POLITECNICO DI TORINO

Department of Mechanical and Aerospace Engineering
Class LM-33 (DM270)

Master's Thesis
in
Automotive Engineering

Slip Controller Development for an Electric 4WD Formula Student Race Car

Tutors:
Prof. Andrea Tonoli
Eng. Stefano Favelli

Candidate:
Federico Battaglino

Academic Year 2021/2022

Dedicated to..

Everyone that helped my growth as an engineer but mostly as a person during my years at Politecnico di Torino and during those as a member in the Squadra Corse team.
I want to start from my family and my love, giving my thanks to who is here and who is not, knowing that everyone of you has always stayed by my side. Than I want to thank my friends all: the closest first, but also those who have been with me during even a small portion of the journey. I hope that I have been able to give back to all of you at least the half of what you gave me during these years. A special thanks goes to Professor Andrea Tonoli, for the help and the support given to Squadra Corse in these years and also to me for this work. Finally, I want to cite the essential help of Stefano, my first team leader, my mentor and a great friend: I wish you the best for your bright future. This would have not been possible without all of you.

This Thesis represents my passion, my experience and work in Formula Student and my competence. I hope that someone will find this useful or inspiring: that would be my deepest satisfaction.

Federico

Contents

Abstract **6**

1 Introduction **8**
1.1 Vehicle Dynamics Controllers 8
1.2 Squadra Corse PoliTO Formula Student Race Car 12
1.3 VI-grade CarRealTime Simulation 16
1.4 Thesis Outline 19

2 Theoretical Background **20**
2.1 Slip Controllers for Electric Vehicles: State of Art 20
2.1.1 PID Controller 21
2.1.2 Explicit Nonlinear Model Predictive Controller 23
2.1.3 Sliding Mode Controller 24
2.1.4 Fuzzy Logic Controller 25
2.2 Vehicle Data Estimation 28
2.2.1 Vehicle States Estimation 29
2.2.2 Tires Forces Estimation 37
2.3 Slip Control Model 43

3 Methodology **46**
3.1 Vehicle Control System 46
3.2 PID Slip Controller 51
3.3 Fuzzy Logic Slip Controller 53
3.3.1 Fuzzy Logic Controller Structure 53
3.3.2 Fuzzy Logic Controller Tuning 56
3.4 Results analysis methodology 64

4 Results **72**
4.1 KPI analysis on simulations 72
4.2 KPI analysis on Real vehicle application 82
4.3 Conclusions 92

List of Figures **94**

List of Tables **96**

References **98**

Abstract

Slip Control is a common topic on high performance race cars vehicle dynamics. Differently from common passenger cars, in this application the main goal is focused on the vehicle dynamic performances improvement. This can be achieved enhancing the vehicle control system capability to properly deliver the right amount of torque to the tires in every working condition.
The subject of this analysis is the development of the Slip Controller for the Formula Student 4-Wheel-Drive (4WD) electric race car of Squadra Corse PoliTO. The prototype is embedded with 4 independently controlled electric motors, which drive one wheel each, giving very high flexibility to the torque delivery. Due to the non-linearity of the tire contact patch interaction with the ground, a proper Controller should be implemented to guarantee good slip control performances also in non-linear conditions.

This work aims to propose a solution to improve the vehicle slip control with the development of a Fuzzy Logic (FL) Slip Controller. This solution has been designed to overcome the limitations imposed by the usage of model-based controllers, which require higher amounts of data, sensors and track validations to achieve similar performances. This makes the FL Slip Controller a cost effective solution both in terms of required testing and computational effort. A Proportional-Integral-Derivative (PID) Slip Controller is also proposed as a benchmark, being the most simple, diffused and yet a quite effective solution, despite its linear approach.
The different solutions are implemented and tested in a Matlab-Simulink and VI-grade CarRealTime co-simulation environment. The quality of this approach is studied and evaluated through simulations and correlation analysis with the real vehicle track data.

Comparative analysis have been performed between the proposed solutions. In order to quantify the performances improvement brought by the proposed controllers, several Key Performances Indicators (KPI) have been introduced. The selected KPIs embrace basic physics, vehicle dynamics and lap-time simulation aspects. Since the analysis is focused on the behaviour of the controller on different working conditions, each of them is characterized by different key aspects. Overall results of the vehicle in normal track drive situations, represented by the different Formula Student (FS) Events, are also analysed.
The comparisons have been performed between the non-controlled vehicle, the PID Slip controller version and the FL Slip Controller one. The results show the improvement brought by the FL Controller in particular, in every KPI. The same KPIs are also used to study the application of the FL Slip Controller on the real race car. They show performances improvement with respect to the non-controlled vehicle, as lap-time improvement in the FS Acceleration of the 5% and the increase of the combined acceleration during cornering of the 6%.

1 Introduction

The design of a race car is mainly focused on the performance enhancement of the vehicle. The majority of the design choices in this application can be summarised in a trade-off between costs, lightweight design and reliability. Today, due to the increase of electronically-controlled devices and electric actuators, it is possible to bring great improvement in vehicle dynamic performances, with enhanced control strategies. Furthermore, the introduction of new clever controllers can be done exploiting already available technologies, optimizing costs and without the addition of more weight.
The subject of this work is the analysis of the performances that a new controller, a Slip Controller, can bring to a full electric race car. The vehicle under analysis is the Formula Student race car of the Politecnico di Torino racing team, Squadra Corse PoliTO. With a 4-independent-motors powertrain, the vehicle controllability is at the center of the design, being one of the most important reason to choice such architecture, rather than a lighter and cheaper 2WD solution.
The development and tuning processes of the control systems are fundamental to extract the most from such solutions. To do so, several tools can be exploited. The V-model is proposed as software development organising tool, while cutting-edge simulation technologies, as those proposed by VI-grade, are used to improve and optimise the controller tuning and subsequent testing.

1.1 Vehicle Dynamics Controllers

Integrated Chassis Control (ICC) is a central topic in modern vehicle dynamics. The increasing number of on-board chassis actuators is enhancing the possibilities towards integration of controllers. The main topics are referred to improvement in safety, comfort and performance. The spreading of ICC is being boosted also by the introduction of electric motors, in hybrid and electric vehicles, that increase the number of the possible control strategies and controllers performances because of the electric motor high controllability.

The ICC can be classified in two principal categories: Downstream Architectures and Upstream Architectures[1].
The former category refers to an horizontal structure of the control hierarchy, in which the control systems work independently until being coordinated at the lowest level of the structure, i.e. the actuators level. This is a bottom-up architecture which guarantees higher recovery possibilities, since the malfunctioning of a controller does not prevent the correct working of the others.

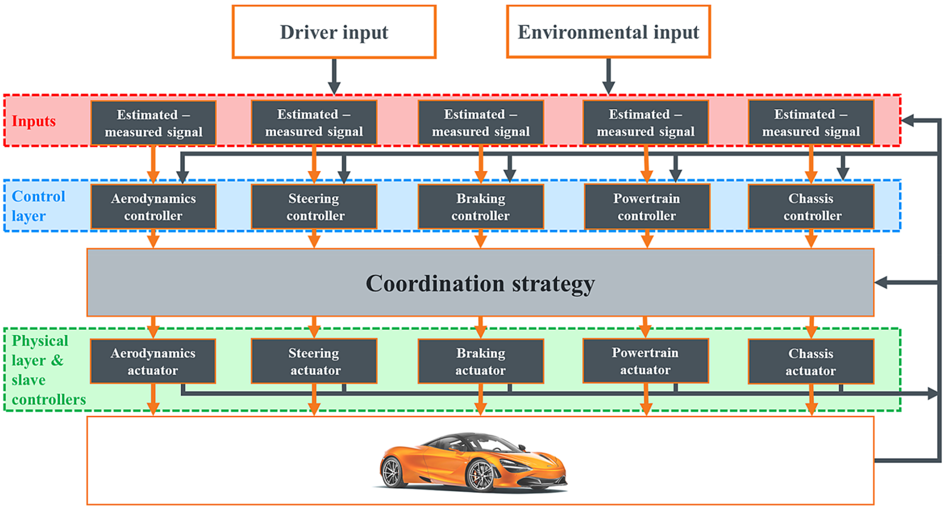

Figure 1: Downstream ICC Architecture Example[1]

The latter category, instead, refers to top-down architectures, in which high-level multivariable controllers are placed between the sensor/state estimations and the actuators levels[1]. An example of multivariable control can be the multi-layer coordination, in which the controllers coordination and monitoring is subdivided in a control stack as the one that follows[1]:

1. The supervision strategy: it defines the control mode and computes the signal references, dependently on the states
2. The high-level controller: computes the control actuations to follow the references
3. The coordination strategy: it selects the control logic dependently on the decisions performed by the supervision strategy
4. The CA strategy: it distributes the control actuation signals to the actuators systems
5. The individual actuator control: it delivers the final control action to each selected actuator
6. The physical layer: here the actuators perform what the previous layers have asked them

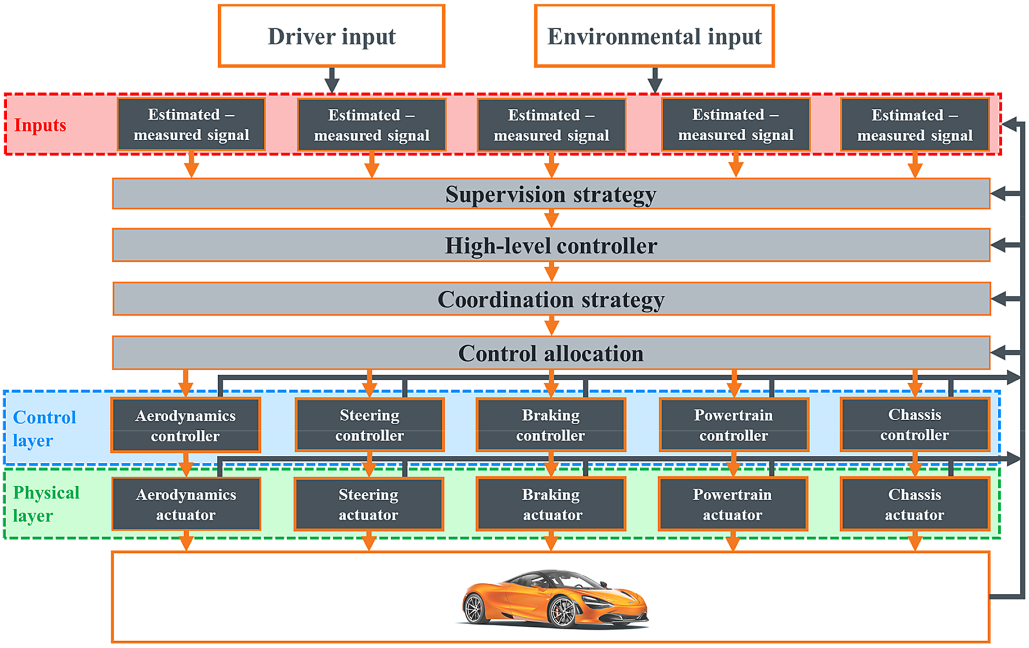

Figure 2: Upstream ICC Architecture Example[1]

The Control System that will be analysed in this work better suits the Downstream approach. The different controllers are, in fact, coordinated at the end of the torque control pipeline, with a logic of minimum: the Chassis Control actions are guaranteed unless they overcome the limits imposed by the Power Control or the Slip Control.
It follows a summary of the most diffused ICC strategies.

Attitude control

Attitude control consists into the improvement of ride comfort and ride performances, reducing the vehicle body motions, i.e. the roll, heave and pitch motions. It includes Active Suspension Systems (ASS), Continuous Damping Control (CDC) and Active Roll Control (ARC). The most common actuators exploited to perform these actions are the semi-active and the active suspension systems, with which is possible to dynamically act on the suspension stiffnesses and/or damping values.

Slip Control

Slip Control embraces all the control strategies that act with the goal of longitudinal slip reduction of the tires of the driving wheels. The two main categories in which the Slip Controllers divide are: Traction Control Systems (TCS) and Antilock Braking Systems (ABS). These controllers work with signals coming from wheel rotational speed measurement and tire longitudinal slip estimators to detect the onset of tire slippage or tire locking conditions, which are critical for the safety of the vehicle passengers, and to correct them.
While the ABS commonly works acting on the braking system, the TCS is able in some cases to properly control also the motor torque. These controllers usually work with proper actuation routines, that are adapted dependently on the estimated road conditions, to increase as much as possible the on-board safety. They also find application in sports cars, increasing the traction and braking capabilities of the vehicles.

Yaw Control

This category of controllers is characterised by the tracking of chassis yaw rate and side slip angle references to ensure vehicle stability and to enhance the vehicle dynamic performances. The two main typologies of controllers that apply these strategies are the Torque Vectoring (TV) and the Electronic Stability Program (ESP).
The TV commonly works with the application of positive torque on the driving wheels. The torque distribution is controlled in a way that a torque unbalance between the two sides of the vehicle is generated, causing a resultant moment. This yaw moment is exploited modifying the vehicle balance attitude towards understeer or oversteer. With common passenger cars the tendency is to obtain an understeering behaviour, which is a safer and easier to control by non-professional drivers. With sports cars, instead, the goal can be to have a vehicle balance as near as possible to the neutral steering, leaving higher degrees of freedom to the drivers to describe the desired trajectories. This control system is quite advanced and frequently requires specific actuators, e.g. electronically controlled differentials of electric motors dedicated to single driving wheels. For this reason, its diffusion is quite narrow in current vehicles, but it is likely to grow in the near future.
The ESP usually works similarly with respect to the TV, but acting on the brake system. This allows to implement this control system on the most of the currently developed vehicles, exploiting technologies already available, for example, for ABS applications. However, differently from the TV, the ESP is mainly applied for safety reasons.

1.2 Squadra Corse PoliTO Formula Student Race Car

Formula Student

Squadra Corse PoliTO is the FS racing team of Politecnico di Torino, for the Electric Vehicles category. Formula Student is an engineering competition in which more than 200 University teams from more than 60 countries challenge. The competition is commonly divided into three main categories: Combustion Vehicles, Electric Vehicles and Driverless Vehicles. As mentioned, Squadra Corse PoliTO competes in the Electric Vehicles class, which has become the widest and most competitive in the last years.

Being an engineering competition and not a racing championship, Formula Student is highly comprehensive of the different engineering disciplines. In fact, every Formula Student competition is divided into two main parts: Static Events and Dynamic Events.

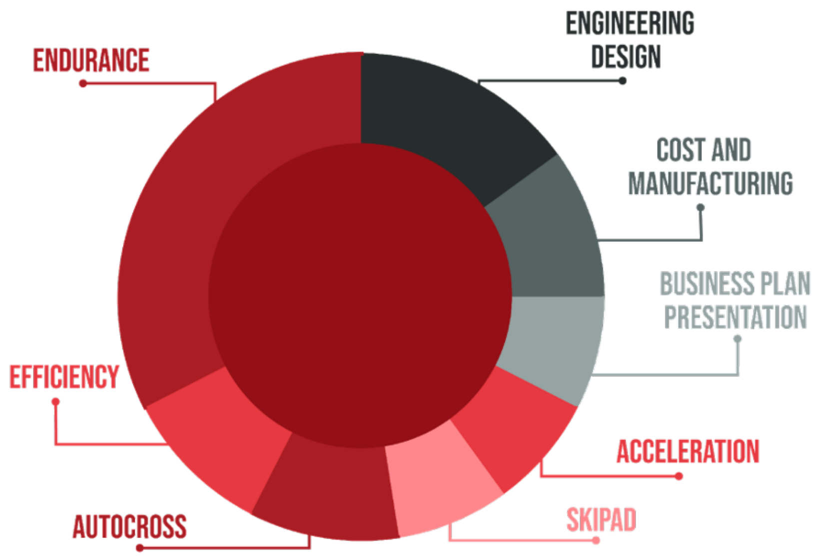

Figure 3: Formula Student Events

The Static Events are three: the Engineering Design, the Cost and Manufacturing and the Business Plan Presentation.
The Engineering Design Event consists into the presentation of the engineering choices performed to design the race car. These choices must be explained and justified to a group of judges, who are experts belonging to different areas of the automotive industry.
During the Cost and Manufacturing event, it is asked to the teams to explain and justify their cost related decisions and the approach to an effective and sustainable manufacturing.
Finally, the Business Plan Presentation consists in a simulation of a real Business Plan case study. The goal is to find the best and innovative business idea to position the team's race car on the market. The entire business process must be defined, from the market evaluation and the financial analysis to the company organization.

The Dynamic Events, instead, are related to the real race cars that compete on the race track against each others. It must be clarified that no wheel-to-wheel racing is allowed in Formula Student and, in fact, the score of almost every event depends only on the relative time spent to complete the test with respect to the best one. The only exception is made by the Efficiency Event, in which the score is based on the energy efficiency of the team during the Endurance Event.
The other Dynamic Events are four: the Acceleration Event, the SkidPad Event, the Autocross Event and the Endurance Event.
The Acceleration Event consists into an acceleration from standing still on a 75 meters long straight line.
The SkidPad event is characterized by two adjacent circles of 9.125 meters of radius and a track width of 3 meters. The driver has to complete two times the right hand side circle and than two times the left hand side one. The score is the average between the time spent to complete the second circle of each side.
The Autocross Event is a single lap of a defined race track, with a length of around 1 kilometer and a minimum track width of 3 meters.
Finally, the Endurance Event, the most important event. It is divided into two stints of 11 kilometers each, each driven by a different driver. In this event, not only the vehicle performances are tested, but also the vehicle overall reliability and its energy management capabilities.

Squadra Corse PoliTO Prototype

Vehicle main data	
Mass without driver	211 kg
Front mass repartition	47.5 %
Wheelbase length	1.525 m
Track width	1.2 m
CoG height (from the ground)	0.28 m
Tires and wheels	185/40 R13 slick tires on 13” OZ Racing magnesium-aluminum alloy rims
Aerodynamic Cl*A	4.78
Aerodynamic Cd*A	1.48
Nominal High Voltage (HV) battery pack capacity	7.7 kWh
Maximum HV battery pack voltage	574.2 V

Table 1: Squadra Corse PoliTO prototype main data

Figure 4: Squadra Corse PoliTO SC22 ”Aurora”

The 2022 Squadra Corse PoliTO prototype is an open-wheel single-seater formula-like race car. The vehicle structure is characterized by a CFRP and aluminum honeycomb sandwich panels monocoque; a steel tubes and aluminum tubes roll-over protection structure and an aluminum honeycomb crashbox for front impact protection. The fully adjustable pushrod suspensions are made of CFRP tubes and are connected with racing uniball joints to the monocoque and to the machined aluminum uprights.
The electric powertrain is the AMK Formula Student racing kit. It consists of 4 IGBT inverters that supply one SPM-IPM Electric Motor each. Each motor is able to generate a maximum of 21 Nm of Torque and it is limited at 20000 rpm. The powertrain is able to develop up to 140 kW at 600 V DC, but it is limited at 80 kW of Electric Power at the HV battery pack output due to Formula Student regulations[2]. The motors are mounted outboard on dual-stage planetary transmissions that reduce the motors speed with a final ratio of 14.69:1.
The HV battery pack is composed by 2 parallels of 132 series of pouch Li-Po cells. The battery pack has a capacity of 15.2 Ah or 7.7 kWh at the nominal voltage.
The on-board communications run on four Controller Area Networks (CAN) that are handled by the Vehicle Control System on the dSpace MicroAutobox II Electronic Control Unit. The Control System is also responsible for the dialogue with the HV Battery Monitoring System (BMS) and with the inverters.

1.3 VI-grade CarRealTime Simulation

The workflow proposed to analyse and develop any control system with Matlab-Simulink in co-simulation with VI-CarRealTime follows the V-model approach:

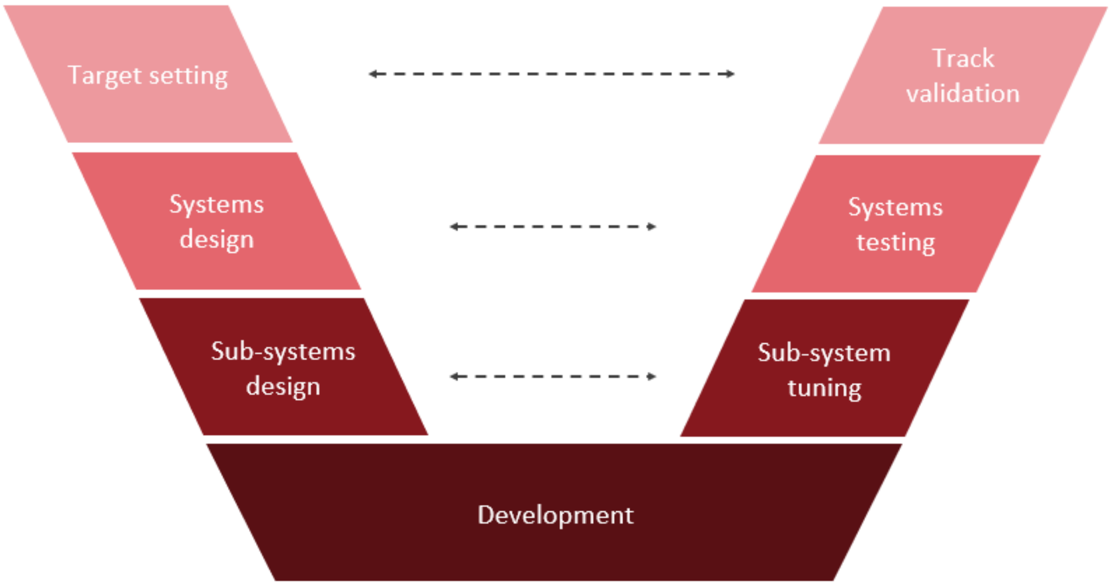

Figure 5: Project V-model

The V-model is a development and testing method. It consists into a sequence of actions: from a general approach, the design goes down to the particular and then it goes back in the opposite direction, during the testing. The advantages brought by this method are related to the complete project understanding, since also the testing and validation phases are defined before the beginning of the development[3].
Some of the main sections of the V-model, from the Development to the System testing, exploit Matlab-Simulink and VI-CarRealTime environments.

Matlab-Simulink is considered as commonly known and no further descriptions of the software are reported.
VI-grade defines VI-CarRealTime as a "virtual modelling environment targeted to a simplified 4 wheels vehicle model". With VI-CarRealTime it is possible to create a high-fidelity digital twin of the real vehicle, with the possibility to assemble the real car system through its most important subsystems.

The VI-CarRealTime digital twin of Squadra Corse PoliTO is characterized by:

- Real mass and mass distribution values of: sprung masses, unsprung masses and driver mass
- Full vehicle CAD estimated inertias
- Real vehicle overall dimensions: wheelbase length, track width, CoG position, etc...
- Aerodynamic Forces maps in function of the front and rear ride heights as results of the validated CFD model
- Full suspensions and steering system elasto-kinematic model imported from MSC Softwares Adams Car environment
- Full powertrain model: motors outboard positioning, traction, coasting and braking torque vs speed maps, powertrain efficiency maps, rotors inertia, transmission ratio and transmission efficiency
- HV battery model (on which correlation analysis have been performed)
- Tire Pacejka Magic Formula 6.1 model with data coming from the Pirelli TIR file
- Real mechanical braking system data

VI-CarRealTime offers the possibility to perform mainly three typologies of simulations: the standardized maneuvers, the file driven events and the max performance events.
The standardized maneuvers represent the common automotive benchmark maneuvers, which are characterized by precisely defined sequences of actions that the driver model has to perform, in open or closed loop. The file driven events are completely customisable: the user can define the sequence of actions, the path, the speed profile and many other variables of the simulation.
Finally, there are the max performance events. These simulations are characterized by a reference path, subdivided in sectors, that the driver model has to follow and a maximum distance that defines the maximum deviation that the vehicle can reach from the reference path. The driver model can be tuned with 3 different parameters in these events, called Performance Factors (PF): Longitudinal PF, Lateral PF and Braking PF. Increasing a PF means increasing the driver exploitation of the vehicle performance limit in that condition. If the vehicle fails to remain within the maximum distance from the path, the simulation is stopped. Then, the PF that caused the limit to be exceeded are reduced of a predefined quantity and the simulation is started again from the end of the previous sector: this is called iteration. It can be noticed that the number of iterations, during comparative analyses between controllers, can be an indicator of the higher or lower controllability of the vehicle. In fact, a lower number of iterations means that the driver model has been capable of running the same sector with higher PF without overcoming the maximum distance threshold.

Simulink and VI-CarRealTime co-simulations

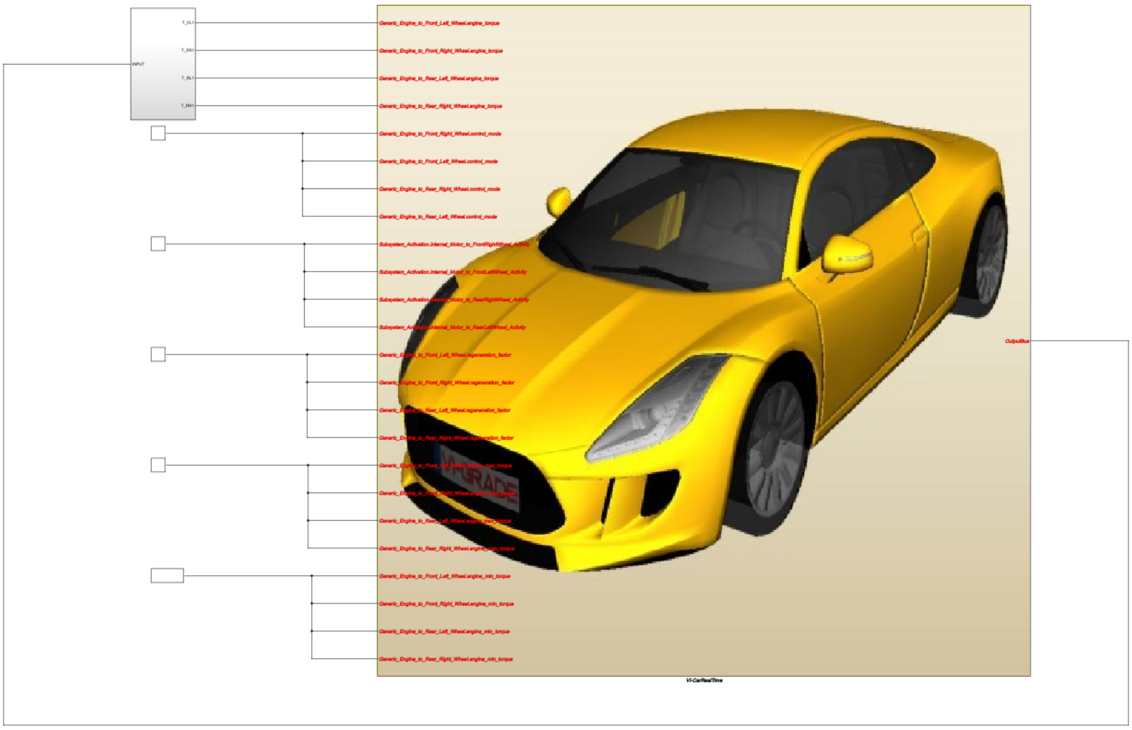

Figure 6: VI-CRT and Matlab-Simulink co-simulation

The VI-CarRealTime and Simulink co-simulation environment permit to import into the Simulink environment the vehicle model. During the simulation, the CarRealTime defined event is run, but the user can select a part of the input signals necessary for it and override them with the Simulink model produced signals. Furthermore, the CarRealTime model returns in feedback a huge amount of vehicle states signals. Some examples of the most important data are: vehicle chassis velocities and accelerations, tires forces, High Voltage battery data (voltage, current, State Of Charge (SOC), etc...) and suspensions data, both kinematics and dynamics. These data represent an even higher number of signals with respect to those coming from the real vehicle sensors, increasing the possibilities of tuning and comparatives of the different controllers. It is important to focus on the fact that these tuning possibilities can be obtained months before the real vehicle is ready to be tested, leaving to the race track only the finest tuning.

1.4 Thesis Outline

This Thesis project is structured as follows:

1. Chapter 2 presents the necessary theoretical background to develop and analyse the discussed topics. It is mainly focused on vehicle kinematics, vehicle dynamics and automatic control logics
2. Chapter 3 presents the methodologies applied on the development of the studied controller and the strategies related to the analysis of the results
3. Chapter 4 is the conclusion of this work. The final analysis of the results of the simulations data and the real track application data are shown, following the methods previously discussed

2 Theoretical Background

In order to proper develop and implement a control strategy as the Slip Control, it is necessary to understand which typologies are already available and diffused, and which of them can best suit in the scope of application.
Once analysed the different control solutions, it is important to study the sequence of mathematical equations that are necessary to estimate the controller input signals, which can be not available from sensors readings. This is due to the high costs and complexity of the necessary instruments, that can make them not applicable in situations as the analysed one in this work.
Finally, a physical model is proposed as a possible solution to implement a model-based Slip Controller. The quality of the estimations produced by the available data and the chosen equations is discussed. This study is concluded with an analysis of the reasons which have led to the decision to concentrate the development of the Slip Controller on a peculiar solution as the Fuzzy Logic Controller.

2.1 Slip Controllers for Electric Vehicles: State of Art

Battery Electric Vehicles (BEV) can rely on the possibility of proper torque control algorithms to perform control strategies, as slip controllers, with the direct actuation on the electric motors. With respect to a conventional Internal Combustion Engine (ICE), the electric powertrain has the potential to improve the performances of wheel slip control in traction and braking conditions. This is due to the higher precision in torque modulation with respect to the hydraulic/electro-hydraulic braking systems and the ICE[4].
The electric motors bring with them several additional advantages, as the possibility to reduce the wear of the braking system components and to recover kinetic energy with the regenerative braking. This brings relevant improvement in the energy efficiency of the vehicle, other than the increase in performance and safety. For example, the actuation of an electric motor used to perform some slip control can be either a simple input torque reduction or a regenerative torque application. Another example can be the modulation of the regenerative braking torque used to perform some ABS: the wheel lock avoidance permits to maintain the possibility to regenerate the energy, that would not be possible with no rotational speed of the motor.

In this chapter, several Slip Control algorithms are discussed, which represent the actual state of art regarding the control strategies applied on BEV.

2.1.1 PID Controller

The Proportional-Integrative-Derivative (PID) is commercially the most diffused controller. The reasons can be found on its simplicity and its overall good performances, that make it results in a cost and complexity effective solution. The PID controller actuation is dependant on three different actions:

- Proportional: it returns a contribute proportional to the reference error. It is mainly used to reduce the steady-state error and the rise time but it increases the overshoot.
- Integrative: it returns a contribute proportional to the integral of the reference error. It is mainly used to eliminate the steady-state error but it increases the overshoot.
- Derivative: it returns a contribute proportional to the derivative of the reference error. It works with a damping effect on the signal, reducing the overshoot and the oscillations around the reference. It highly increases the command activity.

The complete control linear problem results in the following:

$$C(t) = -K_p e(t) - K_i \int e(t)\, dt \ - K_d \frac{de(t)}{dt} \tag{1}$$

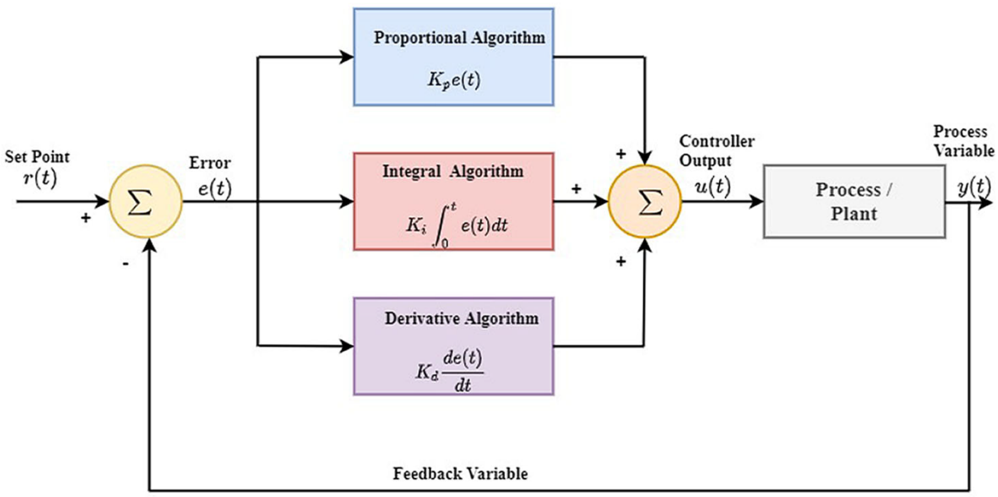

Figure 7: PID Controller Structure[5]

The controller tuning can be performed with different methods. The most simple tuning methodology is the Trial and Error one[6]. It consist into an iterative process in which the tuning parameters are manually changed after each simulation in order to obtain the controller desired behaviour. The tuning starts from the Proportional parameter until the rise time and the reference tracking are acceptable.

Subsequently the final controller behaviour at steady-state can be tuned, introducing and modifying the Integrative parameter. The Derivative parameter is finally tuned to find the proper trade-off between rise time and overshoot and also to reduce the steady-state oscillations around the reference.
Other examples of the most diffused PID Controller tuning methods are those introduces by Ziegler-Nichols: the Step Response Method and the Frequency Response Method[5]. Finally, it is important to cite the online tuning procedure that can be implemented. Through model-based approach, it is possible to exploit Matlab-Simulink Graphic User Interfaces (GUI) tools that allow to manually o automatically tune the PID contoller, with the only input of the desired phase margin.

The proposed PID Controller for Slip Control application works on a tire longitudinal slip signal. The estimation of the longitudinal slip and the selection of the longitudinal slip reference will be discussed in chapter 3.3.2.

Since the PID Controller is a Single Input Single Ouput (SISO) Controller, it can control only a single longitudinal slip signal. The Controller input, which is the longitudinal slip error, is the result of the sum of the slip reference and the slip signal, the latter is taken with the negative sign. If the longitudinal slip error becomes negative, this means that the longitudinal slip signal is above the threshold and so a controller action dependant on the entity of the error and on the tuning parameter of the controller will be produced.
The controller action works as a torque reduction signal that summed up with the actual torque request generates the final torque request.

2.1.2 Explicit Nonlinear Model Predictive Controller

The recent Control Science literature regarding the Slip Control topic shows growing interest in model-based control, with focus on the model predictive control (MPC)[4]. A model based controller heavily depends on mathematical equations to represent the system to be controller.
The MPC, in particular, works predicting the evolution of the system in a finite period of time, called the prediction horizon, and computes the control action dependently on the references it has to track, with the performances that has been decided during the tuning procedure of the controller.

Generally, due to the highly nonlinear behaviour of the tire slip, a nonlinear approach with MPC (NMPC) shows better performances. In the most common NMPC implementation, the implicit one, a nonlinear programming (NLP) problem is solved on-line at each sampling time. If the required sampling time is high, as for what regards the real-time applications, it can be difficult to implement this kind of controller due to the high computational load[4]. For this reason, explicit approach to the NMPC can be preferable. In this case, the solution is computed offline with the usage of a multiparametric (mp) quadratic programming (QP) with numerical approximation of the NLP problem. Than, the control action can be computed online in real time with the current system states and parameters.
The explicit NMPC requires the formulation of an optimization problem for a finite prediction horizon, usually including constraints both on controller inputs and outputs. It can consists into the minimization of a cost function. The parameters of the cost function are the states and the inputs of a model that describes the problem. The most common model and its equations will be discussed in the following chapter 2.3.

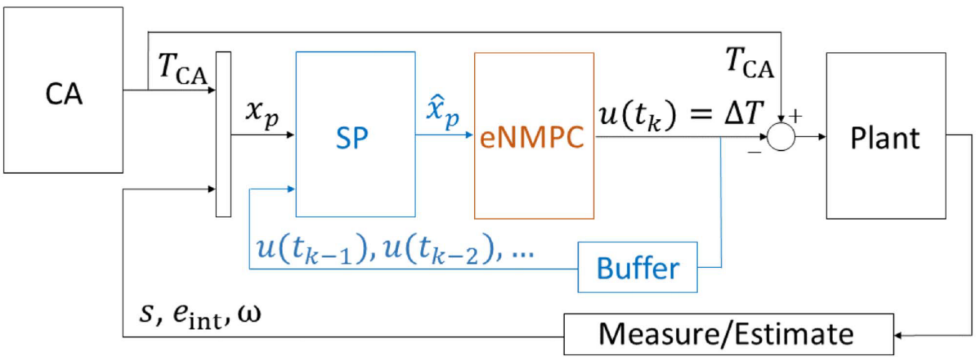

Figure 8: Explicit NMP Slip Controller Structure[4]

2.1.3 Sliding Mode Controller

The Sliding Mode Controller is a Variable Structure Controller (VSC), which includes different functions that translates the plant states into a control surface. The switching between the different functions is determined by the switching function[7]. A general structure for a sliding mode controller exploits a switching function which causes the change of the control action sign dependently on the two sides of it. The threshold that defines the passage from a side to another is called sliding surface. The sliding surface can have a certain thickness, called boundary layer, that causes a smoother passage from a condition to another: this reduces the chattering around the sliding surface. The control strategy works moving the system trajectory towards the sliding surface, guaranteeing the system stability. For this reason, the thickness of the boundary layer defines the robustness of the control method.

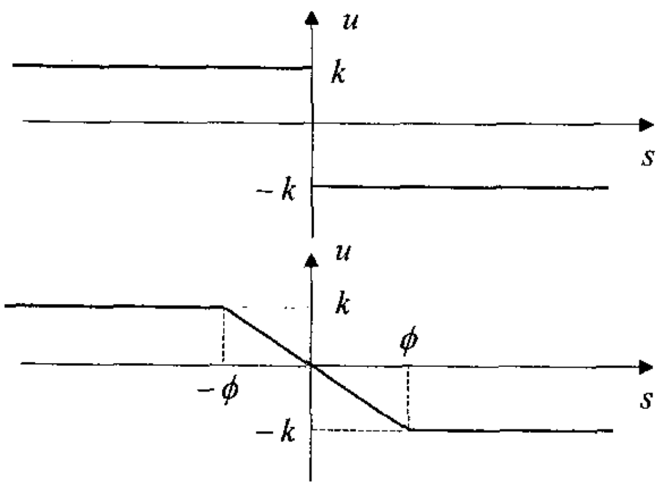

Figure 9: Various Sliding Mode Controllers[7]

The above figure shows the two discussed typologies of sliding surfaces: the upper one belongs to a controller without boundary layer, while the lower one to a controller with boundary layer.

The application of the sliding mode controller on the Slip Control problem is based on the longitudinal slip control. The controller, in fact, minimizes the slip when it overcomes a certain reference, bringing it back to the slip stable region.

2.1.4 Fuzzy Logic Controller

FL control has become an important approach regarding the design of non linear controllers because of many factors. It is, in fact, quite simple to design, does not rely on complex mathematical models and equations and its implementation is usually helped by commercially common tools[8]. The first development of a FL controller was possible thanks to Mamdani work in 1974[9] and its mathematical fundation was stated by Wang in 1994[10].
The FL approach is based on a knowledge-based approach that exploits language variables. This permits an easier design of the problem when a physical and mathematical model is not present or the quality of the data to describe the model is not valid enough. Furthermore, it also permits to rely on initial data present only in a qualitative form. To do so, with FL is possible to exploit fuzzy rules to derive control actions from the inference system inputs. A fuzzy rule is a conditional statement, expressed in the form IF-THEN[8], while the deduction of the rule is the inference.

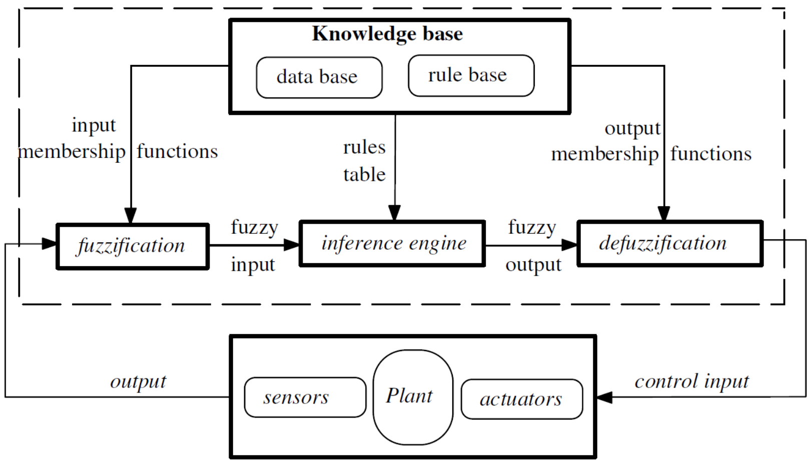

Figure 10: Fuzzy Logic Structure[8]

The general structure of a Fuzzy Inference System (FIS) is characterized by three main blocks: *fuzzification*, *inference engine* and *defuzzification*. The *fuzzification* and the *defuzzification* are the procedures that permit the translation of a numerical variable into a fuzzy variable. They are performed through some membership functions sets. A fuzzy variable is a number, that usually lies between 0 and 1 or even -1 and 1, that is the result of the degree of membership of the variable to every membership function of its set.

A set of a membership functions depends on the language variables that designer chooses for the description of that input variable. There are different typologies of membership functions, dependently on their shape: triangular, trapezoidal, bell and Gaussian. The triangular function is the most common because of its computational efficiency and simplicity[8], that makes it convenient for real time applications.

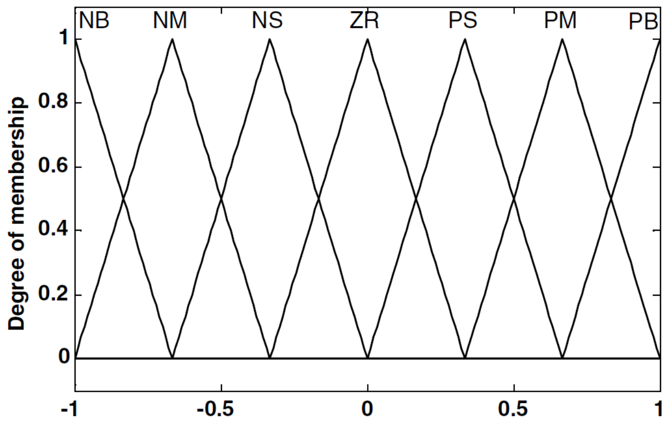

Figure 11: Fuzzy Logic Controller triangular Membership Functions set[8]

The *inference engine* describes the set of fuzzy rules that translate the fuzzy input variables into one or more fuzzy output variables. The most diffused approach for what regards the input variables is to exploit the controlled variable error with respect to a reference and the derivative of that error. For what regards the outputs, instead, they represent the controller actions.

Finally, the definition of the rules can be approached with two main methods: the Mamdani approach and the Sugeno-Takagi approach. Usually Mamdani inference is recommended for use because it produces stronger control actions in certain cases[8]. With this method, a rule is defined for every combination of the inputs language variables. It follows the construction of a table, as the one proposed in the following example:

Change of error $ce(k)$	Error $e(k)$						
	NB	NM	NS	ZR	PS	PM	PB
PB	NS	NS	NS	PS	PL	PL	PL
PM	NS	NS	NS	PS	PL	PL	PL
PS	NM	NM	NS	PS	PM	PL	PL
ZR	NL	NM	NS	ZR	PS	PM	PL
NS	NL	NL	NM	NS	PS	PS	PM
NM	NL	NL	NL	NS	PS	PS	PS
NB	NL	NL	NL	NS	PS	PS	PS

Table 2: Fuzzy Logic Controller General Table of Rules[8]

The most diffused tuning procedures are related to algorithms as the Genetic or the Pattern Search ones, with the goal of finding local minimum points in the control surface, that represent the stability of the systems.

For what has been discussed in this chapter, the FL Controller suits particularly for a control strategy as the Slip Control. The reasons can be found in the lack of experimental data for what regards the contact patch forces, with the consequence of poor estimations performed by the model. This and the structure of the proposed FL Slip Controller will be analysed with higher depth in chapters 2.3 and 3.3.1.

2.2 Vehicle Data Estimation

The implementation of a Slip Controller requires input signals that can properly represent the onset of the tires slippage or blocking conditions. It is common to estimate the tire states, as the tire side slip angle or tire longitudinal slip, to obtain these signals that are very complex and expensive to measure on track. This chapter proposes a sequence of vehicle physical estimations, that are necessary to obtain proper longitudinal slip signals.
Furthermore, a possible method is proposed to estimate the tire contact patch forces. These estimations are necessary for the implementation of the vast majority of the model-based Slip Controllers. Since in chapter 2.3 it is presented a model to perform such implementation, also the evaluation of the necessary data to build the model is shown.

Every estimation performed in this chapter is compared with the output of the VI-CarRealTime and Simulink co-simulation. The simulation is based on a FS Skidpad Event, in which it is possible to appreciate both transient and steady state vehicle behaviour. These make this event very useful to study both longitudinal and lateral dynamics, in an environment that can be reproduced during the track tests for correlation analysis. The blue curves represent the simulation output, while the orange curves represent the new estimations results.

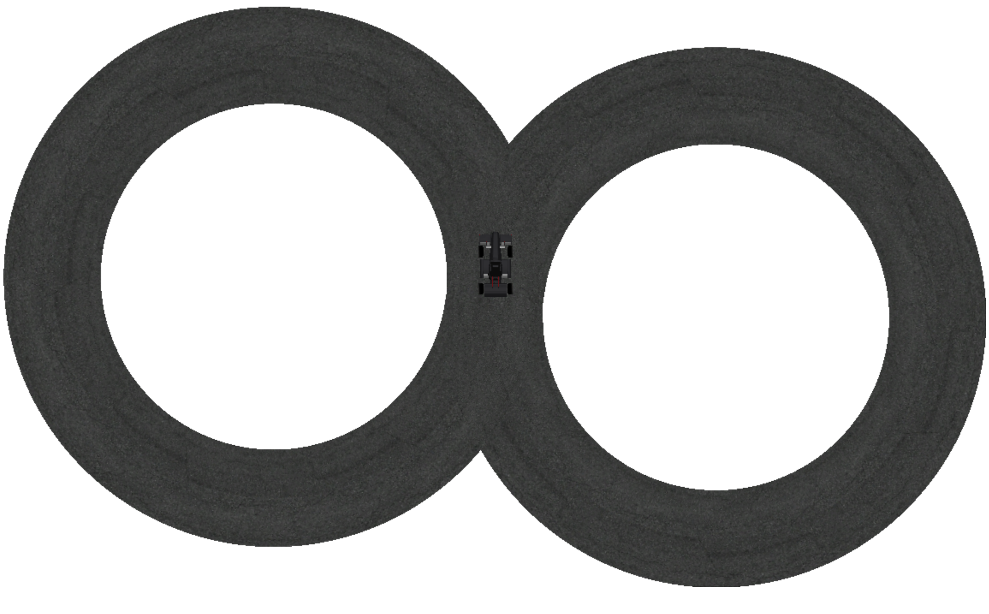

Figure 12: FS Skidpad Event track

2.2.1 Vehicle States Estimation

Dual Track Model

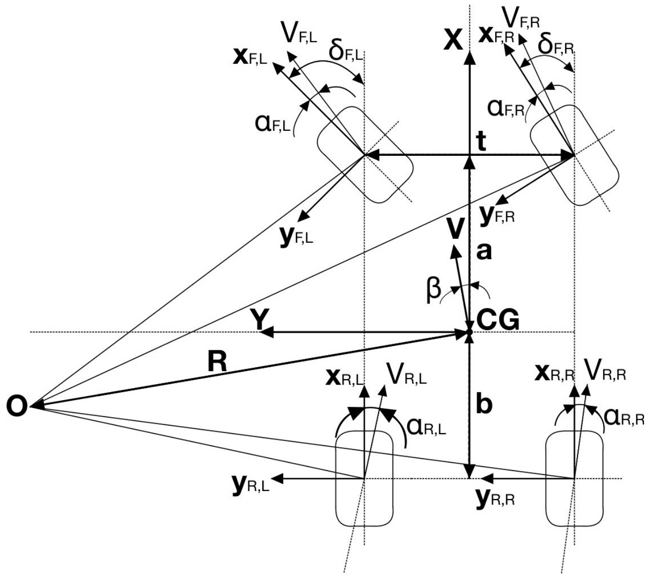

Figure 13: Dual track model

The kinematic model exploited to perform the vehicle states estimation is the dual track model. In vehicle dynamics is common to exploit single track model but, for FS applications, some of its hypothesis are not suited. In particular, the wheelbase and track dimensions are not negligible with respect to the most common values of cornering radius of the competitions. For this reason it is not possible to collapse the outer wheels into a center one (one for each track) to simplify the model. The vehicle side slip angle is represented with the letter β, the wheels steering angles with δ, the tires side slip angles with the letter α and the tires longitudinal slip ratios with the letter σ.

Vehicle Side Slip Angle

The vehicle side slip angle, also known as attitude angle, is defined as the angle between the vehicle CoG velocity vector and the vehicle XY plane[11]. The attitude angle is usually estimated, since its measurement is complex and expensive. The common practice consists into the application of states estimators as the Extended Kalman Filter (EKF).
The race car of Squadra Corse PoliTO is embedded with a SBG Systems Ellipse-N, a system composed by an Inertial Measurement Unit (IMU) and a GPS. This tool is embedded with an EKF that is aided by the velocity measurement provided by the single antenna GPS.
A simplified approach regarding the estimation of the attitude angle is proposed in this chapter.

- Vehicle Side slip angle:

$$\beta(k) = (\frac{a_{centr}(k)}{V_{CG}(k)} - r(k))(t(k) - t(k-1)) \tag{2}$$

- Centripetal acceleration:

$$a_{centr}(k) = a_y(k)cos(\beta(k-1)) - a_x(k)sin(\beta(k-1)) \tag{3}$$

The proposed formulas approach the side slip angle estimation relying only on measured signals, in particular the vehicle longitudinal and lateral accelerations. The signal is produced from the discrete integration of its derivative. The attitude angle derivative is the result of the difference between the centripetal acceleration and the yaw rate. In this approach, the centripetal acceleration is calculated with an iterative process, exploiting its previously estimated value.

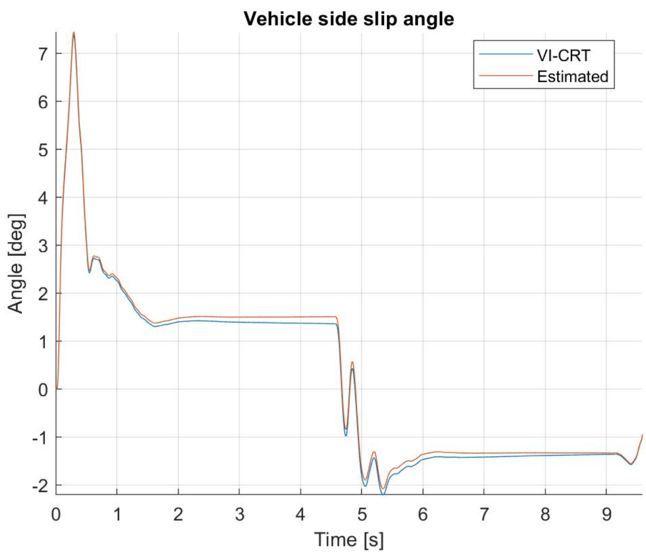

Figure 14: Side Slip Angle comparison

Figure 14 shows the comparative analysis performed between the side slip angle resultant from the VI-CarRealTime simulation and the output of the proposed approach. The plot shows a good correlation between the two signals.

Tire Side Slip Angle

The generation of a lateral force due to the contact between the tire and the ground can occur only due to a lateral deformation of the tire itself.

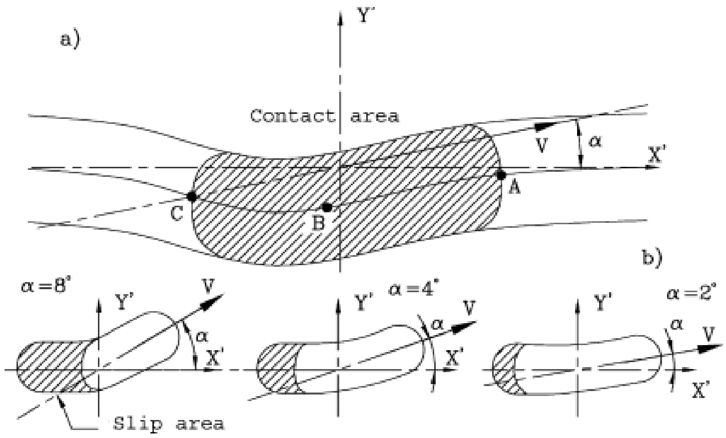

Figure 15: Tire Side Slip Angle[12]

The tire side slip angle is the angle that generates between the XY plane of the tire contact patch reference frame and the tire velocity vector, due to the mentioned deformation. It is possible to compute this tire angle with a kinematic analysis of the dual track model, starting from the vehicle side slip angle and the wheel steering angle. The proposed formulas to obtain the side slip angle are the following:

- Front Tires Side Slip Angle:

$$\alpha_i = \arctan(\frac{V_y + r * x_i}{V_x - r * y_i}) - \delta_i \tag{4}$$

- Rear Tires Side Slip Angle:

$$\alpha_i = \arctan(\frac{V_y + r * x_i}{V_x - r * y_i}) \tag{5}$$

The results of this estimation are depicted in the following figure, in comparison with the CarRealTime simulation output.

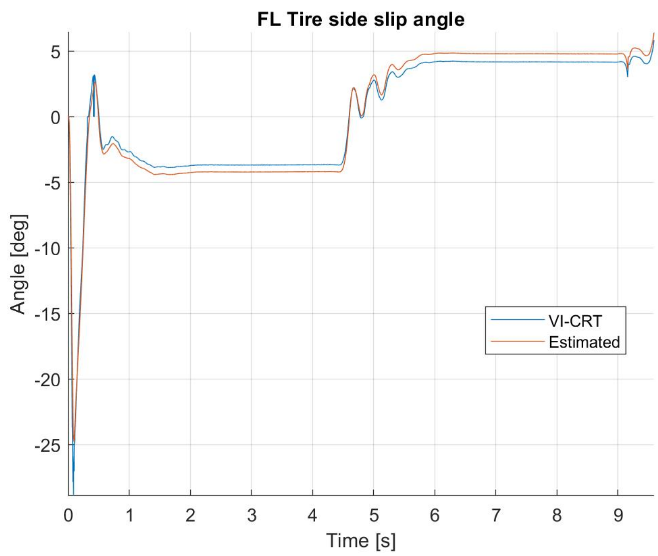

Figure 16: Tire Side Slip Angle comparison

The quality of the estimation is acceptable, both in transient and in steady state conditions.

Wheel Longitudinal Velocity

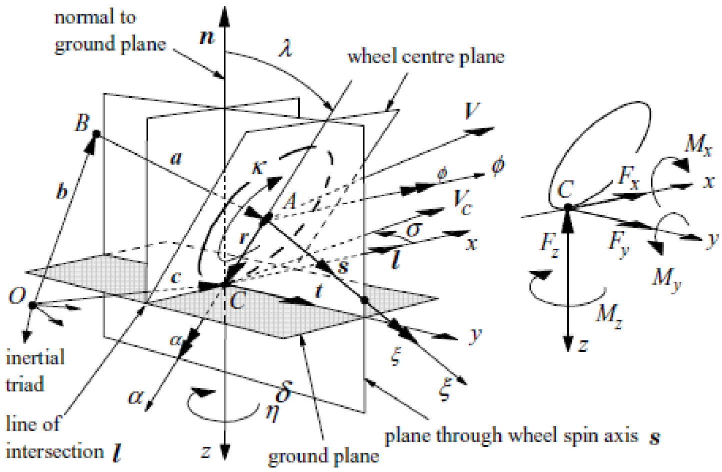

Figure 17: Wheel Reference Frame and angles[13]

The wheel longitudinal velocity is necessary for the longitudinal slip estimation that wheel be discussed in chapter 2.2.1. It represents the velocity of the origin of the wheel reference frame. This velocity can be evaluated from a kinematic analysis of the dual track model. In the Appendix it is possible to find the mathematical steps necessary to define the following formulas.

- Front wheels longitudinal velocity:

$$V_{x,i} = r * cos(\alpha_i) * \left(\frac{R*cos(\beta) - x_i}{cos(\delta_i) - cos(\alpha_i)}\right) \tag{6}$$

- Rear wheels longitudinal velocity:

$$V_{x,i} = r * cos(R * cos(\beta) - x_i) \tag{7}$$

The quality of the estimation is analysed comparing the signal with the correspondent output of the CarRealTime simulation.

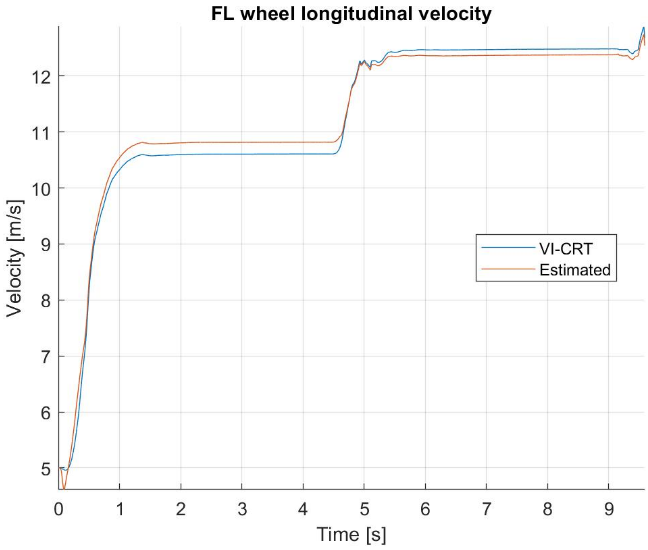

Figure 18: Wheel Longitudinal Velocity

Also in this case, the comparison shows positive results, since the correlation is good.

Tire Longitudinal Slip

As for what regards the generation of cornering force, the tire deforms in order to develop the longitudinal force.

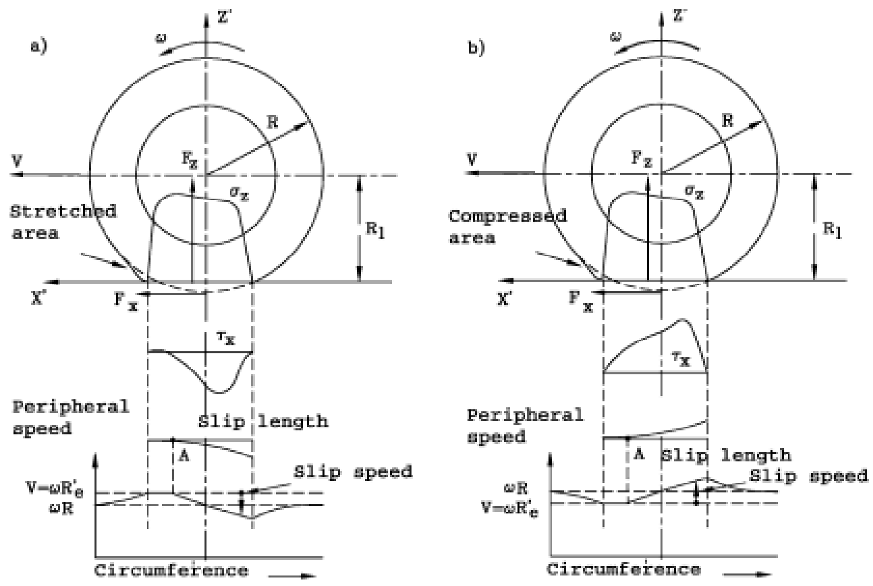

Figure 19: Tire Longitudinal Slip[12]

In traction conditions, for example, the foremost part of the contact zone is compressed. The value of the effective rolling radius $R_e^{'}$ is smaller than the one characterizing the free rolling and is usually smaller than the loaded radius R_l. The angular velocity Ω of the wheel becomes grater than the pure rolling one Ω_0. In this case, it is possible to define a longitudinal slip as follows[12]:

- Tire Longitudinal Slip:

$$\sigma = \frac{\Omega}{\Omega_0} - 1 \tag{8}$$

In this work, however, it has been decided to exploit another definition used to represent this physical aspect. Since the FL Controller works with variables that represent percentages, it has been decided to follow this approach to represent the slip. To do so, many different formulas are present in literature. The chosen one, mentioned in [14], is the following:

- Tire Longitudinal Slip* Ratio:

$$\sigma_i^* = \frac{\omega_i * R_i - V_{x,i}}{\max(\omega_i * R_i, V_{x,i})} \tag{9}$$

Since this formula does not represent the proper definition of the longitudinal slip, in this work it will be reported as longitudinal slip* ratio. The proposed formula exploits V_x, the estimated wheel longitudinal velocity that has been previously discussed.

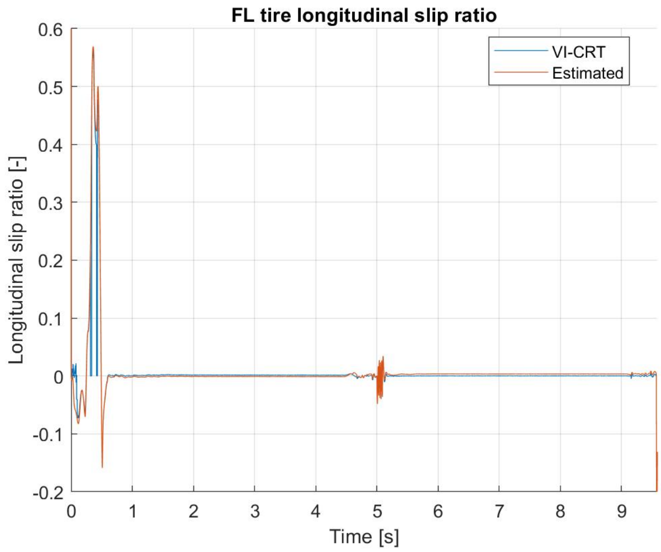

Figure 20: Tire Longitudinal Slip* Ratio comparison

Figure 20 shows the correlation of the estimation with the CarRealTime simulated signal. The quality of the correlation is good, especially in transient conditions in which this phenomenon is quite difficult to estimate with a proper precision.

2.2.2 Tires Forces Estimation

Tire Contact Patch Forces - Normal Force

The estimation of tire forces is fundamental in vehicle dynamics control strategies. In chapter 2.3, the physical model proposed as a possible approach to model based slip control is presented. This model exploits the contact patch forces estimation that is going to be shown in this section.
The force estimation starts with the normal force, which is used as input for the estimation of the other contact patch forces. The normal force is not directly dependant on the tire characteristics, differently from the other forces that are analysed in this chapter. Considering a flat road, it depends on two main typologies of forces: the aerodynamic forces and the vehicle loads. In a simplified analysis as the one that follows, the aerodynamic contribute is represented by the lift force, characterized by the lift coefficient C_z. With vehicle loads, it is possible to refer to the vehicle static load (caused by its mass) and to the load transfers. In particular, the longitudinal load transfer is due to the longitudinal acceleration and the lateral load transfer is due to the lateral acceleration.

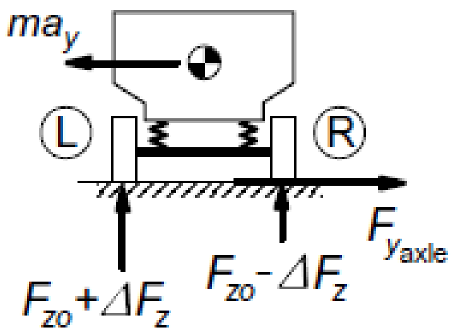

Figure 21: Example of load transfer: lateral[13]

The formulas exploited for the computation of the normal forces of each contact patch are the following:

- Aerodynamic Forces:

$$F_{z,aero} = \frac{1}{2}\rho_{air} C_z A V_x^2 \tag{10}$$

- Longitudinal load transfer:

$$F_{z,long} = \frac{mass * a_x * h_{CG}}{wheelbase} \tag{11}$$

- Longitudinal load transfer:

$$F_{z,lat} = \frac{mass * a_y * h_{CG}}{track} \tag{12}$$

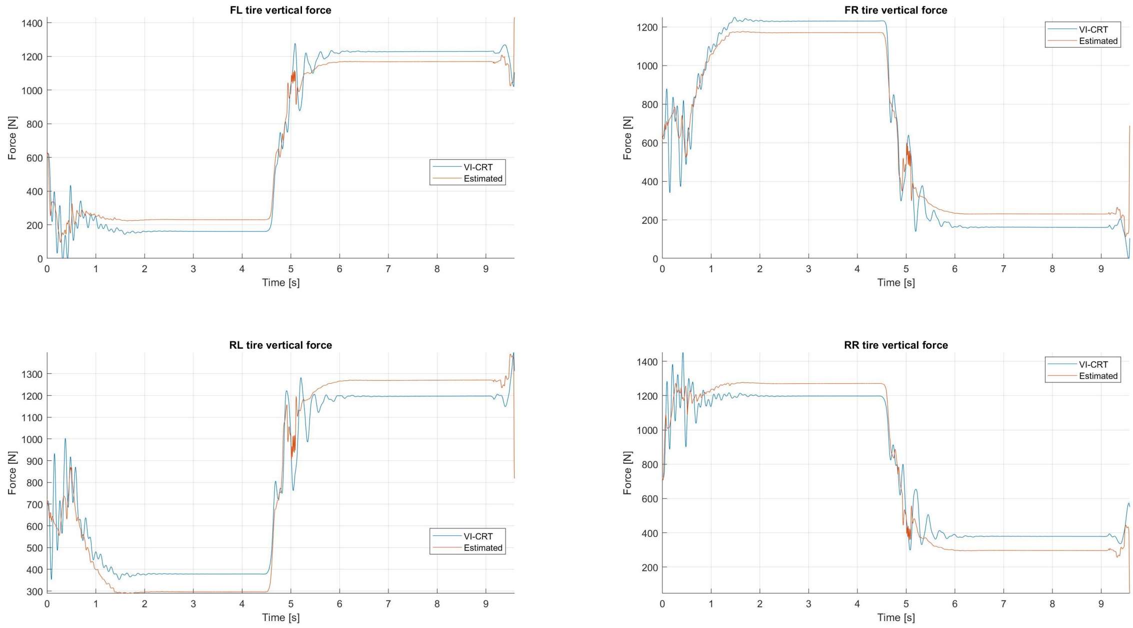

Figure 22: Estimated tires Vertical Forces

Figure 22 shows a comparison between the CarRealTime simulated normal forces and the estimated ones. Even if the trends look correct and the error between the two signals is relatively quite small, it is still relevant and this error will affect also the following estimations.

Tire Contact Patch Forces - Longitudinal Force

As it has already been mentioned, the longitudinal force is caused by the deformation of the tire thread. Hans B. Pacejka introduced in the early 90s several formulas, known as Magic Formulas, to represent the phenomena of the forces developed due to tire and ground interaction. These formulas have been evolved during the years, increasing their complexity and, consequently, their fidelity with the reality. The most common formulation of a Magic Formula is the following[13]:

$$F_{xo} = D_x \sin[C_x \arctan\{B_x\kappa_x - E_x(B_x\kappa_x - \arctan(B_x\kappa_x))\}] + S_{Vx}$$

Figure 23: Tire Longitudinal Force Magic Formula[13]

The Magic Formulas are characterized by many different parameters, that are computed through data fitting techniques of experimental data; an example can be the non-linear least squared method. These parameters are usually collected for industrial application in files called "TIR files". Typical results of the output of the Pacejka formulas are reported in the following figure:

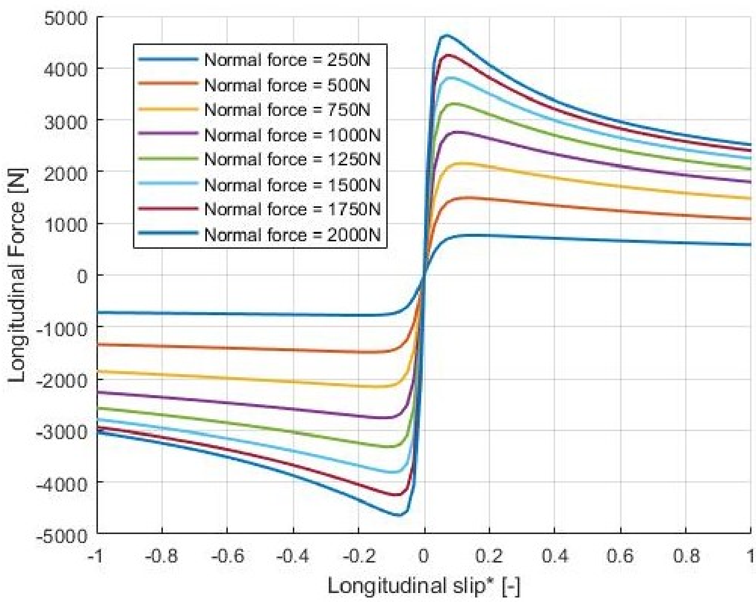

Figure 24: Tire Longitudinal Force

The tire longitudinal force estimation proposed in this work exploits the Magic Formula in its 6.1 version. It takes into account the dependence on the normal force, the tire longitudinal slip, the wheel velocity, the tire pressure and the combined slip.

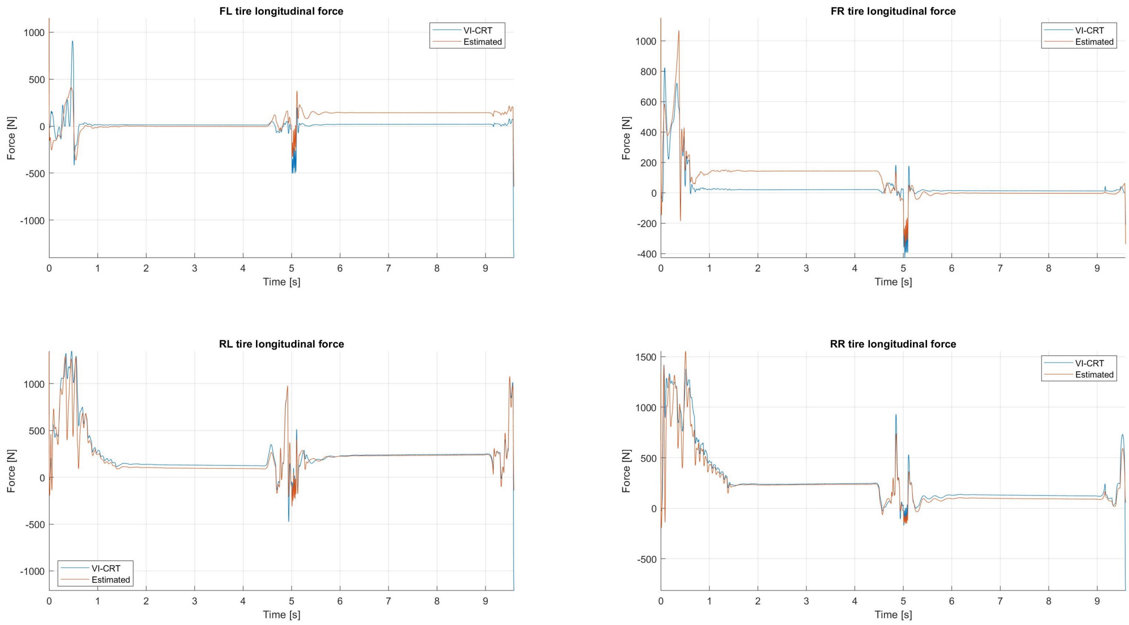

Figure 25: Estimated tires Longitudinal Forces

The previous Figure shows the correlation between the estimated forces and the CarRealTime simulated ones. The error between the two signals is generally small, but in some local cases it becomes quite relevant. As already mentioned, the estimation is affected from the errors already present in the inputs, as for example in the contact patch normal forces.

Tire Contact Patch Forces - Lateral Force

The tire lateral force is the resultant of the stresses caused by the deformation of the tire thread[12].

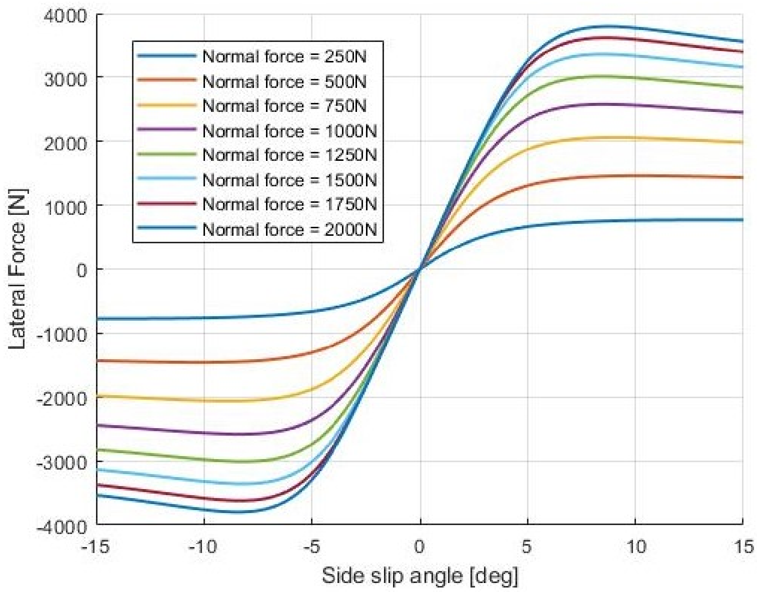

Figure 26: Tire Lateral Force

The approach to the estimation of the lateral force is the same of the longitudinal one. It takes into account the dependence on the normal force, the tire side slip angle, the wheel velocity, the tire pressure and the combined slip.

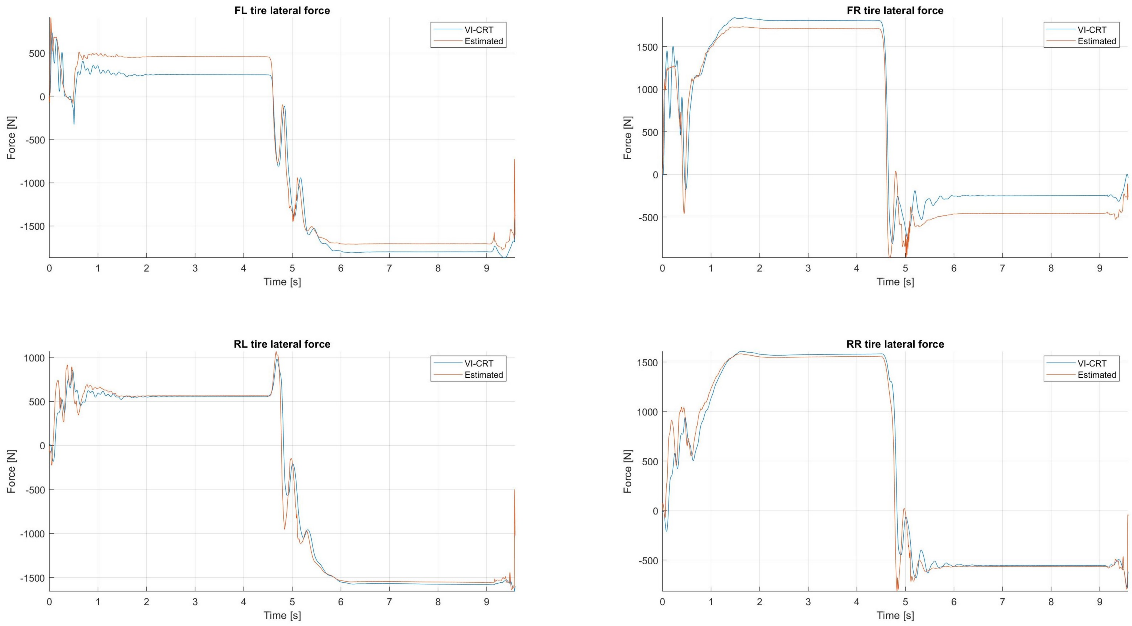

Figure 27: Estimated tires Lateral Forces

The comparison between the two sets of forces signals shows very good results for the estimation of the forces at the rear tires, while it is worse at the front. The fact that the most of the estimation error is at steady state, as it happened also with longitudinal forces estimation, shows the strong dependence of the tire forces on the normal load on the contact patch.

2.3 Slip Control Model

The model-based controllers work exploiting mathematical models to define the desired control action. A possible solution, as already mentioned, for the slip control strategy can be also a model-based controller. In this chapter, it is proposed a model that can be suitable for such application.
The model is based on a starting equation, which is the derivative of the longitudinal slip*. The slip* derivative is mainly dependant on the slip* itself, on the wheel angular acceleration and on the longitudinal acceleration of the vehicle. The former is evaluated with the wheel moment balance, while the latter with a simplified longitudinal dynamics model of the vehicle.

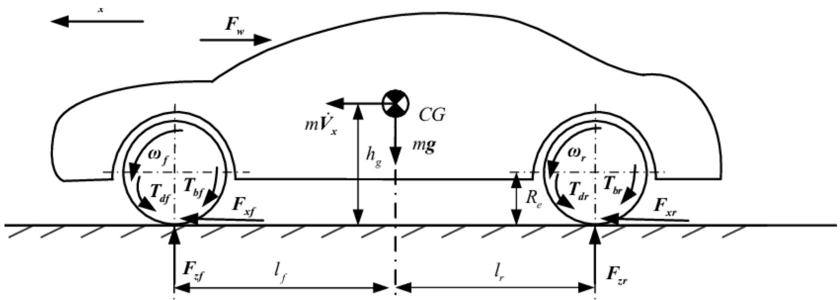

Figure 28: Longitudinal dynamics and wheel moment balance models[15]

The mathematical equations used to describe these models are the following:

- Longitudinal slip* derivative:

$$\tfrac{d}{dt}\sigma(t) = r\tfrac{d}{dt}\omega_w(t) - \tfrac{d}{dt}V_x(t) \tag{13}$$

- Wheel moment balance:

$$\tfrac{d}{dt}\omega_w(t) = \tfrac{1}{J_w}(T_d - T_b - F_x r) \tag{14}$$

- Vehicle longitudinal dynamics:

$$\tfrac{d}{dt}V_x(t) = \tfrac{1}{mass}(\textstyle\sum_i F_{x,i} - F_{x,aero}) \tag{15}$$

$$F_{x,aero} = \tfrac{1}{2}\rho_{air} C_x A V_x^2 \tag{16}$$

The model-based controllers performances strongly depend on the quality of the model, in particular this is related to its capability to represent the physical phenomena in the most accurate way. The accuracy depends on the quality of the data used to build the model and on the complexity of the equations it exploits. It is important to focus, however, on the fact that higher complexity equations can require data that are difficult to obtain with a proper precision and they can also increase the computational load required to solve them.

In order to evaluate the quality of the model and the quality of the estimated data necessary to solve it, several comparison have been performed between the VI-CarRealTime simulations and the model outputs, in the same conditions. The same Skidpad event already discussed in chapter 2.2 is exploited, since these estimations derive from the ones shown there.

Wheel Angular Acceleration

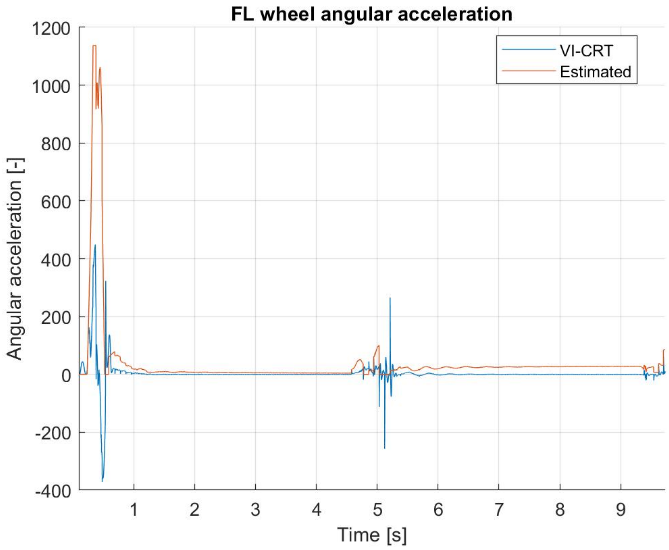

Figure 29: Wheel Angular Acceleration

Figure 29 shows the model behaviour in the estimation of the wheel angular acceleration. The first part of the simulation occurs in transient conditions and it is possible to analyse the huge difference between the two behaviours. In steady state conditions, instead, the correlation is much higher. The reason is dependant on the high sensitivity of this equation on the values of the transmitted ground and the tire longitudinal force. As it has already been shown, the precision with which the longitudinal forces are estimated is not always acceptable. In this equation, this problem is enlarged, since small errors on the forces estimation can lead to very high errors in the angular acceleration estimation. This problem is so evident, that in certain cases the angular acceleration can result negative even if the vehicle speed is growing.

Tire Longitudinal Slip* Derivative

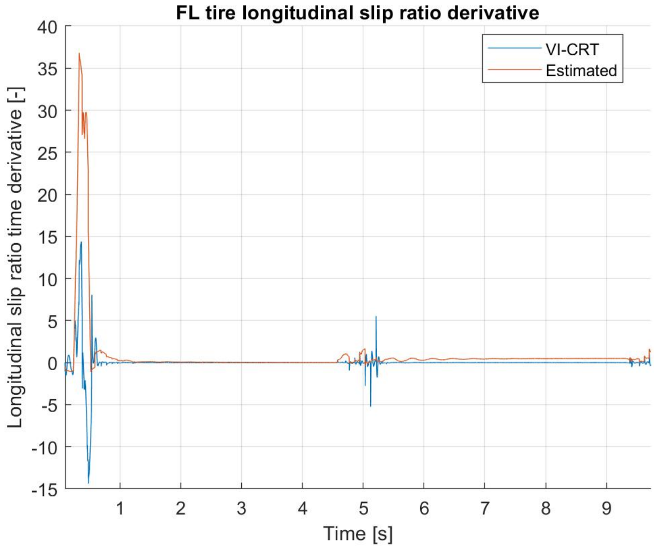

Figure 30: Tire Longitudinal Slip* Ratio Derivative

Figure 30 shows the model behaviour in the estimation of the longitudinal slip* derivative. Also in this case, the problems are the same of the wheel angular acceleration estimation: the transient dynamic is not correctly represented. This is due to the strong dependence of the slip* derivative on the wheel angular acceleration, causing the propagation of the previously discussed error.

The main problem with these estimations is that their worst performances occur in transient conditions. The slip controller, however, works exactly in these conditions, under strong non-linearities of the phenomena. For these reasons and also due to the relevant computational load required for non-linear model-based controllers, it has been preferred to concentrate on different solutions. The Fuzzy Logic controller, as already mentioned, is well suited to be applied on situations in which some difficulties in the model estimations suggest to discard the model-based solutions.

In chapter 3.3, the development of the Fuzzy Logic controller is discussed. The first part works as a framework for the contest in which the FL slip controller is positioned inside the Control System of Squadra Corse PoliTO. A PID slip controller is also proposed as a benchmark, as already discussed.

3 Methodology

This chapter main section consists into the presentation of the methods to develop the FL Slip Controller. Following the V-model approach, it is necessary to introduce the context, at system level, in which the new Slip Controller will operate: the Vehicle Control System. The way in which the torque request for the inverter is produced and the way in which the motors are controller are so presented.

Furthermore, a benchmark controller is briefly introduced: the PID controller. The reasons behind this choice and a short description of its implementation are proposed.

The FL controller development, as mentioned, follows the V-model. The subsystem design, development and tuning procedures are shown in detail. The chapter concludes with the presentation of the final phase, that consists into the validation and the results analysis methods. The validation analysis are performed on the VI-CarRealTime model in comparison with real data from the track. Since the entire process is performed in co-simulation with that model, it is fundamental to validate its result, before the final testing and the final results analysis. The results analysis are useful to show the performances improvement brought by the new controller, in comparison with the non-controlled vehicle and the benchmark controller.

3.1 Vehicle Control System

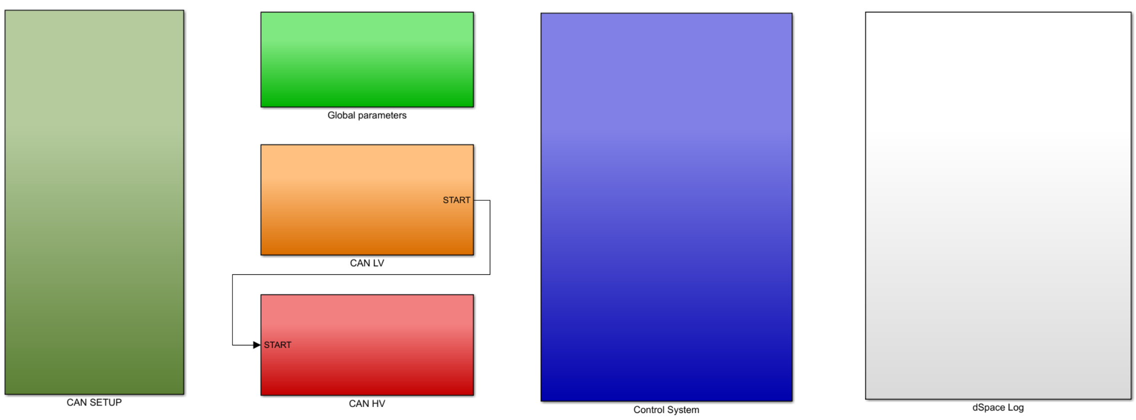

Figure 31: Vehicle Control System

The Vehicle Control System is the responsible for the handling of the information coming from the four on-board Controller Area Networks (CAN). The first one is dedicated to the Low Voltage system communication, in which electronic boards compliant with Formula Student rules and sensors acquisition boards dialogue. The second one is dedicated to the HV BMS. The third and the fourth CAN are dedicated to two inverters each, a couple per each side of the vehicle. The Vehicle Control System is completely developed in Matlab-Simulink environment and it is compiled in C programming language, with dSpace proprietary compiler, to run on the Electronic Control Unit. The Control System works merging the information coming from the four CAN in order to produce the necessary signals to correctly close the High Voltage circuit, with the actuation of the three High Voltage Battery relays, and the signals necessary for the inverters to produce the desired speed control of the motors, depending on the driver requests through the accelerator and brake pedals.

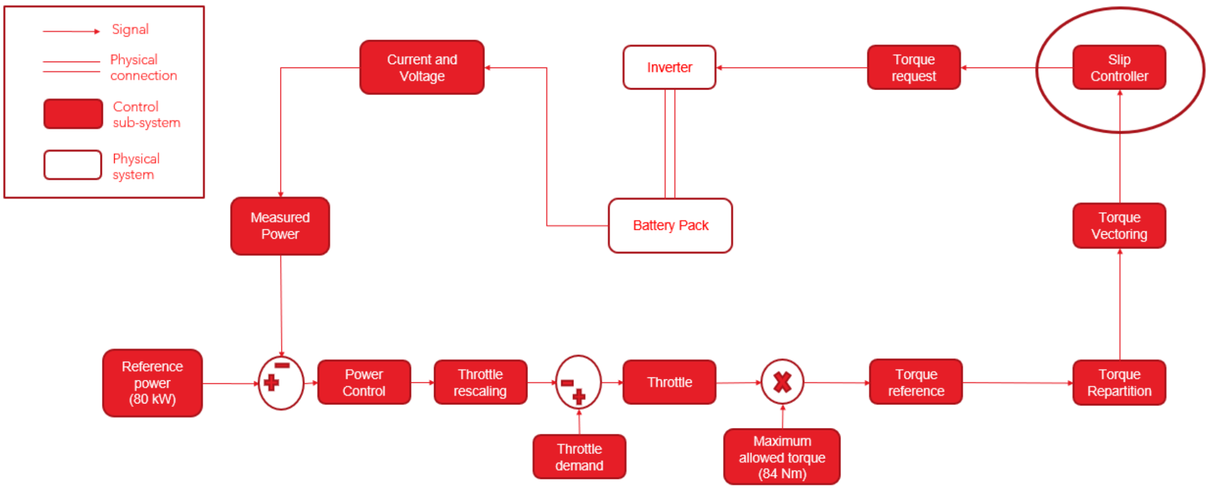

Figure 32: Control System Structure

The Control System most important feature is the torque reference generation, i.e. the production of the torque request for the inverters, starting from the driver's actuation on the accelerator and brake pedals. The driver's demand is translated into a signal called "throttle demand", which represents the percentage (positive for traction and negative for regenerative braking) of accelerator pedal travel or brake pedal pressure (with the maximum levels that are measured during proper bench tests) respectively. The throttle demand is reduced dependently on the Power Control.

The Power Control is a PI controller that receives in feedback the HV battery electric output power. It must reduce the power in order to fulfil the Formula Student regulations of 80 kW of maximum electric power. The limit must not be exceeded by the power signal after a sliding-window moving average filter of 500 ms is applied on it[2]. The controller tuning has been performed with the trial and error method, exploiting VI-CarRealTime and Matlab-Simulink co-simulations. Finer tunings have been performed on real test bench facilities and, finally, on the race track. The final throttle signal, coming from the sum of the throttle demand and the power control signal, rescales the maximum powertrain torque between 0 Nm and the maximum desired torque.

Subsequently, the powertrain total torque request is distributed to the four motors with a front/right repartition. The torque distribution is static in traction conditions, while it is dynamically modified during the regenerative phase, in order to balance the growth of the mechanical braking torque. After that, and only in traction condition, the Torque Vectoring (TV) control automatically modifies both the front/rear and the left/right torque repartitions. The generated torque unbalance is the result of the actuation signal coming from the TV adaptive PI controller, which adapts the tuning parameters of the PI controller dependently on the working condition. The working conditions are defined as combinations of vehicle CoG (Center of Gravity) velocity, longitudinal and lateral acceleration. The goal of the controller is the correction of oversteering and understeering behaviour of the vehicle, by tracking a kinematic neutral steering yaw rate reference. The tuning of the almost 5000 working conditions PI controller parameters have been performed with automatic model-based tuning.

The Slip Controller is the final phase of the Torque Control. Its goal is the reduction of the tire longitudinal slip* when it overcomes the desired threshold, that can be modified dependently on the working condition. It works both in traction condition, i.e. when the tire longitudinal slip* is positive, and in braking condition, i.e. when the tire longitudinal slip* is negative. The chapter 3.3 of this study concentrates on the development, tuning and testing of this control system.
Finally, after the Slip Controller, the final torque request signals are produced (one per each motor), keeping under considerations the limitations due to over-temperatures of the whole powertrain (including also HV battery over-temperature, over-current and voltages protections) and motors field weakening. The torque requests are sent to the inverters, that will be responsible for the speed control of the electric motors.

Motors Speed Control

The motors speed control is simpler to implement with respect to a torque control, since once the imposed speed limit is reached, the inverter autonomously controls the motor torque to maintain the target speed. It is important to specify that the Control System produces a torque request, not a speed request. This is due to the fact that the inverter requires three input signals to control its motor: the motor maximum speed and the minimum and maximum torque that the motor can produce to reach the speed limit. The motor maximum speed is set dependently on the situation (e.g. specific test, dynamic event, etc...) and kept constant if in traction conditions or set to 0 in braking conditions. The motor minimum torque is kept constant at the maximum (in absolute value) regenerative torque that it is desired from the motor. Finally, the maximum torque is exploited as torque request signal: it is the result of the entire Control System.

When the motor is at a different speed from the set limit, the inverter will supply the motor in order to produce the maximum positive torque to accelerate in traction conditions, or the maximum negative torque to brake in regenerative torque mode. With the dynamic modification of the maximum torque limit by the control system, the motor will produced the requested torque in almost every situation, exception made by the already mentioned situation in which the motor is effectively at the maximum speed. In this way, the motors torque will be dependent on the driver requests. This behaviour can be checked with a feedback signal coming from the motor: the excitation current. AMK, the powertrain manufacturer, calls this signal "AMK TorqueCurrent". By dividing the signal for the motor torque constant, which is 0.26, it is possible to record the feedback signal of the real motor output torque.

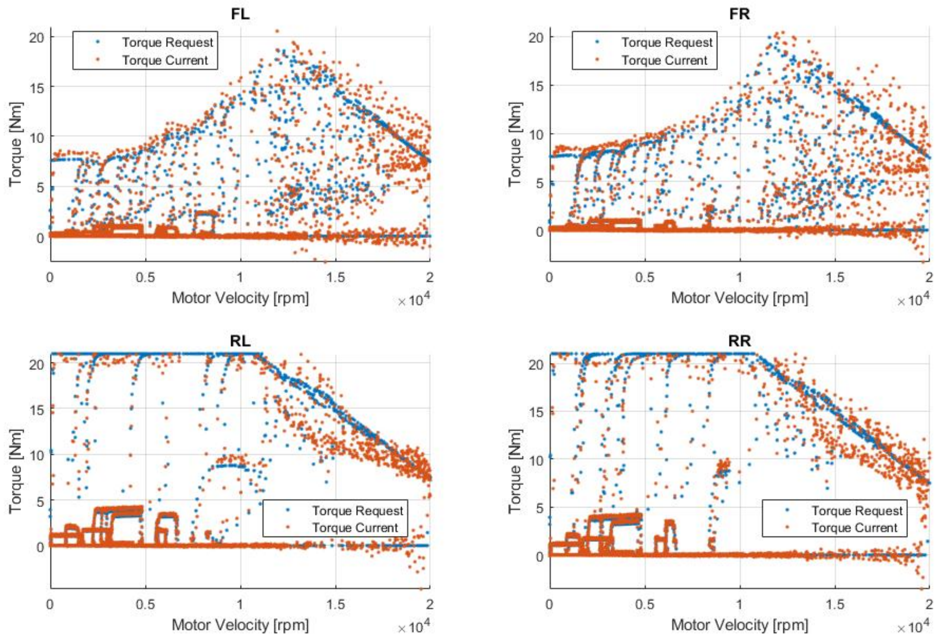

Figure 33: Motors Speed Control

Figure 33 shows how the torque request signal is followed by the motors output torques with a good precision. The blue dots represent the requested torque while the orange dots represent the output torque from the torque current signal; both the signals are plotted in function of the motor speed. The good overlay between the two signals show the correct functioning of the motors speed control, since the motors return what the Control System asks them.

3.2 PID Slip Controller

The PID Controller is the most diffused controller in industrial applications because, even if its simple to implement and tune, it is able to guarantee acceptable control performances in most of the conditions. For these reasons, it has been selected as benchmark solution in this work.
To implement the controller, it has been exploited the Simulink block of the discrete varying PID.

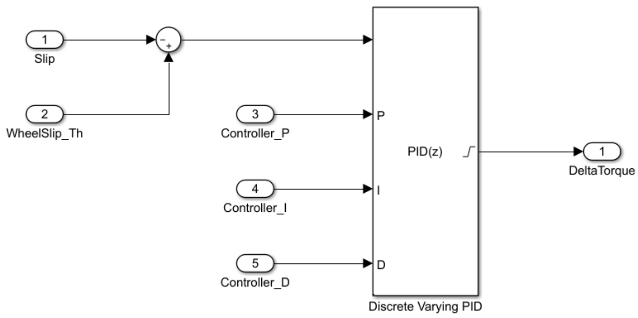

Figure 34: PID Slip Controller

The tuning procedure has been performed with a Simulink model that worked as test bench. The model discussed in chapter 2.3 has been exploited to work as a plant in closed-loop.

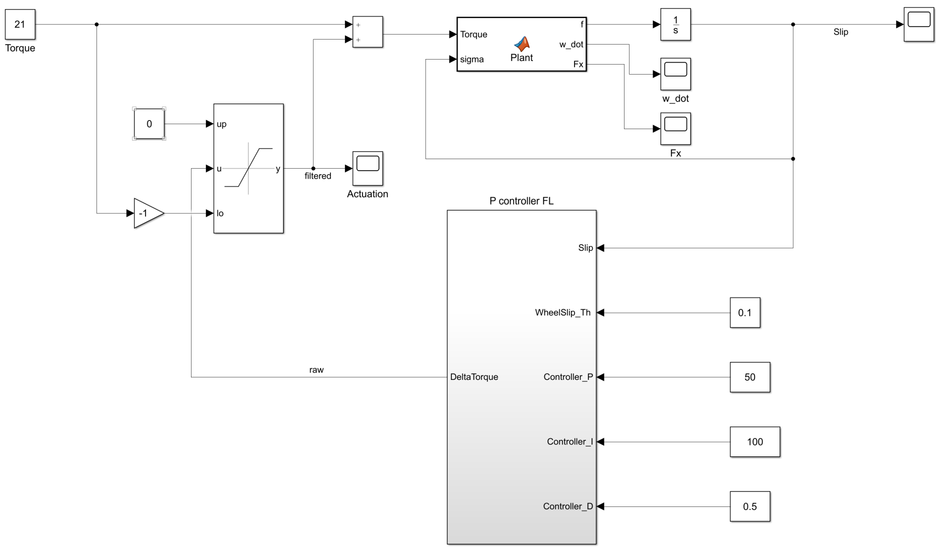

Figure 35: PID Slip Controller Tuning Model

The working condition selected to tune the controller has represented what is considered to be the worst condition for the slip controller: the starting phase

of an Acceleration run, for a front tire. In this condition, full torque is put in input by the motor, due to the driver full throttle request, but the tire has a low vertical force on it due to the longitudinal load transfer caused by the longitudinal acceleration. This causes that the input torque is maximum when the available longitudinal force is minimum and this rapidly causes the tire slippage.
The tuning procedure has been performed with the trial and error method.

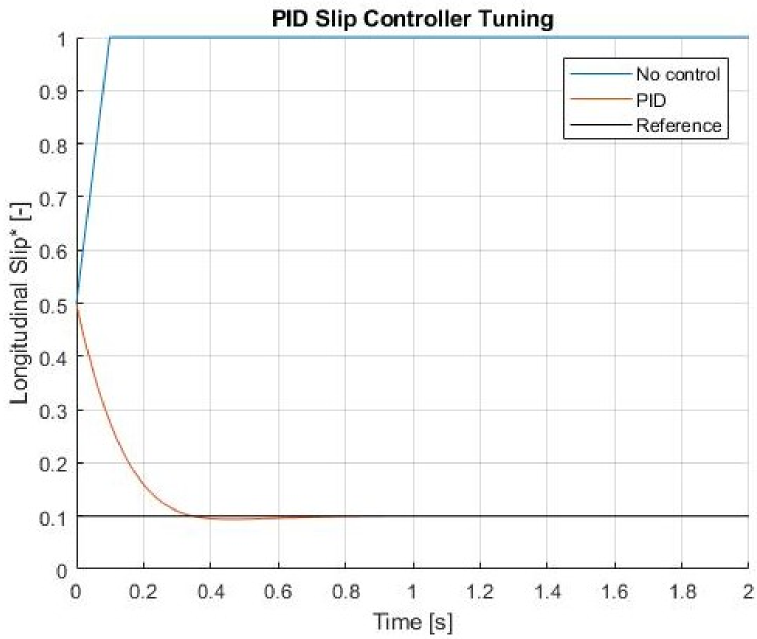

Figure 36: PID Slip Control Tuning Results

Figure 36 shows the PID controller tuning results. It shows how, with respect to the non-controlled situation, in which the longitudinal slip* rapidly increases towards the full slippage, the PID is able to quickly control the slip*, guaranteeing good rise time and low overshoot.

3.3 Fuzzy Logic Slip Controller

3.3.1 Fuzzy Logic Controller Structure

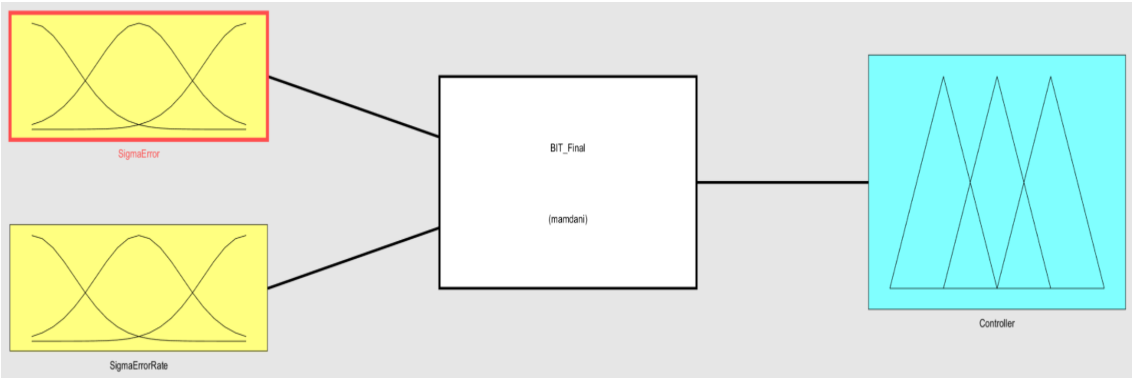

Figure 37: Fuzzy Logic Controller Structure

The structure of the proposed Fuzzy Logic Controller is presented in Figure 3.3.1. The input variables are the slip* error and its derivative, the slip* error rate. The MFs of the slip* error work between 0, no error, and 1, maximum error. The MFs of the slip* error rate work also with negative values, since a negative slip* error represents a situation in which the error is reducing.

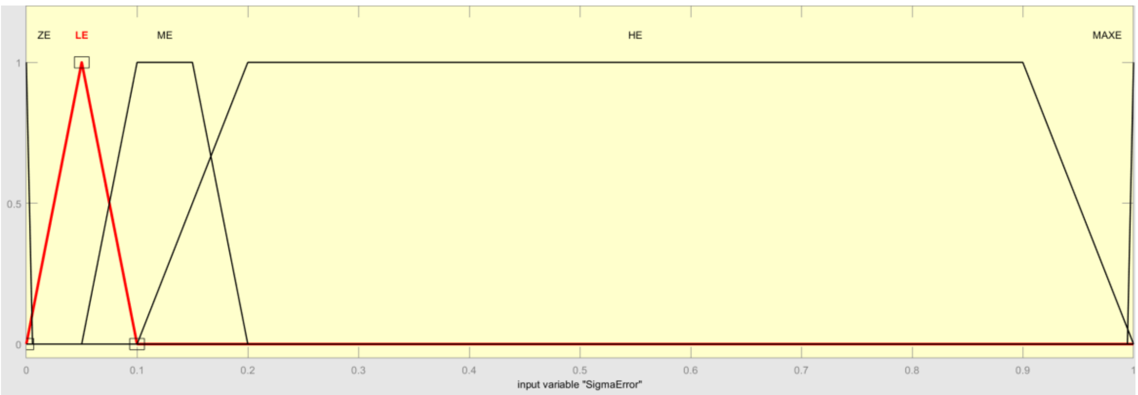

Figure 38: Longitudinal Slip Error Membership Functions

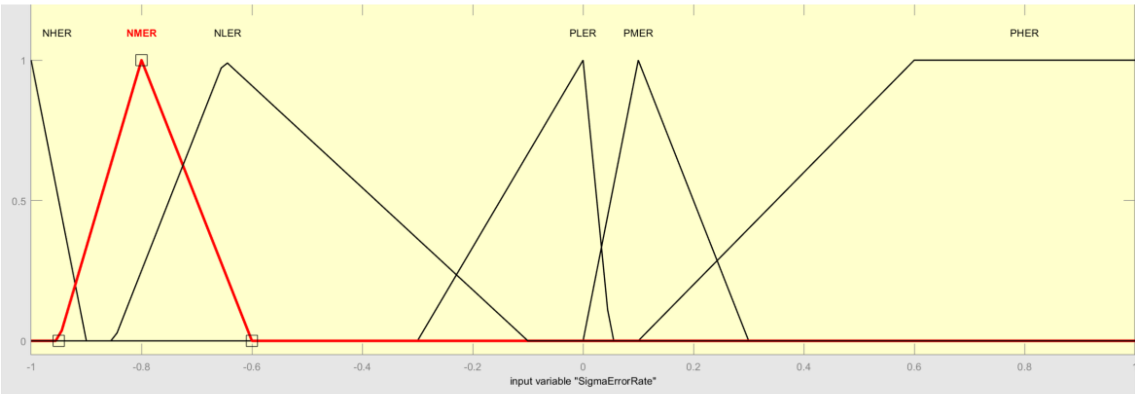

Figure 39: Longitudinal Slip* Error Rate Membership Functions

Four levels of slip* error have been identified, each one represented by a MF: Low Error (LE), Medium Error (ME), High Error (HE) and Maximum Error (MAXE). Six levels, instead, have been defined for the slip* error rate: Negative High Error Rate (NHER), Negative Medium Error Rate (NMER), Negative Low Error Rate (NLER), Positive Low Error Rate (PLER), Positive Medium Error Rate (PMER) and Positive High Error Rate (PHER).

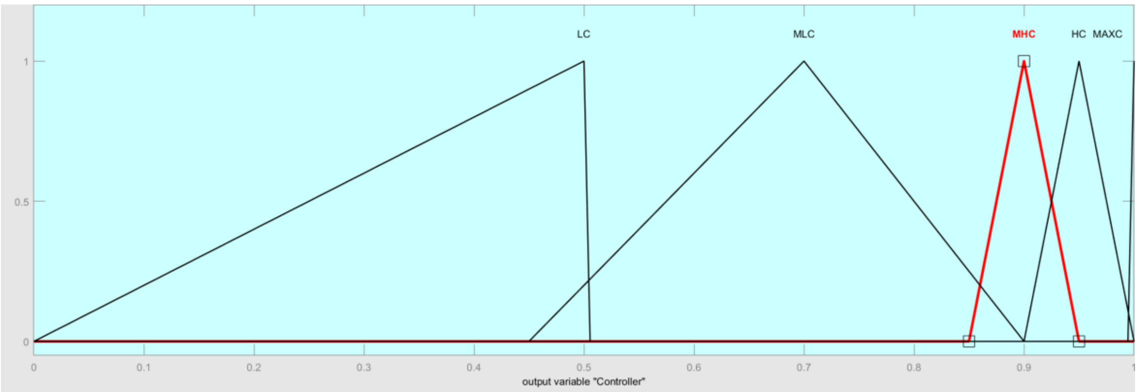

Figure 40: Fuzzy Logic Controller Membership Functions

The controller MFs define the controller action. Also these MFs work between 0 and 1, since the goal of the control action is to reduce the input torque: at most it will be reduced of its entire quantity, with a controller action equal to 1. Five controller MFs have been defined: Low Control (LC), Medium Low Control (MLC), Medium High Control (MHC), High Control (HC) and Maxium Control (MAXC).

Fuzzy Logic Rules						
Slip* Error Rate vs Slip* Error	NHER	NMER	NLER	PLER	PMER	PHER
LE	LC	MLC	MHC	MLC	MHC	HC
ME	M LC	MHC	HC	MHC	HC	HC
HE	MHC	HC	HC	MAXC	MAXC	MAXC
MAXE	HC	HC	MAXC	MAXC	MAXC	MAXC

Table 3: Fuzzy Logic Inference System Table of Rules

The table of rules, as it has already been mentioned, is based on the Mamdani method[9]. To every combination of slip* error MF and slip* error rate MF, a controller MF is linked. The resultant table of rules defines the inference engine that links the inputs and the output: a Low Error that has a Positive High Error Rate, will result in a High Control action.

All the possible combinations between the inputs and the output generate a control surface, in which all the possibilities are represented. The following figure shows an example of a fuzzy inference system:

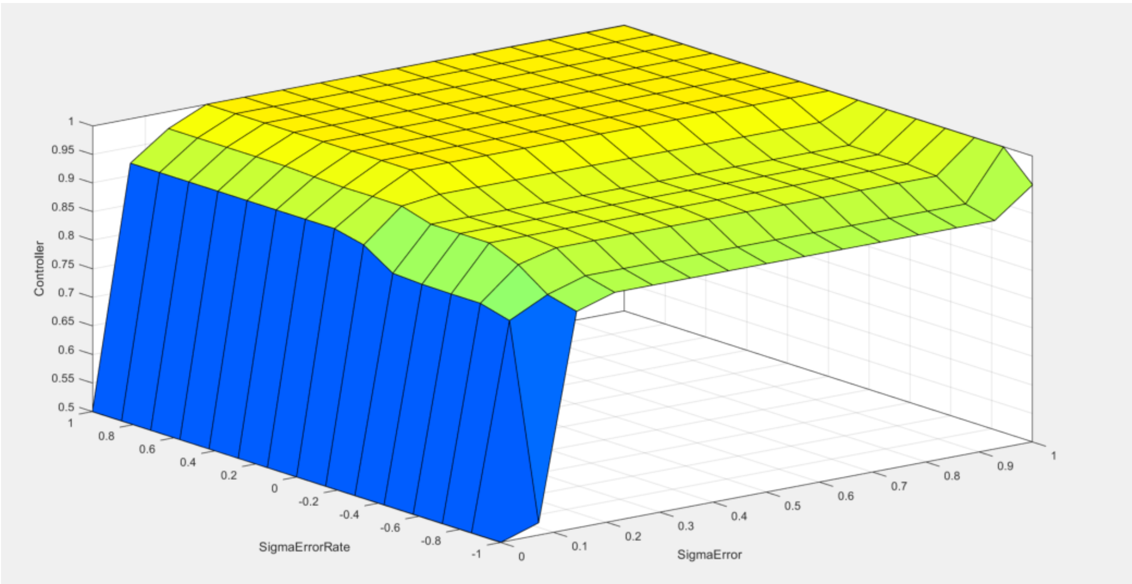

Figure 41: Fuzzy Logic Straight Acceleration Inference System

3.3.2 Fuzzy Logic Controller Tuning

Tuning Procedure

One of the most important goals of this work is the proposal of a tuning procedure for the FL Slip Controller in analysis. Exploiting the already mentioned FL Toolbox of MATLAB, it is possible to manually tune the controller, modifying the shape and width of the different MFs but also to modify the rule that link them. The proposed method is based on the following passages:

1. Definition of a Cost Function
2. Definition of the Standardized Events
3. Controller Optimization

   Minimization of the cost function for:

   - Slip* Error Membership Function
   - Slip* Error Rate Membership Function
   - Fuzzy Inference System Rules
   - Controller Membership Function
4. Slip* Reference Fine Tuning

The order of the MFs tuning has been decided in order of dependence from the other, i.e. the more independent MFs are tuned earlier. For example, the Slip* Error MF is tuned earlier than the Slip* Error Rate MF, due to the dependence of the latter from the former.

Cost Function definition

The cost function is a tool that permits to quantify the performances of the studied controller. The selected cost function for this work is the following:

- Cost:

$$\int_{t_0}^{t_i} (s^*_{actual} - s^*_{reference}) dt \tag{17}$$

The cost function represents the integral of the slip* error with respect to the time, where the slip* reference is modified dependently on the working condition. This cost function has been selected because of its simplicity and effectiveness. The reduction of the integral of the error guarantees the reduction of the overshoot, the steady state error and the oscillations around the reference, since it represents the reduction of the area of the slip* error in function of the time.

In order to reduce as much as possible the dependence on the chosen initial slip* reference, two different slip* references have been used during the tuning procedure. One represents a low slip* condition and the other one a medium slip* condition. The goal was to find similar trends on the cost reduction for both the slip* references, to have good indications on the performance improvement of the controller.

Standardized Events

The Standardized event is a set of maneuvers that must represent the specific working conditions in which the controller will be tuned. The development of the tuning environment for the FL Slip Controller has required a different event for everyone of the four working conditions in which the controller is studied. In order to achieve this result, four VI-CarRealTime standardized maneuvers have been exploited. Every maneuver definition has required an iterative process to identify the proper boundary conditions.

Straight Acceleration

- Initial speed: 5 m/s
- Final speed: not defined
- End time: 2.5 s

The tuning of the Straight Acceleration Fuzzy Inference System has been performed in a straight acceleration simulation environment. In order to avoid the speed estimation errors, which can lead to slip* estimation errors, the initial velocity has been set to 5 m/s. The simulation termination condition has been decided to be the simulation time of 2.5 seconds. The simulation duration has been decided in order to stop it after the return of the longitudinal slip* of every tire of the non-controlled vehicle, which is the worst case, in the stable region of the curve. After this, no more controller action would be needed in any case.

Acceleration in turn

- Initial speed: 15 m/s
- Final speed: not defined
- End time: 15 s
- Turn radius: 20 m

The tuning of the Acceleration in Turn FIS has been performed in a Constant Radius Cornering simulation environment. The acceleration in turn working condition has been simulated with a constant increasing vehicle longitudinal speed profile. The goal of the mentioned iterative approach has been the identification of the correct combination of initial speed, turn radius and simulation end time in order to stop the simulation after reaching the critical speed due to an oversteering behaviour of the non-controlled vehicle. This is then most critical situation in acceleration in turn conditions.

Straight Braking

- Initial speed: 33.33 m/s
- Brake demand: 70
- Brake demand ramp-up time: 0.2 s
- End time: 3 s

The tuning of the Straight Braking FIS has been performed in a straight braking simulation environment. The choice of the simulation parameters has been performed in order to simulate a realistic braking condition that could lead to the full locking of the four tires. The performed iterations have been necessary to find a proper trade-off between the mechanical and the regenerative braking torques. This is necessary in order to avoid locking conditions due to the mechanical braking, that could not be recovered by the Slip Controller that can only rely on the electric motors torques.

Braking in turn

- Initial speed: 22 m/s
- Target deceleration: 2g
- Turn radius: 30 m
- End time: 2 s m

The tuning of the Braking in Turn FIS has been performed in a braking in turn simulation environment. The choices performed to select the simulation parameters have been done with the same goal of the straight braking ones. Furthermore, particular attention has been payed on the avoidance of too strong cornering instabilities that would have caused the failure of the simulation.

Controller optimization

In this chapter, the analysis is focused on the optimization of every section of the Fuzzy Logic Controller. Every membership function or set of rules is tuned on its own, maintaining the others constant.
The selected optimization method works similarly to a simplified mono-dimensional Pattern Search (PS) algorithm. The multidimensional PS algorithms generally define a starting point and generate different iterations of the tuning parameters, each one at the same distance from the starting point, but along different directions [16]. Once found the best solution, new iterations start with the same logic of the previous phase, but taking the previous solution as starting point. When, after a certain number of simulations, it is not possible to find a better solution with respect to the starting point, the distance between the parameters is reduced and the process is repeated again. The convergence to the local minimum can be found and the distance from it depends on the computational efficiency that it is desired to obtain.

The method exploited for the tuning of each FL MF is much simpler and it is based on the possible modifications that can be performed on it. The tuning starting point, which is represented by the initial shape of the MF, has been defined with symmetric MFs, all with the same dimensions. The only exceptions are the Zero Slip* Error and Max Slip* Error functions which represent unitary values of slip* equal to 0 and 1 respectively.
The iterations has been performed with two sequential modalities. The first one consists into the horizontal displacement of the MFs, without the modification of MFs shapes or relative distances. Moving the functions towards the right hand side leads to a decrease of the controller activity, since higher values of slip* will belong to lower severity MFs. The opposite happens if the MFs are moved to the left. After no more horizontal shifting is possible, due to the reach of the borders, the second modality is exploited. It consist into the modification of the the relative distance between the MFs and their width, in order to push the controller activity even further towards the increase or the reduction.
The order of the iterations has been defined following the simplified mono-dimensional PS approach that has already been mentioned. It consists into three different passages:

1. The first batch of iterations is performed at constant distance from the starting point, with some simulations towards the control action increase and the others in the opposite direction. The iterations that follow the first one of every direction are performed at a constant distance from the previous one.
2. The second batch of iterations is performed along the direction that has shown the best trend in the previous phase. In this phase the horizontal shift is performed at first with constant distance between the MFs until it is possible. When necessary, also a reduction of distance can be exploited.
3. The final batch of simulation is performed by modifying the MFs width and shape to reach the desired convergence.

It is important to focus on the fact that FL is highly dependant on the designer choices and experience, so the acceptability of the convergence to the solution

of the optimization problem depends also on this aspect.
The order, with which the different entities of the FIS have been tuned, starts with the two input membership functions: the Longitudinal Slip* Error one and the Longitudinal Slip* Error Rate one. After these, the focus is put on the set of Rules that links the two inputs with the output, which is the controller action. Finally, the last tuning consists into the optimization of the Controller membership function.

In the next Figures, from Figure 42 to Figure 45, the tuning procedure of the Straight Acceleration FIS is taken as example: every one of the other three FIS has been tuned following the same procedure. In the mentioned plots, the FL controller will be represented with a continuous line, while the PID controller, taken as benchmark, is represented with a dashed line.

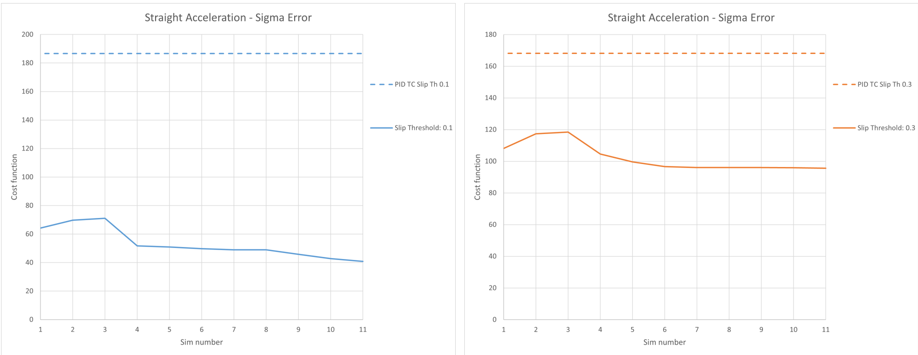

Figure 42: Fuzzy Logic Controller Tuning: Slip* Error in Straight Acceleration

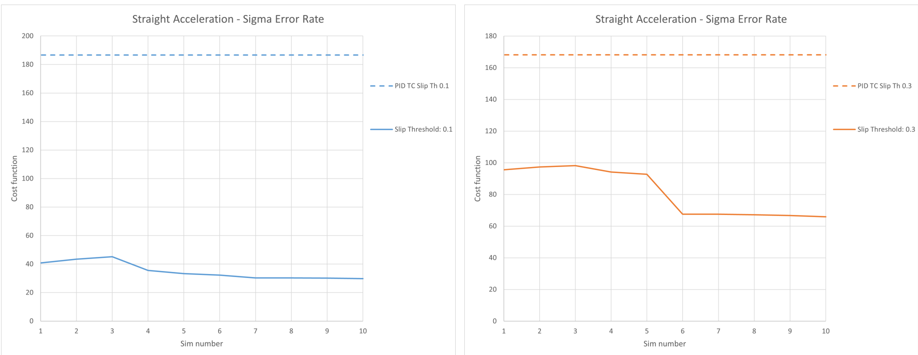

Figure 43: Fuzzy Logic Controller Tuning: Slip* Error Rate in Straight Acceleration

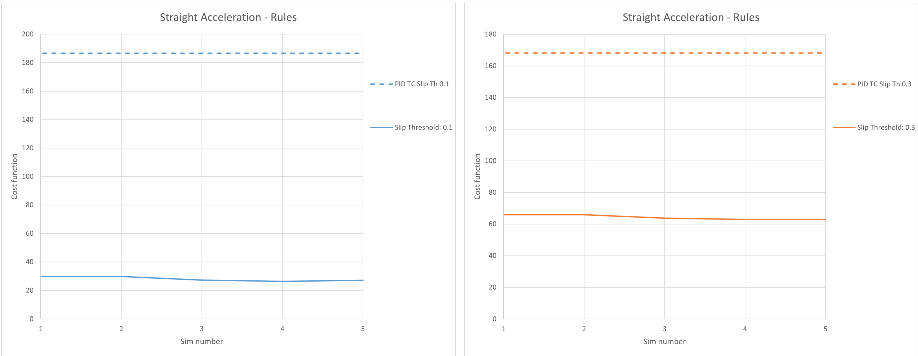

Figure 44: Fuzzy Logic Controller Tuning: Rules in Straight Acceleration

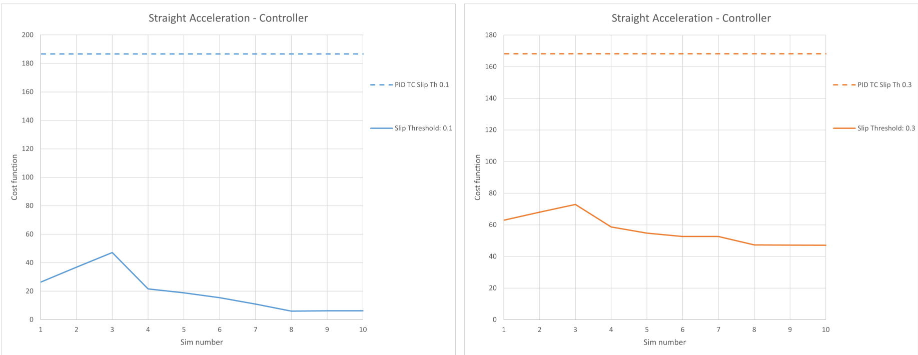

Figure 45: Fuzzy Logic Controller Tuning: Controller in Straight Acceleration

As it has already been mentioned, two slip* reference have been defined to analyse the trends differences. The selected values are 0.1 for the low slip* reference and 0.3 for the medium slip* reference. It is possible to notice how the trends of the cost function reduction are similar for the two different references, in every passage of the tuning process.

The plots also show how the FL controller is already capable of minimizing the cost function from the tuning starting point, with respect to the PID controller. The increase of cost function in the first two iterations is caused by the application of the first phase of the tuning, in which some simulations are performed modifying the MFs towards a controller action reduction. Every set of MFs has shown performances improvement increasing the controller action with respect to the starting point. The complete tuning process caused the reduction of the cost function of an entire order of magnitude.

Slip* Reference Fine Tuning

The final phase of the proposed tuning procedure consists into the tuning of the fine tuning of the slip* reference for every working condition. In this passage, the most relevant tuning parameter is the Key Performance Indicator (KPI) selected for that specific condition. The selection of the KPIs will be explained in detail in chapter 3.4. The cost function is exploited also in this phase, as secondary tuning parameter.

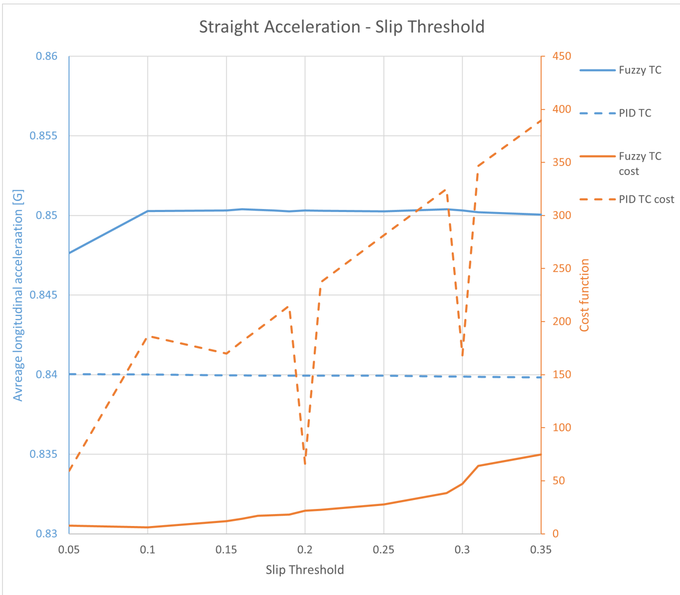

Figure 46: Straight Acceleration Slip* Reference Tuning

Figure 46 shows the fine tuning procedure of the slip* reference of the Straight Acceleration inference system. The blue data refer to the selected KPI for this working condition, which is the average longitudinal acceleration, and the orange data represent the cost function. Also in this plot, the continuous lines describe the FL controller data, while the dashed lines the PID controller. The plot shows how the FL controller is able to guarantee higher average acceleration with respect to the PID one, while always maintaining a lower cost function.

3.4 Results analysis methodology

Model validation with VI-grade VI-CarRealTime simulations

The VI-CarRealTime and Simulink co-simulations are able to produce many output signals. As it has already been shown in chapter 2.2.1, some of those output signals have been used to verify the quality of the introduced estimation. Since also the CarRealTime model is a representation of the real vehicle, it is important to perform correlation studies between the CarRealTime model and the real vehicle data. Only after having obtained a good correlation, the CarRealTime model can be considered reliable for what regards the states estimation and the vehicle dynamic behaviour simulation.
New sets of data have been introduced in order to perform the mentioned correlation analysis. Thanks to the GPS data recorded by the real vehicle, it was possible to reproduce on CarRealTime the same race track layout that the vehicle ran on during track tests or Formula Student events. Two sets of data are shown, to represent different approaches to correlation.
The first analysis consists into an open-loop simulation in which several input signals for the model are taken directly from the raw track data coming from a Skidpad session in FS Alpe Adria 2022. In particular, the input data are: the steering wheel angle, the brake input and the four motors output torques.

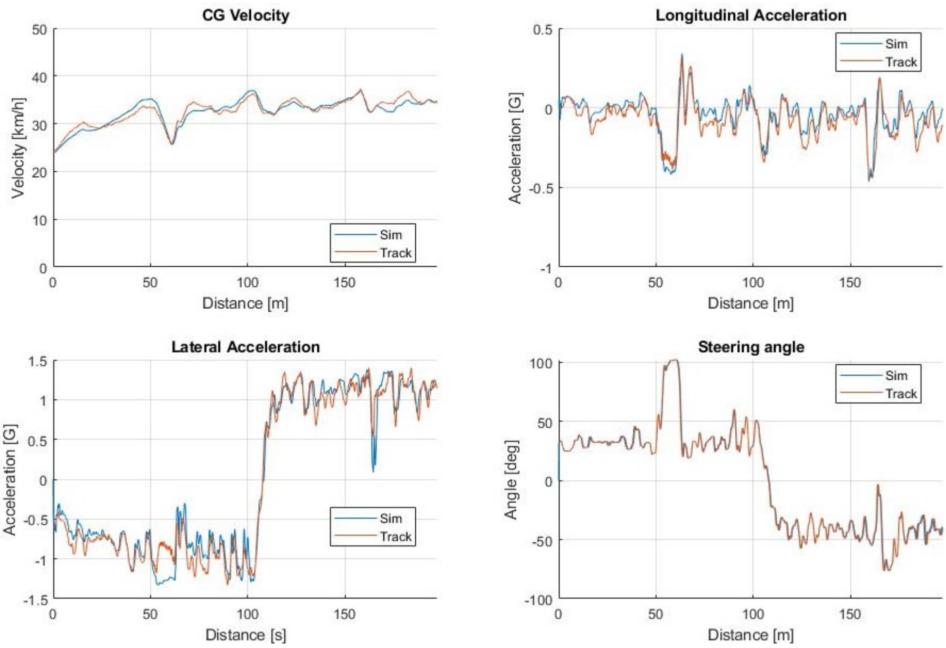

Figure 47: VI-CRT vs Track Data correlation - Open-loop Skidpad

The correlation results quite good, even though there are a few local divergences in the lateral acceleration. They are mainly caused by quite strong correction maneuvers by the driver, which are not perfectly followed by the simulation.

The second method exploits data coming from the Formula ATA 2022 Endurance. The chose data represent the most significant 10 km of the race. The choice of this particular set-up depends on the necessity to perform not only correlation between the model and the race car dynamic behaviour, but also energy consumption and battery discharge profile analysis. Finally, the data taken during a real Formula Student, or Formula SAE in this case, event that perfectly fit the regulations can be considered of an higher quality and interest.

To perform the validation, the same vehicle set-up of the race has been reproduced:

- Masses in racing conditions: 212 + 67 kg (vehicle + driver)
- Suspension characteristic set-up angles in static conditions
- Motors maximum exploited torque: 7 Nm
- Vehicle maximum speed: 60 km/h
- Aerodynamic pack configuration: low drag
- HV battery initial SOC: 88%
- Tires cold pressures

A max performance event was chosen for the simulation. The maximum path distance was selected in order to maintain the simulated vehicle as close as possible to the real driver racing line and the PF have been softly tuned in order to represent a general starting set of parameters. This was done in order to have a more general validity of the analysis, avoiding a too deep customization of the driver model on the specific event.
To perform the vehicle dynamic behaviour correlation analysis, based on the real vehicle available data, it was chosen to select an Endurance Event lap. The lap lateral acceleration profile and speed profile are taken under analysis. Side slip angle data are not reliable enough, since they tend to have strong divergences when the vehicle undergoes strong accelerations. It is important to mention that in everyone of the following figures, the blue data represent the real vehicle track data while the orange ones represent the VI-CarRealTime simulation output.

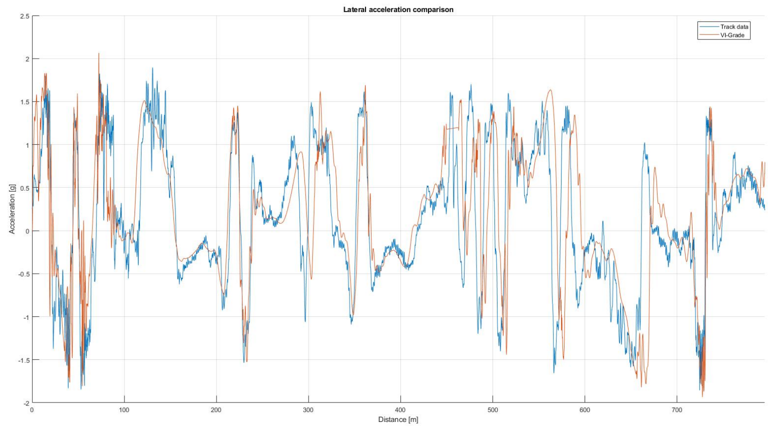

Figure 48: VI-CRT vs Track Data correlation - Lateral Acceleration

Figure 48 shows the correlation between the real vehicle and the model simulation lateral acceleration profile on an Endurance Event lap. It shows how both the signal transient behaviour, in particular the slopes of the signal variations, and its maximum values are well reproduced by the simulation. It is interesting to notice how a real signal has higher ripple with respect to the simulated one. Due to these reasons, the correlation quality is considered to be good. The small differences of the distance position of a few peaks can be due to small racing line differences due to the narrow but still present allowed path distance in the max performance event simulation.

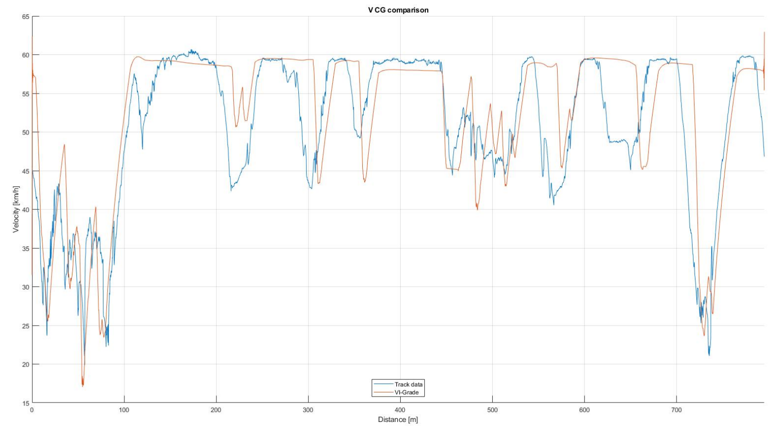

Figure 49: VI-CRT vs Track Data correlation - CoG Velocity

Figure 49 shows the correlation between the real vehicle and the model simulation CoG speed profile on an Endurance Event lap. This plot correlation is slightly worse with respect to the lateral acceleration one. The speed profile representation of the simulation is still quite good, especially the slopes of the speed variations look very similar. However, in some cases the steady state speed is not perfectly overlapping. This can be due to the different lift-and-coast strategy that the real driver and the driver model performed. It must be taken into account that Formula Student drivers are not professionals and their behaviour can be not perfectly repeated on every lap. Finally, also from this plot the obtained correlation is considered to be good.

The following Figures 50-51-52 show the correlation of the Energy consumption model and the High Voltage Battery model. Differently from this chapter previous Figures, they show the behaviour of the signals along the entire data window. It must be clarified how these analysis are highly influenced by the external environment conditions (e.g. external temperature) that are not modelled in the Squadra Corse PoliTO VI-CarRealTime model. Furthermore the real vehicle HV battery SOC is only a rough estimation. In fact, since the cells are not characterized, the SOC is only a cell capacity loss measurement, without any thermal effect or SOH (State of Health) consideration. Finally, the vehicle energy consumption has a very high dependence on the real drive behaviour, which repeatability has already been discussed. With all of these factors taken under consideration, the shown correlations are still considered quite acceptable, but it is difficult to identify their validity.

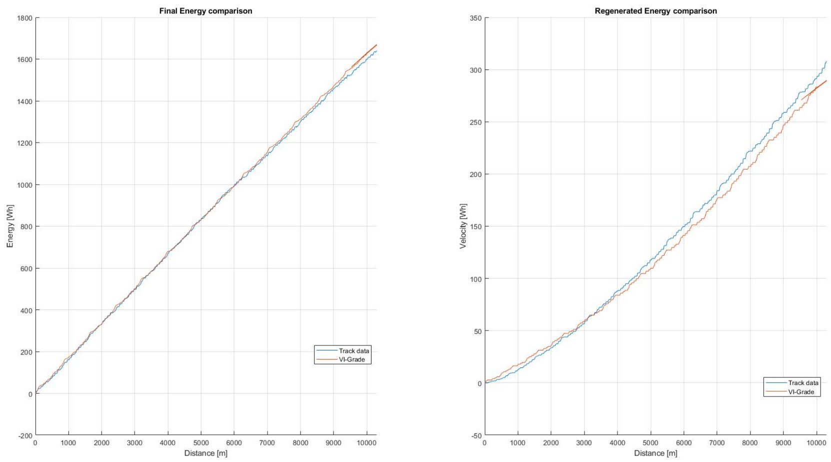

Figure 50: VI-CRT vs Track Data correlation - Energy consumption

The left hand side plot shows the net consumed energy during the Endurance Event, while the right hand side plot is focused only on the regenerated energy.

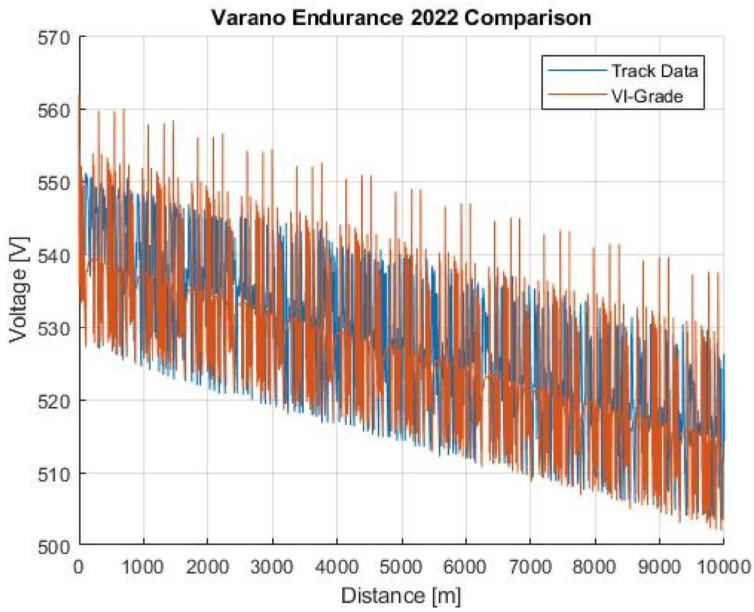

Figure 51: VI-CRT vs Track Data correlation - Battery Pack Voltage

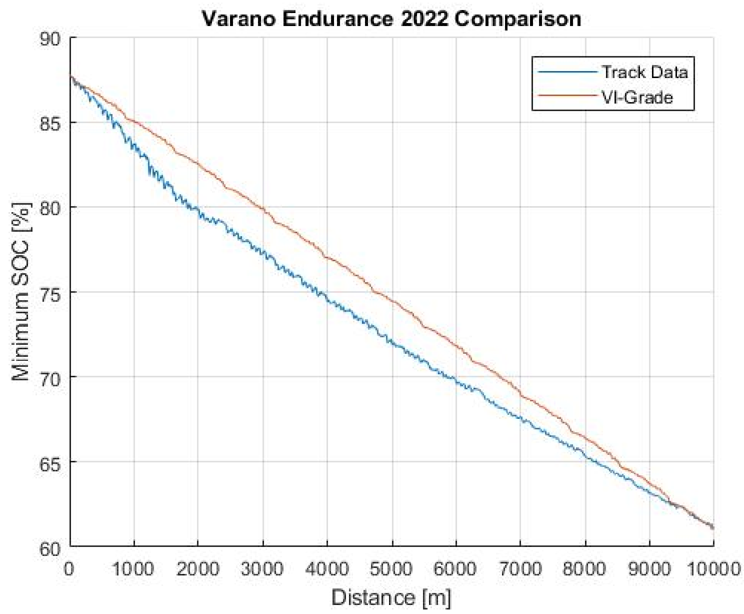

Figure 52: VI-CRT vs Track Data correlation - HV battery pack SOC

KPI analysis

Once the simulation model can be considered validated, it is possible to perform the discussion about the results of those simulations. However, before this discussion, it is necessary to introduce the method with which the performance improvements brought by the different proposed solutions have been measured and compared between them.
The Key Performance Indicators (KPI) are parameters that are used to quantify the results and improvements that a development brings to a project. In this study, the KPIs are used into two ways. The first one consist into the quantification of the quality of the controller fine tuning together with the cost function method. In the second one, the KPIs are exploited as tools for the comparatives performed to analyse the improvements that the new developed controller brings with respect to the benchmark controller and the non-controlled vehicle.

Since the controlled tuning is performed in different working conditions, different KPIs are studied for every situation. Since the main goal of the Slip Controller is the improvement of the torque transmission to ground, the selected physical quantities, thought as better representative, are the average accelerations. In the same conditions and on the same vehicle, higher average accelerations are directly caused by higher transmitted forces.
For what regards the Straight Acceleration and the Straight Braking working conditions, the selected KPI is the developed average longitudinal acceleration. Instead, during the Acceleration in Turn and Braking in Turn conditions, the selected KPI is the combined acceleration. In these last couple of maneuvers also the lateral capabilities of the vehicle are at the center of the analysis. An understeering behaviour of the vehicle would lead to a loss of lateral acceleration, leading to a loss of combined acceleration. An oversteering behaviour, instead, would lead to a slight increase in lateral acceleration, but a strong loss in longitudinal acceleration, still causing an overall loss in combined acceleration. For these reasons, an improvement of combined acceleration would represent not only an increase of transmitted force, but also an improvement of the vehicle balance. In order to show this behaviour, two additional KPIs are selected for the Acceleration in Turn working condition, where this behaviour is straight-forward to show: the side slip angle in function of the CoG velocity and the side slip angle in function of the lateral acceleration. These two KPIs have been selected also to show the stability improvement that the controller cause. The higher stability guarantees lower velocity and lateral acceleration reduction due to a reduced oversteering behaviour.

Finally, the overall behaviour of the vehicle during normal track drive situations is tested. To do so, several Formula Student Events are simulated with Max performance events. The KPIs selected to show an overall improvement of the vehicle performances in the track drive are: the number of simulation iterations and the lap-time. As it has already been mentioned in the introduction, the Iterations are very well representative of the vehicle controllability improvement, due to the fact that the driver model is able to perform the track drive with higher PF, which means that even pushing more the vehicle limits, the vehicle is still capable of maintaining an acceptable distance from the ideal path. It must be clarified that, with different numbers of iterations, which means

different PF, the lap-time comparison is not valid anymore. The vehicle with reduced PF can be slower due to the lower vehicle limits exploitation or faster due to a cleaner racing line, that can be better in some conditions. This makes the lap-time an unsuitable parameter to measure the improvements brought by the only controller modifications.
For these reasons, in order to properly compare different lap-times, two types of simulations are performed. The first typology consist into simulations with equal starting PF for the different tested vehicle set-ups, in order to simply compare the number of iterations. In the second typology, instead, every simulation set of PF is properly tuned in order to have the highest possible parameters values that cause 0 iterations: in this way it is possible to compare the lap-times with every set-up pushed at its limits. In this way, faster lap-times will be caused by higher vehicle limits exploitation due to better control systems behaviour.

KPI Summary	
Working condition	**KPI**
Straight acceleration	Average longitudinal acceleration
Straight braking	Average longitudinal acceleration
Acceleration in turn	Average combined acceleration
Acceleration in turn	Side slip angle vs CoG velocity
Acceleration in turn	Side slip angle vs Lateral acceleration
Braking in turn	Average combined acceleration
Overall	Simulation Iterations
Overall	Lap-time

Table 4: KPI Summary Table

4 Results

The final chapter of this work shows the results of the performances brought by the developed Fuzzy Logic Slip Controller. As explained in chapter 3.4, the KPI analysis have been exploited to evaluate the improvements regarding both the simulations and the application on the real vehicle. Differently from the tuning procedure, in which only one example have been brought to represent the method, the results of every FIS on its relative working condition is going to be shown. Furthermore, analysis of the whole controller in complete lap-time simulations are proposed.
For what regard the real vehicle data, instead, the approach is slightly different. The working conditions are extracted from the data of a couple of laps, selecting only the sections of those data in which the specific working condition is verified. Regarding the straight acceleration, however, also an analysis performed on the FS Acceleration Event is shown, being an easily repeatable environment on which different track tests can be performed with the same input signals and external conditions.
The real vehicle data are extracted from track tests, which are performed in Cerrina Race Track, near Turin. It is a track studied for go-kart racing and, for this reason, it properly represents a possible FS track. In order to simulate even better the real racing conditions, the wider corners and straights are limited by cones.

Finally, a conclusive discussion on the work and the possible future developments is proposed.

4.1 KPI analysis on simulations

The KPI analysis start with the standardized events in which the controller is tuned. These are useful to show the final results of the tuning procedure, in order to compare the behaviour of every FIS separated from the others. With the Acceleration in turn condition, also simple stability analysis are performed. The CarRealTime simulations permit to modify the ground friction coefficient, in order to simulate different grip conditions. All the simulations performed during the tuning phase have been performed with the standard friction coefficient, which is 1. In order to perform the results analysis in different conditions, the friction coefficient has been changed to 0.4, to represent low grip conditions as, for example, wet tarmac.
Finally, in order to test the overall behaviour of the different controls on the general vehicle performances, simulations iterations and lap-time are analysed as explained in chapter 3.4.

Straight Acceleration

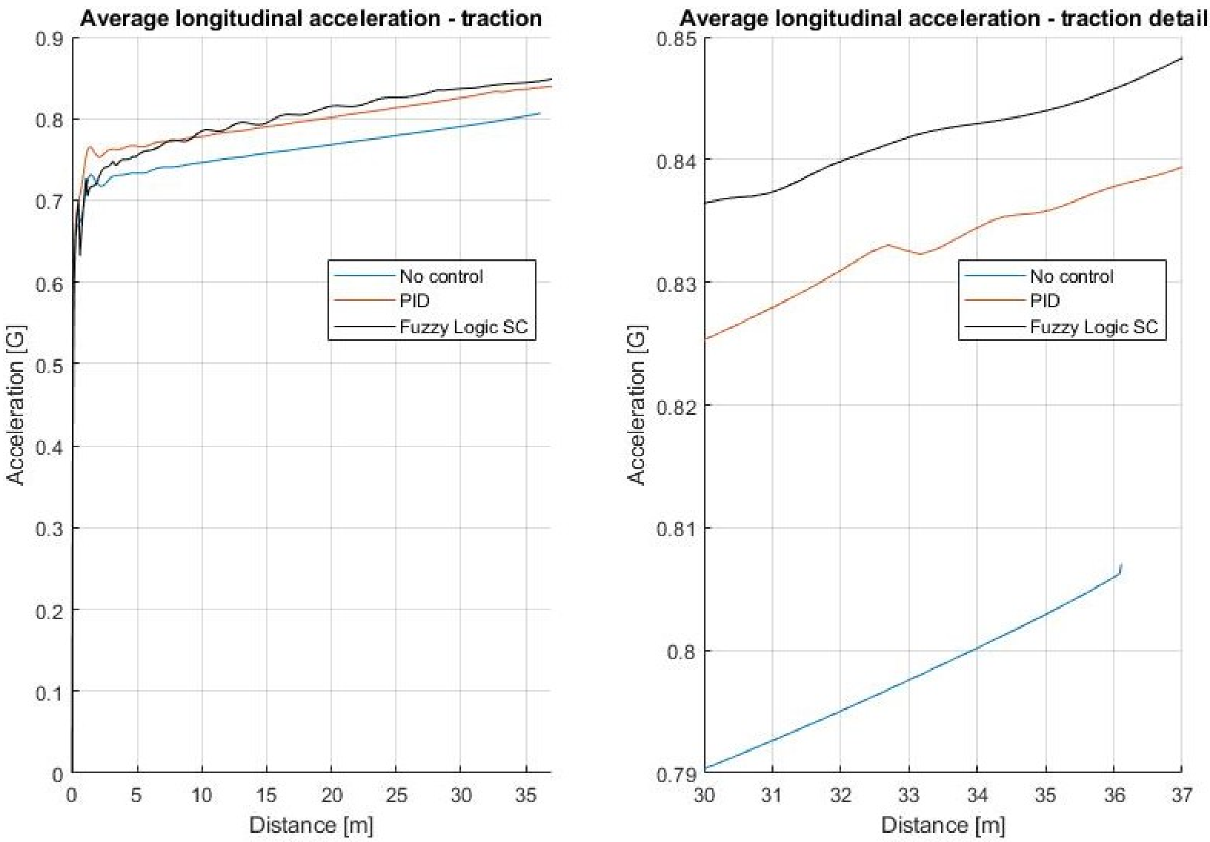

Figure 53: Straight Acceleration KPI

Figure 53 shows the straight acceleration KPI: the average longitudinal acceleration increase. This figure, as the similar ones that follow, shows the entire simulations at the left, while at the right it is present the detail of the final meters of the simulations. Both the controllers are able to improve the performances with respect to the non-controlled vehicle. Furthermore, the FL controller shows better performances than the PID.

Straight Braking

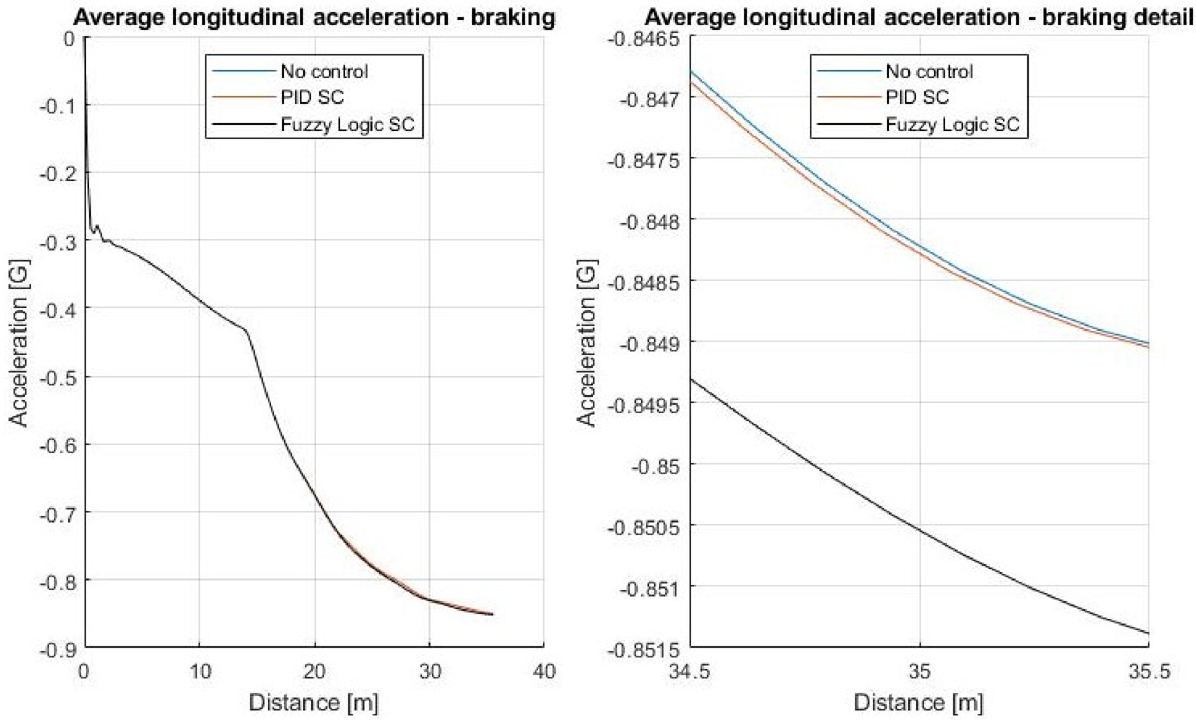

Figure 54: Straight Braking KPI

Figure 54 shows the straight braking KPI: the average negative longitudinal acceleration increase. In this condition, none controller is able to clearly improve the performances. The reason can be found on the mechanical braking torque that becomes predominant as the velocity decreases, reducing the aerodynamic normal load on tires and thus their grip. This causes that the performances improvement that is possible to achieve only acting on the electric motors torques is limited.
Nonetheless, the FL is able to slightly improve the performances, while the PID has basically the same ones of the non-controlled vehicle.

Acceleration in turn

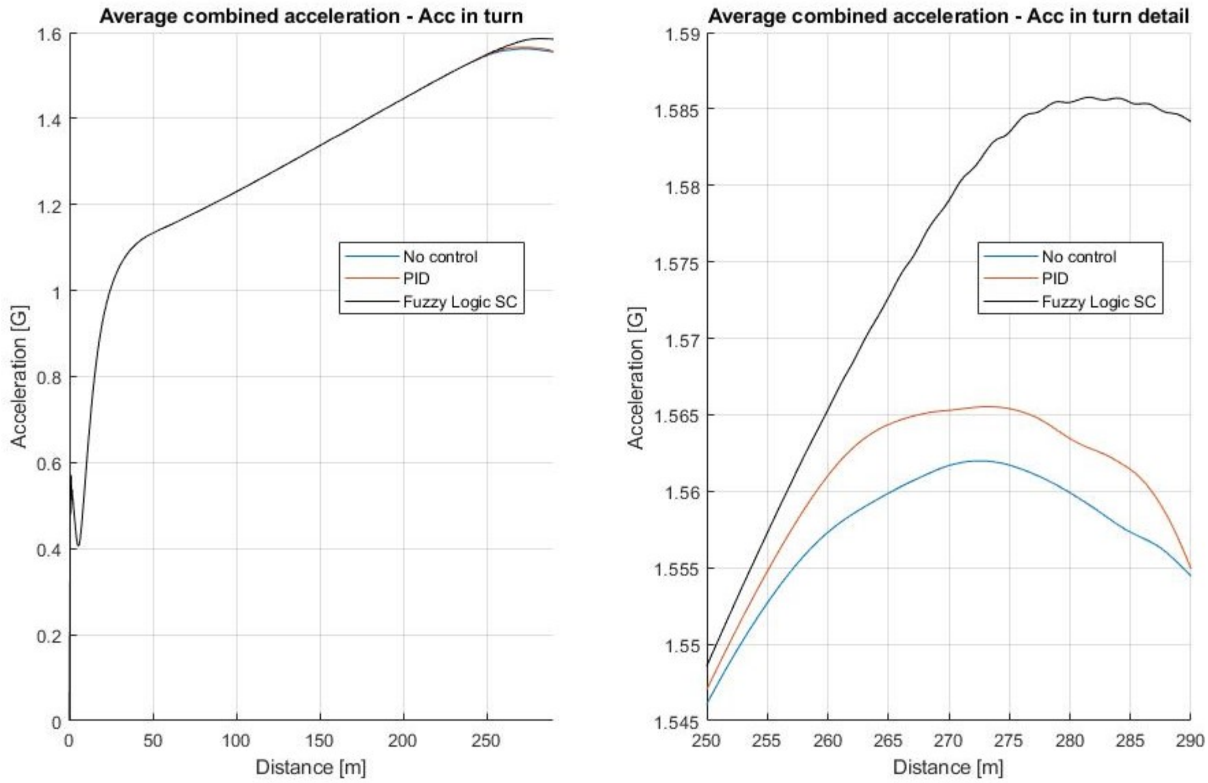

Figure 55: Acceleration in turn KPI - Combined Acceleration

Figure 55 shows the first analysed acceleration in turn KPI: the average combined acceleration increase. In this condition, the PID controller is able to delay the loss of grip by the vehicle, but at the final meters of the simulation its performance rapidly enworse. The FL controller, instead, is able to guarantee higher performances further from both PID controlled vehicle and non-controlled vehicle.
In this simulation, it is important to concentrate on the fact that the differences can be caught only in the final meters. This is due to the progressive increase in speed that brings the vehicle towards its limit conditions only at the end of the simulation.

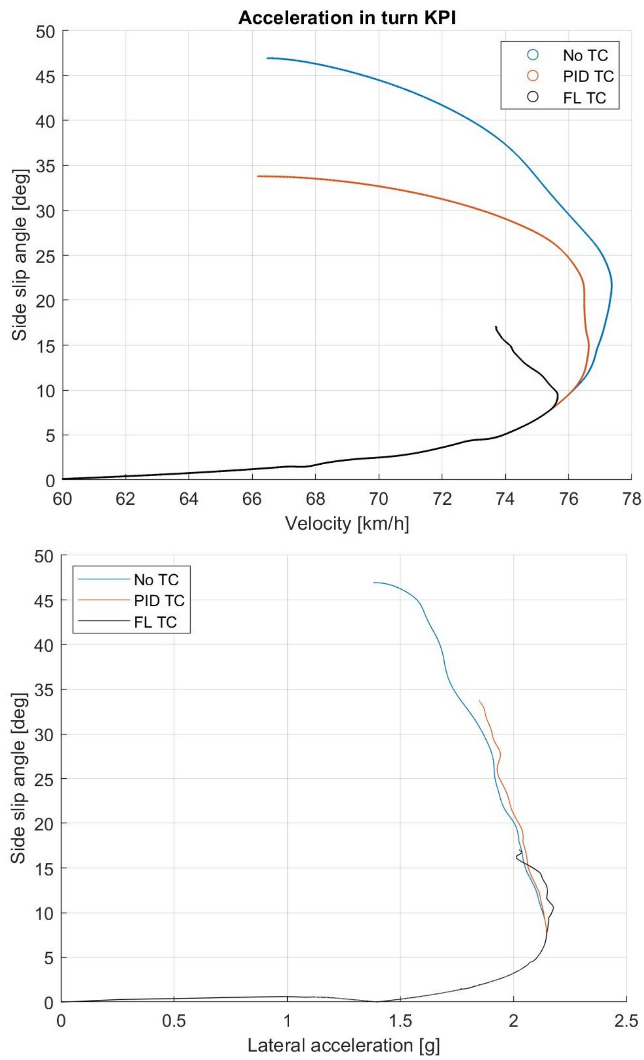

Figure 56: Acceleration in turn KPI - Stability

Figure 56 shows the other two KPIs of the acceleration in turn: vehicle side slip angle in function respectively of vehicle CoG velocity and lateral acceleration. The two plots are truncated in the moment in which the side slip angle would have started to decrease: in this situation the vehicle has recovered the grip, with the reduction of the oversteer. The two slip controllers are able to increase the stability, reducing the side slip angle reached before the grip recovery. The FL controller, in particular, is able to recover the oversteer at a much lower attitude angle: this causes much lower reduction of CoG velocity and lateral acceleration, improving the performances.

Braking in turn

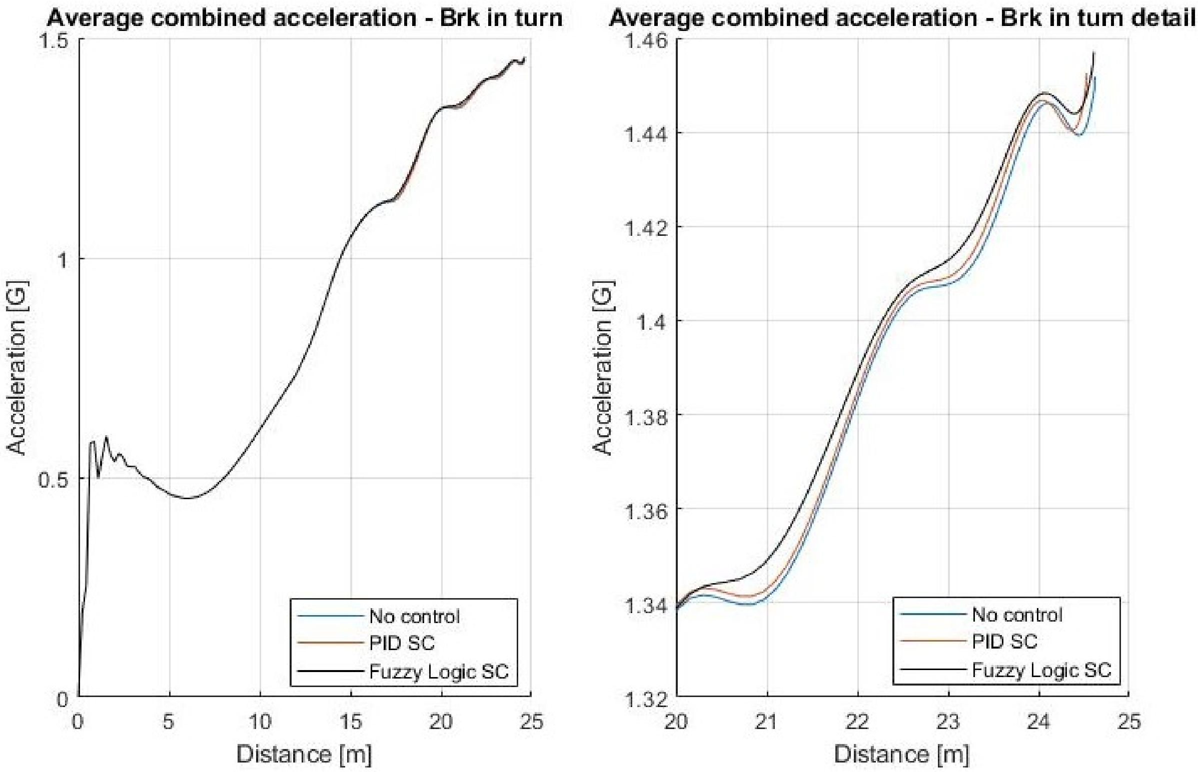

Figure 57: Braking in turn KPI

Figure 57 shows the analysed braking in turn KPI: the average combined acceleration increase. As for the straight braking, the increase of performance in this condition is lower. Furthermore, also in this case, the PID improvement are small with respect to the non-controlled vehicle, while the FL controller ones are a little higher.

Iterations

As already discussed in chapter 3.4, the iterations are caused by the driver model that fails to maintain the vehicle within the maximum allowed distance from the reference trajectory. An higher number of iterations means that a stronger reduction of the PF of the simulation occurred. For this reason, in the exact same simulation conditions, the number of iterations can be exploited as a KPI to show the improvements caused by the different controllers.
The plots that follow will show both lap-time and iterations of each simulation in analysis. As reported by the plots legends, in orange it is represented the simulation lap-time, while in blue the iterations.

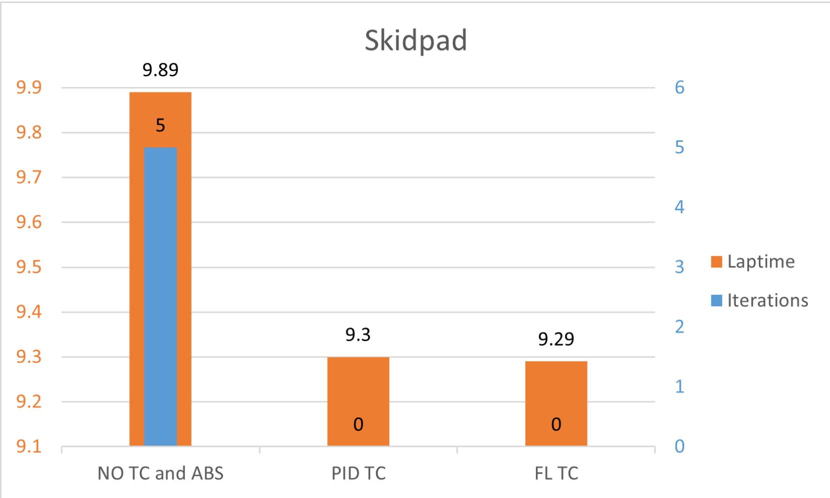

Figure 58: SkidPad KPI - Iterations Comparative

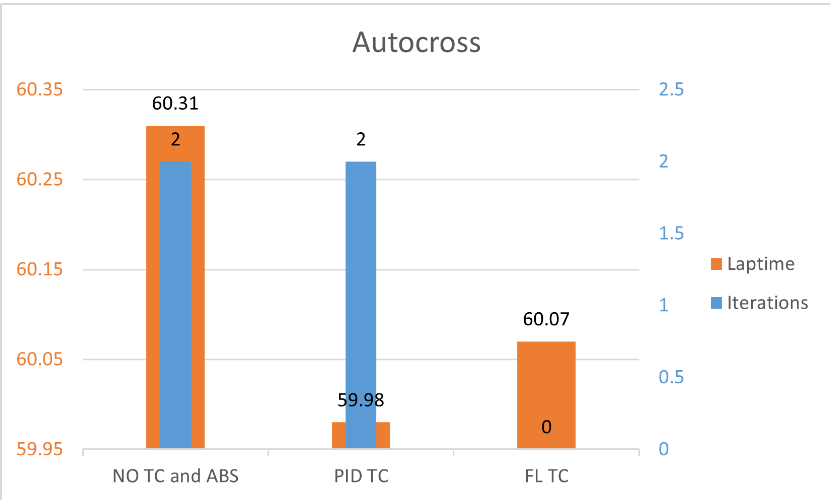

Figure 59: Autocross KPI - Iterations Comparative

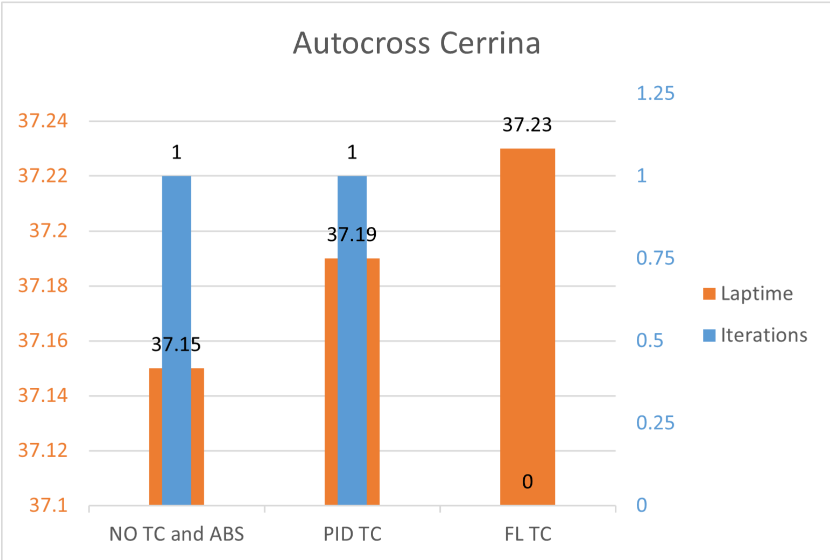

Figure 60: Autocross Cerrina KPI - Iterations Comparative

Figures 58-59-60 show the iterations reduction KPI in different VI-CarRealTime max performance events. The max performance events simulate the correspondent FS Events: a a Skidpad and two Autocross. Two different race tracks have been exploited for the Autocross event: the standard CarRealTime Race track and the Cerrina race track. The latter is built from the GPS data of the race track where Squadra Corse PoliTO performs the track tests on the real vehicle.

The figures show how the FL slip controller reduces the iterations in every simulation, even in those where the PID controller is not able to do so. As already mentioned, it is shown how with different number of iterations, the lap-time comparison loses significance, since the driver model behaviour changes. In particular, wider thus slower trajectories are run by the vehicle, but they become not wide enough to cause an iteration because of the better control. With lower PF, the vehicle limits will not be overcome, so the trajectory will become narrower and faster.

Lap-Time

To perform the lap-time comparatives, a proper approach with the max performance events has been used. In order to bring the vehicle performance, in every set-up, at its limit, the PF of every simulation have been specifically tuned. In particular, every PF is tuned so that an increase of each one of them of a single hundredth, would cause an iteration. In this way, the driver model pushes at the limit the vehicle. The faster lap-time will be the result of the higher PF that the control has made it possible to handle with zero iterations.
In this analysis, also the Acceleration Event is included. This is not a max performance event, for this reason it was not studied during the iterations analysis.

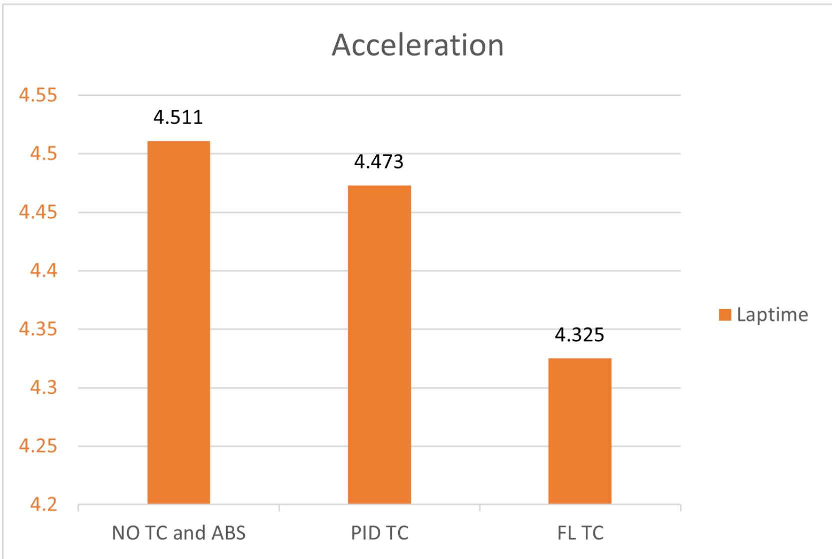

Figure 61: Acceleration KPI - Lap-Time Comparative

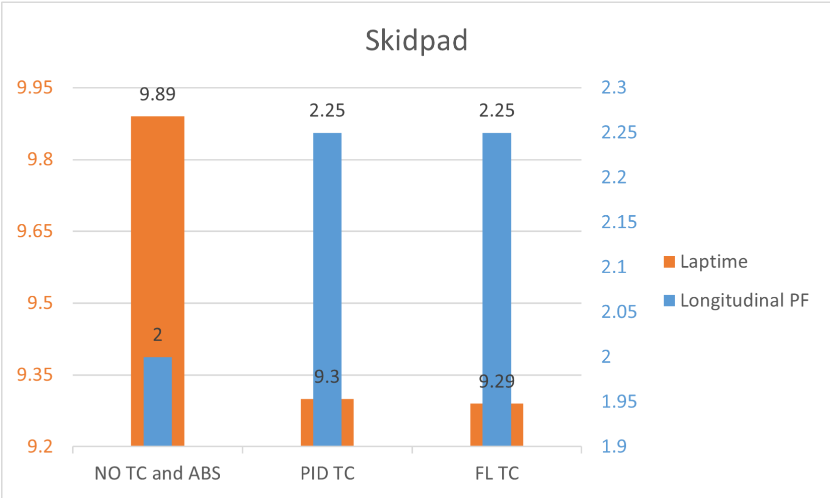

Figure 62: SkidPad KPI - Lap-Time Comparative

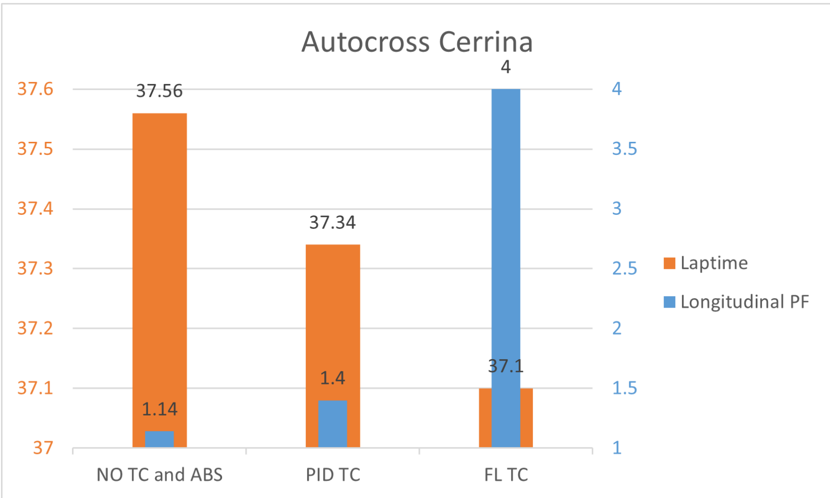

Figure 63: Autocross Cerrina KPI - Lap-Time Comparative

Figures 61-62-63 show the plots of the lap-time KPI. Only the longitudinal PF is represented in the plots, since it is the most involved in the tuning of the traction performances. The braking PF is tuned, instead, to represent a proper racing braking profile.

The PID controller is already capable of lap-time improvement, handling higher longitudinal PF. However, the FL controller shows even better performances, especially in the Acceleration and the Autocross events.

4.2 KPI analysis on Real vehicle application

Two different data sets have been exploited to show the KPIs in the real vehicle data: a set of FS Acceleration Event simulations and a track drive. The latter has been divided into two sections, both with the same driver and track layout. In the first section, the FL Slip Controller was fully functioning, while in the second one it was disabled. Due to the limited time on track and the complexity provided by the swap of two different controllers (much slower with respect to the simple turning on and off of one of it), it has not been possible to implement and test the performances with the PID controller on the real vehicle.
It is important to remember that the driver is not a professional, so the possibility that his confidence with the vehicle set-up and the track layout increased during the session should not be excluded. For these reasons, it has been decided to choose as first section the one with the functioning controller, making it work in worse conditions. The track evolution during the session can be neglected, since the sessions were short and only the Squadra Corse PoliTO's vehicle was present on track.

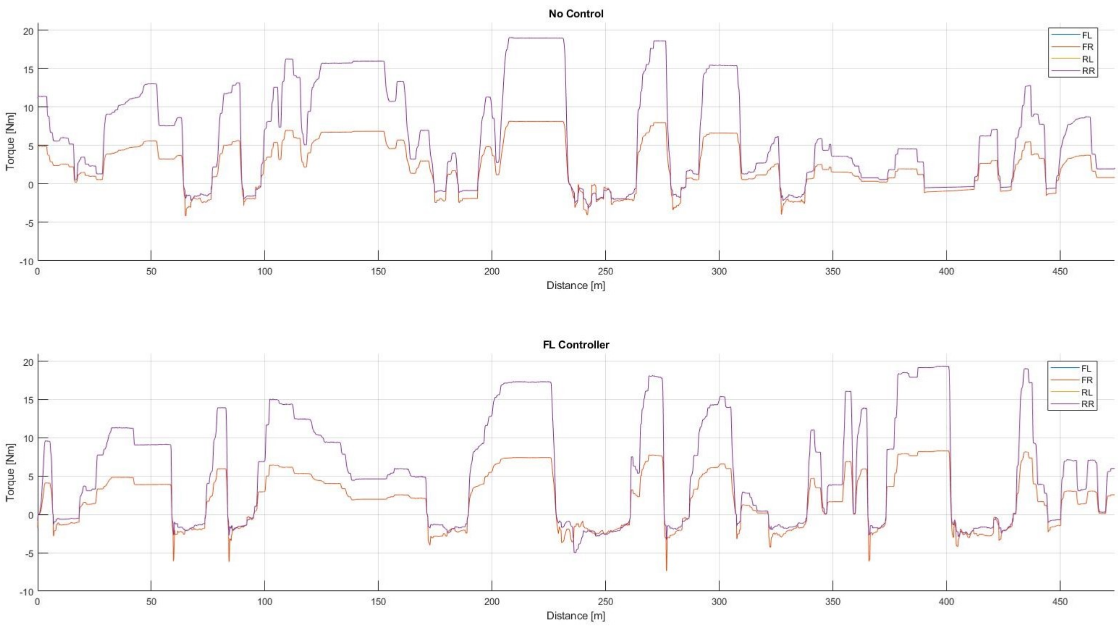

Figure 64: Real vehicle application - Initial Torque Request

In order to perform the comparative analysis, one lap per each session has been selected. The choice criterion consists into the highest similarity between the torque requests of the two sessions, especially for what regards the maximum torque values. This is done in order to have comparable inputs, so that the different output behaviours can better show the influence of the controller.

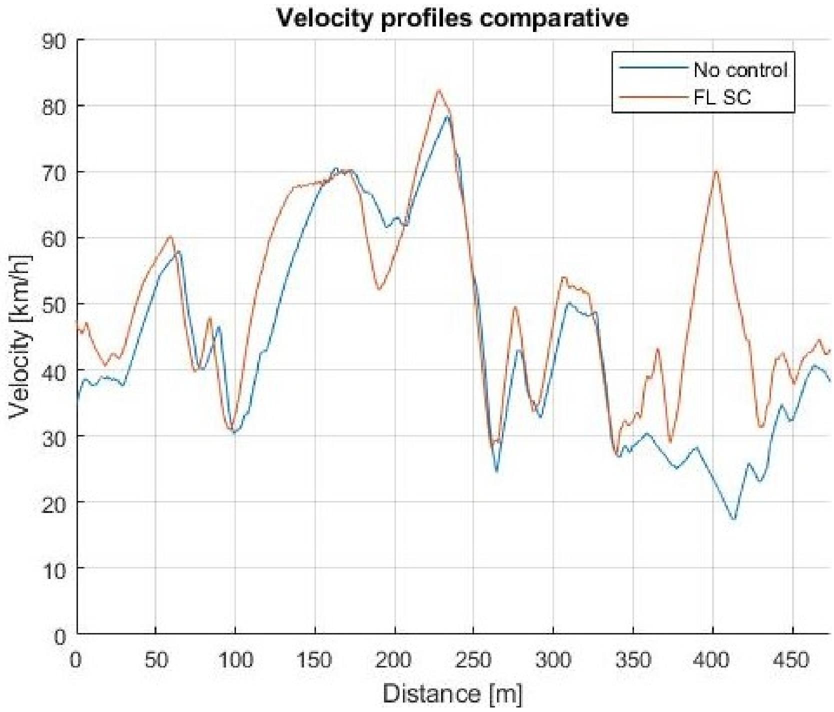

Figure 65: Real vehicle application - Speed Profile

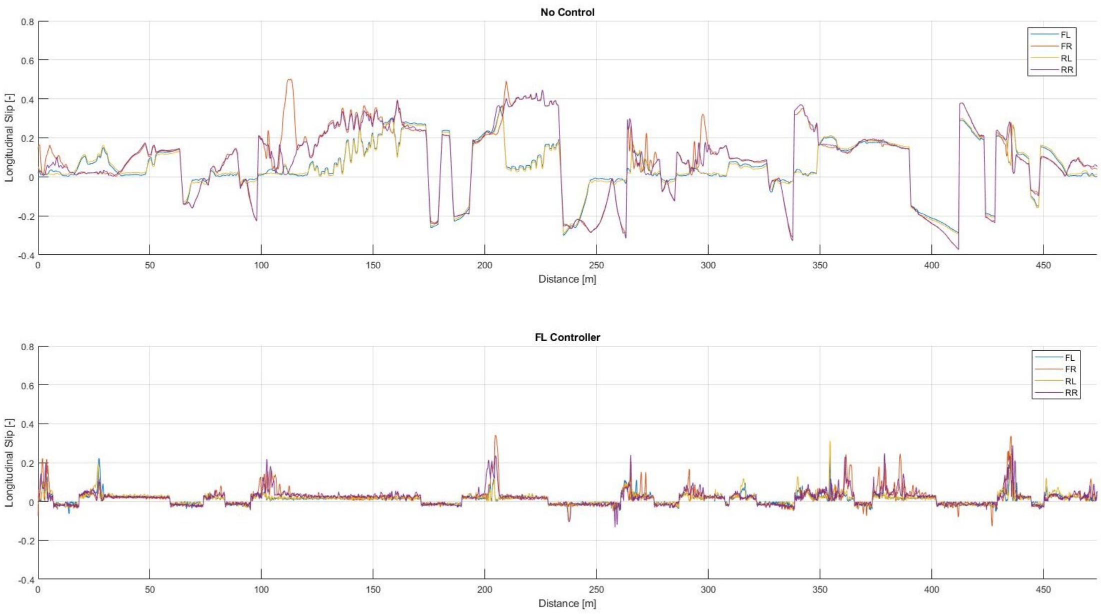

Figure 66: Real vehicle application - Longitudinal Slip*

The first overall results of these analysis are shown by Figures 65 and 66.
In the first one, the two laps speed profiles are compared. It is possible to analyse how the vehicle set-up with the active controller permits higher accelerations, both in traction and braking conditions (steeper speed variations). It is possible to suggest also higher velocities in corners, since the speed profile of the vehicle with the active controller is almost always beneath the other.
The strong difference around the 400 meters cannot be caused only by better capabilities of the controlled vehicle, but probably there is also a driver mistake.

For what regards the second Figure, it shows the comparative between the two output slip* signals. It is possible to analyse how the general trend shows a strong slip* reduction overall and even the signals spikes are smaller and recovered in a faster way.

Straight Acceleration

The straight acceleration working condition is at first analysed alone, since it permits a more controllable environment and a smaller number of variables. The torque request of the two simulations are identical, since basically the accelerator pedal remains flat-out from the beginning to the end of the 75 meters.

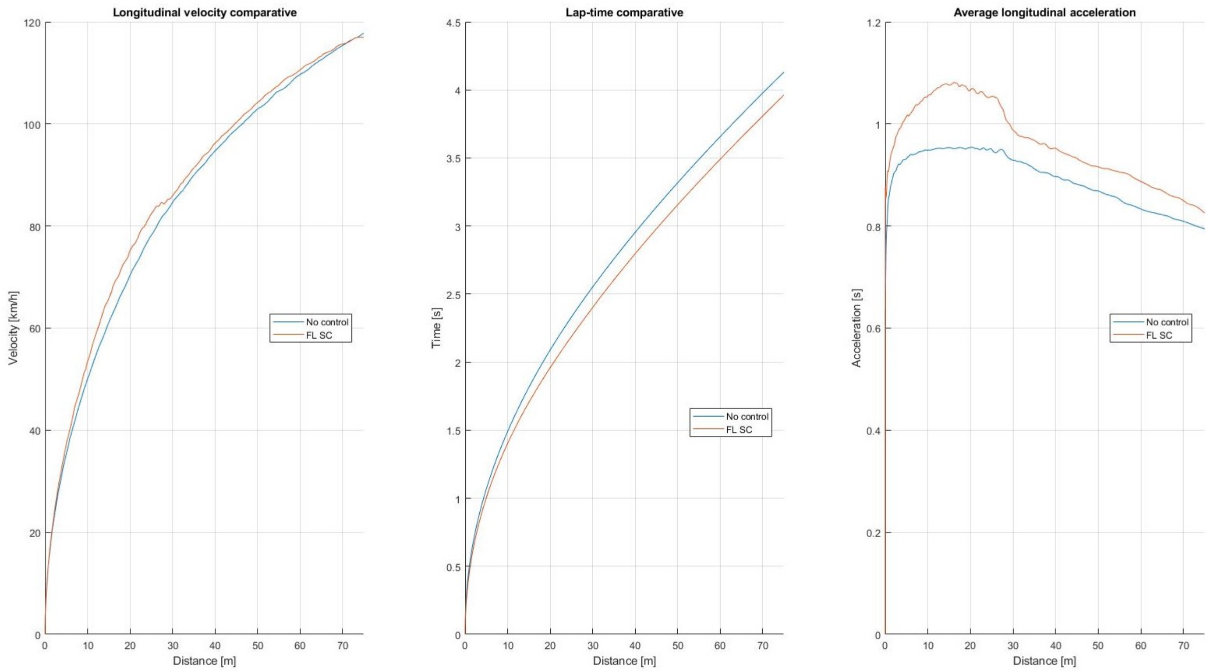

Figure 67: Real vehicle application - Straight Acceleration KPI 1

The left plot of Figure 67 shows how the gain in velocity occurs in the first part of the acceleration, in which the grip limit on the tires is stronger due to the low aerodynamic load at low velocities. It is exactly in these conditions that the Slip Controller can show its capability to increase the transmitted torque to the ground. This is well shown, as already explained, by the higher average acceleration (right hand side plot), that leads to a smaller lap-time: the reduction falls near to the 5%. In order to give a simple term of comparison, with the same forces transmitted to the ground, a reduction of 5% of lap-time would have required a reduction of 26.8 kg of vehicle mass.

Regarding the other data set, a different approach with respect to the previously shown has been used. In order to separate the single working condition, the average accelerations have been computed only in the sections of the lap in which the respective condition was fulfilled. In all the points that are outside the working condition, the computation of the average acceleration does not occur and the signal is kept constant.
Every working condition is shown with a GG plot, in which the vehicle longitudinal acceleration is depicted in function of the lateral one, both measured in G (for this reason the plot is called GG).
In all the following plots, the non-controlled vehicle data are represented in blue, while the FL controller ones in orange. A general overview of the acceleration profiles during the laps is also reported in comparative.

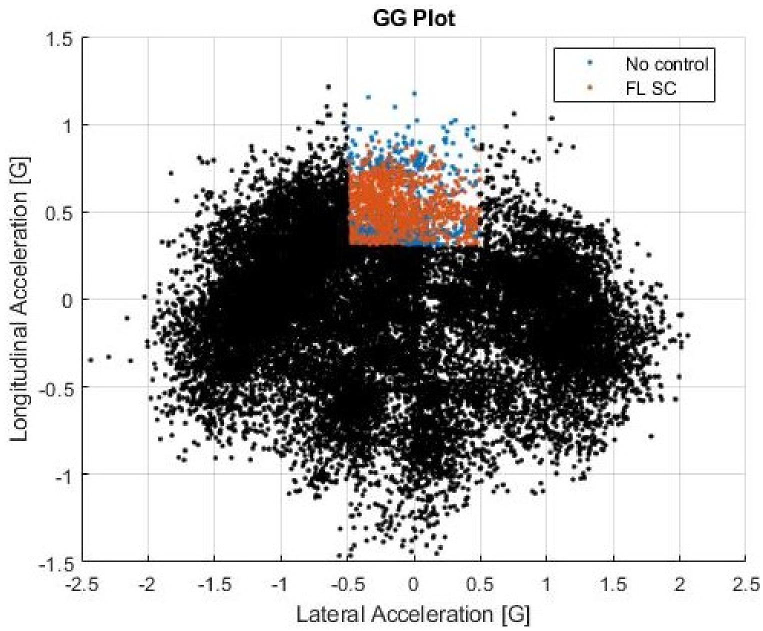

Figure 68: Real vehicle application - Straight Acceleration GG Plot

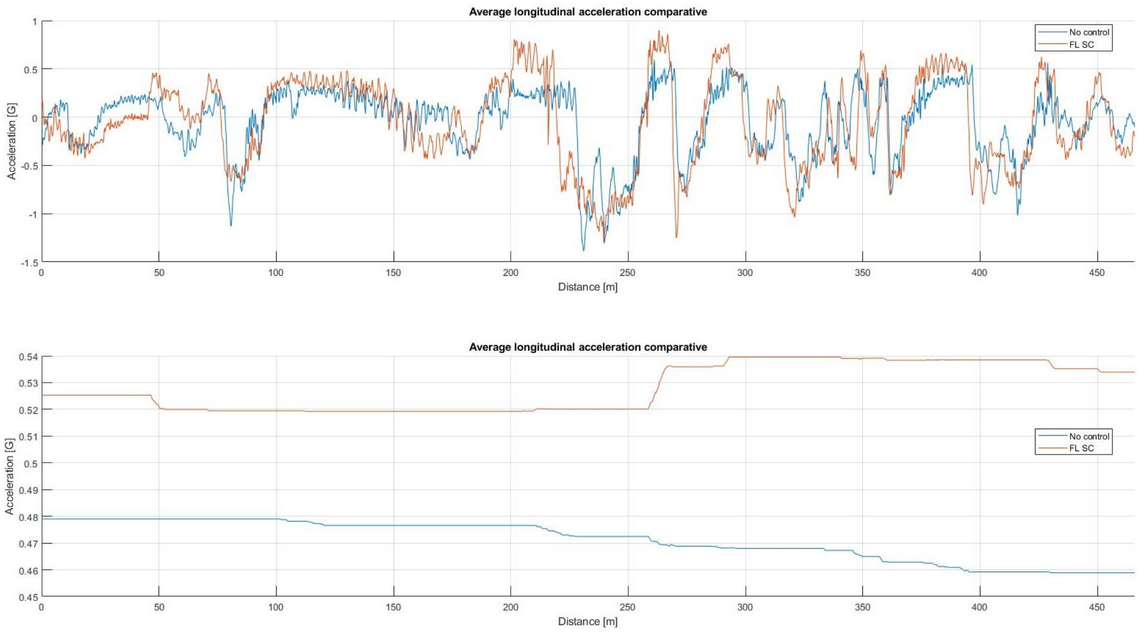

Figure 69: Real vehicle application - Straight Acceleration KPI 2

Also in this analysis, it is possible to observe how the performances of the vehicle improve in the straight acceleration working condition, because of the FL controller. In fact, the signal of the average longitudinal acceleration of the vehicle with the controller active is always above the other set-up signal.

Straight Braking

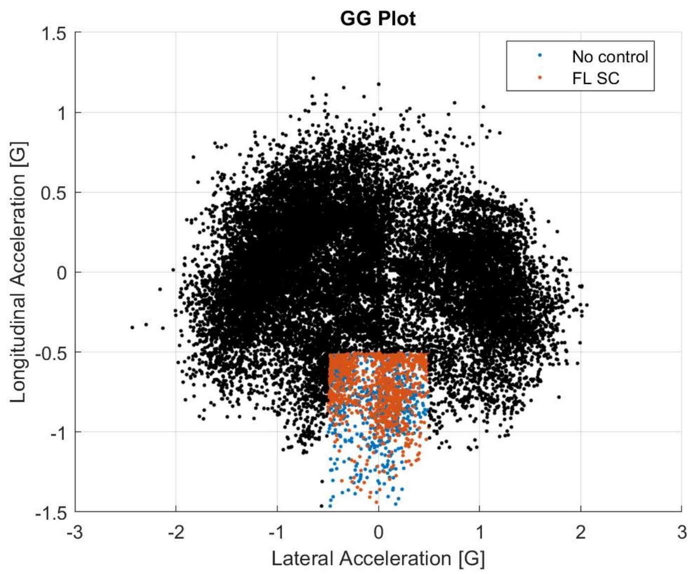

Figure 70: Real vehicle application - Straight Braking GG Plot

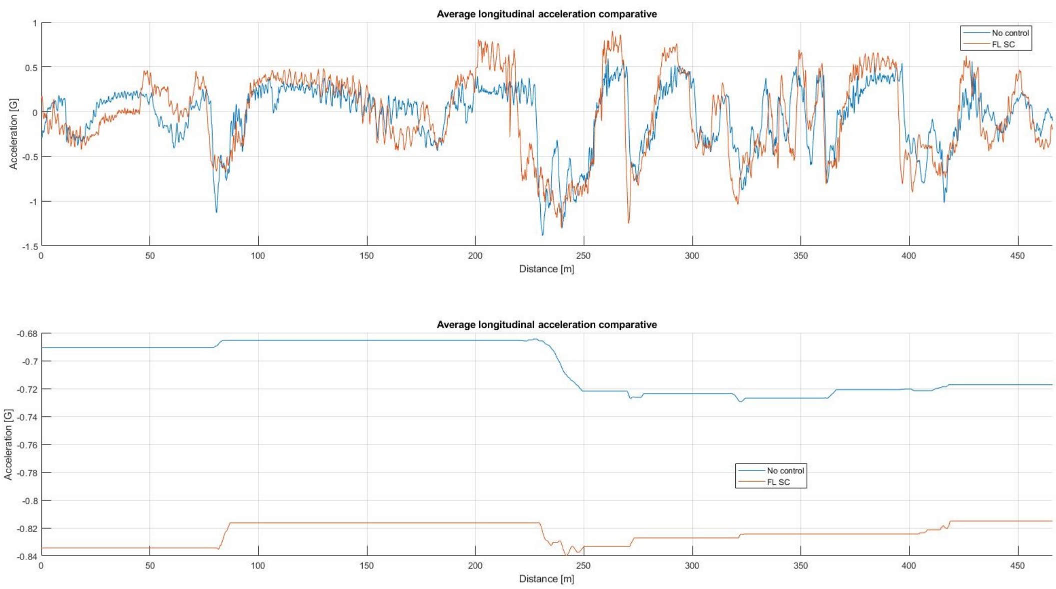

Figure 71: Real vehicle application - Straight Braking KPI

The straight braking KPI shows higher differences between the non-controlled vehicle and the one with the Slip Controller active with respect to simulations. The FL controller is again able to guarantee higher average longitudinal negative acceleration, with a wider gap from the non-controlled vehicle than expected.

Acceleration in turn

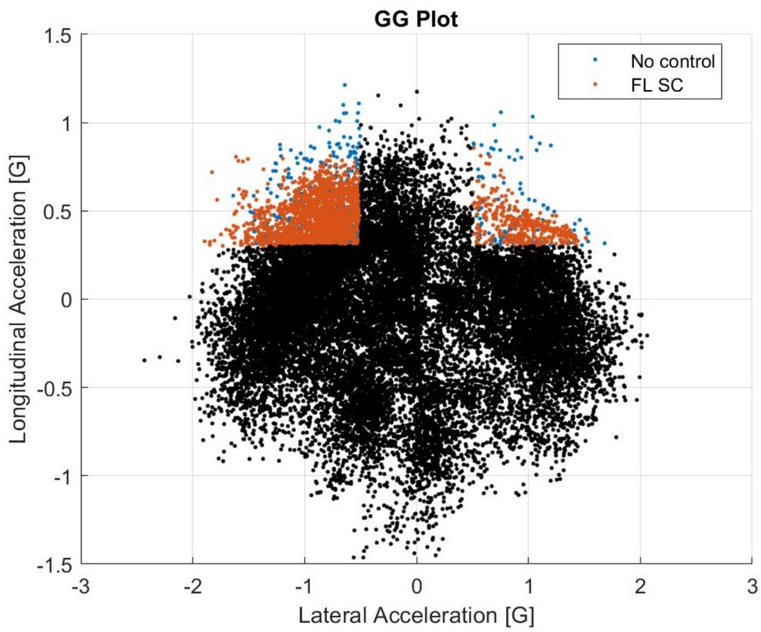

Figure 72: Real vehicle application - Acceleration in turn GG Plot

Figure 68 and 72 GG plot show very well an important point on which it is interesting focus. The GG plot is a simple and very useful tool to compare two drivers or two vehicle performances, but its outliers data must be kept particularly under attention. In fact, they can be caused by the noise of the accelerometers and from local high accelerations point that do not represent the real overall performance. In these GG plot, both vehicles show outliers, but the non-controlled vehicle have some more that could suggest false higher performances.

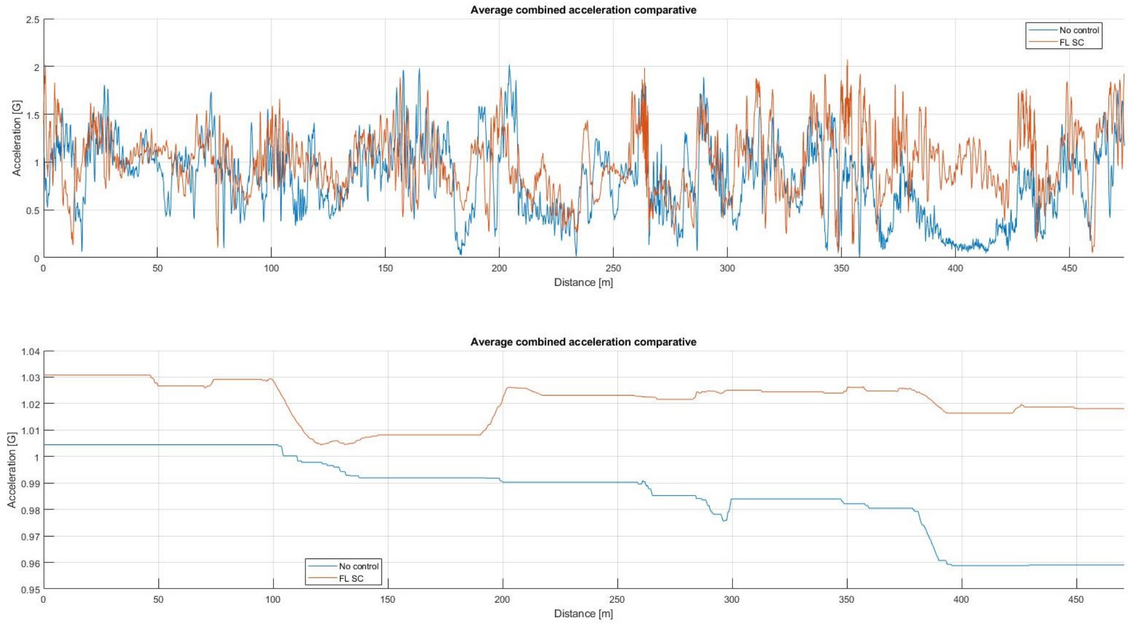

Figure 73: Real vehicle application - Acceleration in turn KPI

In Figure 73 it is shown how the average combined acceleration, the acceleration in turn KPI, is always higher in the vehicle in which the FL slip controller is active. This confirms also the hypothesis that the higher acceleration points in the GG plot of the non-controlled vehicle were mostly outliers, that have low weight on the computation of the average acceleration.

Braking in turn

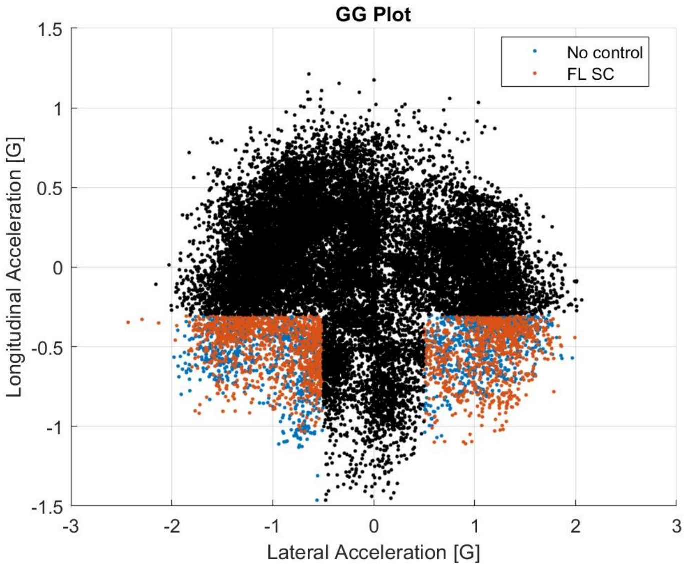

Figure 74: Real vehicle application - Braking in turn GG Plot

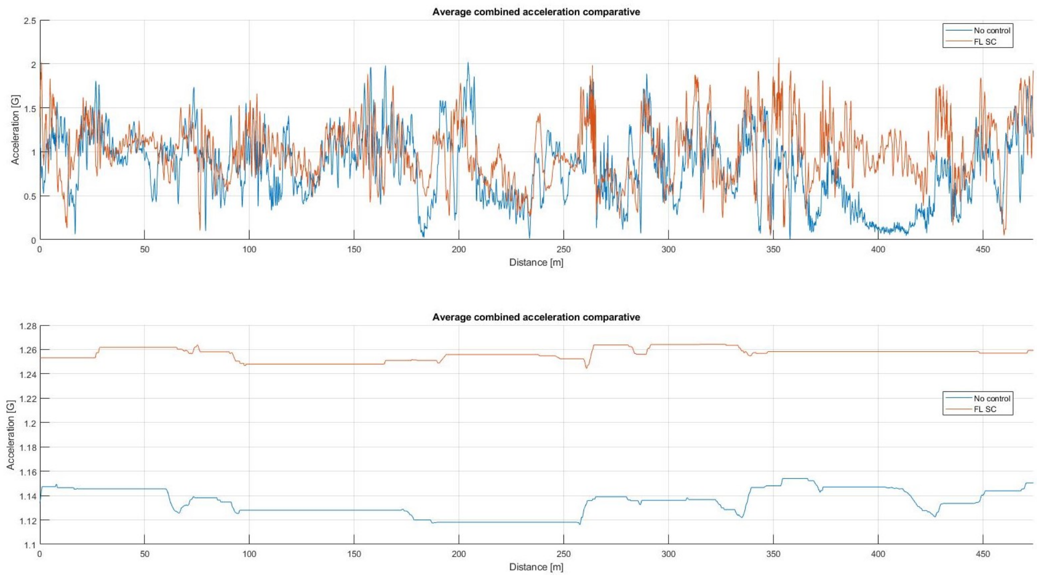

Figure 75: Real vehicle application - Braking in turn KPI

Finally, also in the case of the braking in turn working condition, the vehicle with the FL slip controller shows higher average combined acceleration with respect to the non-controlled vehicle. The result again is better than what expected from simulations.

4.3 Conclusions

This work aimed to propose a method to design and develop a Slip Controller for the application on Electric Vehicles.
In order to develop a proper controller, it is necessary to estimate or measure several vehicles states. The estimation of many of those states, which measurement can be complex and expensive, has shown good results. However, the output of the model proposed to implement a model-based controller were not acceptable to do so. For these reasons and after several analysis on different technologies, the solution that has been implemented consists in a Fuzzy Logic Controller. The developed controller has been tuned in co-simulation between Matlab-Simulink and VI-grade VI-CarRealTime and tested on the Formula Student race car of Squadra Corse PoliTO.

The development and tuning procedure has shown good results, improving the controller performances and enhancing the vehicle dynamics, with the controller active. In order to quantify the improvements brought by the controller in all the different studied situations, some KPIs have been properly defined and exploited. These analysis have been performed in several working conditions, comparing in simulation environment the new controller with the non-controlled vehicle and a benchmark controller, represented by a PID Slip Controller. The simulations have shown performances improvement in every condition, even if in the braking ones the improvement is not so relevant.
The analysis on the real vehicle, instead, have been performed only comparing the non-controlled vehicle and the FL Slip Controller. Also in these cases the FL controller has shown good improvements, even better than in simulations, especially in the braking conditions.

Beyond the data and the KPI metrics, the different drivers have reported strong and positive feedback. They had the capability to feel the torque reduction caused by the controller especially during corner exit, while along the straights they reported a smoother behaviour. Without the controller, their trust in the vehicle grip during the accelerations was much lower and they frequently reported the necessity of releasing the accelerator pedal to avoid strong slippage, especially during corner exit, precisely were the controller had the highest control activity.

Future developments

Future possible improvements regarding the FL Slip Control on its own are difficult to identify, since the solution has been pushed at its limits. Higher time on the project could be spent on the race track, with the goal of mapping the different slip* references for different conditions of ground and tires. It could be also very useful for the drivers to have several maps of different intensities (that could be related with different grip conditions), giving them the possibility to have higher live customization.

Regarding possible developments, that could include the Slip Control, the first step should be the improvement of the model used and the related estimations. A possible starting point can be a change in approach regarding the tire forces estimation method. A solution as the one proposed in [17] can be of interest, since it does not rely on TIR files. These files, in fact, need proper tuning on track, with the scaling coefficients, to have higher adherence with the real conditions of the tire. However, these tests can be expensive and can require complex set-ups of tests and sensors. The proposed method, instead, relies only on already available information, as the accelerations, and it is based on a validated approach.
The following step could be the passage to a 7 Degrees Of Freedom (DOF) model, in which a 3 DOF rigid vehicle model works in parallel with the slip derivative model presented in this work. The four remaining DOF, in fact, are exploited to describe the dynamics of each of the four wheels. Doing so, with a well designed model, it could be possible to integrate a TV Control with the Slip Control, with just one controller. Additional references would be required, as yaw rate and attitude angle references.
A suitable controller for this complex application can to be the eNMPC that has been presented in this work in chapter 2.1.2. In fact, it could guarantee the proper performances, with a multi-reference control problem, still requiring low computational power, because of the explicit formulation of the controller.

List of Figures

1 Downstream ICC Architecture Example[1] 9
2 Upstream ICC Architecture Example[1] 10
3 Formula Student Events 12
4 Squadra Corse PoliTO SC22 ”Aurora” 14
5 Project V-model 16
6 VI-CRT and Matlab-Simulink co-simulation 18
7 PID Controller Structure[5] 21
8 Explicit NMP Slip Controller Structure[4] 23
9 Various Sliding Mode Controllers[7] 24
10 Fuzzy Logic Structure[8] 25
11 Fuzzy Logic Controller triangular Membership Functions set[8] . 26
12 FS Skidpad Event track 28
13 Dual track model 29
14 Side Slip Angle comparison 30
15 Tire Side Slip Angle[12] 31
16 Tire Side Slip Angle comparison 32
17 Wheel Reference Frame and angles[13] 33
18 Wheel Longitudinal Velocity 34
19 Tire Longitudinal Slip[12] 35
20 Tire Longitudinal Slip* Ratio comparison 36
21 Example of load transfer: lateral[13] 37
22 Estimated tires Vertical Forces 38
23 Tire Longitudinal Force Magic Formula[13] 39
24 Tire Longitudinal Force 39
25 Estimated tires Longitudinal Forces 40
26 Tire Lateral Force 41
27 Estimated tires Lateral Forces 42
28 Longitudinal dynamics and wheel moment balance models[15] . . 43
29 Wheel Angular Acceleration 44
30 Tire Longitudinal Slip* Ratio Derivative 45
31 Vehicle Control System 46
32 Control System Structure 47
33 Motors Speed Control 50
34 PID Slip Controller 51
35 PID Slip Controller Tuning Model 51
36 PID Slip Control Tuning Results 52
37 Fuzzy Logic Controller Structure 53
38 Longitudinal Slip Error Membership Functions 53
39 Longitudinal Slip* Error Rate Membership Functions 54
40 Fuzzy Logic Controller Membership Functions 54
41 Fuzzy Logic Straight Acceleration Inference System 55
42 Fuzzy Logic Controller Tuning: Slip* Error in Straight Acceleration 61
43 Fuzzy Logic Controller Tuning: Slip* Error Rate in Straight Acceleration 61

44 Fuzzy Logic Controller Tuning: Rules in Straight Acceleration . . 62
45 Fuzzy Logic Controller Tuning: Controller in Straight Acceleration 62
46 Straight Acceleration Slip* Reference Tuning 63
47 VI-CRT vs Track Data correlation - Open-loop Skidpad 64
48 VI-CRT vs Track Data correlation - Lateral Acceleration 66
49 VI-CRT vs Track Data correlation - CoG Velocity 66
50 VI-CRT vs Track Data correlation - Energy consumption 67
51 VI-CRT vs Track Data correlation - Battery Pack Voltage 68
52 VI-CRT vs Track Data correlation - HV battery pack SOC . . . 68
53 Straight Acceleration KPI 73
54 Straight Braking KPI 74
55 Acceleration in turn KPI - Combined Acceleration 75
56 Acceleration in turn KPI - Stability 76
57 Braking in turn KPI 77
58 SkidPad KPI - Iterations Comparative 78
59 Autocross KPI - Iterations Comparative 78
60 Autocross Cerrina KPI - Iterations Comparative 79
61 Acceleration KPI - Lap-Time Comparative 80
62 SkidPad KPI - Lap-Time Comparative 81
63 Autocross Cerrina KPI - Lap-Time Comparative 81
64 Real vehicle application - Initial Torque Request 82
65 Real vehicle application - Speed Profile 83
66 Real vehicle application - Longitudinal Slip* 84
67 Real vehicle application - Straight Acceleration KPI 1 85
68 Real vehicle application - Straight Acceleration GG Plot 86
69 Real vehicle application - Straight Acceleration KPI 2 87
70 Real vehicle application - Straight Braking GG Plot 88
71 Real vehicle application - Straight Braking KPI 88
72 Real vehicle application - Acceleration in turn GG Plot 89
73 Real vehicle application - Acceleration in turn KPI 90
74 Real vehicle application - Braking in turn GG Plot 91
75 Real vehicle application - Braking in turn KPI 91

List of Tables

1 Squadra Corse PoliTO prototype main data 14
2 Fuzzy Logic Controller General Table of Rules[8] 26
3 Fuzzy Logic Inference System Table of Rules 55
4 KPI Summary Table 70

References

[1] Victor Mazzilli, Stefano De Pinto, Leonardo Pascali, Michele Contrino, Francesco Bottiglione, Giacomo Mantriota, Patrick Gruber, and Aldo Sorniotti. Integrated chassis control: Classification, analysis and future trends. *Annual Reviews in Control*, 51:172–205, 2021.

[2] Formula Student Germany. *Formula Student Rules 2022.* 2021.

[3] Ganesh B Regulwar, PR Deshmukh, RM Tugnayat, PM Jawandhiya, and VS Gulhane. Variations in v model for software development. *International Journal of Advanced Research in Computer Science*, 1(2):134–135, 2010.

[4] Davide Tavernini, Mathias Metzler, Patrick Gruber, and Aldo Sorniotti. Explicit nonlinear model predictive control for electric vehicle traction control. *IEEE Transactions on Control Systems Technology*, 27(4):1438–1451, 2018.

[5] John G Ziegler, Nathaniel B Nichols, et al. Optimum settings for automatic controllers. *trans. ASME*, 64(11), 1942.

[6] Rakesh P Borase, DK Maghade, SY Sondkar, and SN Pawar. A review of pid control, tuning methods and applications. *International Journal of Dynamics and Control*, 9(2):818–827, 2021.

[7] Mahdi Jalili-Kharaajoo and Farhad Besharati. Sliding mode traction control of an electric vehicle with four separate wheel drives. In *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 03TH8696)*, volume 2, pages 291–296. IEEE, 2003.

[8] Ilyas Eker and Yunis Torun. Fuzzy logic control to be conventional method. *Energy conversion and management*, 47(4):377–394, 2006.

[9] Ebrahim H Mamdani. Applications of fuzzy algorithms for control of simple dynamic plant. *Proc. Iee*, 121:1585–1588, 1974.

[10] Li-Xin Wang. *Adaptive fuzzy systems and control: design and stability analysis.* Prentice-Hall, Inc., 1994.

[11] Giancarlo Genta and Lorenzo Morello. *The automotive chassis: vol. 2: system design.* Springer, 2009.

[12] Giancarlo Genta and Lorenzo Morello. *The Automotive Chassis.* Springer, 2009.

[13] Hans Pacejka. *Tire and vehicle dynamics.* Elsevier, 2005.

[14] Daniel Bohl, Nikolaos Kariotoglou, Andreas B Hempel, Paul J Goulart, and John Lygeros. Model-based current limiting for traction control of an electric four-wheel drive race car. In *2014 European Control Conference (ECC)*, pages 1981–1986. IEEE, 2014.

[15] Xianjian Jin, Guodong Yin, and Nan Chen. Advanced estimation techniques for vehicle system dynamic state: A survey. *Sensors*, 19(19):4289, 2019.

[16] Charles Audet and John E Dennis Jr. Analysis of generalized pattern searches. *SIAM Journal on optimization*, 13(3):889–903, 2002.

[17] Flavio Farroni. Trick-tire/road interaction characterization & knowledge-a tool for the evaluation of tire and vehicle performances in outdoor test sessions. *Mechanical Systems and Signal Processing*, 72:808–831, 2016.