Actuators Coordination of Heavy Vehicles using Model Predictive Control Allocation

by

Andrea Sinigaglia

This thesis, in accordance with the T.I.M.E. program, has been submitted in partial fulfilment of requirements for the double joint Master's degree in Automation Engineering at the Università degli Studi di Padova (Padova, Italy) and Automotive Engineering at the Escola Tècnica Superior d'Enginyeria Industrial de Barcelona (Barcelona, Spain). The thesis has been carried out at the Department of Chassis Strategies and Vehicle Analysis, Volvo Group Trucks Technology (Göteborg, Sweden).

October 2015

Abstract

This report proposes the use of a novel method called Model Predictive Control Allocation (MPCA) in order to conveniently coordinate the different actuators present on a heavy vehicle.

The actuators analysed in this report are disc brakes, powertrain and rear active steering. All these actuators can technically be controlled by an external electronic device and their utilization has an impact on the planar dynamics of the vehicle.

The actuators are designed so that, if the driver wants to modify the vehicle behaviour, there are several ways of using the actuators that lead to the same requested behaviour. This property identifies the vehicle as an over-actuated system. Considering the nature of all the actuators and their effects on the vehicle is essential for the designated method to coordinate the actuators.

The method used for the coordination merges the characteristics of two different types of controllers: Model Predictive Control (MPC) and Control Allocation (CA). The potential of a model predictive control method resides in its ability to explicitly take into account the nature of the actuators for a certain time horizon ahead before deciding the control action to be applied to the system. The control allocation, on the other hand, is a suitable method to decide how to combine the actuators in order to modify the behaviour of the vehicle.

The peculiarity of these controllers lies in the way they compute the control input to the system. Unlike a classical PID controller, in fact, they use a cost function, which has to be iteratively minimized, in order to find out the best input for the system. Common issues related to this class of controllers are the robustness and speed of the algorithms used to solve the problem. The problem defined by the MPCA controller belongs to the class of Quadratic Programming (QP) problems for which several methods have been developed. A primal-dual interior-point method with Mehrotra's predictor-corrector is used by the solver selected to deal with the QP problem.

In order to evaluate the performance of the controller, three test scenarios have been analysed: split-μ braking, split-μ acceleration and brake blending. In each one of the scenarios there is a need to precisely coordinate the actuators in order to improve the vehicle's dynamics. The expected behaviour of the controller when facing the three different situations has firstly been analysed and explained. Later, the controller has been validated using simulations and tests on a real vehicle.

Both simulations and tests have shown promising results. The controller is able to effectively deal with each one of the situations leading to a satisfactory enhancement of the vehicle dynamics. The controller has also been compared with other methods, a Control Allocation formulation with rate limits and a vehicle without rear active steering (RAS). In general, better performances can be observed during the transitions when using the MPCA formulation rather than the CA formulation and improved stability can be achieved on the vehicle when the RAS is introduced. The different behaviours of the vehicle for every different scenario have been presented and explained.

Acknowledgements

My most grateful thanks go to Kristoffer Tagesson, my supervisor at Volvo. I have really appreciated his unwavering support and sheer enthusiasm in this project; the freedom he gave me to develop the method used in this thesis while keeping me on the path, avoiding unnecessary loss of precious time; the encouragements, genuine comments and helpful feedbacks I have received during the writing of the thesis; his patience and experience during the tests on the truck that have been indispensable to complete the most exciting part of this project.

I would also like to thank Paolo Falcone and Bengt Jacobson, the professors at Chalmers that have taken part to this thesis. I would like to thank Paolo Falcone for his helpful insides and suggestions about model predictive control, while I would like to thank Bengt Jacobson for his detailed and accurate observations on many topics of this report.

I would also like to thank Maria Elena Valcher and Ana Barjau, the professors in Italy and Spain that, although being distant, have always supported me, promptly answering to all my emails, solving questions and doubts.

Contents

1. Introduction

This chapter introduces the relevant aspects of the thesis: motivations, vehicle configuration, scope of the thesis and methods used to achieve the scope.

1.1. Background

Vehicles are becoming safer and safer. In the last few years, driving assistance systems have made vehicles more reliable and easier to drive. Active safety systems, defined as all the devices made to reduce the risk that an accident occurs, are becoming more and more popular, making little steps towards a vision of road traffic safety. Both, trucks and cars, are today protagonists of significant changes in the way they behave under risky circumstances. Many common dangerous situations can today be handled without asking too much effort from the driver.

Many different active safety systems can today be found as a basic equipment for vehicles and some of them, such as the ABS (Antilock Braking System) and the ESC (Electronic Stability Control), have become mandatory in all new vehicles. The active safety systems have an impact on the dynamics of the vehicle. This means that they influence the behaviour of the vehicle in response to an input from the driver and depending on the state of the vehicle/environment. In general, it can be said that the scope of every active safety system is to ensure stability and controllability of the vehicle as long as it is physically possible.

Today's active safety systems have considerably reduced the amount of accidents due to harsh weather driving conditions, driver's inexperience or distraction. Nevertheless the "*vision zero*", that is having no more people victims or affected for the rest of their lives by car accidents, is still far away from reality. Statistics from 2010 claim that *the total number of road traffic deaths remains unacceptably high at 1.24 million per year* [1]. This means that about 3400 people die every day in road crashes. Because of this remaining gap, innovative safety systems are needed.

Heavy vehicles play a fundamental role in today's society. The widespread of the roads has permitted heavy vehicles to reach almost any place in the world, providing them with a network that is superior to any other means of transport of goods currently available. Moreover, the highly personalisation of heavy vehicles, i.e. number of axles, maximum load, trailers, etc. make them configurable to meet customers demand that have to use the vehicle in very different work environments. Unfortunately, heavy vehicles are still involved in a significant percentage of severe accidents that can end up being more critical than normal car accidents. Therefore, there is a need for ensuring heavy vehicles safety in as many situations as possible.

Accidents caused by vehicle instability still play a significant role in every-day roads. Common situations that provoke instability are uneven roads when there is one or more wheels of the vehicle that can easily lose the grip with the ground. Such situations, where there is a need to maintain the vehicle controllable, have been analysed in this thesis.

1.2. Vehicle Configuration

The vehicle considered in the thesis is a $6x2$ solo truck. As depicted in Figure 1.1, $6x2$ means that the vehicle is composed of three axles, one of whom is the driven axle. The truck is then equipped with 6

independent disc brakes, one diesel engine with engine brake and a tag axle with rear active steering (RAS). A tag axle is defined as an axle situated after the driven axle that can be elevated when there is no need to use it, while RAS means that an external controller is responsible of turning the wheels of the tag axle. The truck has an open differential at the driven axle and the wheels are numbered so that the front left wheel is the number 1, the front right wheel is the number 2 and so on.

The vehicle and its coordinate system are represented in Figure 1.1. With this coordinate system, the longitudinal force is defined as the total force produced in the x direction, the lateral force is defined as the total force produced in the y direction and the yaw moment is defined as the moment at the z-axis of the vehicle.

An independent coordinate system can be taken for each of the wheels in the same way as for the vehicle. The definitions of longitudinal and lateral forces are analogous.

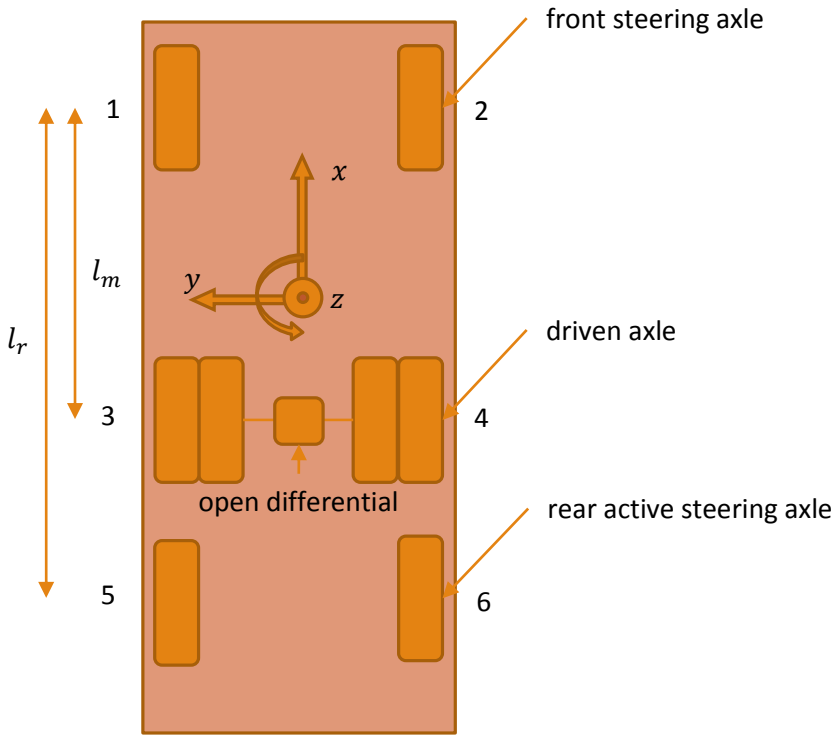

Figure 1.1. Image from the top of the vehicle configuration

1.3. Scenarios

In this thesis, particular attention has been paid to situations where not all the wheels of the vehicle are in contact with the same type of ground. Specifically the situation taken into account is the one where the wheels on one side of the vehicle are in contact with high friction coefficient ground (e.g. dry asphalt) while the wheels on the other side are in contact with low friction coefficient ground (e.g. ice). This situation commonly goes under the name of "*split-μ road*". In such a scenario, it is easy to lose the control of the vehicle leaving the driver to cope with a dangerous and difficult manoeuvre.

Two natural challenges occur when driving on split-μ road stretches: (1) braking the vehicle from an initial speed to zero and (2) accelerating the vehicle when it is at a standstill. Both situations should

be handled in a way so that they are easy to manage even for an unexperienced driver. These are two cases that will be treated in this thesis.

Braking on a split-μ road is complicated due to the trade-off between an acceptable braking distance and vehicle stability. In order to minimize the braking distance every wheel of the vehicle should brake as much as possible. When doing so the wheels in contact with high friction ground brake much more than the wheels in contact with low friction ground and an undesired yaw moment is produced. This results in an unstable condition for the vehicle (Figure 1.2-A). On the other hand, if stability has to be maintained all the wheels on the same axle should generate the same amount of braking force. This would limit the amount of braking force based on the wheels in contact with lower friction ground and could make the braking distance unacceptable (Figure 1.2-B).

This trade-off can be solved if rear axle steering is considered and used when braking. A steered rear axle is usually used to improve the dynamics of the vehicle, for example to avoid wheels from sliding during steady-state cornering. In the case of a split-μ road a vehicle equipped with RAS can take advantage of turning the rear axle wheels when braking. RAS in fact permits to brake more with the wheels on the high friction side because the undesired yaw moment that would be produced by the brakes can be compensated by an opposite moment that can be produced by turning the rear axle wheels. This results in an acceptable braking distance without loss of stability (Figure 1.2-C).

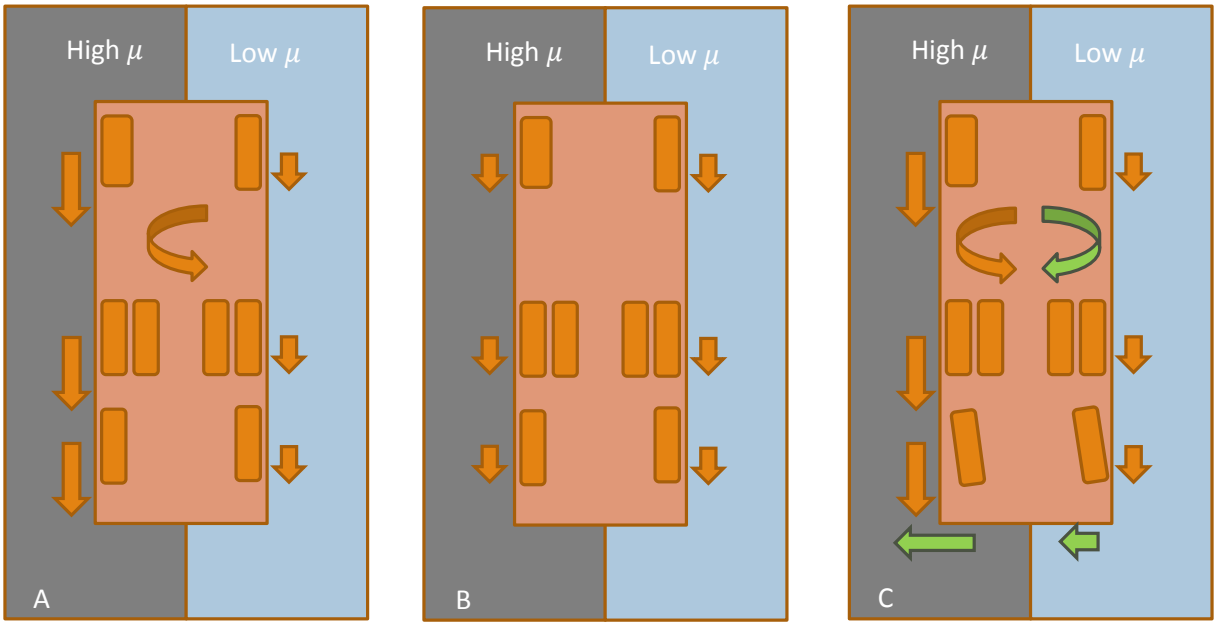

Figure 1.2. (A) The braking distance is minimized but the brakes on the left side produce an undesired yaw moment. (B) No yaw moment is produced but the total braking force is low and the braking distance could be unacceptable. (C) The braking distance is minimized and the yaw moment produced by the left side brakes is compensated by the opposite yaw moment generated by turning the rear wheels.

Keeping in mind the situation C of Figure 1.2, it is clear that the key point of using the rear active steering while braking is the coordination of every one of the forces produced by the various actuators on the vehicle. The correct coordination between longitudinal forces produced by the brakes and lateral forces produced by the rear axle prevents the vehicle from an undesired yaw moment.

Now, if the vehicle is stopped on a split-μ road and the driver wants to start moving the vehicle again, a similar situation occurs.

When accelerating the engine torque is distributed by the open differential 50-50 to the driven wheels. At some point, the driving force on the wheel in contact with lower friction ground will reach its limit and it will start slipping (Figure 1.3-A). Because of the open differential no bigger torque can be transmitted to the other wheel, thus the result is an insufficient total driving force to move the vehicle. One way traction control systems avoid this situation is by braking on the wheel on the low friction side to create a virtual resistance between wheel and ground (Figure 1.3-B). Then, the wheel in contact with high friction ground can generate a bigger driving force than the wheel on the opposite side. As a result this generates a yaw moment that again can be compensated with RAS (Figure 1.3-C).

Overall, it is now clear that it is important to coordinate all the actuators of the vehicle in order to generate the desired longitudinal force without inducing too much unwanted yaw moment.

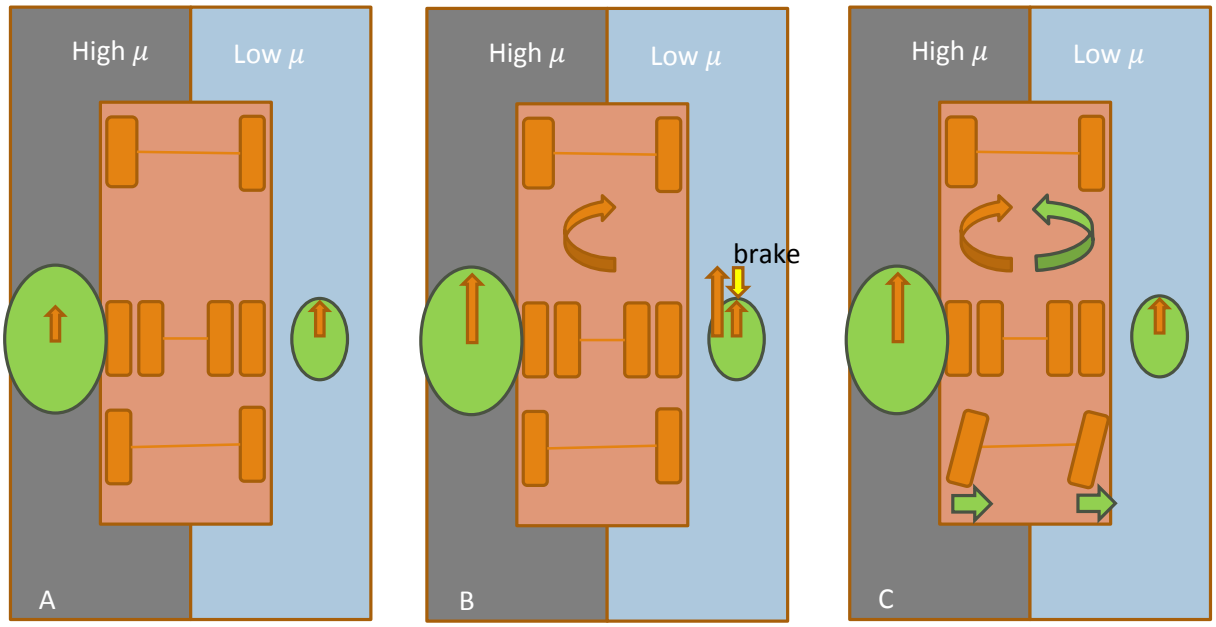

Figure 1.3. (A) The torque is equally split and the right wheel has reached its maximum driving force. (B) Right brake is used to generate a resistance on the right wheel so that more torque can be transmitted on the left wheel. This generate a yaw moment on the vehicle. (C) RAS is used to counteract the yaw moment generated by the driven axle.

Another interesting situation where it is convenient to coordinate the actuators of the vehicle is in the so called brake blending scenario. This scenario is defined as a braking event under normal conditions (e.g. all the wheels on dry asphalt), where disc brakes and engine brake are used together. Engine brake in fact has a slower response than disc brakes but it is preferable to use the engine brake as much as possible because it does not present the typical problems of the disc brakes, fading and wear. Coordination between engine brake and disc brakes makes it possible to use the disc brakes at the very beginning to have a fast response from the vehicle and then slowly decreasing the use of the disc brakes as the engine brake starts to produce the required braking force.

These three scenarios: split-μ braking, split-μ acceleration and brake blending are the test cases considered in the thesis. Building a controller that is able to cope with each one of these three situations is the objective of the thesis.

1.4. Control algorithm design

Over-actuated systems often appear in automotive, aerospace and maritime industry. A system is called over-actuated if there are various actuators that can produce the same global effect for the considered system (Figure 1.4). An actuator is defined as a device that is able to produce specific forces and moments on the system as requested by the control signal. Thus, brakes, motors, propellers, etc. are actuators. An example related to the topic of this thesis is the following: in order to produce the global effect "braking force", the system "truck" can brake with the front axle brakes or with the rear axle brakes or with all the brakes together. Therefore, there are various ways to use the actuators "brakes" to produce the same global effect "braking force" on the system "truck".

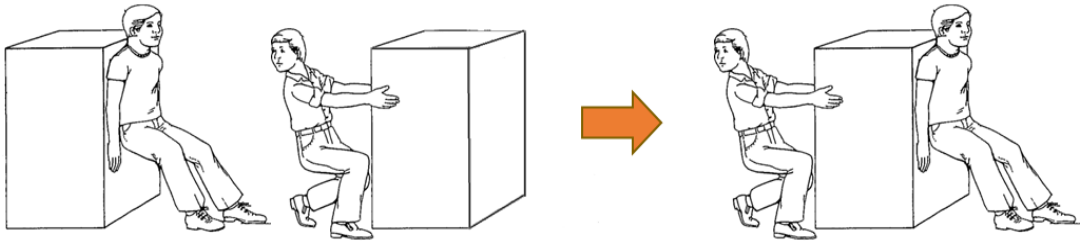

Figure 1.4. Over-actuated systems philosophy: to move the box, it can be pushed or pulled. When it is both pushed and pulled there are several ways to coordinate the two actuators (the two guys) in order to produce the same desired global force on the box. The image is from www.siliconbeachtraining.co.uk.

Over-actuated systems are useful as they increase the fault tolerance of the system and give more freedom on how to use the actuators in order to achieve the desired global effect. The way the actuators are used can change depending on external factors too.

A common method to deal with over-actuated systems is called *Control Allocation*. Control allocation algorithms coordinate the different actuators of a system so that they collectively produce the desired global effect on the system. A control allocation algorithm is solved several times during a short period of time in order to continuously adapt the actuators usage to the current situation.

Usually control allocation methods do not take into account the proper dynamics of the actuators; this means that the algorithms do not have precise information about how the output value of an actuator evolves once a commanded input is sent to the actuator. Instead a static relation between input and output of the actuator is considered in the algorithms. However, this assumption is not always sufficient to ensure a satisfactory coordination of the actuators. This is particularly true when the coordination is done with actuators that have very different time responses to a specified input.

To cope with this problem, a new method has recently been developed so that the dynamics of the actuators can be explicitly incorporated in the algorithm. This method makes use of the *Model Predictive Control* theory and it is usually referred in the literature as *Model Predictive Control Allocation* (MPCA) [2] [3] [4] [5] [6] [7]. According to the best understanding of the author, this method has not previously been used in commercial road vehicles.

A model predictive control allocation approach has been used in this thesis to coordinate the vehicle's actuators of interest, namely brakes, powertrain and rear active steering. The development of the controller is based on simulations in Simulink using a non-linear model for the vehicle

described in section 1.2. The vehicle model is part of VTM (Volvo Transports Model) library and it has been set up to meet the configuration specified in section 1.2.

To evaluate the effectiveness of the developed controller, three test scenarios have been analysed: split-μ braking, split-μ acceleration and brake blending. Split-μ braking has been chosen to comprehend how the RAS can be used to minimize the stopping distance without compromising the vehicle stability. Split-μ acceleration shows how the controller can work as a traction control system that, apart from conveniently adjusting the engine torque between the wheels at the driven axle, uses the RAS to compensate the generation of possible undesired yaw moments. Finally, the objective of the brake blending scenario is to understand how disc brakes and engine brake can be combined together in order to produce the desired braking force and to minimize the use of the disc brakes.

The final test for the controller has been the evaluation of its performance on a real truck. Due to time constraints the performed tests are all related with the split-μ braking scenario. This scenario has proven to be the most interesting one to evaluate the performances that can be achieved with a precise coordination of the considered actuators. The implementation of the controller in a real environment raises questions that are not so evident during the simulations and it contributes to a better understanding of the whole system. The implementation from a virtual to a real environment has been performed via dSPACE. The software speeds up the transfer of a controller designed in Simulink into its correspondent code loadable on a specific hardware called MicroAutoBox. The advantage of using MicroAutoBox is that it can directly communicate with the actuators and sensors of the vehicle.

1.5. Goals

The aim of this thesis is to evaluate the effective coordination of the actuators of a heavy vehicle that can be achieved using a model predictive control allocation formulation. During all the simulations, the priority has been given to the vehicle stability. Once the stability is ensured, the controller can deal with other issues, such as braking and accelerating for the split-μ scenarios, use of the disc brakes for the brake blending scenario.

An additional purpose of the thesis is to compare the MPCA method with both a CA method and a vehicle without RAS. All the comparisons are made to understand what are the benefits and drawbacks of using the MPCA formulation when the vehicle faces one of the described scenarios.

1.6. Limitations of the Study

The list below clarifies which aspects have not been considered within this thesis:

- It is assumed that the MPCA controller has continuous access to some vehicle parameters, namely the vertical load on each wheel, the wheel angles of the front and rear steering axles and the friction coefficient between wheels and ground. Among all these parameters, the estimation of the friction coefficient is today the most critical issue and currently an intensive field of research. While vertical loads and wheel angles can, in fact, be known from sensors situated respectively in the air suspensions and steering axles, not consolidated methods exist for real-time friction estimation.

- Only solo truck configurations have been taken into account. This means that no trailers dynamics have been included in the study. The variety of heavy vehicles configurations is definitely an important point for the industry and it is what makes them a competitive means of transport in the market. The idea is that the developed controller can be used as a basis for a wider family of trucks configurations. Extensions to simpler or more complicated configurations should be possible without changing the substantial nature of the controller.

- During the test cases, a limited number of situations have been considered. In particular, only flat segments of roads without curves have been taken into account. The degree of complexity introduced when considering uneven roads makes it important, at the beginning, to understand the vehicle dynamics when exposed to simple manoeuvres. However, other situations such as split-μ braking during a steady-state curve are of particular interest to validate the strength of the designed controller.

1.7. Outline of Report

This report is organized as follows:

- Chapter 2 presents an introduction to the control allocation theory and model predictive control theory. The chapter ends with the formulation of the designed controller.
- Chapter 3 describes in detail the three test cases considered. In this chapter it is explained how to overcome some problems that can arise during the manoeuvres.
- Chapter 4 explains what methods and algorithms can be used to solve the MPCA problem, together with the software that has been chosen to compute the solution of the problem.
- Chapter 5 shows the results of the simulations for every one of the analysed scenarios and compares the MPCA method with the other methods.
- Chapter 6 explains the implementation of the controller in a real vehicle, the chapter ends with the results of the split-μ braking tests.
- Chapter 7 summarizes all the conclusions of the thesis and provides some suggestions for future work.

2. Model Predictive Control Allocation

This section describes the method used by the controller that has been designed during the thesis.

2.1. Problem Formulation

Vehicle controllers should help the drivers to make their vehicles behave as they think they should. Usually, the controllers developed for the vehicle receive some inputs from the driver and, based on the vehicle conditions, they decide which are the forces and/or moments on the vehicle that make it behave as requested by the driver.

As previously stated, the vehicle considered along this thesis is an over-actuated system, which means that there are several combinations of the actuators that can produce the same global forces on the vehicle. To cope with over-actuated systems, two different philosophies can be followed so that the actuators can together generate the requested global forces on the vehicle.

The simplest way to cope with over-actuated systems is to limit the number of actuators used to generate the global forces on the vehicle so that it is possible to have a bijective mapping between the selected actuators and the global forces for every different situation of interest that can occur. The strength of this method is the simplicity of its implementation but, on the other hand, it is not optimal when considering either fault tolerance or dynamic adaptability to different situations.

For instance, imagine that only one brake on each side of the vehicle is designated to correct possible oversteering or understeering behaviours of the vehicle. It can happen that, during the manoeuvre, the designated brake fails to work or that the corresponding wheel encounters a surface with low friction coefficient. In both cases, there would be no solution to help the driver correcting the undesired vehicle behaviour.

Another way to cope with over-actuated systems, which is the way followed in this thesis, is to have a method that divides the control into two levels:

1. A high level motion control algorithm that computes the forces requested on the vehicle in order to make it behave as desired by the driver. These forces are called global forces in the thesis;
2. A low level coordination control algorithm that finds a suitable actuators coordination and usage in order to produce the global forces computed by the high level motion control.

The global forces computed by the high level controller are forces and moments that have an impact on the vehicle dynamics. To describe the vehicle dynamics some convenient states of the vehicle are considered such as yaw rate, lateral and longitudinal speed, etc. This often leads to describe the evolution of the states as an affine system:

$$\dot{x} = f(x) + g(x)v \qquad \textit{Eq. 2.1}$$

where x is the vector of the considered states (usually, $x = \left[v_x\ v_y\ \omega_z\right]^T$ that are respectively the vehicle longitudinal speed, lateral speed and yaw velocity) and v is the vector containing the global forces on the vehicle (usually, $v = \left[F_x\ F_y\ M_z\right]^T$ that are respectively the resulting longitudinal force, lateral force and yaw moment on the vehicle). From (Eq. 2.1) it is possible to understand the influence of the forces and moments on the behaviour of the vehicle and thus it is possible to design

a high level controller that is able to find suitable values for v so that the driver's intentions are met and the vehicle behaves as requested by the driver.

Once v has been generated by the high level controller, it is sent to the low level controller that is responsible for mapping v into adequate values for the actuators used on the vehicle. Vector v can then be thought as the *virtual input* for the low level controller. Defining δ as the vector containing the output values for each one of the actuators, the low-level controller finds δ such that $v = f(\delta)$. This relation between virtual input and actuators outputs can often be approximated by a linear system of the form:

$$v = B_f \delta \qquad \textit{Eq. 2.2}$$

As the system is over-actuated, $q := \dim(v) < \dim(\delta) =: n$. The matrix $B_f \in \mathbb{R}^{q \times n}$ that maps the actuators usage into the global forces is called effectiveness matrix.

A topic that has not been mentioned yet is that the relation between the commanded input from the low level controller and the real output of the actuators is not static, i.e. every actuator needs some time to reach the value commanded by the controller. It is then important to make a clear distinction between:

- δ_{cmd} the vector computed from the low level controller that is used as input for the actuators;
- δ the vector that describes the actual value at the output of the actuators.

In most cases the relation between these two vectors (δ_{cmd} and δ) can be explained through a linear dynamic system:

$$\dot{\delta} = A\delta + B\delta_{cmd} \qquad \textit{Eq. 2.3}$$

It can be noted that as long as δ does not reach the value of the commanded input δ_{cmd}, (Eq. 2.2) cannot be satisfied and the global forces requested by the high level controller are not met.

The division of the control system into these two different controllers offers the advantage of a modular design. This means that on the one side the high level controller can be designed without specific information about the actuators behaviour and, on the other side, the low level controller can be developed without knowing the relation between global forces and vehicle states. Each controller deals with a specific problem that is independent of the others. In this configuration it is the low level controller that takes care of both solving fault problems on the vehicle and dynamically

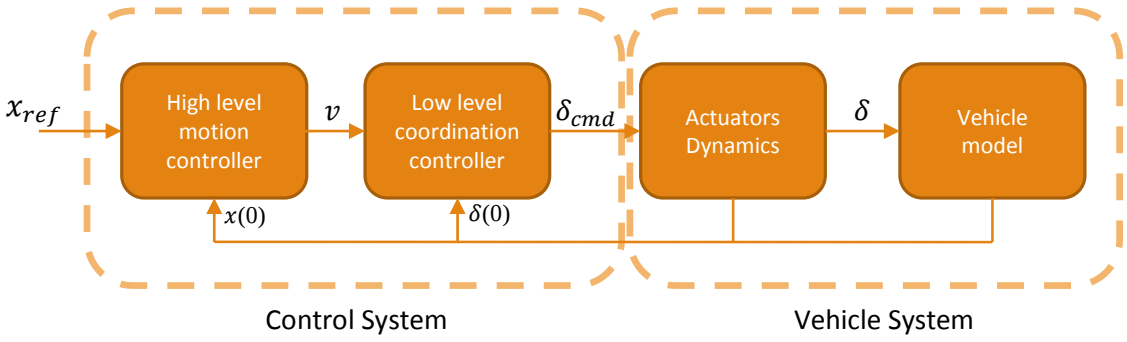

Figure 2.1. Structure of the overall system

adapting to different external situations.

To sum up, the entire structure of the system can be illustrated as in Figure 2.1.

The signal x_{ref} stands for "reference" and it usually comes from another block responsible for interpreting the driver intentions and transforming them into convenient values for the states of the system (yaw rate, lateral/longitudinal speed). This block is referred to as *Driver Interpreter* and instances of its implementation can be found in [8] [9].

A simple example will help understanding the structure of the overall system. Imagine a driver wants to brake on a straight street and to do that the driver uses the brake pedal. The *driver interpreter* will translate the position of the brake pedal into a desired speed for the vehicle v_{ref}. At this point the signal is sent to the *high level controller* that, comparing the requested speed with the actual speed, transforms the signal into the global longitudinal force needed to brake the vehicle F_x. This force is then the virtual input v for the *low level controller* which is responsible for taking care of the brakes dynamics and find a convenient configuration for the inputs δ_{cmd} so that the brakes can together generate the requested global force F_x.

The focus of this thesis is to coordinate the actuators of a heavy vehicle and hence to design a low level controller that can compute suitable commanded values δ_{cmd} for the actuators in different situations of interests. To do that the controller receives the virtual input v from the high level controller and, aware of the different actuators dynamics, it computes δ_{cmd} so that v can be achieved.

In the following sections the methods used to set-up the low level controller are explained.

2.2. Control Allocation (CA)

The primary objective of a control allocation algorithm is to find a value for δ so that $B_f\delta = v$. Control allocation algorithms do not explicitly consider actuators dynamics, thus there is no distinction between the control input computed by the low level controller (δ_{cmd}) and the actual value at the output of the actuator (δ). Therefore the assumption is:

$$\delta = \delta_{cmd} \rightarrow v = B_f\delta = B_f\delta_{cmd} \qquad Eq.\ 2.4$$

As every actuator has its own saturation limits, a solution for the linear system (Eq. 2.4) is not always guaranteed. It could happen that the global forces requested by the high level controller cannot be met by the actual capabilities of the actuators.

In order to cope with this problem the control allocation formulation is often rewritten as an optimization problem in the form:

$$\begin{aligned} &\delta = argmin \left\| B_f\delta - v \right\|_{W_v}^2 \\ &subject\ to: \delta_{\min} \leq \delta \leq \delta_{\max} \end{aligned} \qquad Eq.\ 2.5$$

where the norm in the objective function is a quadratic form, that is $\|a\|_W^2 = a^T W a$, W_v is a weighting diagonal matrix that indicates which one of the global forces in v has the priority in the minimization. $\delta_{\min}$ and $\delta_{\max}$ are respectively the lower and upper limits for the actuators saturation. In case a solution does not exist for (Eq. 2.4), the new formulation finds a δ so that $B_f\delta$ is as close as possible to v.

In over-actuated systems $\dim(v) < \dim(\delta)$, therefore if a solution exists for (Eq. 2.4) it is probably not unique. The possibility to have more than one solution for (Eq. 2.4) can be seen as an advantage in the formulation of (Eq. 2.5). With this new formulation, in fact, it is possible to add a second objective in the minimization function that is of interest for the considered problem. This second objective is usually related to actuators usage and reflects the wish that the optimal solution should make more use of some actuators than others. The second objective is incorporated in (Eq. 2.5) as follows:

$$\begin{gathered}\delta = argmin(\left\|B_f\delta - v\right\|_{Wv}^2 + \gamma\|\delta - \delta_d\|_{W_\delta}^2) \\ subject\ to: \delta_{\min} \leq \delta \leq \delta_{\max}\end{gathered} \qquad \textit{Eq. 2.6}$$

where W_δ is a weighting diagonal matrix that prioritizes some elements of the norm $\|\delta - \delta_d\|^2$, γ is a scalar that is chosen small in order to prioritize the first objective in the minimization function ($\left\|v - B_f\delta\right\|_{W_v}^2$) and δ_d is read as "desired δ" and it can be set to a determined value if it is convenient that the actuators stay as close as possible to that value in the solution. For example, when the vehicle brakes on a straight road the RAS should not be used and $\delta_{d_{RAS}}$ should be set $\delta_{d_{RAS}} = 0$. As in (Eq. 2.5), the constraints $\delta_{\min} \leq \delta \leq \delta_{\max}$ define the saturation limits of the actuators. Apart from the saturation limits, another type of constraints, called rate constraints, can be added to (Eq. 2.6). Being aware of the fact that the controller is a digital system, actuators rate constraints can be included by limiting the change in the control $\Delta\delta$ between one sample time and the following one:

$$\Delta\delta_{\min} \leq \Delta\delta(t) = \frac{\delta(t) - \delta(t - T_s)}{T_s} \leq \Delta\delta_{\max} \qquad \textit{Eq. 2.7}$$

where T_s is the sampling period chosen for the digital controller, $\Delta\delta_{\max}$ and $\Delta\delta_{\min}$ are the maximum increment and decrement allowed to the controller at each step, respectively. The rate constraints can be included in (Eq. 2.6) as:

$$\underline{\delta}(t) \leq \delta(t) \leq \overline{\delta}(t) \qquad \textit{Eq. 2.8}$$

where:

$$\begin{gathered}\overline{\delta}(t) = \min(\delta_{\max}, \delta(t - T_s) + T_s\Delta\delta_{\max}) \\ \underline{\delta}(t) = \max(\delta_{\min}, \delta(t - T_s) + T_s\Delta\delta_{\min})\end{gathered} \qquad \textit{Eq. 2.9}$$

Rate constraints are useful when the systems contain both slow and fast actuators. As the control allocation does not distinguish between commanded input (δ_{cmd}) and actual output (δ) of the actuators, having reasonable values for $\Delta\delta_{max}$ and $\Delta\delta_{min}$ permits to have the actual output of the actuator that is following the commanded input, $\delta_{cmd} \approx \delta$. This is surely an advantage because the purpose of the control is having δ so that $B_f\delta = v$ but what it is actually computed by the low level controller is δ_{cmd} so that $B_f\delta_{cmd} = v$.

2.3. Model Predictive Control (MPC)

Model predictive control is a technique that aims to predict the possible future states of the controlled system in order to find the optimal input to control the system. Model predictive control algorithms are optimization-based and the predictions of the possible future states are made over a

finite time horizon using a dynamic model that approximates the behaviour of the system. The model predicts the possible values of the future states based on the current values of the states and the combination of available control inputs.

Model predictive control methods are usually based on digital controllers. At each sampling time, an optimal control problem is solved over a finite time horizon (N steps). Once the optimization has been solved and a sequence of optimal inputs has been found ($\delta_{cmd}(0), \dots, \delta_{cmd}(N-1)$), only the first input signal $\delta_{cmd}(0)$ is applied to the system. At the next sample time, the current states are updated and the optimal control problem is solved again. A classical formulation of the model predictive control problem is as follows:

$$\delta^*_{cmd}(0) = argmin \sum_{k=1}^{N} \delta(k)^T Q \delta(k) + \sum_{k=0}^{N-1} \delta(k)^T_{cmd} R \delta(k)_{cmd}$$
$$subject\ to: \quad \delta(k+1) = A\delta(k) + B\delta_{cmd}(k) \qquad \textit{Eq. 2.10}$$
$$\underline{\delta}(k)_{cmd} \leq \delta(k)_{cmd} \leq \overline{\delta}(k)_{cmd}$$
$$\underline{\delta}(k) \leq \delta(k) \leq \overline{\delta}(k)$$

where $Q > 0$, $R \geq 0$ are weighting matrices for the states and the commanded inputs respectively. $\delta(k+1) = A\delta(k) + B\delta_{cmd}(k)$ is a set of constraints that comes from the discretization of the continuous models used to describe the system. $\underline{\delta}(k)_{cmd}$ and $\overline{\delta}(k)_{cmd}$ are upper and lower limits for the commanded input, respectively, while $\underline{\delta}(k)$ and $\overline{\delta}(k)$ are the upper and lower limits for the states of the system.

Looking at (Eq. 2.10), two are the key features of a model predictive control formulation:

- During the minimization, it explicitly considers the dynamics of the states of interest in the system;
- It can naturally handle constraints on both the control input and the states of the system.

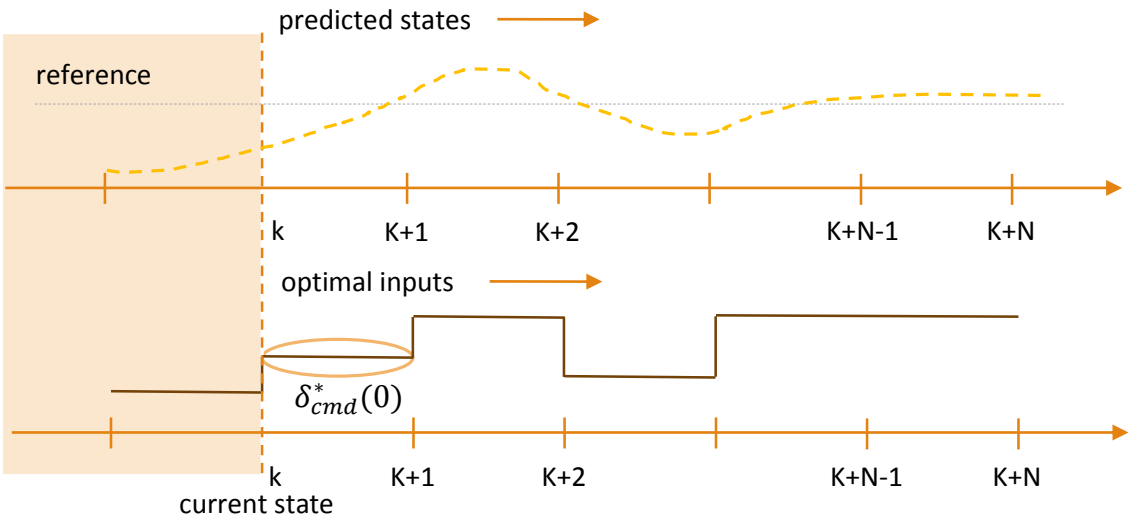

Figure 2.2. Model Predictive Control approach

For these reasons a model predictive control formulation can improve the previously introduced weak points of a control allocation formulation (no explicit consideration of actuators dynamics, no distinction between δ_{cmd} and δ).

Taking into account that:

$$\delta^T Q \delta = \|\delta\|_Q^2 \qquad \textit{Eq. 2.11}$$

It is then straightforward to rewrite the control allocation formulation of (Eq. 2.6) into a model predictive control formulation as:

$$\begin{aligned} \delta^*_{cmd}(0) = argmin \sum_{k=1}^{N} \left\|B_f \delta(k) - v\right\|^2_{Wv} + \gamma \sum_{k=0}^{N-1} \|\delta_{cmd}(k) - \delta_d\|^2_{W_\delta} \\ subject\ to: \quad \delta(k+1) = A\delta(k) + B\delta_{cmd}(k) \\ \underline{\delta}(k)_{cmd} \le \delta(k)_{cmd} \le \overline{\delta}_{cmd}(k) \\ \underline{\delta}(k) \le \delta(k) \le \overline{\delta}(k) \end{aligned} \qquad \textit{Eq. 2.12}$$

The formulation of (Eq. 2.12) can be explained as follows:

Objective function: the scope is to find the optimal control input $\delta^*_{cmd}(0)$ minimizing a cost function.

The first block of the cost function $\sum_{k=1}^{N}\left\|B_f \delta(k) - v\right\|^2_{Wv}$ allocates the actuators usage in order to produce v. As every actuator has its own dynamics, specified by the constraints, the optimal input $\delta^*_{cmd}(0)$ of this formulation is the one that brings $B_f\delta(k)$ towards v in as less steps as possible during the considered time horizon (N steps).

The second block $\gamma \sum_{k=1}^{N}\|\delta(k) - \delta_d\|^2_{W_\delta}$ is multiplied by γ, a scalar with a small value, and it starts to have an impact on the cost function only when the first block has become really small, that is $B_f\delta(k) \approx v$. The aim of the second block is to decide which actuator should be used more in order to produce v.

In case a combination of the actuators that produces v does not exist, the first block tries to stay as close as possible to v, prioritizing the global forces that have relatively higher weights in W_v. The second block is not taken into account during the minimization because its contribution to the objective function is negligible with respect to the first block.

Constraints: there are three different types of constraints.

The first block of constraints $\delta(k+1) = A\delta(k) + B\delta_{cmd}(k)$ describes the dynamics of each actuator. Depending on the initial state, these constraints trace a path of values that $\delta(k)$ can reach during the considered time horizon when a specific $\delta_{cmd}(k)$ is applied.

The second block of constraints $\underline{\delta}_{cmd}(k) \le \delta_{cmd}(k) \le \overline{\delta}_{cmd}(k)$ limits the magnitude of the values that can be used as inputs for the actuators. This reflects the saturation limits of the actuators inputs.

The third block of constraints $\underline{\delta}(k) \le \delta(k) \le \overline{\delta}(k)$ bounds the actual allowable value that an actuator can reach. The constraints take into account that, in a particular moment, because of external conditions, the threshold of permitted values for an actuator can be different from its saturation limit.

This new formulation that merges the control allocation method with the model predictive control method goes under the name of Model Predictive Control Allocation (MPCA). Here again the only computed input that is sent to the actuators is $\delta_{cmd}(0)$, the first optimal input calculated for the selected time horizon. At the next step, the states and the constraints are updated and the entirely process is repeated, shifting the time horizon of the predictions one step ahead.

2.4. MPCA tailored for intended vehicle

The method used during this thesis to coordinate the vehicle actuators is the model predictive control allocation. This section explains how the objective function, the constraints and the variables have been configured to achieve the scopes presented in (1.3).

The MPCA formulation used is very similar to the one of (Eq. 2.12):

$$\delta^*_{cmd}(0) = argmin \sum_{k=1}^{N} \left\| B_f \delta(k) - v \right\|^2_{Wv} + \gamma \sum_{k=1}^{N} \left\| C_f \delta(k) + \delta_e \right\|^2_{W_\delta}$$
$$subject\ to: \quad \delta(k+1) = A\delta(k) + B\delta_{cmd}(k) \qquad \textit{Eq. 2.13}$$
$$\underline{\delta}_{cmd}(k) \leq \delta_{cmd}(k) \leq \overline{\delta}_{cmd}(k)$$
$$\underline{\delta}(k) \leq \delta(k) \leq \overline{\delta}(k)$$

The only difference resides in how the second term has been formulated, that is in which way it has been given a cost to the utilization of the actuators. The reasons why the second term of the objective function has been formulated in this way will be explained during the description of the brake blending scenario (section 3.3).

2.4.1. Variables Description

The actuators that can be controlled in the vehicle are: 6 independent disc brakes, 1 engine with engine brake and the rear active steering for a total of eight controllable actuators.

The vehicle has pneumatic brakes, which means that compressed air is used to move the piston that, via the brake pad, generates the frictional force on the brake disc. A convenient way to describe the brakes as actuators is then by using the commanded pressure to the brake as control input and the actual pressure in the brake as output. Therefore the variables are defined as:

- $\delta_{cmd_i}(k), i = 1, \dots, 6$. Commanded pressure at the brake on the wheel i;
- $\delta_i(k), i = 1,\dots,6$. Actual pressure at the brake on the wheel i.

The variables are measured in bars $[bar]$.

The diesel engine with engine brake is connected to the driven axle via the gearbox and the open differential. Both gearbox and differential transform the torque provided by the engine into a suitable torque for the driven axle. The open differential splits the torque 50-50 between the two sides of the driven axle. The variables used to describe the powertrain as actuator are the requested torque at the driven axle and the actual torque present at the driven axle:

- $\delta_{cmd_7}(k)$ commanded torque at the driven axle;
- $\delta_7(k)$ actual torque at the driven axle.

The variables are measured in Newton metres $[Nm]$.

The rear active steering turns the wheels of the rear axle via the movement of a piston situated inside a chamber and directly connected to the rear axle. The piston divides the chamber into two volumes and compressed oil is sent to one of the two volumes of the chamber in order to move the piston and so the rear wheels. A controller receives as input the desired angle for the rear wheels and decides how much pressure to apply on the piston in order to move the wheels. Taking the entire system (controller plus piston) as a block, the variables used to describe the RAS system as actuator are:

- $\delta_{cmd_8}(k)$ commanded angle at the rear wheels;
- $\delta_8(k)$ actual angle at the rear wheels.

The variables are measured in radians $[rad]$ and the same angle for both wheels is assumed.

2.4.2. Effectiveness Matrix $\boldsymbol{B_f}$

The global forces chosen to control the planar dynamics of the vehicle are the resulting longitudinal force and yaw moment of the vehicle:

$$v = \begin{bmatrix} F_{X,tot} \\ M_{Z,tot} \end{bmatrix} \qquad Eq.\ 2.14$$

The total lateral force $F_{Y,tot}$ on the vehicle has not been included in the formulation because it was not of any use for the analysed scenarios.

B_f is the effectiveness matrix that maps the actuators usage into the global forces, hence $B_f \in \mathbb{R}^{2\,x\,8}$. To fill in the various elements of B_f the following relations between variables and forces have been considered.

The first approximation is that the wheels angles of the steering axles are small so that brakes and engine only produce longitudinal forces on the vehicle while the wheels on the first and third axle produce pure lateral forces on the vehicle when they are turned without braking on those wheels.

The vehicle is equipped with disc brakes, this means that the relation between applied pressure and generated moment on the wheel is linear:

$$\Gamma_i = k_b \delta_i \quad i = 1, \dots, 6 \qquad Eq.\ 2.15$$

where k_b is a constant expressed in $\left[\frac{Nm}{bar}\right]$ and Γ_i is the moment generated on the wheel i.

The braking force produced by each brake can then be expressed as:

$$F_{x_i} = \left(\frac{k_b}{r_j}\right)\delta_i \quad i = 1, \dots, 6 \;\; j = 1,2,3 \qquad Eq.\ 2.16$$

where r_j is the effective wheel radius of the wheels at the axle j.

Considering the engine torque and its relative torque at the driven axle modified by the gearbox and differential gear ratio, the longitudinal force produced by the driven axle is expressed as:

$$F_{x_7} = \frac{1}{r_2}\delta_7 \qquad \textit{Eq. 2.17}$$

where F_{x_7} includes the forces generated on both sides of the driven axle, that is on the wheels 3 and 4.

The formulation used for the lateral forces produced by RAS has required a more complicated model. A nowadays widely used model to describe the forces produced by tyres is the so called Magic Formula by Hans B. Pacejka. The Pacejka's tyre model has been used as basis to describe the relation between the actuator output δ_8 and forces produced by RAS.

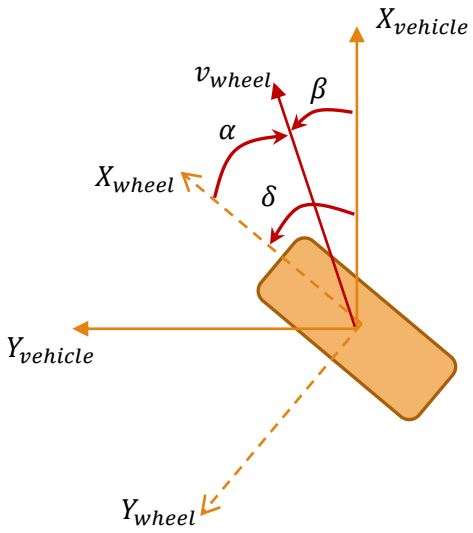

Figure 2.3. Illustration of the slip angle α, the side-slip angle β and the wheel angle δ.

The key variable to describe lateral forces in the Magic Formula is the slip angle, defined as:

$$\alpha = \beta - \delta \qquad \textit{Eq. 2.18}$$

where α is the slip angle, β is the side-slip angle and δ is the wheel angle. These angles are illustrated in Figure 2.3 where v_{wheel} is the velocity of the centre of the wheel.

From Figure 2.3 it is clear that the slip angle can also be defined as:

$$\alpha = \arctan\left(\frac{v_{y,wheel}}{v_{x,wheel}}\right) \qquad \textit{Eq. 2.19}$$

The fundamental result of the Magic Formula for lateral forces is to describe the magnitude of the lateral force produced by a wheel as a function of its slip angle:

$$F_y = -D_y \sin[C_y \arctan\{B_y\alpha - E_y(B_y\alpha - \arctan(B_y\alpha))\}] \qquad \textit{Eq. 2.20}$$

From here, if the value of α is small, the approximations $\arctan(x) \approx x$ and $\sin(x) \approx x$ hold and the relation in (Eq. 2.20) can be rewritten as:

$$F_y = -D_y C_y B_y \alpha = -C_\alpha \alpha \qquad \textit{Eq. 2.21}$$

where C_α is commonly defined as the cornering stiffness of the wheel. Both equations are shown in Figure 2.4 when the friction coefficient is $\mu = 1$ and the load on the wheel is $F_z = 35\ kN$.

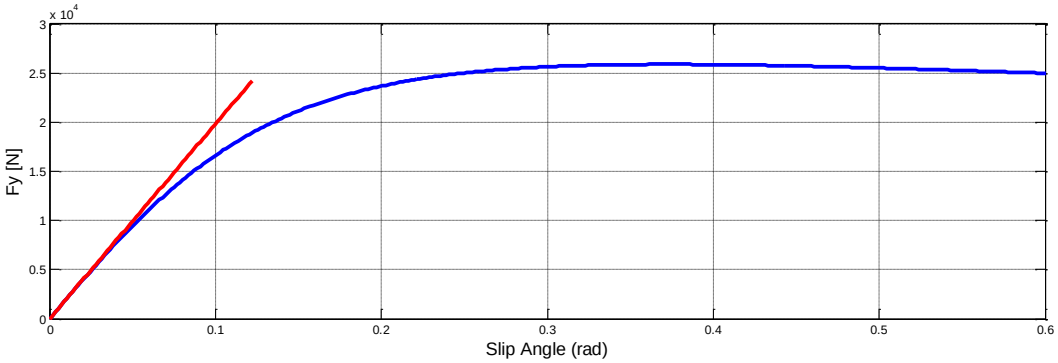

Figure 2.4. Relation between lateral force and slip angle for both the formulations in Eq. 2.20 and Eq. 2.21.

In case one of the RAS wheels is in contact with a low friction surface, the characteristic $F_y(\alpha)$ significantly changes, Figure 2.5 shows the characteristics for the same wheel with the same load ($F_z = 35\ kN$) but different friction coefficients $\mu_1 = 0.1$ and $\mu_2 = 0.7$.

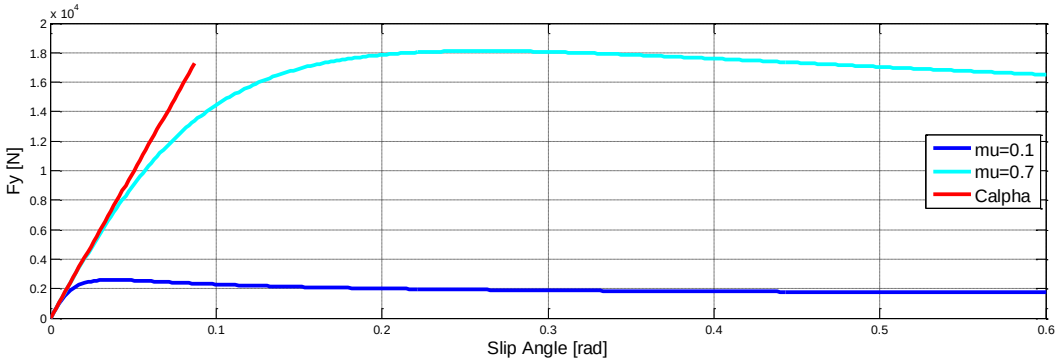

Figure 2.5. Lateral force characteristics for $\mu_1 = 0.1$ and $\mu_2 = 0.7$.

From Figure 2.5 it is straightforward to notice that, for the wheel on low friction surface, the approximation $F_y = -C_\alpha \alpha$ holds only for really small values of α. In order to overcome this problem, the following equations have been used, in general, to describe the lateral forces produced by the wheels:

$$\begin{cases} F_y = -C_\alpha \alpha & if \quad -D_y \leq -C_\alpha \alpha \leq D_y \\ F_y = D_y & if \quad -C_\alpha \alpha > D_y \\ F_y = -D_y & if \quad -C_\alpha \alpha < -D_y \end{cases} \qquad \textit{Eq. 2.22}$$

where $D_y = \mu_y F_{z_i}$ is the peak value in the Magic Formula. (Eq. 2.22) is shown graphically in Figure 2.6 for $\alpha > 0$.

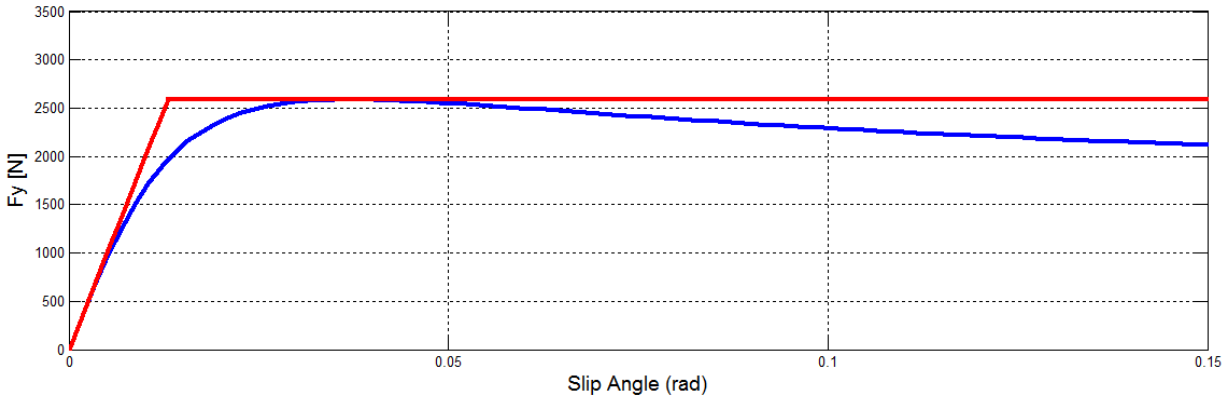

Figure 2.6. Linearization of the lateral forces.

During the considered scenarios, no steady-state curves have been analysed. In such a situation, the side slip angle is almost zero, that is $\alpha = \beta - \delta \approx -\delta$, and (Eq. 2.22) can be rewritten as:

$$\begin{cases} F_y = C_\alpha \delta & if \quad -D_y \leq C_\alpha \delta \leq D_y \\ F_y = D_y & if \quad C_\alpha \delta > D_y \\ F_y = -D_y & if \quad C_\alpha \delta < -D_y \end{cases} \qquad Eq.\ 2.23$$

As long as both wheels of the RAS do not reach their peak value, the relation between RAS wheels angle and lateral force is linear:

$$F_{y_8} = (C_{\alpha_5} + C_{\alpha_6})\delta_8 \qquad Eq.\ 2.24$$

where C_{α_i} is the cornering stiffness at the wheel i. When one of the wheels reaches its peak value, that wheel will approximately contribute only with a constant value to the lateral force produced by the RAS. If, for example, the sixth wheel reaches its peak value, the lateral force will become:

$$F_{y_8} = C_{\alpha 5}\delta_8 + D_{y_6} \qquad Eq.\ 2.25$$

Those equations have been used to describe the relation between the considered variables of the actuators and the force that they can produce.

With these relations in mind it is straightforward to find the correspondence between v and δ. In fact, using the linear momentum and angular momentum theorems to calculate the total longitudinal force and yaw moment on the vehicle that the actuators can produce, one gets:

$$\begin{cases} F_{x,tot} = F_{x_1} + F_{x_2} + F_{x_3} + F_{x_4} + F_{x_5} + F_{x_6} + F_{x_7} \\ M_{z,tot}(COG) = -F_{x_1}\left(\frac{w_1}{2}\right) + F_{x_2}\left(\frac{w_1}{2}\right) - F_{x_3}\left(\frac{w_2}{2}\right) + F_{x_4}\left(\frac{w_2}{2}\right) \\ \qquad -F_{x_5}\left(\frac{w_3}{2}\right) + F_{x_6}\left(\frac{w_3}{2}\right) - F_{y_8}L_r \end{cases} \qquad Eq.\ 2.26$$

where w_j is the track width of the axle j and L_r is the distance between the third axle and the updated centre of gravity of the vehicle, as explained in Appendix B **– Parameters**. It is then simple to find out what is the structure of the effectiveness matrix B_f:

$$B_f = \begin{bmatrix} \frac{k_b}{r_1} & \frac{k_b}{r_1} & \frac{k_b}{r_2} & \frac{k_b}{r_2} & \frac{k_b}{r_3} & \frac{k_b}{r_3} & \frac{1}{r_2} & 0 \\ -\frac{k_b}{r_1}\frac{w_1}{2} & \frac{k_b}{r_1}\frac{w_1}{2} & -\frac{k_b}{r_2}\frac{w_2}{2} & \frac{k_b}{r_2}\frac{w_2}{2} & -\frac{k_b}{r_3}\frac{w_3}{2} & \frac{k_b}{r_3}\frac{w_3}{2} & 0 & -(C_{\alpha_5} + C_{\alpha_6})L_r \end{bmatrix} \qquad Eq.\ 2.27$$

In the last element of the matrix (element (2,8)) C_{α_5} and/or C_{α_6} are set to zero if the wheel has reached its peak value. If this happens, the constant moment produced by that wheel on the vehicle is taken into account with the use of a constant term that is directly subtracted to $v(2)$.

2.4.3. Constraints

The minimization function has to deal with three different types of constraints:

1. Constraints on the actuators dynamics: $\delta(k+1) = A\delta(k) + B\delta_{cmd}(k)$
2. Constraints on the control inputs: $\underline{\delta}_{cmd}(k) \leq \delta_{cmd}(k) \leq \overline{\delta}_{cmd}(k)$
3. Constraints on the actuators outputs: $\underline{\delta}(k) \leq \delta(k) \leq \overline{\delta}(k)$

All these constraints have to be simultaneously satisfied in the solution of the minimization problem.

In order to define the constraints on the actuators dynamics, a model of the actuators has been built. The first observation is that the dynamics of every actuator is independent of the states or inputs of the other actuators. Secondly, the states defined for the models are the same as the outputs of the actuators.

The behaviour of every actuator has been modelled as a first order system:

$$W(s) = \frac{K}{\tau s + 1} \qquad \textit{Eq. 2.28}$$

where $W(s)$ is the transfer function from $\delta_{cmd_i}(k)$ to $\delta_i(k)$, τ is the time constant of a specific actuator and K accounts for possible errors of $\delta_i(k)$ in reaching the steady-state value.

The models of the actuators have been based on real data collected from a heavy vehicle. The data have been collected sending a step input to a specific actuator (red lines of Figure 2.7 and Figure 2.8) and observing its output over a time horizon (blue lines of Figure 2.7 and Figure 2.8). The parameters that define the behaviour of the first order system have been found using a standard Least Square method:

$$(K, \tau) = argmin \left\| f\left(K, \tau, t, \delta_{cmd_i}\right) - y_{data} \right\|_2^2 \qquad \textit{Eq. 2.29}$$

where $f(K, \tau, t, \delta_{cmd})$ is the set of possible time responses to the input δ_{cmd_i}, parametrized by K and τ. y_{data} is the log data collected from the vehicle.

Figure 2.7 shows the response read from the sensor situated in the brake chamber when a step input is sent to the brake as commanded input. The figure also shows the response of the first order system model to the same input. The data have been analysed for every brake and no significant differences have been noted for the time responses of the different brakes when the same step input has been sent to the different brakes. On the other hand, little changes in the value of τ_b, the brakes time constant, have been observed when different magnitudes of the step input have been sent to the same brake. The final value for the brakes time constant τ_b has been chosen as an average of all these values.

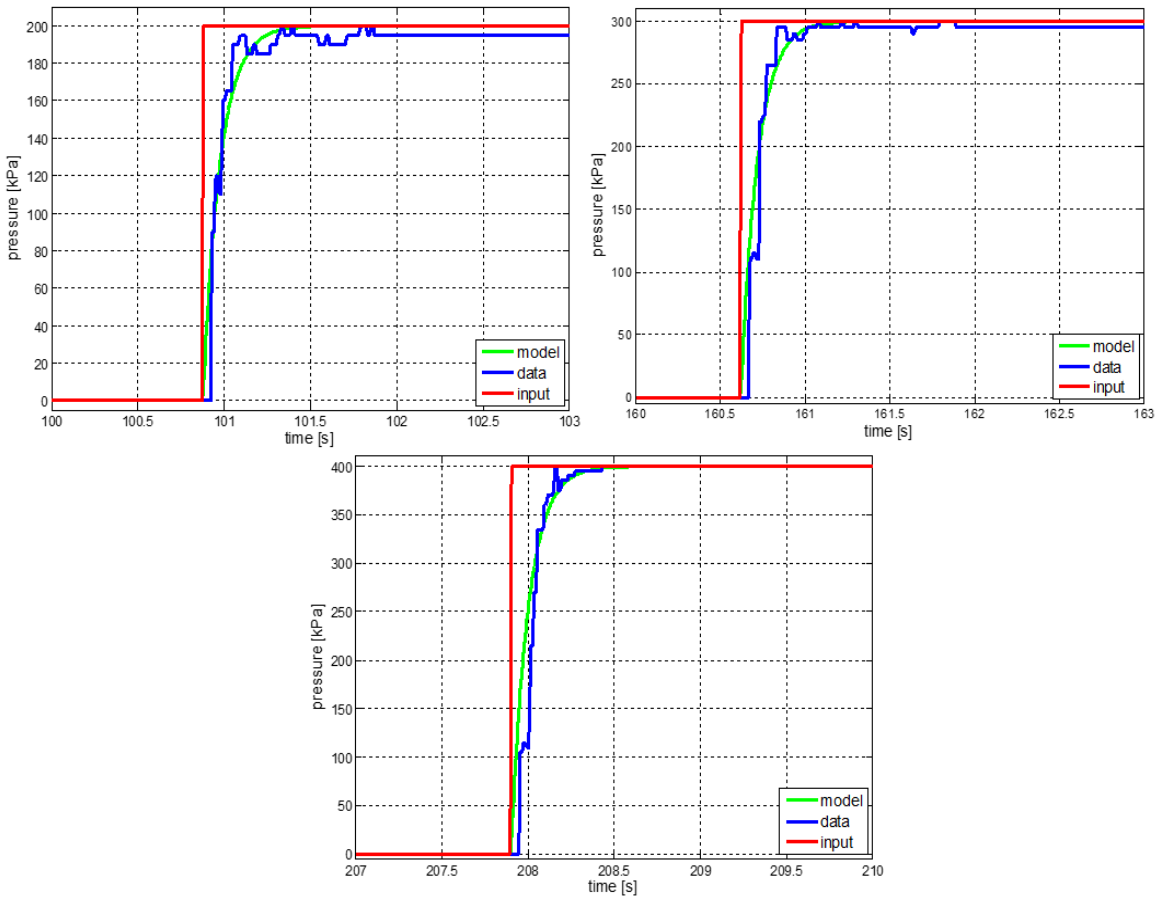

Figure 2.7. Model of the brakes (green) vs real data (blue). The commanded input (red) is 2, 3 and 4 bar.

Figure 2.8 shows the real step response of RAS and the step response that comes from the first order system model. As for the brakes, the RAS too presents slightly different time constants when different values are set for the step input. The time constant of the RAS, τ_{RAS}, has been chosen as the average of these values.

No data were available for the powertrain, so the model has not been based on real measurements. Nevertheless the engine as an actuator has been modelled as a first order system in previous works [9] [10] and it is known to be an actuator slower than the brakes and faster than the RAS. The powertrain time constant, τ_p, has been defined as an intermediate value between τ_b, the time constant of the brakes and τ_{RAS}, the time constant of the RAS.

Once the models of the actuators have been defined, they need to be discretized in order to be implemented in the MPCA. There are two parameters that play a key role in the MPCA performance: the sample time for the models discretization T and the number of steps that define the horizon of the objective function N.

The value of T defines the precision with which the continuous model is converted into a discrete model. If the model has been discretized with a small value of T, it is able to describe all the dynamics defined by the continuous model.

The time horizon N defines how far in the future the actuators outputs are considered by the controller. In particular, the product TxN determines what is the period of time that the controller takes into account during the minimization of the objective function. It is suitable to have a period of

time so that the dynamics of all the actuators can be observed until they are close to their steady-state value.

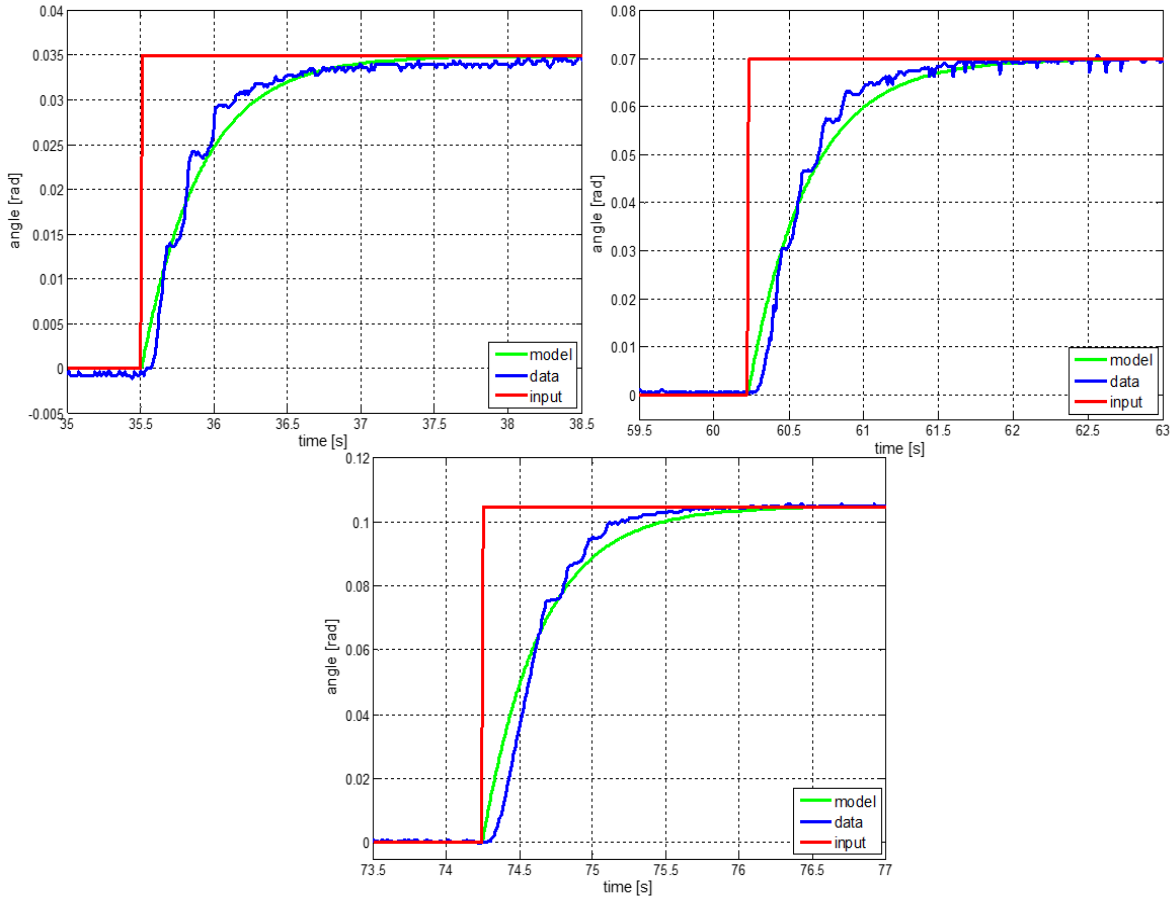

Figure 2.8. Model of RAS (green) vs real data (blue). The commanded input is 2, 4 and 6 degrees.

It is clear that, for a fixed N, having a small value for T makes the controller less predictive in the sense that the controller considers the actuators outputs only for a short period of time in the future. On the other hand, for a fixed T, increasing the value of N dramatically increases the computational time of the MPCA algorithm.

The values for T and N have been determined as a trade off among different instances: precision in the actuators dynamics description, period of time considered during the minimization and computational cost to solve the MPCA problem at every step.

The equivalent discrete-time system of (Eq. 2.28) can be written in the time domain as:

$$\delta_i(k+1) = \kappa\delta_i(k) + (1-\kappa)\delta_{cmd_i}(k) \qquad \textit{Eq. 2.30}$$

where:

$$\kappa = e^{-\left(\frac{\tau}{T}\right)} \qquad \textit{Eq. 2.31}$$

Combining the actuators together, the constraints have been defined as:

$$\delta(k+1) = A\delta(k) + B\delta_{cmd}(k) \quad k = 1, \dots, N \qquad \textit{Eq. 2.32}$$

where:

$$A = diag(\kappa_{br}, \kappa_{br}, \kappa_{br}, \kappa_{br}, \kappa_{br}, \kappa_{br}, \kappa_{pw}, \kappa_{RAS}) \qquad \textit{Eq. 2.33}$$

$$B = diag(1-\kappa_{br}, 1-\kappa_{br}, 1-\kappa_{br}, 1-\kappa_{br}, 1-\kappa_{br}, 1-\kappa_{br}, 1-\kappa_{pw}, 1-\kappa_{RAS}) \qquad \textit{Eq. 2.34}$$

$$\delta(k) = [\delta_1(k) \dots \delta_8(k)]^T; \quad \delta_{cmd}(k) = \left[\delta_{cmd_1}(k) \dots \delta_{cmd_8}(k)\right]^T \qquad \textit{Eq. 2.35}$$

with κ_{br},κ_{pw} and κ_{RAS} respectively the discrete time constant for brakes, powertrain and RAS.

These constraints are included in the MPCA formulation as a set of equality constraints. As the solution of the objective function fulfils this set of constraints, it is ensured that the dynamics of every actuator has been taken into account during the minimization.

The constraints on the commanded input deal with saturation limits of the actuators and limit the maximum usage of some actuators. In particular:

- As the maximum pressure that can be reached in the brakes chamber is $p_{\max}$, the commanded inputs have been limited to:

$$0 \leq \delta_{cmd_i}(k) \leq p_{\max} \quad i = 1, \dots, 6 \qquad \textit{Eq. 2.36}$$

- Considering dec_{en} as the maximum value for the deceleration when braking with the engine, the maximum torque that can be requested to the driven axle is the one that produces a deceleration equal to dec_{en}. The commanded input has then been limited as:

$$m_{veh} dec_{en} r_2 = \Gamma_{dec,\max} \leq \delta_{cmd_7}(k) \leq 0 \qquad \textit{Eq. 2.37}$$

  where m_{veh} is the total mass of the vehicle and r_2 is the dynamic radius of the second axle's wheels.

  When the vehicle is accelerating, the maximum torque that can be commanded to the driven axle depends on the maximum engine torque and the torque conversion made by gearbox and differential. The constraints for the engine when accelerating are then:

$$0 \leq \delta_{cmd_7}(k) \leq \Gamma_{acc,\max} = \Gamma_{en,\max}\, n_{gb_j} n_{df} \qquad \textit{Eq. 2.38}$$

  where $\Gamma_{en,\max}$ is the engine maximum torque, n_{gb_j} is the conversion made by the gearbox at the j-th gear and n_{df} is the torque conversion made by the differential.
- The RAS wheels angles have been limited within $[\gamma_{min}, \gamma_{max}]$. The range considered is wide enough to generate high lateral forces on the vehicle and it ensures that the linear approximations used to describe the lateral forces produced by the wheel ($F_y = C_\alpha \delta_8$, $F_y = \pm D_y$) are precise. The limitations are written as:

$$\gamma_{\min} \leq \delta_{cmd_8}(k) \leq \gamma_{\max} \qquad \textit{Eq. 2.39}$$

These constraints pertain the low level controller outputs and they ensure that the possible input values for the actuators never lay outside the above-defined range.

The third type of constraints, constraints on the actuators outputs $\underline{\delta}(k) \leq \delta(k) \leq \overline{\delta}(k)$, has been used to limit the amount of force generated at the wheel-ground interface and so to prevent the wheels from sliding. The relations between forces and actuators variables have already been illustrated in the "effectiveness matrix B_f" section 2.4.2.

In order to prevent the wheels form sliding, the friction ellipse of every wheel has been taken into account. The friction ellipse graphically explains how the longitudinal and lateral forces generated by a wheel can be combined together without making the wheel slip. The friction circle sets the maximum values for the total force produced by the wheel and it is the graphical representation of the following equation:

$$\frac{F_x^2}{D_x^2} + \frac{F_y^2}{D_y^2} \leq 1 \qquad \textit{Eq. 2.40}$$

where $D_x = \mu_x F_z$ and $D_y = \mu_y F_z$ are the peak values for the longitudinal and for the lateral forces, respectively, in the Pacejka's Magic Formula.

Obviously, the constraints in (Eq. 2.40) are not linear so they cannot be used in the MPCA formulation of (Eq. 2.13). To overcome this problem the idea is to approximate the friction ellipse with linear constraints.

The wheels of the first and third axle can produce lateral forces and negative longitudinal forces. The friction ellipse has then been approximated by linear constraints as shown in Figure 2.9

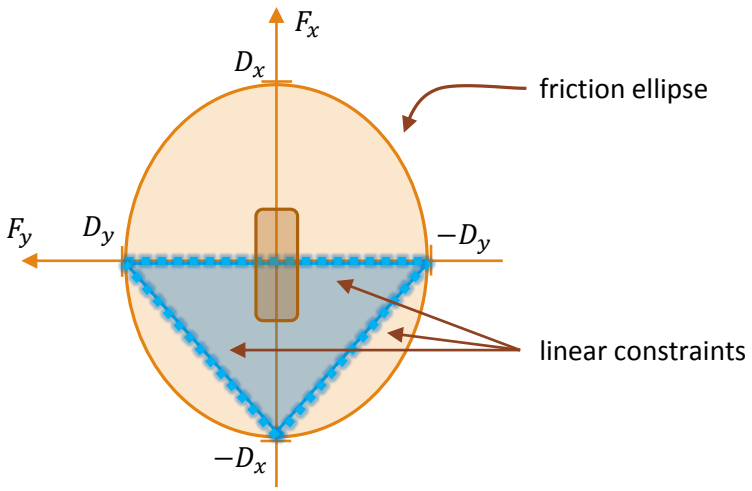

Figure 2.9. Approximation of the friction ellipse with three linear constraints.

The linear constraints set in the MPCA formulation approximate the lower semi-ellipse of the friction ellipse with the triangle inscribed in the semi-ellipse. The constraints have been defined as:

$$\begin{cases} F_{x_j} \leq 0 \\ F_{x_j} \geq \dfrac{D_{x_j}}{D_{y_j}} F_{y_j} - D_{x_j} \\ F_{x_j} \geq -\dfrac{D_{x_j}}{D_{y_j}} F_{y_j} - D_{x_j} \end{cases} \qquad j = 1,2,5,6 \qquad \textit{Eq. 2.41}$$

where j is the number of the wheel of the vehicle.

Regarding the second axle, it can produce both negative and positive longitudinal forces but it does not steer and so the constraints have been set as:

$$-D_{x_j} \leq F_{x_j} \leq D_{x_j} \qquad j = 3,4 \qquad \textit{Eq. 2.42}$$

From (Eq. 2.41), (Eq. 2.42) and knowing what is the force that every actuator can produce (Eq. 2.16), (Eq. 2.17), (Eq. 2.23), the constraints of (Eq. 2.41) and (Eq. 2.42) can be transformed into constraints on the actuators outputs:

$$wheel\ 1,2 \quad \begin{cases} \dfrac{k_b}{r_1}\delta(k)_i \leq 0 \ \rightarrow \ \delta(k)_i \leq 0 \\ \dfrac{k_b}{r_1}\delta(k)_i \geq \dfrac{D_{x_j}}{D_{y_j}} C_{\alpha_j} \delta_{SWA} - D_{x_j} \\ \dfrac{k_b}{r_1}\delta(k)_i \geq -\dfrac{D_{x_j}}{D_{y_j}} C_{\alpha_j} \delta_{SWA} - D_{x_j} \end{cases} \quad \begin{matrix} i = 1,2 \\ j = 1,2 \end{matrix} \qquad \textit{Eq. 2.43}$$

where δ_{SWA} is the angle of the first axle wheel steered by the driver. With this formulation it can be noted that priority has been given to the steering. δ_{SWA} is, in fact, an external constant coming from the steering wheel that is introduced in the constraints to limit the use of the brakes. It means that if the combined longitudinal and lateral forces of one wheel are at the limit of the approximated friction ellipse, in order to fulfil all the constraints, the brake pressure on that wheel will be decreased. The purpose is to ensure that, during a risky manoeuvre, the priority is given to the driver who has the possibility of controlling the vehicle through the steering wheel.

$$wheel\ 3,4 \quad \begin{cases} -D_{x_j} \leq \dfrac{k_b}{r_2}\delta(k)_i + \dfrac{0.5}{r_2}\delta(k)_7 \leq 0 \ \ when\ braking \\ 0 \leq \dfrac{k_b}{r_2}\delta(k)_i + \dfrac{0.5}{r_2}\delta(k)_7 \leq D_{x_j} \ \ when\ accelerating \end{cases} \quad \begin{matrix} i = 3,4 \\ j = 3,4 \end{matrix} \qquad \textit{Eq. 2.44}$$

In (Eq. 2.44), the forces produced by brakes and powertrain are combined together because they act on the same wheel. The term "0.5" is to take into account that an open differential has always been considered. In this case the torque available at the driven axle is always equally split between the two sides of the vehicle. Expression "when braking" means when the global longitudinal force requested by the high level controller (first element of v) is negative. Expression "when accelerating" means the opposite ($v(1) > 0$).

For the last axle, the simplest formulation is to set the constraints as in (Eq. 2.43) where δ_{SWA} is replaced by $\delta(k)_8$, the angle steered by the RAS. The weak point of this formulation is that the RAS wheels angle is the same for both wheels and thus the maximum value for $\delta(k)_8$ is limited by $D_{y_{min}}$.

Under the hypothesis $F_{z_5} = F_{z_6}$, $D_{y_{min}}$ is the maximum lateral force that the wheel with lower friction coefficient, $\mu_{\min} = \min(\mu_5, \mu_6)$, can reach.

In a situation where one wheel could produce much more lateral force than the other wheel (e.g. $F_{z_5} = F_{z_6}\ and\ \mu_5 \gg \mu_6$), it has been noted that it is useful to let one wheel saturate and limit the maximum value for $\delta(k)_8$ to the wheel that can generate greater lateral force. Following the example, the formulation for the wheel 5 is then:

$$wheel\ 5 \begin{cases} \frac{k_b}{r_3}\delta(k)_5 \leq 0 \ \rightarrow\ \delta(k)_5 \leq 0 \\ \frac{k_b}{r_3}\delta(k)_5 \geq \frac{D_{x_5}}{D_{y_5}} C_{\alpha_5}\delta(k)_8 - D_{x_5} \\ \frac{k_b}{r_3}\delta(k)_5 \geq -\frac{D_{x_5}}{D_{y_5}} C_{\alpha_5}\delta(k)_8 - D_{x_5} \end{cases} \qquad Eq.\ 2.45$$

These constraints limit $\delta(k)_8$ based on the peak value D_{y_5} of the wheel with high friction coefficient. To take into account that the other wheel too is generating some lateral force and then to limit the amount of braking force on that wheel, the following constraints are set for the wheel 6:

$$wheel\ 6 \begin{cases} \frac{k_b}{r_3}\delta(k)_6 \leq 0 \ \rightarrow\ \delta(k)_6 \leq 0 \\ \frac{k_b}{r_3}\delta(k)_6 \geq F_{y_6} - D_{x_6} \\ \frac{k_b}{r_3}\delta(k)_6 \geq -F_{y_6} - D_{x_6} \end{cases} \qquad Eq.\ 2.46$$

where F_{y_6} is defined as

$$F_{y_6} = \begin{cases} C_{\alpha_6}\delta(0)_8 & if\ -D_{y_6} \leq C_{\alpha_6}\delta(0)_8 \leq D_{y_6} \\ D_{y_6} & if\ C_{\alpha_6}\delta(0)_8 > D_{y_6} \\ -D_{y_6} & if\ C_{\alpha_6}\delta(0)_8 < -D_{y_6} \end{cases} \qquad Eq.\ 2.47$$

and where $\delta(0)_8$ is the value of the wheels angle currently read by the sensor. In case $\mu_6 > \mu_5$ the constraints are dynamically set in the opposite way before computing the MPCA problem.

The general formulation is then:

$$\begin{array}{l} if\ D_{y_5} \geq D_{y_6} \rightarrow a_1 = b_2 = 1; a_2 = b_1 = 0 \\ if\ D_{y_5} < D_{y_6} \rightarrow a_1 = b_2 = 0; a_2 = b_1 = 1 \end{array} \qquad Eq.\ 2.48$$

$$wheel\ 5 \begin{cases} \frac{k_b}{r_3}\delta(k)_5 \leq 0 \ \rightarrow\ \delta(k)_5 \leq 0 \\ \frac{k_b}{r_3}\delta(k)_5 \geq a_1\frac{D_{x_5}}{D_{y_5}} C_{\alpha_5}\delta(k)_8 + a_2 F_{y_5} - D_{x_5} \\ \frac{k_b}{r_3}\delta(k)_5 \geq -a_1\frac{D_{x_5}}{D_{y_5}} C_{\alpha_5}\delta(k)_8 - a_2 F_{y_5} - D_{x_5} \end{cases} \qquad Eq.\ 2.49$$

$$wheel\ 6 \begin{cases} \dfrac{k_b}{r_3}\delta(k)_6 \leq 0 \ \rightarrow \ \delta(k)_6 \leq 0 \\ \dfrac{k_b}{r_3}\delta(k)_6 \geq b_1 \dfrac{D_{x_6}}{D_{y_6}} C_{\alpha_6}\delta(k)_8 + b_2 F_{y_6} - D_{x_6} \\ \dfrac{k_b}{r_3}\delta(k)_6 \geq -b_1 \dfrac{D_{x_6}}{D_{y_6}} C_{\alpha_6}\delta(k)_8 - b_2 F_{y_6} - D_{x_6} \end{cases} \qquad \textit{Eq. 2.50}$$

3. Scenarios

This chapter introduces the test cases that have been analysed in the thesis along with the expected behaviour of the MPCA algorithm.

Three scenarios have been analysed during this thesis, namely:

- Split-μ braking
- Split-μ acceleration
- Brake blending

For all these scenarios there is a need to coordinate the actuators of the vehicle in order to achieve determined global performance such as vehicle stability, stopping distance, requested acceleration, etc.

3.1. Split-$\boldsymbol{\mu}$ braking

When a vehicle comes across a split friction road, the wheels on one side of the vehicle are in contact with high friction ground while the wheels on the other side are in contact with low friction ground. Under such circumstances, the passengers of the vehicle will face a dangerous situation if the vehicle has to brake in a short distance while maintaining the vehicle stability. Being able to successfully manage such a situation has gained more and more importance and it is now regulated by the Economic Commission for Europe.

The UNECE regulates the split-μ braking in the Regulation 13 – Annex 13. The Annex describes the conditions that the ABS system implemented on a heavy vehicle has to meet.

In particular, "*… when the right and left wheels of the vehicle are situated on surfaces with differing coefficient of adhesion (k_H and k_L) where $k_H \geq 0.5$ and $\frac{k_H}{k_L} \geq 2$, the directly controlled wheels shall not lock when the full force is suddenly applied on the control device at a speed of $50\ km/h$*".

Moreover, "*the braking rate (z_{MALS}) for laden power-driven vehicles shall be:*

$$z_{MALS} \geq 0.75\frac{4k_L + k_H}{5} \ \ and \ \ z_{MALS} \geq k_L" \qquad Eq.\ 3.1$$

With k_H, k_L respectively the side with high and low coefficient of adhesion and the braking rate defined as $z = T_b/F_z$, T_b = brake force at tyre/road interface, F_z = normal reaction of road surface on the vehicle under static conditions.

And "*During the tests … steering correction is permitted, if the angular rotation of the steering control is within $120°$ during the initial two seconds, and no more than $240°$ in all.*"

From the regulation it is clear that the two key parameters to consider when evaluating the efficacy of a split-μ braking are the generated braking force and the effort made by the driver in order to maintain the stability of the vehicle. The vehicle should be equipped with a system that minimizes the stopping distance while not demanding too much effort from the driver.

The MPCA algorithm has been designed so that it can cope with split-μ braking. The idea behind the algorithm is to exploit the brakes on the high friction side to generate the majority of the requested braking force while using the RAS to maintain the stability of the vehicle. The RAS angle is limited by the wheel on the high friction side so that a greater amount of lateral force can be generated on that wheel.

During this manoeuvre, it is expected that the RAS wheels start turning to one side in order to counteract the yaw moment generated by the brakes. At some point the wheel on the low friction side will saturate and the wheel will approximately generate the same constant amount of lateral force $D_{y_{min}}$. This force no longer depends on the RAS angle $\delta(k)_8$, so the yaw moment that the wheel produces on the vehicle is treated as a constant, called ef, in the MPCA formulation of (Eq. 2.13). The objective function of (Eq. 2.13) can then be seen as:

$$\delta^*_{cmd}(0) = argmin \sum_{k=1}^{N} \left\| B_f \delta(k) + ef - v \right\|^2_{Wv} + \gamma \sum_{k=0}^{N-1} \left\| C_f \delta(k) + \delta_e \right\|^2_{W_\delta} \qquad \textit{Eq. 3.2}$$

where $ef = \left[0 \ \pm D_{y_{min}} L_r\right]^T$ takes into account the moment generated by the constant lateral force of the saturated wheel during the manoeuvre. As long as the wheel does not saturate it is $ef = [0\ 0]^T$ and the yaw moment generated by both wheels is taken into account in $B_f \delta(k) = v$.

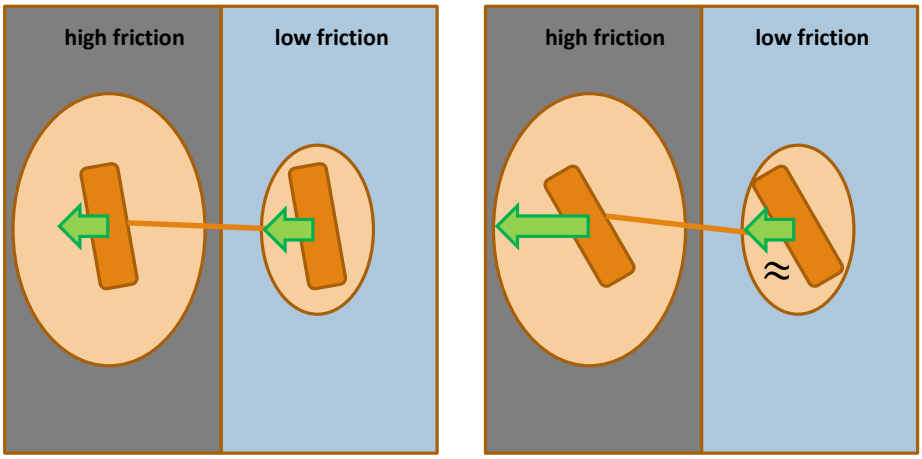

Figure 3.1. Difference in the generated lateral forces by the RAS when the RAS angle is limited by the wheel on the low friction side (left) and when it is limited by the wheel on the high friction side (right).

3.2. Split-μ acceleration

The second scenario considered is when the vehicle tries to accelerate from a standstill situation. The nature of the constraints in the MPCA formulation already ensures that the torque sent to the driven axle always prevents the wheels form slipping.

Although the driver asks for a large amount of longitudinal force, the global longitudinal force reached during the minimization of the objective function in (Eq. 2.13), that is the first element of $B_f \delta(k)$, is such that the wheels of the driven axle do not slip.

Nevertheless, another case that is of interest in today's vehicles is the vehicle acceleration from a standstill on a split friction road. The vehicle described in section 1.2 is equipped with an open differential, such a differential limits the torque at the driven axle to the maximum amount of torque that can receive the wheel on the low friction side, multiplied by 2.

The torque transmitted to the ground could then be insufficient to move the vehicle. One solution to overcome the problem is to create a resistant moment on the lower traction wheel so that more torque can be transmitted to the ground through the wheel in contact with high friction ground.

One way to create this resistant moment is to use the brake of the low friction wheel in order to counteract the tendency of the wheel to slip, this is the so called Traction-Control-by-Brake.

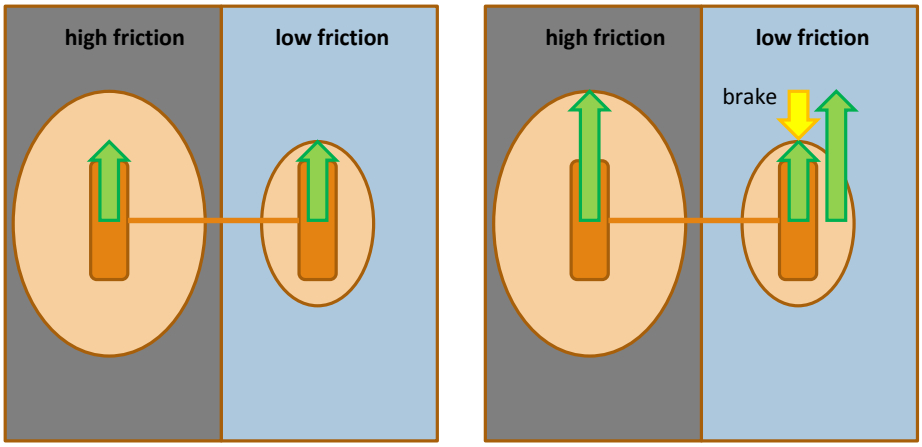

Figure 3.2. Maximum force that can be transmitted to the ground without using the brakes (left) and with the use of the brakes (right).

This is how the MPCA controller manages to cope with a split-μ acceleration from a standstill situation. Suppose that the right wheel at the driven axle has low friction surface while the wheel on the left side has high friction surface. If no brakes are used, the constraint:

$$0 \leq \frac{k_b}{r_2}\delta(k)_4 + \frac{0.5}{r_2}\delta(k)_7 \leq D_{x_4} \qquad \textit{Eq. 3.3}$$

becomes:

$$0 \leq \frac{0.5}{r_2}\delta(k)_7 \leq D_{x_4} \qquad \textit{Eq. 3.4}$$

The maximum torque that can be sent to the driven axle $\delta(k)_7$ is limited by 2 times the maximum torque that the lower traction wheel can transmit to the ground D_{x_4}. This limit is acceptable as long as the global requested force satisfies $v(1) = F_{x,tot} \leq 2D_{x_4}r_2$.

If $F_{x,tot} > 2D_{x_4}r_2$, the braking force in (Eq. 3.3) has to be used in order to increase the maximum value achievable by $\delta(k)_7$ and so succeed in generating the requested $F_{x,tot}$. The constraint is

$$\delta(k)_7 \leq 2(D_{x_4}r_2 - k_p\delta(k)_4) \qquad \textit{Eq. 3.5}$$

As soon as $\delta(k)_4 \neq 0$, $\delta(k)_7$ can be increased and half of the increment is transmitted to the ground through the higher traction wheel. The difference between the longitudinal forces produced at the two sides of the driven axle $\Delta F_{x,trac} = F_{x_3} - F_{x_4}$ depends on how much the brake has been used on the right side. Looking at Figure 3.2 it is clear that $\Delta F_{x,trac} = -\frac{k_b}{r_2}\delta(k)_4$ and that $\Delta F_{x,trac}$ generates a negative yaw moment $M_{z,trac} = -\Delta F_{x,trac}\frac{w_2}{2}$.

As $M_{z,trac} = \frac{k_p}{r_2}\frac{w_2}{2}\delta(k)_4$, the magnitude of the yaw moment is automatically taken into account in B_f as soon as $\delta(k)_4 \neq 0$ and it is compensated by using the RAS. In fact, as no other brakes are used, the explicit expression for the second element of $B_f\delta(k)$ is:

$$B_f\delta(k)(2) = \frac{k_b}{r_2}\frac{w_2}{2}\delta(k)_4 + \left(C_{\alpha_5} + C_{\alpha_6}\right)\delta(k)_8 \qquad \textit{Eq. 3.6}$$

And $\delta(k)_8$ can be used to have $v(2) = M_{z,tot} = 0$.

The downside of this strategy is that the brake can undergo wearing problems if it is used too extensively. For this reason, when accelerating, the method is limited up to the speed of $20\ km/h$. During an acceleration, if $v_{veh} > 20\ km/h$, the brakes are no more allowed to be used and the maximum torque sent to the driven axle is limited by (Eq. 3.4).

3.3. Brake Blending

The term brake blending is understood here as the combined use of engine brake and disc brakes in order to produce the global desired longitudinal force. The analysed situation is a typical mild braking on a normal road, e.g. a braking event that does not saturate any actuator while the vehicle is driving on a dry asphalt road. The idea is to receive from the driver a desired deceleration for the vehicle, transform it into a suitable value for $v(1) = F_{x,tot}$ and let the low level controller decide how to coordinate the actuators in order to satisfy the driver's desires while prioritizing the use of some actuators.

The prioritization is made by assigning suitable values to the scalar γ and the weighting matrix W_δ. Three aspects have been considered to decide how to prioritize the actuators:

1. The desired deceleration should be reached as fast as possible.
2. The engine brake should be used as much as possible because it does not present the problems of a typical disc brake: wear and fading.
3. The wheels should brake proportionally to their available maximum braking force $D_{x_i} = \mu_{x_i}F_{z_i}$ to avoid wheel slip.

The first issue is solved by setting γ small so that the first term of the objective function has the precedence in the minimization. In particular, the minimization $B_f\delta(k)(1) - v(1)$ will have the priority on the second term of the objective function.

The second point in the list is accomplished by setting the weighting value for the powertrain equal to zero in the second term of the objective function: $W_\delta(7,7) = 0$. This means that if there are various solutions for $B_f\delta(k) = v$, the chosen solution will use the engine brake as much as possible because it is the only actuator that will not increase the cost of the objective function.

To deal with the last considered issue, C_f, δ_e and the remaining W_δ have been set as follows:

$$C_f = diag\left(\frac{k_b}{r_1}, \frac{k_b}{r_1}, \frac{k_b}{r_2}, \frac{k_b}{r_2}, \frac{k_b}{r_3}, \frac{k_b}{r_3}, 1, 1\right) \qquad \textit{Eq. 3.7}$$

$$\delta_e = \left[0\ 0\ 0.5\frac{\delta_7(0)}{r_2}\ 0.5\frac{\delta_7(0)}{r_2}\ 0\ 0\ 0\ 0\right]^T \qquad \textit{Eq. 3.8}$$

$$W_\delta = diag\left(\frac{1}{\mu_{x_1}F_{z_1}}, \frac{1}{\mu_{x_2}F_{z_2}}, \frac{1}{\mu_{x_3}F_{z_3}}, \frac{1}{\mu_{x_4}F_{z_4}}, \frac{1}{\mu_{x_5}F_{z_5}}, \frac{1}{\mu_{x_6}F_{z_6}}, 0, a_{RAS}\right) \qquad \textit{Eq. 3.9}$$

During a mild braking, the "less expensive" way to have $v(2) = M_{z,tot} = 0$ is to have symmetric braking on each axle and not to use the RAS. Moreover, during a mild braking, after some time $B_f\delta(k) \approx v$ without saturating any constraint, so the first term of the objective function is zero and only the second term is taken into account during the minimization.

In this situation the first six elements of $C_f\delta(k) + \delta_e$ are the braking forces F_{x_i} produced by the actuators on each wheel. The cost to use the engine brake is zero but the force it produces is taken into account in F_{x_4} and F_{x_5} by using δ_e. This is done to prioritize the use of the engine brake rather than the disc brakes but, at the same time, to take into account the combined forces produced by the engine brake and disc brakes on the second axle. The reason why C_f, δ_e and W_δ have been set with these values is that, in steady-state conditions, the objective function can then be seen as:

$$\delta^*_{cmd}(0) = argmin \sum_{i=1}^{6} \frac{F_{x_i}^2}{\mu_{x_i} F_{z_i}}$$
$$subject\ to \sum_{i=1}^{6} F_{x_i} = F_{x,tot} \qquad \textit{Eq. 3.10}$$

From here, the Lagrangian function associated to (Eq. 3.10) is:

$$\Lambda = \sum_{i=1}^{6} \frac{F_{x_i}^2}{\mu_i F_{z_i}} - \lambda(\sum_{i=1}^{6} F_{x_i} - F_{x,tot}) \qquad \textit{Eq. 3.11}$$

and the minimum coincides with the solution of the following system:

$$\begin{cases} \sum_{i=1}^{6} F_{x_i} = F_{x,tot} \\ 2\dfrac{F_{x_i}}{\mu_i F_{z_i}} = \lambda \end{cases} \qquad \textit{Eq. 3.12}$$

That gives:

$$F_{x_i} = \frac{\mu_i F_{z_i}}{\sum_{i=1}^{6} \mu_i F_{z_i}} F_{x,tot} \qquad \textit{Eq. 3.13}$$

This means that, in steady-state conditions, the commanded inputs are found so that the braking force on each wheel is proportional to the available amount of braking force on that wheel divided by the total amount of braking force available with the ground. This approach to distribute the braking force is the same used in today's heavy vehicle and it is particularly convenient because it ensures that no wheel starts slipping before all the other wheels reach their peak value $\mu_i F_{z_i}$.

4. Solver

This chapter starts with a summary of the methods used to solve quadratic programming problems, followed by a description of the algorithm used by the solver and a description of the considered cases.

4.1. $\boldsymbol{QP}$ background

One of the most important aspects related with the effectiveness of the MPCA algorithm is the solver used to compute the solution of the problem at every step. The MPCA problem as stated above (Eq. 2.13) is part of a larger family of problems, the quadratic programming problems. The quadratic programming problems are conventionally expressed as:

$$\begin{aligned} &minimize && \left(\frac{1}{2}\right) x^T Q x + q^T x \\ &subject\ to && Gx \leq h; \quad Ax = b \end{aligned} \qquad \textit{Eq. 4.1}$$

where $Q \in \mathbb{R}^{n\,x\,n}$ is a symmetric positive definite matrix, $q \in \mathbb{R}^n$, $G \in \mathbb{R}^{p\,x\,n}$, $h \in \mathbb{R}^p$, $A \in \mathbb{R}^{m\,x\,n}$ and $b \in \mathbb{R}^m$. There exist various methods to solve such a family of problems with different philosophies: *Active Set methods*, *Barrier Interior Point methods*, *Primal-dual Interior Point methods*.

An active set method uses a combinatorial approach to iteratively determine the set of constraints active at the optimum [11]. A set of constraints is said to be active when the equality in the constraint is satisfied for that x. The idea of the method is to solve an auxiliary objective function considering only the active set of constraints during the minimization. Once the solution is computed, it gives indications on how to proceed in order to find the solution for the original QP problem. In particular, the solution indicates if the optimum has been reached, if an active constraint has to be removed or how to iterate the procedure in order to get closer to the optimum.

Today interior-point methods are among the most widely used numerical methods for solving convex optimization problems [11] and therefore they have emerged as important and useful methods to solve QP problems.

Barrier interior point methods aim to remove the inequalities constraints in the minimization problem and solve it taking into account only the equality constraints. In order to do that a barrier function $\phi(x)$ is used and added to the original objective function:

$$\begin{aligned} &minimize && \left(\frac{1}{2}\right) x^T Q x + q^T x + \mu\phi(x) \\ &subject\ to && Ax = b \end{aligned} \qquad \textit{Eq. 4.2}$$

where the barrier function has to be $\phi(x) \to +\infty$ when $Gx > h$. If the value of μ can be reduced iteratively at every step, the various solutions $x(\mu)$ at every step tend to the optimal value for the original problem (Eq. 4.1) as μ approaches zero.

Probably the most important class of interior point methods is the so called primal-dual interior-point methods. The basic idea of primal-dual interior-point methods is to compute the KKT conditions using a modified version of the Newton's method. The KKT conditions for the problem stated above (Eq. 4.1) are:

$$\begin{gathered} Qx + q + G^T z + A^T y = 0 \\ Gx + s = h; \quad Ax = b \\ s \geq 0; \quad z \geq 0 \\ s_i z_i = 0 \end{gathered} \qquad \textit{Eq. 4.3}$$

where s denotes the slack variables for the inequality constraints and z is the vector of the associated Lagrange multipliers. The KKT equations represent necessary and sufficient conditions to find a solution for the quadratic program (Eq. 4.1). There are several approaches that iteratively solve the KKT conditions and therefore more than one primal-dual interior-point method has been proposed. In the following, a classical primal-dual path-following method is described, as it is the basis to explain the algorithm used by the software that solves the MPCA problem.

Primal-dual methods modify the basic Newton procedure by solving the following system:

$$\begin{gathered} Qx + q + G^T z + A^T y = 0 \\ Gx + s = h; \quad Ax = b \\ s > 0; \quad z > 0 \\ s_i z_i = \tau \end{gathered} \qquad \textit{Eq. 4.4}$$

Note that in (Eq. 4.4) only the last equation $s_i z_i \neq 0$ is different form (Eq. 4.3). The set of points that are solution of (Eq. 4.4) for every $\tau > 0$ is the central path C. As $\tau \to 0$, the points of C converge to a solution for the quadratic program (Eq. 4.1). The idea of primal-dual algorithms is to take Newton steps towards points in C rather than pure Newton steps towards a solution of (Eq. 4.3). This idea is motivated by the fact that it is usually possible to take longer steps when moving closer to the central path C.

This means that, at every iteration, the linear system that has to be solved is:

$$\begin{bmatrix} Q & 0 & G^T & A^T \\ 0 & Z & S & 0 \\ G & I & 0 & 0 \\ A & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta s^k \\ \Delta z^k \\ \Delta y^k \end{bmatrix} = \begin{bmatrix} -(A^T y + G^T z + Qx + q) \\ -SZe + \tau e \\ -(Gx + s - h) \\ -(Ax - b) \end{bmatrix} \qquad \textit{Eq. 4.5}$$

where:

$$S = diag(s_1, \dots, s_p); \quad Z = diag(z_1, \dots, z_p); \quad e = [1\ 1 \dots 1]^T$$

The variables are then updated:

$$\begin{aligned} &(x^{k+1}, s^{k+1}, z^{k+1}, y^{k+1}) \\ &\qquad \leftarrow (x^k, s^k, z^k, y^k) + \alpha_k (\Delta x^k, \Delta s^k, \Delta z^k, \Delta y^k) \end{aligned} \qquad \textit{Eq. 4.6}$$

with $\alpha_k \in [0,1]$ so that $s^{k+1} > 0$ and $z^{z+1} > 0$.

To describe the different possible search directions of the primal-dual method from the pure Newton step, two parameters are normally introduced:

- *Centering parameter* $\sigma \in [0,1]$
- *Duality measure parameter* $\mu \triangleq \frac{1}{p} \sum_{i=1}^{n} s_i z_i$

These two parameters replace τ in (Eq. 4.4) with $\tau = \sigma\mu$

Then, if $\sigma = 1$ the solution of the system (Eq. 4.5) defines a *centering direction*, that is a Newton step towards the point in C where:

$$s_i z_i = \mu \quad \forall i \qquad \textit{Eq. 4.7}$$

On the other side hand, if $\sigma = 0$ the solution of the system (Eq. 4.5) defines an *affine-scaling direction*, that is a standard Newton step towards the solution of (Eq. 4.3).

Primal-dual methods choose a convenient value of σ in $(0,1)$ depending on whether it is necessary to get closer to C or reduce the value of μ. The figure below illustrates the role of σ in a primal-dual method.

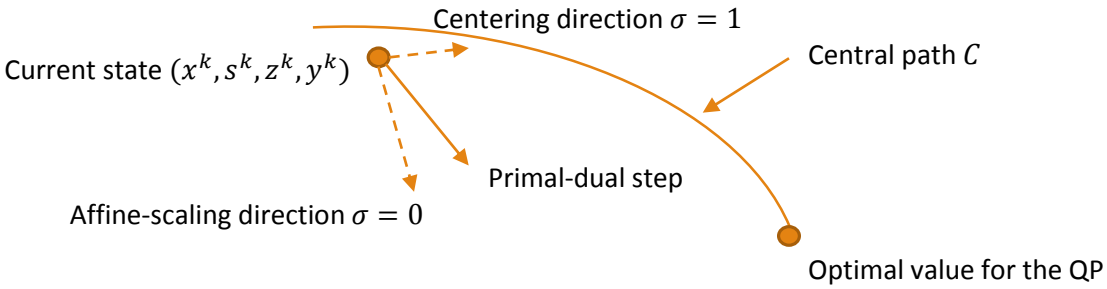

Figure 4.1. Conceptually, the role of σ in the search direction of a primal-dual method.

Computational experiments have proved that usually primal-dual methods are significantly more effective than other interior-point methods. As a consequence, nowadays many software packages implement a primal-dual strategy in their algorithms.

4.2. Solver description

In the last years several different software has been developed in order to implement the methods described above into reliable software algorithms: CVX, YALMIP, ACADO, MATLAB functions, CVXGEN, … .

In this thesis, CVXGEN has been chosen as solver for the MPCA problem. CVXGEN can automatically generate a custom solver for all those convex optimization problems that can be reduced to quadratic programming problems. It has been developed by Jacob Mattingley and Stephen Boyd as a new improved version of CVXMOD, an earlier less effective code generator software developed by the same authors.

It has been shown that the algorithm used in CVXGEN to solve the convex optimization problems has suitable properties regarding speed and robustness. Moreover, it is useful for fast implementations as it automatically generates code from a high-level description of the problem. These are the reasons that justify the choice of CVXGEN to deal with the MPCA problem in the thesis.

The high-level description consists of specifying the structure of the problem: dimensions, parameters, variables, objective function, without defining the values of the parameters. This allows to generate code for a whole family of problems that share the same structure. The language used in the high-level description is intuitive so, for example, the objective function (Eq. 2.13) is defined as:

```
minimize
  sum[t=1..T](quad(Bf*d[t]-v,Wv))+sum[t=1..T](quad(Cf*d[t]+de,Wd))
```

The code generated by CVXGEN is written in C. In the code, the parameters can be dynamically changed at every step in order to solve every time a different QP problem with the same structure.

As a first step, CVXGEN transforms the defined MPCA problem into a quadratic program that has the form of (Eq. 4.1). In order to do that all the variables $\delta(k)_i$ and $\delta(k)_{cmd_i}$ are vertically stacked into a unique variable x, and both the constraints and the objective function are rewritten in terms of Q, q, G, h, A and b.

Once the problem is in the canonical form (Eq. 4.1), a primal-dual interior-point method with Mehrotra's predictor-corrector is used to solve the quadratic program.

Mehrotra's algorithm generates a sequence of iterates (x^k, s^k, z^k, y^k) for which $(s^k, z^k) > 0$. At every step, the computed search direction depends on three different elements:

1. At the beginning an affine-scaling step is computed, that is finding a solution of (Eq. 4.5) with $\tau = 0$. This step is defined as the *predictor* for the algorithm.
2. Based on the predictor, a value for the centering parameter σ is chosen. The value of σ can change at every step.
3. In the end a *corrector* step is computed. The corrector tries to adjust the error that has been made in 1. when the solution for the nonlinear KKT conditions was found with a linear system approximation.

The idea behind 2. is to exploit the information from the predictor step in order to choose a convenient centering parameter for the current iteration. This means that if the predictor step manages to significantly reduce the duality measure μ, little or no centering is needed for the iteration. On the other hand, if no progress in reducing μ has been made, it is convenient to use the iteration to have at the next step a point close to the central path C, in this case the value of σ is close to 1 for the current iteration [12].

The drawback of this method is that it is necessary to solve two different linear systems in a single iteration. The first linear system accounts for the predictor step (point 1.), while the second one is solved to take into account the modifications of the predictor step made by the centering step and the corrector step. Luckily the centering step (point 2.) and the corrector step (point 3.) come from two independent linear systems and can be merged together in a unique linear system.

Following [13] the two linear systems that the algorithm has to solve at every step are:

$$\begin{bmatrix} Q & 0 & G^T & A^T \\ 0 & Z & S & 0 \\ G & I & 0 & 0 \\ A & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x^{aff} \\ \Delta s^{aff} \\ \Delta z^{aff} \\ \Delta y^{aff} \end{bmatrix} = \begin{bmatrix} -(A^T y + G^T z + Qx + q) \\ -SZe \\ -(Gx + s - h) \\ -(Ax - b) \end{bmatrix} \qquad \textit{Eq. 4.8}$$

The solution of the equation (Eq. 4.8) is the affine-scaling step defined in 1. while the solution of the following system (Eq. 4.9) is the centering-corrector step defined in 2. and 3.

$$\begin{bmatrix} Q & 0 & G^T & A^T \\ 0 & Z & S & 0 \\ G & I & 0 & 0 \\ A & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x^{cc} \\ \Delta s^{cc} \\ \Delta z^{cc} \\ \Delta y^{cc} \end{bmatrix} = \begin{bmatrix} 0 \\ \sigma\mu e - \Delta S^{aff} \Delta z^{aff} \\ 0 \\ 0 \end{bmatrix} \qquad \textit{Eq. 4.9}$$

where:

$$\mu = \frac{Sz}{p}; \quad \sigma = \left(\frac{\mu^{aff}}{\mu}\right)^3$$

and:

$$\mu^{aff} = \left(\frac{1}{p}\right)\left(s + \alpha\Delta s^{aff}\right)^T\left(z + \alpha\Delta z^{aff}\right);$$
$$\alpha = \max\{\alpha \in [0,1] \mid s + \alpha\Delta s^{aff} \geq 0, z + \alpha\Delta z^{aff} \geq 0\}$$

μ^{aff} can be thought of as the hypothetical value of μ that is reached when computing only an affine-scaling direction (Eq. 4.8) plus a line search.

Eventually, all the variables are updated:

$$\begin{aligned} x^{k+1} &= x^k + \alpha_k(\Delta x^{aff} + \Delta x^{cc}) \\ s^{k+1} &= s^k + \alpha_k(\Delta s^{aff} + \Delta s^{cc}) \\ y^{k+1} &= y^k + \alpha_k(\Delta y^{aff} + \Delta y^{cc}) \\ z^{k+1} &= z^k + \alpha_k(\Delta z^{aff} + \Delta z^{cc}) \end{aligned} \qquad \textit{Eq. 4.10}$$

with:

$$\alpha_k = \min\{1\,, 0.99\max\{\alpha_k \geq 0 \mid s^k + \alpha_k\left(\Delta s^{aff} + \Delta s^{cc}\right) \geq 0, z^k + \alpha_k(\Delta z^{aff} + \Delta z^{cc})\}\}$$

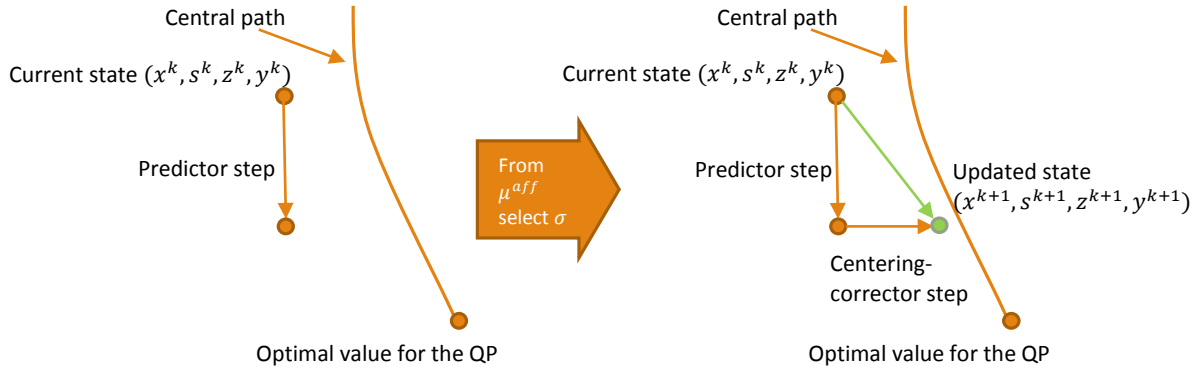

Figure 4.2. . Conceptually the elements that characterize Mehrotra's algorithm: a predictor step plus a centering-corrector step that together make the search direction at every step.

It is clear that the solving time for the algorithm depends almost entirely on the speed at which the two linear systems can be computed. The idea is to take advantage of the structure of the KKT matrix in order to find a solution via a permuted LDL^T factorization:

$$PKP^T = LDL^T \qquad \textit{Eq. 4.11}$$

where K is the KKT matrix defined in (Eq. 4.8) and (Eq. 4.9). The permutation matrix P is important because it defines the number and patterns of nonzero entries in L. This process, called fill-in, has a great impact on the time needed to solve the linear systems: the more zero entries L has, the faster. Finding a suitable matrix P on-line for every specified problem would be however computationally expensive. To save time and make the algorithm more efficient, the permutation matrix has to be chosen off-line. In order to make it possible, the following steps are performed.

First of all, the KKT matrix of the systems (Eq. 4.8) and (Eq. 4.9) is made symmetric so that it becomes:

$$\hat{K} = \left[\begin{array}{cc|cc} Q & 0 & G^T & A^T \\ 0 & S^{-1}Z & I & 0 \\ \hline G & I & 0 & 0 \\ A & 0 & 0 & 0 \end{array}\right] \qquad \textit{Eq. 4.12}$$

$\hat{K}$ is a symmetric quasi-semidefinite[1] matrix and it is not guaranteed to be factorizable. On the other hand, symmetric quasi-definite[2] matrices are guaranteed to be strongly factorizable [14]. A symmetric quasi-definite matrix can be obtained from (Eq. 4.12) through the following regularization:

$$\tilde{K} = \left[\begin{array}{cc|cc} Q & 0 & G^T & A^T \\ 0 & S^{-1}Z & I & 0 \\ \hline G & I & 0 & 0 \\ A & 0 & 0 & 0 \end{array}\right] + \left[\begin{array}{cc|cc} \epsilon I & & 0 & \\ & & & \\ \hline 0 & & -\epsilon I & \end{array}\right] \qquad \textit{Eq. 4.13}$$

with $\epsilon > 0$. The parameter ϵ used in the regularization is one of the settings that the user can change in CVXGEN (`settings.kkt_reg`) and in the thesis it has been set to 10^{-11}. The default value is 10^{-7}, however with the new set value the algorithm tends to converge in a smaller number of iterations for the MPCA problem considered.

As the two linear systems solved with $\tilde{K}$ differ from the two original systems defined with K, CVXGEN performs a number of refinement iterations that can be defined by the user (`settings.refine_steps`). The default value is 1 and no significant changes have been observed when modifying this setting. So, during the simulations, the default value has been used. The refinement steps aim to find a corrector for the previously computed solution with $\tilde{K}$.

4.3. Considered cases

Before implementing the CVXGEN code in the controller, it has been compared with the MATLAB function `quadprog()`. A condensed method has been used to convert the MPCA formulation into the QP canonical form and the default method of the `quadprog()` function has been used to solve the QP problem. The default method used by MATLAB was an active set method. The comparison of the two algorithms ensured that they converge to the same solution. It has also been noted that if CVXGEN does not converge it still manages to roughly follow the path of solutions found with `quadprog()`.

Two different algorithms have been generated using CVXGEN: an MPCA algorithm based on (Eq. 2.13) and a CA algorithm. The CA algorithm is a simplified version of the MPCA algorithm where no

[1] A symmetric matrix is quasi-semidefinite if it has the form:

$$K = \begin{bmatrix} E & A^T \\ A & F \end{bmatrix}$$

where *E* is symmetric positive-semidefinite and *F* is symmetric negative-semidefinite.

[2] A symmetric matrix is quasi-definite if it has the form:

$$K = \begin{bmatrix} E & A^T \\ A & F \end{bmatrix}$$

where *E* is symmetric positive-definite and *F* is symmetric negative-definite.

actuators dynamics are explicitly taken into account and no distinction between commanded values δ_{cmd} and actual values δ is made. The following equation, (Eq. 4.14), describes the CA formulation:

$$\delta = argmin \left\| B_f \delta - v \right\|_{W_v}^2 + \gamma \left\| C_f \delta + \delta_e \right\|_{W_\delta}^2 \qquad Eq.\ 4.14$$
$$subject\ to\ \underline{\delta} \leq \delta \leq \overline{\delta}$$

where the constraints take into account the rate limits of the actuators plus all the other constraints of the MPCA formulation but do not distinguish between δ_{cmd} and δ. The time horizon is not taken into account either in the objective function or in the constraints.

The following statistics directly taken from the CVXGEN web page after the generation of the two codes give an idea of the computational complexity that the MPCA formulation brings with it.

		MPCA	CA
Problem size	original variables	160	8
	variables in solver	180	10
KKT matrix	size	1100x1100	94x94
	original non-zeros	2532	230
	non-zeros after fill-in	3036	232
	fill-in factor	1.2	1.01

Table 4.1. Statistics of the two considered algorithms, the MPCA and CA.

Note that the size of the KKT matrix in the MPCA is bigger and thus more time is needed in order to solve the linear systems explained in section 4.2.

During the real tests explained in section 6.1 and with the set-up explained in section 6.1 some data have been collected for the time needed to the processor in order to solve the MPCA and the CA problem at every step. Figure 4.3 shows the different computational efforts required to the processor of MicroAutoBox.

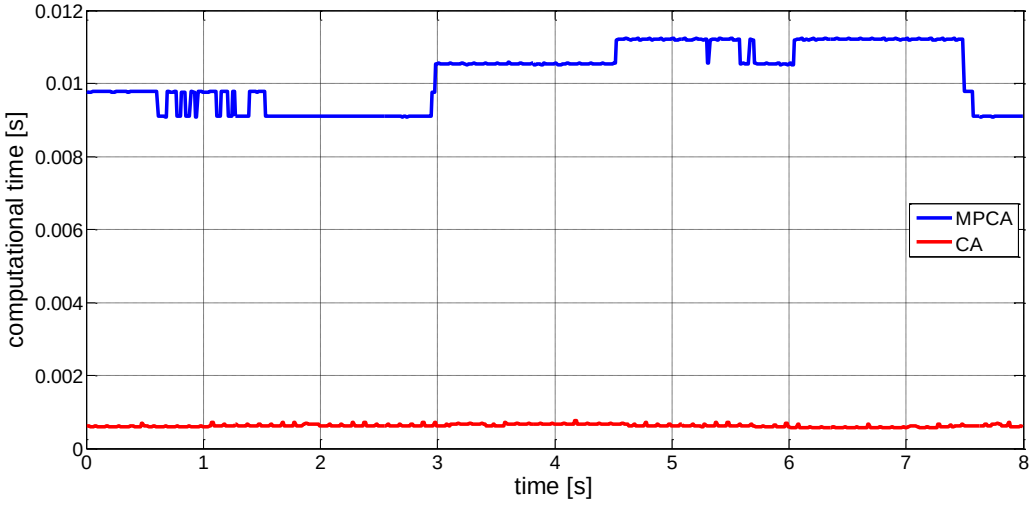

Figure 4.3. Different computational time of the two algorithms during the tests on the real vehicle.

5. Simulations

All the simulations have been performed in Simulink. The vehicle model is a non-linear model that meets the vehicle configuration described in section 1.2 and is part of the Volvo Transport Model library. The vehicle model includes all the important features needed to simulate the dynamics of a heavy vehicle (suspensions, body compliances, magic formula for the tyres model, …).

The low level controller has been implemented in Simulink as a MATLAB function. The MATLAB function, at every step, updates the parameters of the MPCA formulation and calls the execution of the CVXGEN solver in order to get the value of δ_{cmd} to send to the actuators.

The purpose of the simulations is to evaluate the predictive ability of the low-level controller, therefore no information on the vehicle dynamics, such as the yaw rate ($\dot{\psi}$) or the vehicle body side-slip angle (β_{veh}), has been fed back to the controller during the scenarios executions. This feed-forward nature of the controller permits to better reveal limitations and benefits of the predictions and to compare the performance with the CA formulation.

5.1. Split-μ braking

In this scenario, the vehicle drives on a straight but uneven road. Suddenly the vehicle has to perform an emergency braking and the controller has to stop the vehicle in a short distance without losing the stability of the vehicle. The initial speed for the vehicle is $v_0 = 50\ km/h$ while the road condition is $\mu_l = 0.7$, $\mu_r = 0.1$, where μ_l and μ_r are respectively the friction coefficient on the left and right side of the vehicle. The value of μ_l corresponds to a dry asphalt road, while μ_r to an icy road.

During this situation, it is interesting to understand how an unexperienced driver would react to the braking event. To simulate that, an external driver actuating on the steering wheel has been implemented as a smooth PID that tries to follow a straight path.

The following graphs show the behaviour of the vehicle and the driver effort on the steering wheel during the emergency braking.

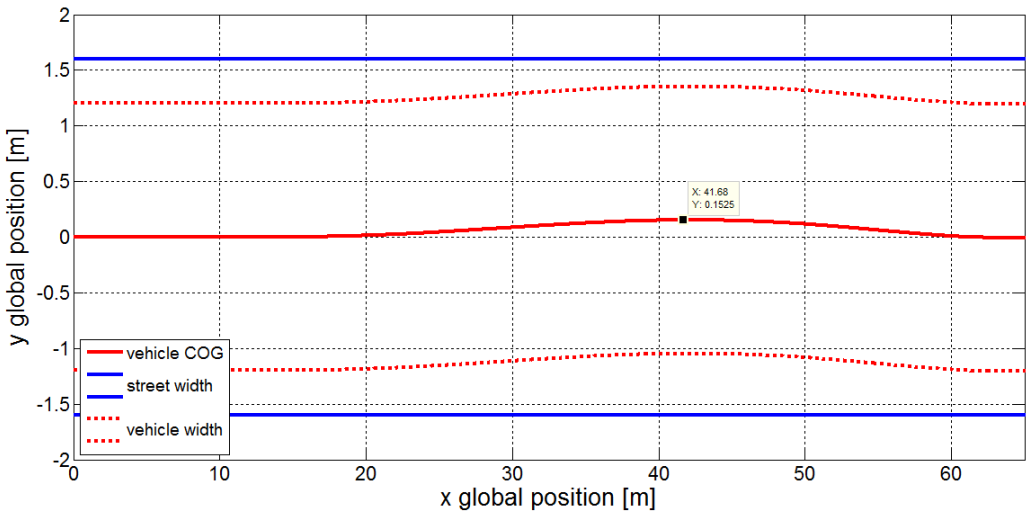

Figure 5.1. Lateral deviation of the vehicle during the braking. The red line represents the COG of the vehicle, the red dotted lines represent the width of the vehicle and the blue lines represent the street width.

In Figure 5.1 the blue lines represent the width of a normal road of $3.2\ m$, while the red dotted lines simulate the vehicle width of $2.4\ m$ and the red line the COG (centre of gravity) of the vehicle.

The braking starts at $14\ m$, and the maximum lateral deviation of the vehicle is less than $16\ cm$ and it is reached about $30\ m$ after the beginning of the braking. During the braking, the vehicle reaches the deceleration of $-0.2\ g$ and it stops in $7.2\ s$.

The blue line of Figure 5.2 represents the driver effort on the steering wheel in order to make the vehicle behave as shown in Figure 5.1.

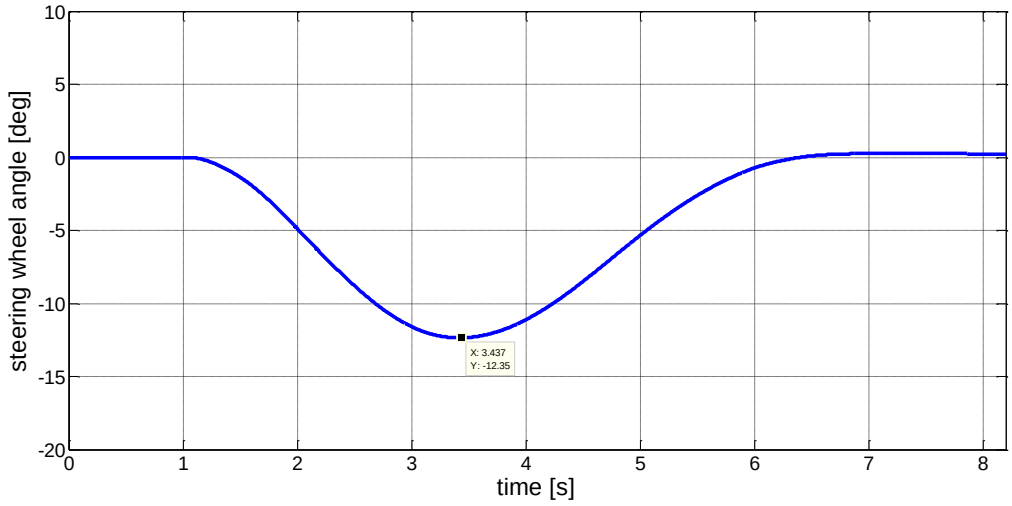

Figure 5.2. Driver effort on the steering wheel.

The braking starts at $1\ s$ and it can be noted that the maximum effort on the steering wheel is less than $15°$ clockwise that is performed by the driver during the first $2.4\ s$ of the manoeuvre.

The two figures highlight the ability of the controller to help the driver during a risky braking on an uneven road. The vehicle is stopped under the control of the driver and without requiring the driver any demanding manoeuvre.

It is clear that the requirements listed in section 3.1 are satisfied too. In fact, the effort on the steering wheel is much less than the established limit of $120°$ and it is kept beneath this limit even after $2\ s$ from the beginning of the braking event. The lower limit for the braking rate z_{MALS} is reached too. In fact, calculating the braking force T_b as:

$$T_b = \frac{1}{t_f - t_0}\int_{t_0}^{t_f} F_{x_{braking}}(t)dt \qquad Eq.\ 5.1$$

where t_0 is the instant when the braking starts and t_f is the time at which the vehicle speed is zero, the resulting braking rate is:

$$z_{MALS} = \frac{T_b}{F_z} = 0.212 \geq 0.165 = 0.75\frac{4\mu_r + \mu_l}{5} \qquad Eq.\ 5.2$$

which means that the requirement for the vehicle deceleration during the braking is met.

To compare the controlled vehicle with a standard vehicle, i.e. with a standard braking system, it has been set $W_v(2,2) = 0$. With this configuration, in fact, the controller ensures that the wheels do not slip but it does not take into account the yaw moment produced by the brakes on the left side. To make the comparison effective, the same braking force reached in the previous simulation has been requested to the vehicle and the same driver model has been used.

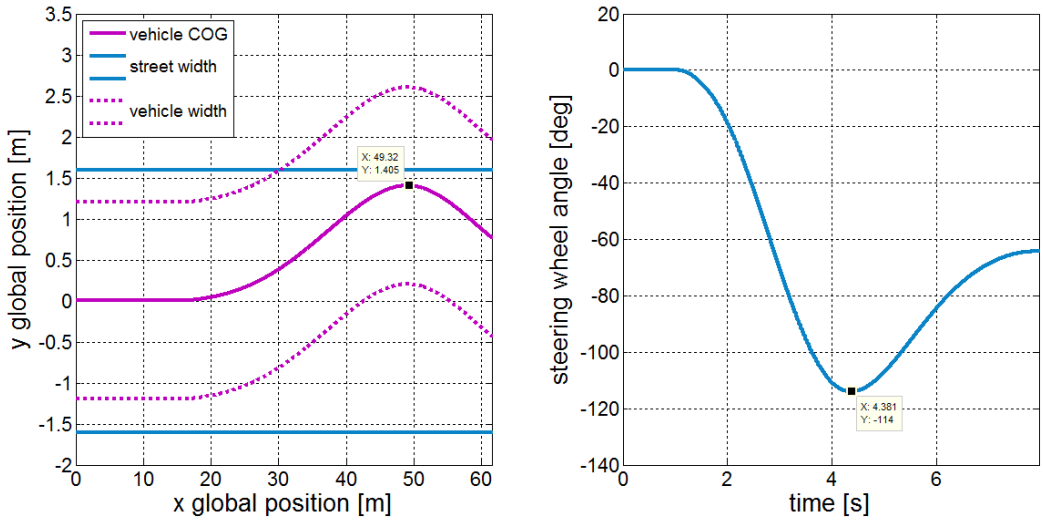

Figure 5.3. Vehicle behaviour during the braking (left) and driver effort on the steering wheel (right) when the controller does not take into account the yaw moment generated by the brakes.

Figure 5.3 shows that the value of the maximum lateral deviation has considerably increased from $0.16\ m$ to $1.4\ m$. A more aggressive driver would have managed to keep the vehicle lateral deviation within reasonable values, nevertheless the magnitude of the steering wheel angle would not have been changed. It can be observed that the driver has to turn the steering wheel by $114°$ in order to make the vehicle re-entry to the straight road.

Lately, the method used in the controller has been compared with a simplified version of itself. The new method is named Control Allocation (CA) in the following and it is based on the formulation of (Eq. 4.14). The idea is to have the same controller with the same constraints but without an explicit formulation of the actuators dynamics. As the vehicle has slow and fast actuators, it is essential to provide the new controller with some information on the actuators dynamics present on the vehicle. This is accomplished by limiting the rate at which the input value for the actuators can increase or decrease.

A rule on how to set the rate for the actuators does not exist, so the limit rate values have been gradually changed depending on the results of the simulations. It has been noted that, in order to have good performance with the CA formulation, the key value is the limit rate set for the RAS. The RAS, in fact, is the slowest actuator on the vehicle and is the responsible of counteracting the yaw moment generated by the brakes. It is then fundamental that the actual value for the RAS wheels angle δ_8 approximates as much as possible its commanded value δ_{cmd_8} at each time instant. If this is not the case, the CA acts as if the commanded input for the RAS is counteracting the brakes while the actual value of the RAS is lower and insufficient to keep the vehicle stable.

Figure 5.4 shows the different approach used by the two controllers when commanding the desired value for the RAS.

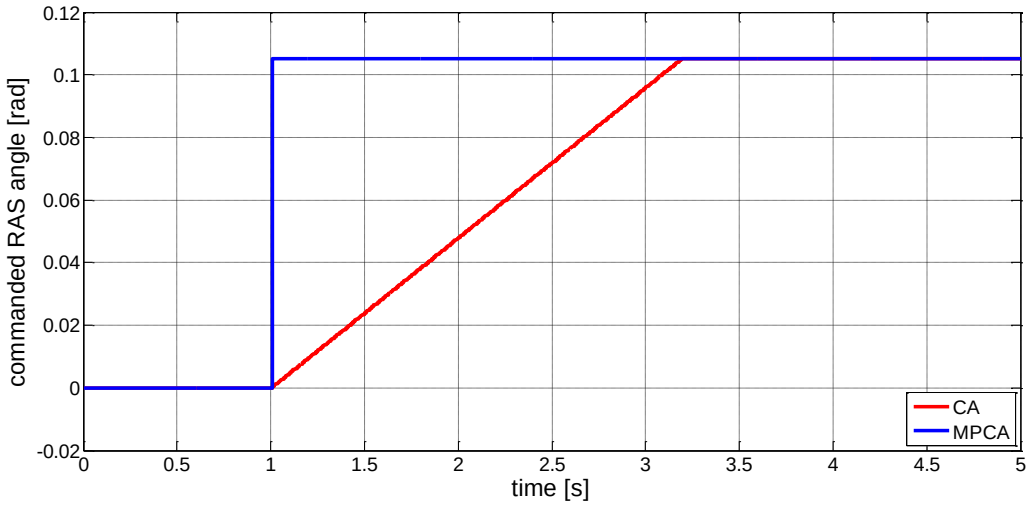

Figure 5.4. Different commanded input to the RAS actuator when using the MPCA formulation (blue) or the CA formulation (red).

The MPCA can command a more aggressive input to the RAS because it knows the dynamics of the RAS and can coordinate the brakes so that they generate the undesired yaw moment accordingly to when the actual value of the RAS can counteract that moment. On the other hand, the CA is more cautious because it can just command an input so that at each next step the actual value of the RAS can approximate the commanded value.

The consequence of these two different approaches is shown in Figure 5.5, which depicts the vehicle deceleration in the two cases.

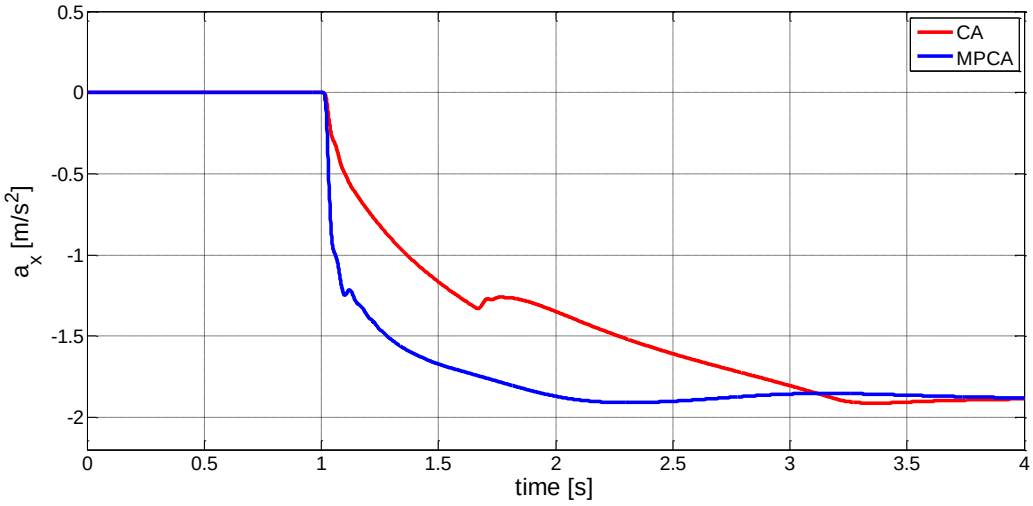

Figure 5.5. Different vehicle longitudinal deceleration dynamics when using the MPCA formulation (blue) or the CA formulation (red).

During the initial transient the MPCA controller manages to reach the desired deceleration faster than the CA controller. This faster transition is translated into a shorter distance covered by the vehicle equipped with the MPCA controller. Looking at the global position of the two vehicles at $t = 3\ s$, it is:

$$x_{MPCA} = 92.131\ m < 93.135\ m = x_{CA} \qquad Eq.\ 5.3$$

That means approximately $1\ m$ is gained during the first two seconds of the braking event.

On the other hand, no significant differences have been noted regarding the maximum lateral deviation of the vehicle and the effort required to the driver on the steering wheel.

It has been observed that the dynamics of the RAS is the factor that most influences the different vehicle behaviours when using the MPCA formulation or the CA formulation. If, for example, the PID that controls the RAS makes the system behave as a second order system, the performances of the CA controller deteriorate when compared with the MPCA controller. The reason why the performances deteriorate is that it is difficult in this case for the CA controller to have $\delta_{cmd_8} \approx \delta_8$ at every step unless a slow rate limit is set for the RAS.

On the other side, if the RAS can be tuned so that it approximates a constant rate system, the performance of the CA controller well approximates the performance of the MPCA controller. Constant rate system is meant here as a system that saturates if a large step input is applied so that its output always increases at a constant rate. In reference to Figure 5.4 it is then clear that there is no gain in sending a step input rather than a ramp input to the RAS because its output will evolve in the same way. In such a situation the braking force that can be applied without losing the vehicle stability is, in any moment, the same for the MPCA controller and the CA controller.

5.2. Split-μ acceleration

In the split-μ acceleration scenario the controller aims to coordinate the brakes, in order to create a resistance on the lower traction wheel, and the RAS, in order to compensate the undesired yaw moment produced by the uneven traction force at the driven axle. The initial speed of the vehicle is $v_0 = 0\ km/h$ and the driver decides to accelerate in order to move the vehicle. The vehicle stands on an uneven road as in the previous scenario ($\mu_l = 0.7$, $\mu_r = 0.1$).

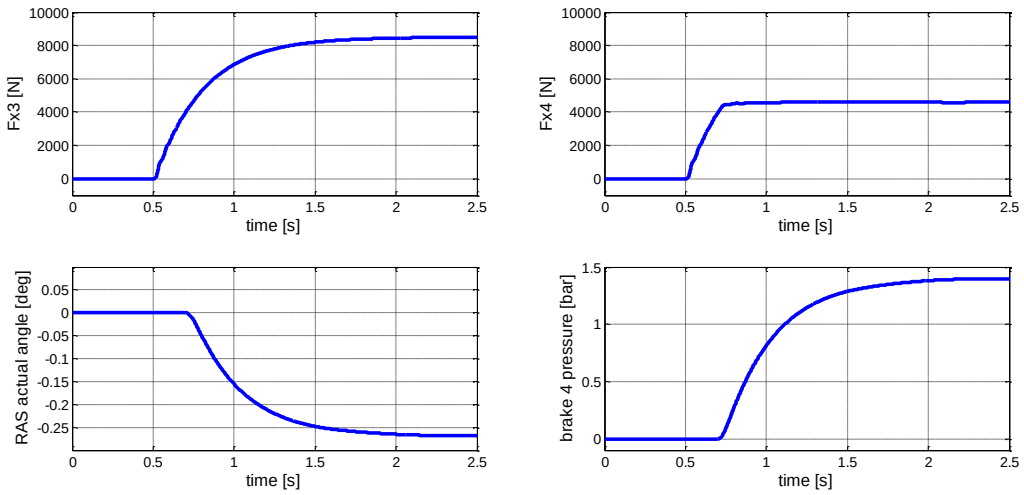

Figure 5.6. Longitudinal force produced by wheel 3 (top left) and wheel 4 (top right). Angle of the RAS wheels (bottom left) and pressure applied on the brake at the 4th wheel.

Figure 5.6 illustrates the behaviour of the controller, in particular the two graphs on the top show how the wheel on the low friction side can produce much less traction force than the wheel on the

high friction side. The two bottom graphs show how the actuators react in order to meet the desired acceleration of the driver, within the physical limit of the ground. During this particular simulation the desired acceleration of the driver has been converted into $v(1) = F_{x,tot} = 14\ kN$ and the controller, braking on the fourth wheel and fulfilling all the constraints, has been able to reach $B_f \delta(k)(1) \approx 13\ kN$ while turning the RAS wheels to counteract the moment produced by the driven axle.

Figure 5.7 also shows the benefit of using the RAS when accelerating on an uneven road. The same acceleration has been tried with the RAS activated and deactivated. The driver has not actuated on the steering wheel during the manoeuvre and it can be seen how the RAS is able to reduce the tendency of the vehicle to deviate form a straight line.

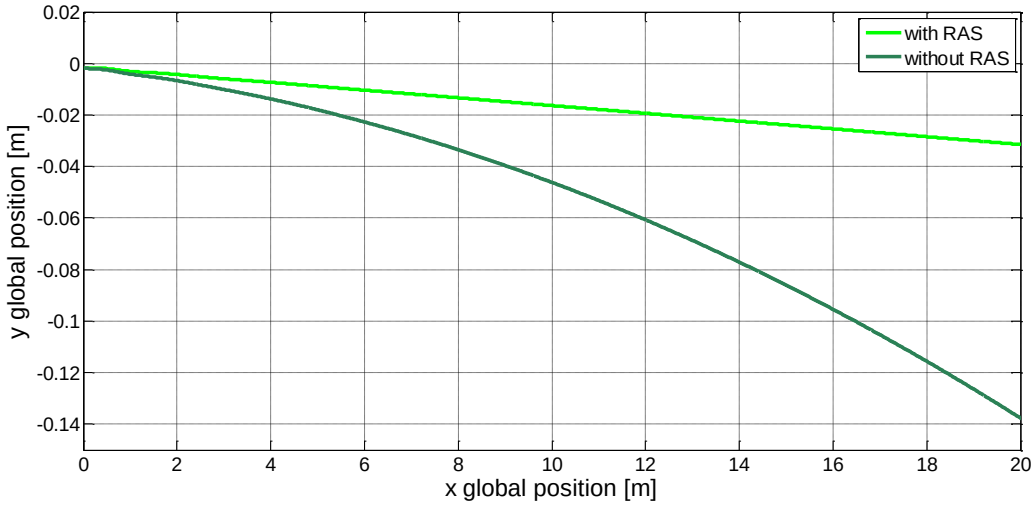

Figure 5.7. Vehicle behaviour when using the RAS (light green) to counteract the moment generated by the driven axle and when the RAS is deactivated (dark green).

The same test case has been tried with the CA formulation but no significant differences have been discovered. During this scenario the dynamics of the actuators are not as critical as in the split-μ braking scenario.

5.3. Brake blending

The objective of the brake blending is to minimize the wear on the brake pedals. Disc brakes are excellent devices to stop the vehicle but their lifetime can easily deteriorate if they are used too often. As the vehicle is equipped with the engine brake, it can be used to reduce the utilization of the disc brakes.

The simulations are run on dry asphalt, $\mu_i = 0.7\ \forall i$, and the vehicle starts braking from an initial speed $v_0 = 50\ km/h$.

The combination of these two types of actuators is especially useful during modest braking when the engine brake can play the primary role in reducing the speed of the vehicle. Figure 5.9 shows how the MPCA can combine together disc brakes and engine brake in order to have a fast response while minimizing the utilization of the disc brakes.

Figure 5.8 and Figure 5.10 show the comparison of a mild braking, when not only the engine brake is used in steady-state conditions, with and without the use of the engine brake. As the braking force is

symmetric at each axle, only the left side wheels are shown in the graphs. In particular, Figure 5.8 illustrates how the braking force has been split among the three axles during the two cases. It can be observed that the use of the engine brake does not modify the way the total braking force is distributed among the axles.

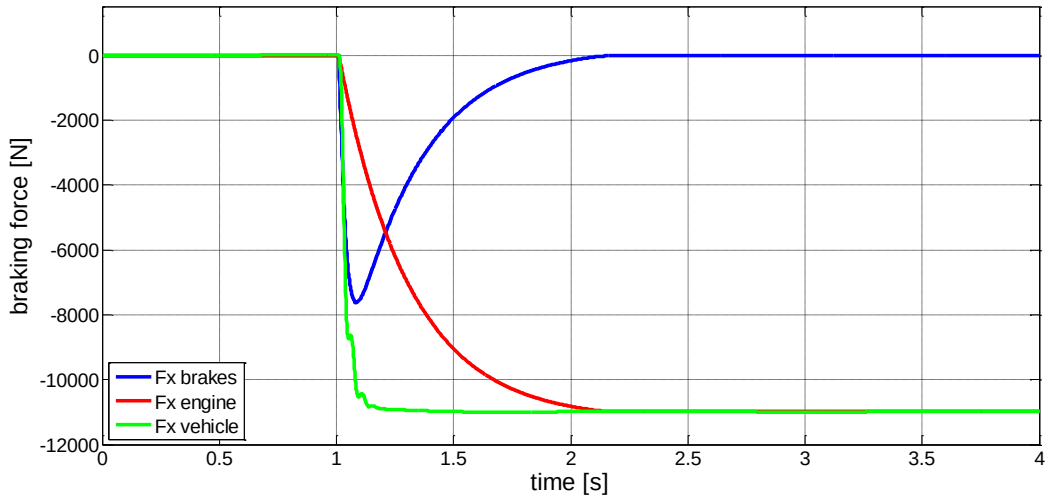

Figure 5.9. Combined braking force of engine and disc brakes in order to reach $F_{x,tot}$ as fast as possible while minimizing the use of the disc brakes.

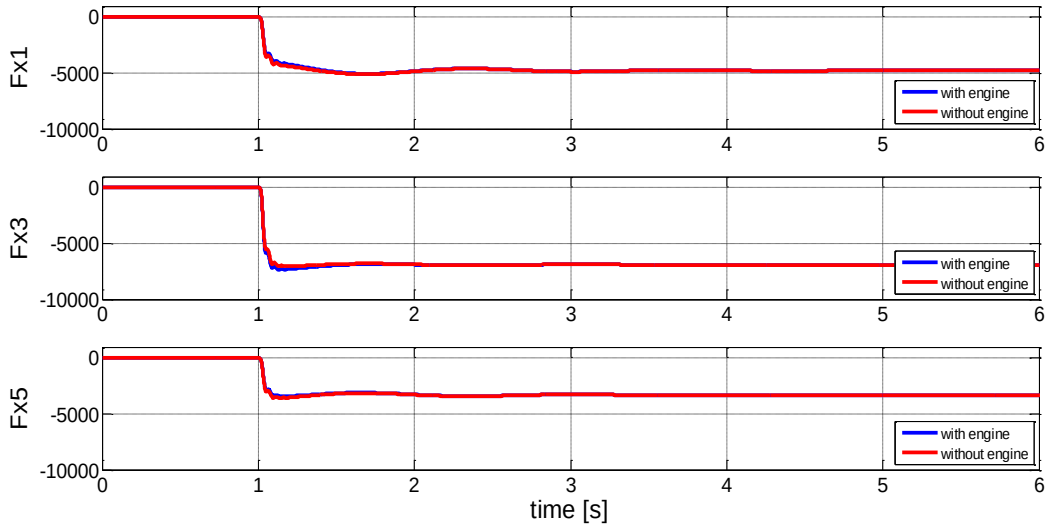

Figure 5.8. Comparison of how the braking force is distributed among the axles when using the engine brake (blue) and without using it (red).

On the other side, Figure 5.10 shows that the disc brakes on the second axle are used much less when the engine brake is activated.

The last observation is made about how much braking force has been allocated on each axle. For what has been explained in section 3.3 and taking into account that the friction coefficient is the same for each wheel $\mu_i = 0.7\ \forall i$, the braking force should be split among the axles proportionally to the load present on each axle. Table 5.1 with the values calculated at $t = 5\ s$ confirms the expectations.

	$\frac{F_{z_i}}{F_{z,tot}}$	$\frac{F_{x_i}}{F_{x,tot}}$
Axle 1	0.318	0.319
Axle 2	0.461	0.461
Axle 3	0.220	0.219

Table 5.1. Distribution of the braking force among the three axles

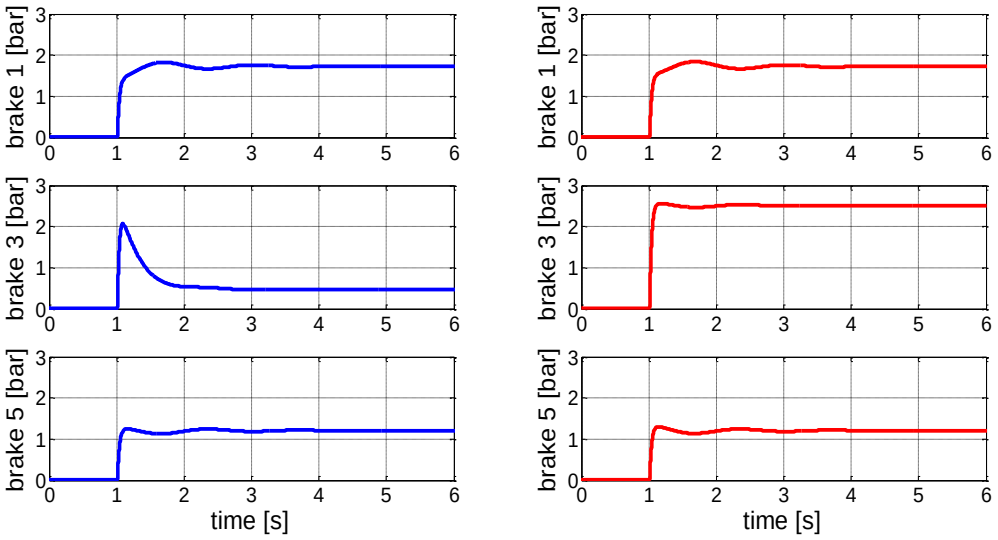

Figure 5.10. Pressure on the disc brakes 1-3-5 when the engine brake is used (left) and when the engine brake is not used (right).

From Table 5.1 it is clear that all the axles use the same amount of available friction during the braking:

$$\kappa_{axle_i} = F_{x_i}/F_{z_i} = 0.12 \ \forall i \qquad Eq.\ 5.4$$

Eventually, the brake blending scenario with the MPCA controller has been compared with the same scenario using the CA controller. Figure 5.11 shows the different results of the two controllers. The left figure shows the combined use of engine brake and disc brakes made by the MPCA controller, while the right figure shows the same braking using the CA controller.

It is worth noting how the MPCA controller can better combine the two different types of actuators during the initial transient. In particular, it manages to reach faster the requested global longitudinal force $F_{x,tot}$ and it has a smoother behaviour when the pressure on the brakes starts to be released and the engine brake torque ramps up.

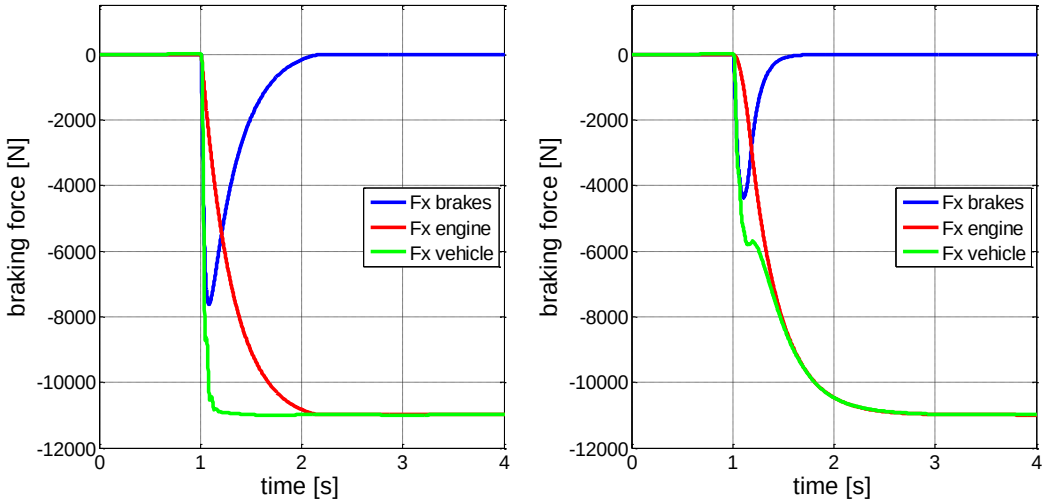

Figure 5.11. Comparison of the brake blending scenario using the MPCA controller (left) and the CA controller (right).

6. Real Tests

This chapter explains the implementation of the controller on a real vehicle and presents the results of the tests covering the split-μ braking scenario.

6.1. Implementation

The rapid implementation of the controller designed in Simulink in a real vehicle has been possible with the use of dSPACE. dSPACE provides tools, both software and hardware, to make a faster development of controllers for real-time applications.

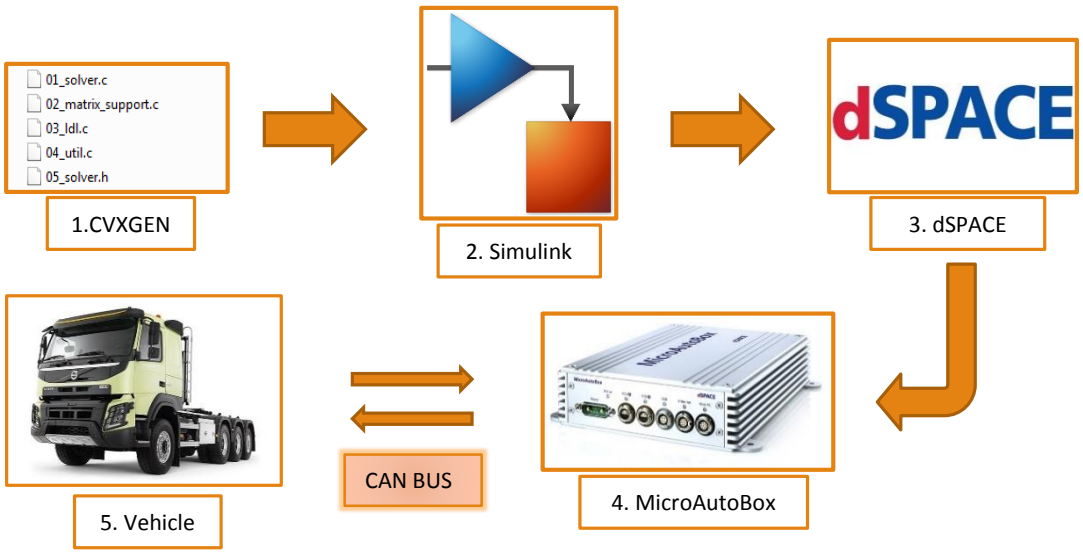

Figure 6.1. Method used to implement the controller into a real vehicle.

Figure 6.1 illustrates the steps followed to implement the controller:

1. The solver for the MPCA problem has been generated using CVXGEN. CVXGEN delivers the solver in the form of C-code.
2. Simulink has been used to build-up the controller. In particular, the new Simulink model does not contain the model of the vehicle but two blocks that are responsible for reading and sending the signals through the vehicle's CAN BUS. These are special blocks provided by dSPACE and they are necessary to directly communicate with other sensors and actuators of the vehicle. Besides these two blocks the new model contains the real core of the controller (Figure 6.2). In particular, a MATLAB function that transforms the input signals into the parameters used by the solver and an S-function builder, which wraps the C-code generated by CVXGEN. The S-function receives the parameters from the MATLAB function and solves the MPCA problem (Eq. 2.13) at every step giving $\delta^*(0)_{cmd_i}\ i = 1, \dots, 8$ as output.
3. dSPACE has been linked to MATLAB in order to generate code from the Simulink model that can be executed on real-time in the dSPACE environment. In particular, the software used as interface for the real-time execution is ControlDesk, a software provided by dSPACE, from where it is possible to manually change the parameters of the code that represents the Simulink model and read the values from the sensors of the vehicle during the tests.
4. MicroAutoBox is the hardware that runs the designed controller. From ControlDesk, the code generated from the Simulink model, has been loaded in MicroAutoBox that can directly send and receive signals to and from the vehicle. MicroAutoBox simulates a real vehicle ECU and it

is directly connected to the vehicle's CAN BUS. It is equipped with a processor that is responsible for solving the MPCA problem at each sample time.

5. The vehicle used during the tests is a Volvo FMX (Figure 6.3). It performs all the tasks required to a commercial vehicle plus it communicates with MicroAutoBox via CAN BUS.

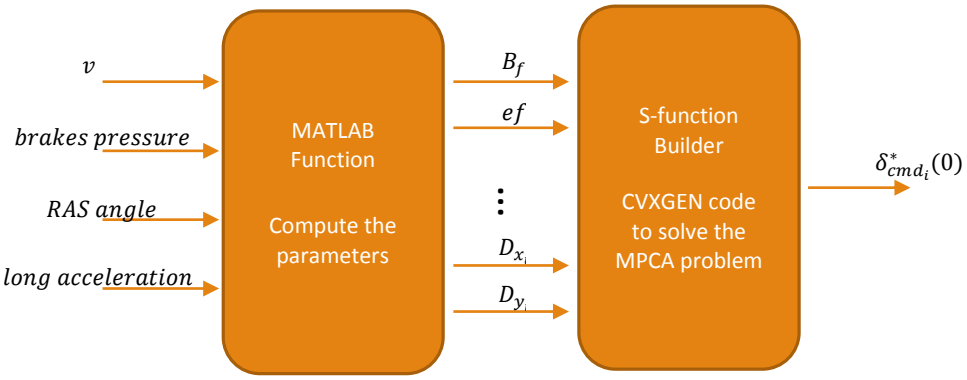

Figure 6.2. Core of the controller.

The configuration of the vehicle used during the tests was slightly different from the vehicle model used to design the controller during the Simulink simulations. In particular, the vehicle for the tests was an 8x4 tag axle with RAS and not a 6x2 tag axle with RAS. Nevertheless, one strong point of control allocation algorithms is the high flexibility to adapt to different vehicle configurations. Changing the dimensions of the matrices in the objective function and adding new constraints to the additional axle have been sufficient to redesign the controller so that it can meet the new vehicle configuration.

Figure 6.3. Photo of the truck used during the tests.

Due to time constraints, the scenario that has been validated during the tests is the split-μ braking. In this scenario the principal role is played by the coordination of the disc brakes with the RAS, while

the engine brake does not have an influence on the vehicle performance. For this reason and to limit the complexity of the tests, the engine brake has been deactivated during the tests.

The facility used for the tests was a handling and braking track that offers the possibility to try the split-μ scenario using a wet basalt surface to simulate the low friction road (i.e. the icy road).

Although the simulations were essential for the fast design of the controller, during the track tests some new challenges had to be faced. In particular, the feasible sample time of the controller to solve the MPCA problem and the calibration of the vehicle parameters had to receive special attention.

The controller sample time is basic to ensure safety and reliability for the action commanded by the controller. A slow sample time could result in an unacceptable response time of the controller and consequently in a degradation of the vehicle dynamics performances. The sample time T_{clk} chosen for the tests was large enough to ensure the computation of the MPCA problem at every step and, at the same time, fast enough to guarantee appropriate vehicle performance.

The calibration of the vehicle parameters was possible by running dedicated tests. As an example, one parameter that needed to be tuned was the relation between brake pressure and generated longitudinal force. To find out this relation, namely $\frac{k_p}{r_j}$ in the MPCA formulation, one axle at a time was braked with different amounts of pressure. Once validated the linearity between brake pressure and longitudinal force, from the knowledge of the brake pressure and the relative vehicle deceleration, the relation $\frac{k_p}{r_i}$ was easy to determine.

6.2. Tests

What follows is a presentation of the tests results during the split-μ braking. The tests have been performed several times to ensure the repeatability of the controller under the split-μ braking scenario.

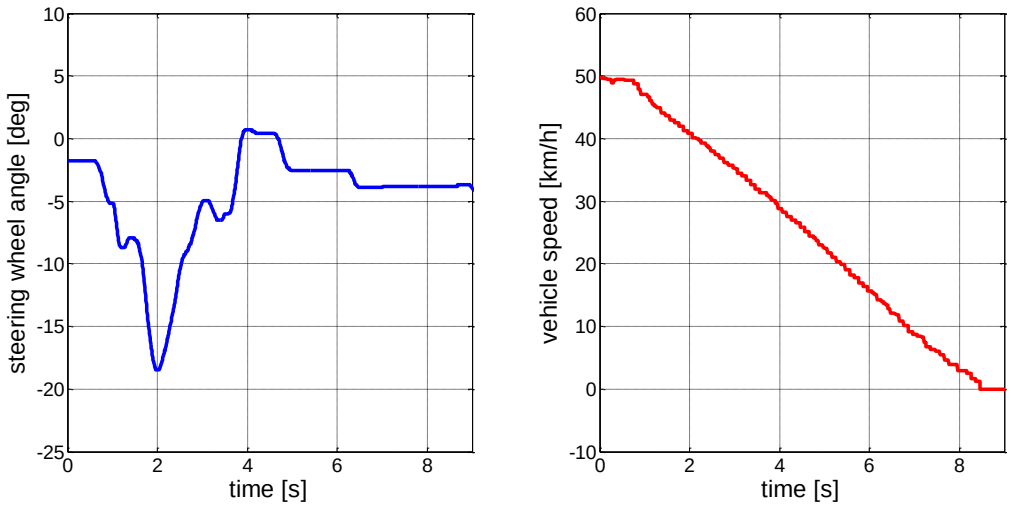

Figure 6.4. Recording of typical driver effort (left) and speed vehicle (right) during the braking.

During the tests, the driver was able to maintain the vehicle stable on a straight line. Particular attention has then been paid to what is the effort made by the driver in order to control the vehicle and how long it takes to the vehicle to stop.

Figure 6.4. shows the typical behaviour of the steering wheel turned by the driver and the vehicle speed during the split-μ braking. In general it has been noted that, during the five repetitions of the split-μ braking, the driver never needed to turn the steering wheel more than $18°$ while the time to stop the vehicle was always around $7.5\ s$.

In Figure 6.5 the same graphs are shown during a split-μ braking without using the MPCA controller, that is when the vehicle is braking using the standard braking system.

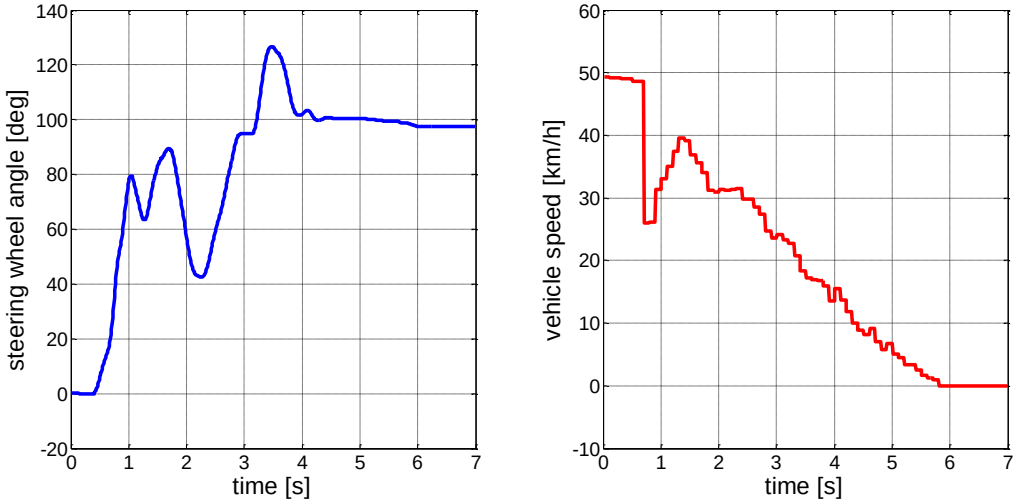

Figure 6.5. Driver effort (left) and vehicle speed (right) during a normal split-μ braking with ABS.

The two different philosophies followed by the two systems, MPCA and standard braking system, can be observed. When the vehicle was braking with the MPCA controller, the vehicle stability, that is avoiding yaw disturbance, always had the priority and, as a consequence, the controller limited the maximum achievable braking force; while with the standard braking system the priority was given to the maximum amount of generated braking force. This system takes care of the vehicle stability only by limiting the different pressure that can be sent to the wheels of the same axle.

As a consequence, the vehicle equipped with the standard braking system took less time to stop, around $2.5\ s$ less compared with the same split-μ braking when using the MPCA controller. Nevertheless, the effort required to the driver in order to maintain the stability of the vehicle was much higher when the MPCA controller was not used. In that case, in fact, the steering wheel had to be turned up to a peak value of $126°$ after $3\ s$ from the beginning of the braking event compared with the maximum $18°$ that the driver had to turn when using the MPCA controller.

The last comparison for the split-μ braking scenario was made using the CA formulation. The difference between the two formulations has already been noted during the simulations and the tests confirmed what was expected.

Figure 6.6 shows the different approaches when using the RAS with the MPCA formulation or the CA formulation. The fact that the MPCA formulation managed to turn faster the RAS wheels is reflected in a faster deceleration of the vehicle during the initial transient.

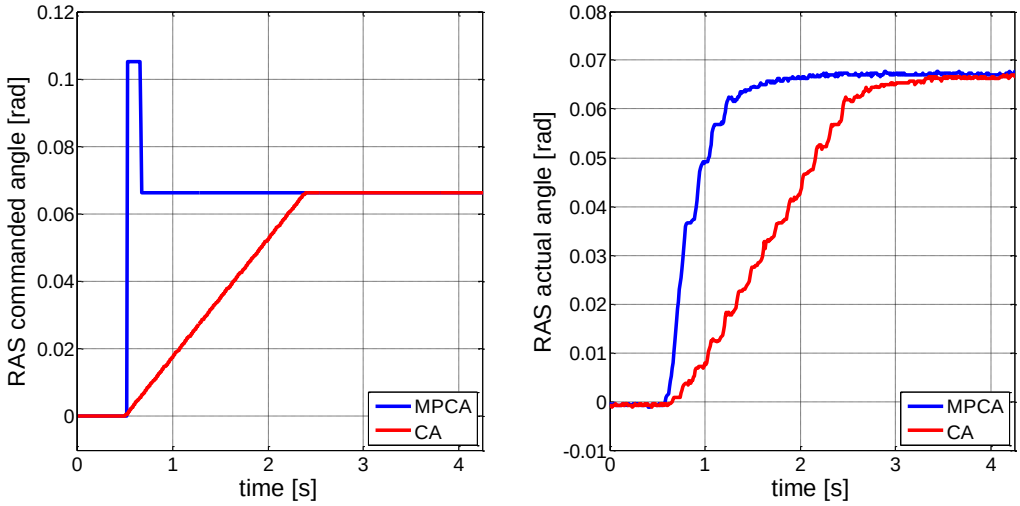

Figure 6.6. Typical recordings of the split-μ braking tests. Commanded RAS angle (left) and actual RAS angle (right). Comparison using the MPCA and CA formulation.

Figure 6.7 shows three different braking events with the MPCA controller compared with other three different braking events with the CA controller. As expected, the MPCA controller is, in general, always faster to reach the desired deceleration for the vehicle.

As a consequence, during the initial transient, the vehicle equipped with the MPCA controller brakes more and covers a shorter distance. This distance can be seen as the braking distance gained when using the MPCA controller and it has been calculated from the accelerations as:

$$\Delta x = \iint_{t_0}^{t_{tr}} a_{x_{CA}}(t)dt - \iint_{t_0}^{t_{tr}} a_{x_{MPCA}}(t)dt \approx 1\ m \qquad \textit{Eq. 6.1}$$

where t_0 is the instant when the vehicle starts braking and t_{tr} is the instant when the initial transient ends. $a_{x_{CA}}(t)$ and $a_{x_{MPCA}}(t)$ are respectively the deceleration of the vehicle when using the CA controller and the MPCA controller.

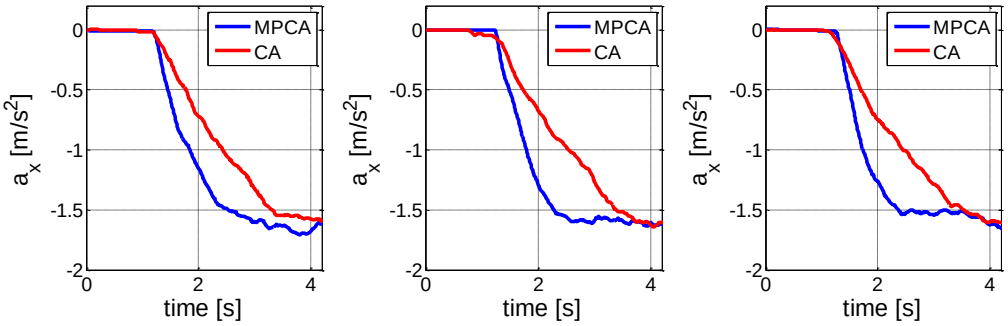

Figure 6.7. Initial vehicle deceleration when using MPCA (blue) or CA (red) formulation.

As already observed for the split-μ braking simulations, the effort requested to the driver during the tests in order to maintain the vehicle on a straight path was almost the same when using the MPCA and the CA controller. Figure 6.8 shows a comparison of how much the steering wheel had to be turned by the driver when the vehicle was equipped with the MPCA controller and the CA controller.

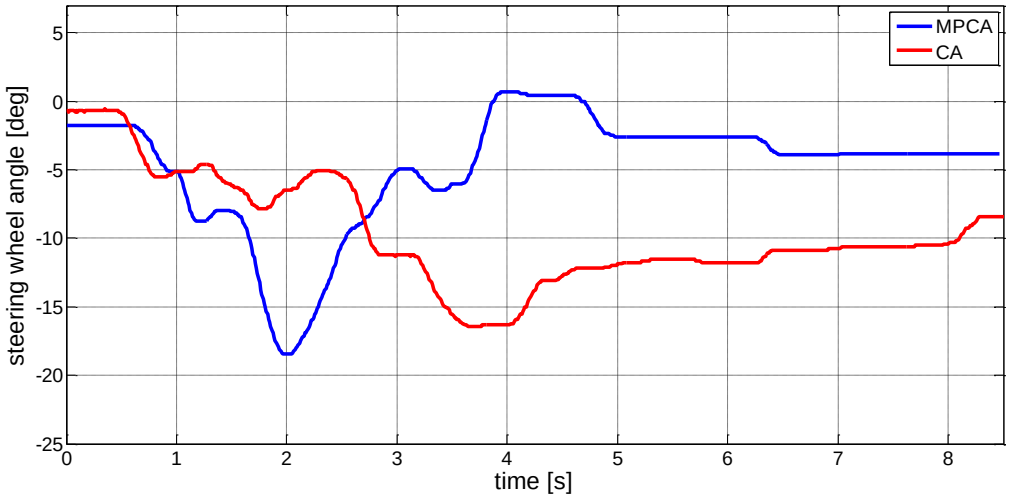

Figure 6.8. Typical driver effort on the steering wheel when using MPCA (blue) and CA (red).

7. Concluding Remarks

This chapter concludes the thesis with some observations on the work done and explains how the new controller can have an impact on the environment and society. It also evaluates the controller from an economic point of view and makes some suggestions for future work.

7.1. Conclusions

The objective of this thesis was to evaluate the potentiality of a Model Predictive Control Allocation method to coordinate the actuators of a heavy vehicle.

Initially, the effects and the dynamics of the actuators have been studied to understand how they behave and influence the movement of the vehicle. Then the controller has been designed to automatically adapt and cope with three different test scenarios: split-μ braking, split-μ acceleration and brake blending.

Simulations in the Simulink environment have been run for every scenario. The split-μ braking scenario has shown how the MPCA controller manages to coordinate the contrary yaw moments generated by brakes and RAS in order to brake the vehicle while maintaining it stable.

The comparison of the same scenario with the same vehicle but without using the RAS enlightened the benefits of being able to counteract the yaw moment generated by the brakes with a steerable axle. It is worth saying that the controller should be effective as long as there is at least one controllable steerable axle in the vehicle, no matter which one.

Moreover, the comparison between the MPCA controller and the CA controller exhibited the major responsiveness of the MPCA formulation with respect to the CA formulation. This responsiveness is reflected in a shorter time needed for the MPCA controller to reach the desired deceleration of the vehicle. It has also been explained why a more complicate dynamics for the RAS would worsen the performance of the CA controller while a simple, constant rate, dynamics of the RAS would make the two controllers indistinguishable. If there is no need to deeply exploit the actuators, setting conservative rate limits in the CA formulation facilitates the handling of the vehicle as forces and moments are produced smoother.

Simulations of the split-μ acceleration have shown the ability of the controller to independently brake the wheel on the low friction coefficient side in order to generate more traction force on the wheel in contact with high friction coefficient road. Furthermore, the controller has succeeded in coordinating the RAS so that it can eliminate the tendency of the vehicle to deviate from a straight path. During this scenario no differences have been noted between the MPCA controller and the CA controller.

Eventually the brake blending simulations have illustrated how to combine engine brake and disc brakes in order to minimize the wear of the disc brakes. A comparison between the MPCA controller and the CA controller has highlighted the faster response of the MPCA formulation to build up the requested braking force. In addition, the transition from disc brakes to engine brake is smoother when using the MPCA formulation.

During a mild braking, when not only the engine brake is used in steady-state conditions, the controller is also able to distribute the braking force among the axles in the same way whether the engine brake is used or not. In both cases, the distribution is proportional to the amount of the

braking force available between wheel and ground. This strategy to distribute the braking force is in accordance with today's braking systems.

After the simulations, the tests on a real vehicle have consolidated the performances of the MPCA controller for the split-μ braking scenario. Even if it needs more time to stop the vehicle when compared with a standard braking system, the MPCA controller is able to significantly reduce the effort required to the driver in order to maintain the vehicle stable.

Moreover, the comparison between the MPCA controller and the CA controller has shown the advantages during the initial transition of using an explicit formulation of the actuators dynamics in the MPCA formulation. The controller can ask for a more rapid input to the RAS without compromising the vehicle dynamics. As a result, a faster deceleration is achieved with the MPCA controller.

During the simulations and tests the MPCA controller has always been set up to prioritize the vehicle stability, nevertheless the trade-off between braking distance and vehicle stability in the MPCA controller can be adjusted and dynamically adapted to different situations depending on the purpose of the controller [15].

The main drawback encountered when using the MPCA formulation rather than the CA formulation is the higher computational effort requested to solve the MPCA problem. During the tests, the average computational time of the MPCA was $10\ ms$ compared with the average $0.7\ ms$ to solve the CA problem. The computational time exhibited by MPCA can represent a critical point when it is necessary to work with arduous sample time.

7.2. Environmental and Social Impact

Analyzing the situations where the controller modifies and improves the vehicle dynamics, some conclusions can be made regarding the effects that the controller can have on the environment and society.

Vehicles have always had an enormous impact on the society, they have radically changed the way people move from one point to another opening infinite opportunities to improve the mobility on short, medium and long distances. Automotive industry is considered among the most important economic sectors around the world and, for many industrialized nations, vehicles production and sales reflect the economic status of a country. It is then fundamental to take care of how of vehicles are integrated in the society.

In general, two are the factors that have a significant impact on the vehicle image as integral part of the society: safety and pollution. Safety implies that a vehicle does not have to cause any damage nor increase the risk to provoke injuries to people both inside and outside the vehicles. Safety is one of the most important aspects related with vehicles industry, and regulations on minimal safety criteria have become more and more tight over the time. Pollution, on the other hand, is a topic that has assumed extremely high relevance in the last two decades. Pollution can be divided into noise and air pollution. Between them, air pollution in the form of maximum vehicle contaminant emissions has been extensively legislated and people have been largely sensitized to the importance of having a low-emissions vehicle.

Starting with the two split-μ scenarios, the objective of the controller is helping the driver successfully overcome dangerous situations. As long as the vehicle remains stable and under the control of the driver, the chances of producing an accident can be drastically reduced. The controller

has then an impact on the society in the sense that it increases the number of situations where a vehicle is perceived as safe, reducing the risk to damage any of the people involved in the scenario. Make the dynamics of the vehicle safe during maneuvers taken in harsh environmental conditions is still a field where there are wide margins for improvement. Control the behavior of the vehicle during these maneuvers represents a challenge to increase the confidence people have in vehicles.

On the other side, taking into account the brake blending scenario, the controller has been designed to reduce the use of the disc brakes while prioritizing the use of the engine brake. Above all during smooth downhill roads, the use of the engine brake prevents the brakes from fading thus increasing the performance and safety of the vehicle in case a suddenly hard braking is needed.

Moreover, the brake blending scenario has proved that the MPCA formulation is an effective method to combine together actuators that produce longitudinal braking force on the vehicle. With the introduction of the hybrid powertrains, the number of actuators able to produce a braking force on the vehicle will further increase. In a hybrid heavy vehicle, in fact, it can be thought that in addition to disc brakes and engine brake also the electric brakes can actively participate in stopping the vehicle. The use of the proposed MPCA method can then be easily extend so that it includes the use of electric brakes in its formulation. The use of electric brakes has a significant impact on the environment as they transform the kinetic energy of the vehicle into reusable electric energy for the motor, thus reducing the fuel consumption and all the contaminant emissions that a diesel engine brings with itself (CO, HC, NO_x, PM, CO_2). Hybrid powertrains for heavy vehicles are the state-of-art solution to reduce fuel consumption and increase the efficiency of the vehicle. It is then essential to have methods that are able to actively and effectively combine together all the actuators of the vehicle, as it does the designed MPCA controller.

7.3. Budget

This paragraph collects some ideas and considerations that are useful when taking into account the development and implementation of the controller from an economic point of view. Although the goal of this thesis was not to design a controller ready to enter the market in a short time, the following points are of concern for the day such a strategy of combining the actuators will be mature and will be used in a commercial heavy vehicle:

- The controller does not make use of any additional hardware for the actuators. No additional sensors, valves or hydraulic circuits are needed in order to coordinate the actuators taken into account during this thesis. The majority of the cost is then related with the development of a reliable software that is able to effectively and robustly communicate with the other sensors and actuators of the vehicle. The cost of the hardware is confined to the dedicated ECU designated of running the MPCA algorithm.

- Heavy vehicles have to continuously adapt to customers demands usually dictated by very different working conditions. One advantage of using the developed controller is its high reconfigurability. Simple modifications of the same algorithm can make them deal with vehicles that have different number of axles, axles distance and tracks width. As a consequence, the development of the software can achieve economies of scale: the same algorithm can be used for a large volume of vehicles thus amortizing the unit cost of the controller.

- It is fundamental to understand what is the role of the MPCA controller when compared with the other controllers of the vehicle. If only the split-μ braking scenario is taken into account, the controller is independent of the other controllers of the vehicle such as the ESP or the EBD and its cost has to be added to the standard cost of the vehicle. If, on the other hand, the control allocation architecture (Figure 2.1) is used as a basis to improve the general dynamics of the vehicle, it has to be clear which other controllers are no longer necessary in the vehicle. As an example, the brake blending strategy can be used as an EBD system, moreover methods to improve closing curves dynamics and to prevent yaw and roll over instability have been presented in [8],[9].

Based on these considerations, the unit cost of the MPCA controller can be calculated as:

$$c_{MPCA} = c_{ECU} + \frac{c_{dev}}{n_{veh}} - c_{ctrl} \qquad Eq.\ 7.1$$

where c_{ECU} is the unit cost of the dedicated ECU that runs the MPCA algorithm, c_{dev} is the cost necessary to develop the software for the MPCA controller, n_{veh} is the volume of the vehicle that will be equipped with the MPCA controller, c_{ctrl} is the unit cost of the other controllers hardware in the vehicle that can be replaced by the MPCA controller.

Supposing that the MPCA controller replaces the functionality of an already existing ECU, that is $c_{ECU} = c_{ctrl}$, the cost of implementing the new MPCA method is reduced to $c_{MPCA} = \frac{c_{dev}}{n_{veh}}$. The development cost can have significant variations depending on how much the existing architecture has to be changed. In an optimistic view where the development cost can be estimated around 1 million euros, if all the vehicles sold in the next three years will be equipped with the same software, it can be estimated $n_{veh} \approx 600000\ units$. These numbers lead to a unit cost for the MPCA controller of $c_{MPCA} \approx 1.7$ €. In this scenario, implementing the new controller would not have an important impact on the vehicle price, nevertheless it is difficult to forecast the development cost, which could be up to one hundred times more expensive and it is more realistic to think that only a portion of the sold vehicles will be equipped with the new controller. These considerations clarify the wide range of values that can be assigned to c_{MPCA} and brighten the necessity of a further, more sophisticated, analysis to understand the real impact that the new controller could have on the final vehicle price.

7.4. Future Work

Although they would have deserved special attentions, there are some topics related with this thesis that have not been covered. In particular, the most interesting topics that should be analysed in the future are:

1. New heavy vehicle configurations that include trailers and semi-trailers. When heavy vehicles are made of more than one rigid body the study of their dynamics increases in complexity.

2. New test scenarios for the vehicle. In particular, a further insight on the split-μ braking during a steady-state curve would be interesting. Moreover, set up some tests that simulate possible actuators faults would help understanding the general adaptive behaviour of the controller.

3. New actuators for the vehicle. With the introduction of the hybrid powertrains it is interesting to understand how the controller, apart from improving the vehicle dynamics, can optimize the energy management of the vehicle. For example, one should understand what is the best way to coordinate disc brakes, engine brake and electric brakes during a mild braking.

4. New model for the powertrain that is able to better approximate the reality. The new model should be able to take into account the dynamics caused by a gear shift too.

5. Finally, a real-time estimator of the tyre/road friction coefficient would probably be the major contribution that helps understanding the potentialities of the controller in a real, every-day environment.

Appendix A – Nomenclature

This appendix lists all the symbols used throughout the report and their respective meanings.

Symbol	Description
$\boldsymbol{v}$	Virtual input
$\boldsymbol{B_f}$	Effectiveness matrix
$\boldsymbol{\delta}$	Actuators outputs
$\boldsymbol{\delta_{cmd}}$	Actuators inputs
$\boldsymbol{A, B}$	Dynamic matrices for the actuators
$\boldsymbol{W_v}$	Weighting matrix for virtual input
$\boldsymbol{W_\delta}$	Weighting matrix for actuators
$\boldsymbol{T_s}$	Controller sample time
$\boldsymbol{T}$	Sample time for system discretization
$\boldsymbol{T_{clk}}$	MicroAutoBox sample time
$\boldsymbol{T_b}$	Braking force of the vehicle
$\boldsymbol{N}$	Time horizon steps
$\boldsymbol{\delta(k)_1, \dots, \delta(k)_6}$	Brake pressure at wheel 1,…,6
$\boldsymbol{\delta(k)_{cmd_1}, \dots, \delta(k)_{cmd_6}}$	Commanded brake pressure at wheel 1,…,6
$\boldsymbol{\delta(k)_7}$	Torque at the driven axle
$\boldsymbol{\delta(k)_{cmd_7}}$	Commanded torque at the driven axle
$\boldsymbol{\delta(k)_8}$	RAS wheels angle
$\boldsymbol{\delta(k)_{cmd_8}}$	Commanded RAS wheels angle
$\boldsymbol{F_{x,tot}}$	Longitudinal force on the vehicle
$\boldsymbol{M_{z,tot}}$	Yaw moment on the vehicle
$\boldsymbol{k_p}$	Relation between wheel torque and brake pressure
$\boldsymbol{r_i}$	Dynamic wheel radius at axle *i*
$\boldsymbol{l_m}$	Distance first-second axle
$\boldsymbol{l_r}$	Distance first-third axle
$\boldsymbol{L_r}$	Distance centre of gravity-third axle
$\boldsymbol{\alpha}$	Slip angle
$\boldsymbol{\beta}$	Sideslip angle
$\boldsymbol{C_{\alpha_i}}$	Cornering stiffness
$\boldsymbol{D_{x,y}}$	Peak value in Pacejka's Magic Formula
$\boldsymbol{F_{z_i}}$	Load on wheel *i*

μ_i	Adhesion coefficient on wheel *i*
w_i	Width of axle *i*
L_r	Distance between the last axle and the centre of gravity of the vehicle
K	Gain first order system
τ	Time constant first order system
τ_b	Brakes time constant
τ_p	Powertrain time constant
τ_{RAS}	RAS time constant
m_{veh}	Vehicle mass
δ_{SWA}	Front steering wheels angle

Appendix B – Parameters

This appendix lists all the values of the parameters explained during the report.

Parameter	***Value***
m_{veh}	$22760\ kg$
k_b	$-1470.6\ Nm/rad$
$[r_1\ r_2\ r_3]$	$[0.53\ 0.534\ 0.54]\ m$
$[w_1\ w_2\ w_3]$	$[2.05\ 1.85\ 2.05]\ m$
l_m	$4.8\ m$
l_r	$6.17\ m$
W_v	$diag(0.1, 100)$
τ_b	$0.1\ s$
τ_p	$0.3\ s$
τ_{RAS}	$0.4\ s$
a_{RAS}	1
K	1
$p_{\max}$	$9\ bar$
$\Gamma_{dec,\max}$	$6000\ Nm$
$\Gamma_{acc,\max}$	$9000\ Nm$
$[\gamma_{min}\ \gamma_{max}]$	$[-6\ 6]\ deg$
N	10
T	$0.05\ s$
T_s	$0.01\ s$
T_{clk}	$0.02\ s$

The distance L_r between centre of gravity and third axle of the vehicle is calculated from F_{zi} and l_m, l_r. In particular, an updated value for the centre of gravity is calculated at every step by the moment balance around the first axle:

$$l_{COG} = \frac{\left(F_{z_3} + F_{z_4}\right) l_m + \left(F_{z_5} + F_{z_6}\right) l_r}{\sum_{i=1}^{6} F_{z_i}}$$

where l_{COG} is the distance between first axle and centre of gravity. Eventually L_r can be easily found as:

$$L_r = l_r - l_{COG}$$

Tyres parameters

The model of the tyres used during the simulations is based on the PAC2002 and the peak values in the Magic Formula are defined as:

$$D_{x_i} = \mu_{x_i} F_{z_i}$$

where:

$$\mu_{x_i} = \left(PDX1 + PDX2 \frac{F_{z_i} - FNOMIN}{FNOMIN}\right) \mu_i$$

and:

$$D_{y_i} = \mu_{y_i} F_{z_i}$$

where:

$$\mu_{y_i} = \left(PDY1 + PDY2 \frac{F_{z_i} - FNOMIN}{FNOMIN}\right) \mu_i$$

The cornering stiffness is defined as:

$$C_{\alpha_i} = -PKY1 \cdot FNOMIN \cdot \sin\left(2\, arctan\left(\frac{F_{z_i}}{PKY2 \cdot FNOMIN}\right)\right)$$

The values of the parameters used during the simulations are:

Parameter	***Value***
$PDX1$	0.9
$PDX2$	$-1 \cdot 10^{-4}$
$FNOMIN$	35000
$PDY1$	0.73957
$PDY2$	-0.075004
$PKY1$	-10.289
$PKY2$	3.3343

All these values strongly depend on the wheel load F_{z_i} and they are updated at every step in order to take into account significant modifications of their starting value.

References

[1] World Health Organization, “Global Status Report on Road Safety 2013,” WHO Press, Luxembourg, 2013.

[2] Y. Luo, A. Serrani and S. Yurkovich, “Model Predictive Dynamic Control Allocation with Actuator Dynamics,” in *American Control Conference*, Boston, Massachusetts, 2004.

[3] Y. Luo, A. Serrani and S. Yurkovich, “Dynamic Control Allocation with Asymptotic Tracking of Time-Varying Control Input Commands,” in *American Control Conference*, Portland, Oregon, 2005.

[4] C. Vermillion, J. Sun and K. Butts, “Model Predictive Control Allocation for Overactuated Systems - Stability and Performance,” in *Decision and Control*, New Orleans, Louisiana, 2007.

[5] C. Vermillion, J. Sun and K. Butts, “Predictive Control Allocation for a Thermal Management System Based on an Inner Loop Reference Model - Design, Analysis and Experimental Results,” *Control Systems Technology, IEEE Transactions,* vol. XIX, no. 4, pp. 772-781, 2011.

[6] M. Hanger, T. A. Johansen, G. K. Mykland and A. Skullestad, “Dynamic Model Predictive Control Allocation using CVXGEN,” in *IEEE International Conference on Control and Automation*, Santiago, 2011.

[7] C. Satzger, R. Castro and T. Bünte, “A Model Predictive Control Allocation Approach to Hybrid Braking,” in *IEEE Intelligent Vehicles Symposium*, Dearborn, Michigan, 2014.

[8] P. Sundström and K. Tagesson, “On Real Time Adaptive and Dynamically Constrained Control Allocation for Stability Control of Heavy Vehicles,” Master of Science Thesis, 2008.

[9] J. Eklöv, “Real-Time implementation of a vehicle motion coordinator for a single unit truck,” Master's Thesis in Systems, Control and Mechatronics, 2013.

[10] P. Lingman, “Integrated Brake Control - Downhill Driving Strategies,” Thesis for the Degree of Doctor of Philosophy, 2005.

[11] F. Borrelli, A. Bemporad and M. Morari, Predictive Control for linear and hybrid systems, March 7, 2014.

[12] S. J. Wright, Primal-Dual Interior-Point Methods, Philadelphia: Siam, 1997.

[13] J. Mattingley and S. Boyd, CVXGEN: a code generator for embedded convex optimization, Springer, 2011.

[14] R. J. Vanderbei, “Symmetric Quasi-Definite Matrices,” *SIAM Journal on Optimization,* vol. V, no. 1, pp. 100-113, 1995.

[15] K. Tagesson, L. Laine and B. Jacobson, “Combining Coordination of Motion Actuators with Driver Steering Interaction,” *Traffic Injury Prevention,* vol. XVI, pp. 18-24, 2015.

[16] S. Kharrazi, M. Lidberg, P. Lingman, J.-I. Svensson and N. Dela, “The effectiveness of rear steering on the yaw stability and responsiveness of a heavy truck,” *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility,* vol. XLVI, pp. 365-372, 2008.

[17] P. Nyman and K. Uhlén, “Coordination of actuators for long heavy vehicle combinations using control allocation,” Master's Thesis in Systems, Control and Mechatronics, 2014.

[18] L. Laine, “Reconfigurable Motion Control Systems for Over-Actuated Road Vehicles,” Thesis for the degree of Doctor of Philosophy, 2007.

[19] T. A. Johansen and T. I. Fossen, “Control allocation - A survey,” *Automatica,* vol. XLIX, no. 5, pp. 1087-1103, 2013.

[20] B. P. Otta, “Numerical Algorithms for Quadratic Programming for Approximated Predictive Control,” Master Thesis, 2013.