwen, and V. Ondrak, "An alternative approach to automotive ESC based on measured wheel forces," en, *Vehicle System Dynamics*, vol. 49, no. 12, pp. 1855–1871, Dec. 2011, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2010.548526.

[103] G. Huijun, S. Weichao, Y. Shen, and O. Kaynak, "Stability control for lateral vehicle motion with uncertain parameters and external nonlinearities," in *Control Conference (CCC), 2013 32nd Chinese*, Jul. 2013, pp. 2733–2738.

[104] E. Velenis, D. Katzourakis, E. Frazzoli, P. Tsiotras, and R. Happee, "Stabilization of steady-state drifting for a RWD vehicle," in *Proceedings of AVEC*, 2010.

[105] E. N. Smith, E. Velenis, D. Tavernini, and D. Cao, "Effect of handling characteristics on minimum time cornering with torque vectoring," en, *Vehicle System Dynamics*, pp. 1–28, Sep. 2017, ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2017.1371771.

[106] G. Mastinu, F. Della Rossa, M. Gobbi, and G. Previati, "Bifurcation Analysis of a Car Model Running on an Even Surface - A Fundamental Study for Addressing Automomous Vehicle Dynamics," en, *SAE International Journal of Vehicle Dynamics, Stability, and NVH*, vol. 1, no. 2, Mar. 2017, ISSN: 2380-2170. DOI: 10.4271/2017-01-1589.

[107] F. Braghin, F. Cheli, S. Melzi, and E. Sabbioni, "Race driver model," en, *Computers & Structures*, vol. 86, no. 13-14, pp. 1503–1516, Jul. 2008, ISSN: 00457949. DOI: 10.1016/j.compstruc.2007.04.028.

[108] B. Nagy and A. Kelly, *Trajectory Generation for Car-like Robots Using Cubic Curvature Polynomials*, /paper/Trajectory-Generation-for-Car-like-Robots-Using-Nagy-Kelly/ea3db89e, 1998.

[109] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, *Neural Network Design*, English. S. l.: s. n., 2016, OCLC: 958124711, ISBN: 978-0-9717321-1-7.

[110] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[111] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[112] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *American Control Conference, Proceedings of the 1995*, vol. 3, IEEE, 1995, pp. 1628–1632.

[113] A. Tesi, F. Vinattieri, R. Capitani, and C. Annicchiarico, "Development of an e-LSD Control Strategy Considering the Evolution of the Friction Torque with the Wear Depth," en, *SAE International Journal of Engines*, vol. 9, no. 3, Apr. 2016, ISSN: 1946-3944. DOI: 10.4271/2016-01-1136.

[114] M. Zanon, J. V. Frasch, and M. Diehl, "Nonlinear moving horizon estimation for combined state and friction coefficient estimation in autonomous driving," in *Control Conference (ECC), 2013 European*, IEEE, 2013, pp. 4130–4135.

[115] M. Guiggiani, *Dinamica del veicolo*, Italian. S.l.: CittàStudi, 2007, OCLC: 800028905, ISBN: 978-88-251-7300-0.

[116] H. Edelsbrunner, "Alpha shapes—a survey," *Tessellations in the Sciences*, vol. 27, pp. 1–25, 2010.

[117] A. Domahidi and J. Jerez, *FORCES Professional*, embotech GmbH (http://embotech.com/FORCES-Pro), Jul. 2014.

[118] A. Zanelli, A. Domahidi, J. Jerez, and M. Morari, "FORCES NLP: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs," en, *International Journal of Control*, pp. 1–17, May 2017, ISSN: 0020-7179, 1366-5820. DOI: 10.1080/00207179.2017.1316017.

[119] W. Liu, L. Xiong, B. Leng, H. Meng, and R. Zhang, "Vehicle Stability Criterion Research Based on Phase Plane Method," Mar. 2017. DOI: 10.4271/2017-01-1560.

[120] G. Genta and L. Morello, *The Automotive Chassis: Volume 2: System Design*, en, ser. Mechanical Engineering Series. Springer Netherlands, 2009, ISBN: 978-1-4020-8673-1.

[121] *ISO 3888-2:2011 - Passenger cars – Test track for a severe lane-change manoeuvre – Part 2: Obstacle avoidance*, https://www.iso.org/standard/57253.html.

[122] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, pp. 238–246, 1980.