CHALMERS

Numerical optimization methods for improving energy efficiency in Battery Electric Vehicles using wheel torques and steering angles

Master's thesis in Automotive Engineering

MANISH SUVARNA
ANAND PRABHU

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2021

MASTER'S THESIS IN AUTOMOTIVE ENGINEERING

Numerical optimization methods for improving energy efficiency in Battery Electric Vehicles using wheel torques and steering angles

MANISH SUVARNA
ANAND PRABHU

Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2021

Numerical optimization methods for improving energy efficiency in Battery Electric Vehicles using wheel torques and steering angles

MANISH SUVARNA
ANAND PRABHU

Master's thesis 2021:45
Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31-772 1000

Chalmers Digitaltryck
Göteborg, Sweden 2021

Numerical optimization methods for improving energy efficiency in Battery Electric Vehicles using wheel torques and steering angles

Master's thesis in Automotive Engineering
MANISH SUVARNA
ANAND PRABHU
Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
Chalmers University of Technology

Abstract

The planar motion of a ground vehicle is affected by controls given to it in the form of the front axle steering angle and the individual wheel torques. A battery electric vehicle with four in-wheel motors is an over-actuated system where the control space has multiple control solutions for allocating the wheel torques and front axle steering angle for achieving the desired planar motion of the vehicle. Of these available solutions, a particular energy efficient control solution can be found to achieve this desired planar motion of the vehicle and ultimately improve the drive range of the BEV. This is achieved by formulating and solving a Control Allocation (CA) problem. The authors of this thesis report have proposed an energy efficient optimization based CA problem which is inspired from Mixed-Optimization CA problem. The control solution obtained by solving this proposed optimization based CA problem is intended to minimize the energy consumption by reducing the energy losses, specifically the electric loss in wheel motors and tire losses such as lateral and longitudinal slip losses and rolling resistance loss, while maintaining the path tracking performance of BEV with minimum lateral deviation. The path tracking algorithm was developed only to dictate the desired behaviour to the BEV in order to track a path, therefore replacing the need for the driver model. The proposed CA method is implemented in two different ways, one is by solving the CA problem to achieve the required behaviour of the BEV at every time step, and the other where the CA problem is solved by considering the required behaviour of BEV for certain upcoming time frame. The former approach is known as Instantaneous optimization based CA and the latter approach known as Predictive CA is implemented using Nonlinear Model Predictive Control (NMPC). The cost function and the constraints involved in optimization based CA problem are nonlinear in nature and is solved as an NLP problem. Two different Newton type optimization algorithms namely Sequential Quadratic Programming (SQP) and Interior Point are explored in order to solve the NLP problem. It was found that the formulated NLP problem is better solved by a large-scale algorithm such as IP rather than solving using SQP which is a small-scale algorithm. Of the two approaches, the predictive method i.e. the NMPC based CA problem's solution was found to have less energy consumption than the solution given by the instantaneous method of solving the CA problem by 3.89 %. The combined electric energy losses in motor and energy losses due to tire dynamics was less by 3.41 % .The allocated torques had least amount of torque vectoring in NMPC based CA compared to instantaneous CA method. The lateral deviation for path tracking was also less in NMPC relative to instantaneous. The front axle steering angle allocated for both the methods had transient behaviour, more prominent in solutions given by instantaneous CA problem. A comparison between energy consumption for torques allocated by NMPC based CA and equal torque distribution (ETD) using IPG driver model controlling the lateral dynamics was done. It was found that the ETD driving scenario had lesser energy consumption by 2.03 % and overall losses less by 2.4 % compared to NMPC based CA. Out of which 65 % of this excess loss in NMPC based CA compared to ETD was due to excess in lateral slip loss due to transient steering angle control allocated by NMPC based CA. Therefore the energy savings given by NMPC based CA can be further improved by regularizing the transient behaviour of the allocated steering angles.

Keywords: Battery Electric Vehicle, Vehicle Dynamics, Control allocation, Numerical optimization methods, Model predictive control, Nonlinear programming

Acknowledgements

We would like to thank Chalmers University and Volvo Cars Corporation for giving us the opportunity to work on this masters thesis which has the potential of improving the energy efficiency of electric ground vehicles and promote sustainable transportation solutions which is indeed the call of the hour. This masters thesis work wouldn't have been accomplished without the constant support and intuitive guidance of our main supervisors at Volvo Cars Corporation- Dr. Derong Yang and Juliette Torinsson. Their inputs have played a very important role in fetching good results from the work involved in this thesis. We are also grateful to our manager Anna Söderlund, for believing in us and allowing us to work on such an interesting and challenging topic and ensuring that our time in Volvo Cars was hassle free. We would also like to extend our immense gratitude to our academic supervisor Dr. Mats Jonasson, for his guidance and advice during the early stages of this work which has helped in channeling this thesis work in the right direction. Last but not the least, our parents, who have been indefinitely supporting us from miles away to stay focused at our goals and reminding us that the only person one needs to be better than is the one he was yesterday.

Contents

List of Figures

List of Tables

1 Introduction

EVER since the emergence of the industrialization era, a world with zero carbon footprint does sound like utopia. Of the many factors contributing to the existing drastic rate of climate change today, harmful emissions from transportation is quite a credible factor. In a time where transport is responsible for almost 30% of CO_2 emissions in EU, of which 72 % comes from road transportation, resorting to a cleaner source of energy for transportation is something that should not be overlooked.

To reduce the carbon emissions emitted by the automobiles, there has to be a shift in the inclination of consumer base from automobiles with IC powered engines to electric vehicles. In the recent decades, many automobile industries have started to acknowledge this impact of transport emissions on the environment and have given significant importance to electrification of vehicles as a sustainable energy solution to power the vehicles. In such times of need, every automobile industry ought to reduce their carbon footprint for better future and Volvo Cars Corporation, on similar lines, has proposed an ambitious plan of having 50% of their global sales to be all-electric cars by year 2025.

To achieve this shift, the electric vehicles have been improved continuously, thanks to the community of research and academia where the goal has been to make the electric vehicles more reliable and promising in terms of its performance and efficiency. The recent explosion of technologies in the fields of high power density batteries and energy management systems [1] [2] has pushed the drive range of BEVs to an acceptable level. If one can look at the powertrain of BEV as a mechatronic control system, with a set of controllers controlling a set of actuators in the BEV's powertrain, there exists multiple windows of opportunity to achieve the desired outcome of the vehicle, in terms of traction control, driving comfort, improved energy efficiency etc. In an all-wheel-driven BEVs, one of such mapping between controllers and actuators can be applied to the in-wheel motors housed in the individual wheels of the vehicle [3][4]. Having individual in-wheel motors allows to control the torques of each wheel independently. From control engineering's vantage point, the electric motors can be controlled much more precisely with shorter control period than IC engines. This leads to improving vehicle features such as antiskid breaking systems (ABS) and traction control system (TCS) [3].

Marching on in the similar direction of research and development of electric cars, the work presented in this thesis at Volvo Cars, is an attempt to contribute towards improving the driving range and the energy efficiency of an all wheel driven battery electric vehicle (BEV) with individual in-wheel motors.

With four independent in-wheel motors contributing to the planar motion of the vehicle, the electric vehicle as a system stands affirmed to the definition of what is known as an over-actuated system [5, 6, 7]. A classical definition of over-actuated system is simply a system with number of actuators being greater than the degrees of freedom of the system[8][9]. An over-actuated system has more than one control signal in its control space to control a state in the state space in order to achieve the desired behaviour of the system.

An event of such over-actuation in BEV occurs while achieving the required longitudinal, lateral and yaw dynamics for the motion of the vehicle. The required lateral motion in terms of yaw moment can be delivered by actuating the wheel angles to steer the vehicle in order to achieve the desired heading and also by employing torque vectoring i.e having differential torques in the wheels on the either sides of the axle. The wheel torques also contribute to the required longitudinal dynamics. Therefore, the BEV has multiple controls in terms of wheel torques and wheel angles, which can provide the required attitude of the vehicle. The energy losses occurring within the frame of wheel motors and tires are a function of the wheel motor torques and wheel's

orientation while driving. If the BEV can be made to understand which of the available combination of controls (wheel torques and steering angles) gives least energy losses and also delivers the required attitude of the vehicle, then the impact on the energy efficiency of BEV can be expected to be positive.

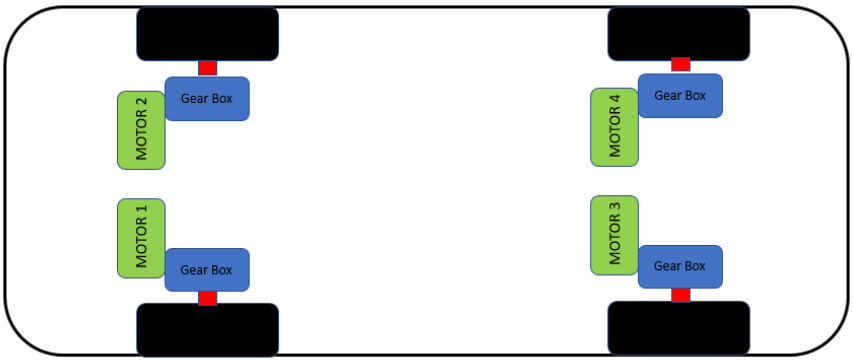


Figure 1.1: *Battery electric vehicle with four in-wheel motors*

1.1 Objective

The objective of this thesis is to develop a numerical method algorithm, which for a given set of desired longitudinal, lateral and yaw motion, should choose a set of wheel torques and front axle wheel angle as the controls out of the available control layers in the over actuated control space of the BEV. This combination of actuation should be employed such that the performance of the vehicle is stable and as desired. Also from the vantage point of energy efficiency, the power consumed from the battery for this allocated set of controls must have minimal electric losses in the in-wheel motor and the tire-losses arising due to lateral slip, longitudinal slip, and rolling resistance.

The approach in this thesis involves developing a cost optimization based control, in which the losses occurring in the wheel motors and tires are modelled as a function of controls i.e. the motor torques and front wheel steering angles. This loss function is the cost to be optimized whenever the torques and the wheel angles are being allocated to the wheels, to achieve the desired behaviour of the vehicle, based on the requested longitudinal acceleration and the yaw moment. This optimal control problem (OCP) is also subjected to a certain set of constraints which reflects the operating range of the actuators in terms of maximum and minimum extent of motor torques and front wheel steering angles actuation in order to have the optimal control signals within this pre-defined operating range of the actuators. This approach is known as 'Control Allocation' (CA) which is a common solution used for allocating controls with a desired objective in an over-actuated system, where controls are allocated such that a function of control signals is minimized [8][9][6].

For the longitudinal acceleration and the yaw rate requests which is usually requested by the driver who is driving the vehicle, the control allocation in this thesis will be made to abide by the requests generated by a path tracking algorithm. The references for the request come in the form of longitudinal velocity and yaw angle, which are generated online, based on the vehicle's current state which helps in making the vehicle know what velocity and heading should it have, in order to stay on track. The algorithm uses the position profile and the velocity profile of a given path in order to generate the requests based on the current state of the vehicle. This implemented path tracking algorithm is also another significant feature that compliments the proposed control allocation approach in this thesis work.

1.2 Research Questions

The research questions that will be addressed while carrying out this thesis work are listed below :

- How to model the non-linear dynamics of the vehicle and the power losses for the objective function for the optimization problem such that the optimization problem has a good affinity for convergence or at least an approximation of an optimal solution, as fast as possible?
- How should the path tracking algorithm be formulated such that it follows the desired path along with control allocation in a parallel manner?
- For path tracking to take place, it needs some references that can trigger the reference motion model to give the desired acceleration in order to follow the desired path. How to generate such references for the desired motion of the vehicle depending on the vehicle's plant model?
- The literature corresponding to previous research work has shown various strategies for implementing control allocation in the systems with each having their respective pros and cons. Which of these strategies is well applicable for the problem in hand?
- Apart from implementing CA for path tracking, using the given longitudinal acceleration and yaw rate requests instantaneously at the current instant of time, there is also a possibility to have a predictive behaviour by implementing the control allocation for path-tracking over a set of future longitudinal acceleration requests and yaw rate requests based on reference parameters generated for a given window into future, required for path tracking. This method can be applied using Non-linear Model Predictive Control (NMPC) method for control allocation. How to formulate the NMPC for the developed optimization problem and which of the two methods i.e. instantaneous CA or predictive CA is an energy efficient approach for the problem within the scope of this thesis work?
- The formulation of control problem begins with discretizing the continuous nonlinear vehicle dynamics model in order to capture the states of the vehicle at every instant of time. What class of discretization method will help in discretizing the non-linear ordinary differential equations such that the non-linear characteristics are captured to the at most extent as possible and not lost to the process of approximation done by the discretization methods, in a time-efficient manner?
- For the numerical optimization of the formulated non-linear constrained- optimal control problem (OCP), it needs to be converted into a Non-Linear Programming (NLP) problem. Different methods exists to convert the OCP to NLP, known as transcription methods. Based on the problem in hand, which type of transcription methods is computationally efficient and feasible to implement?

1.3 Scope

The methods involved and the results of this thesis work will be analysed based on the simulations, in simulation environments like MATLAB Simulink and IPG CarMaker. The focus of this work will only be on the losses occurring between the motors and tire receiving the torques from the motors. Losses like transmission losses and the impact of coupling and decoupling the motors are omitted in the study of this work. For the wheel angles, only the front axle angle is considered as a part of the control vector in Control Allocation problem. The state estimation feedback from the vehicle plant are assumed to be close to the true values of the states such as the longitudinal velocity and yaw moment of the vehicle.

A positive outcome of this thesis project can provide a method of finding an energy efficient control solutions for allocating wheel torques and front axle steering angle in an over actuated control space of a BEV. This contributes towards improving the BEVs driving range by improving its energy efficiency and promoting sustainability.

The following part of the report emphasizes on the work conducted in this thesis in 5 comprehensive chapters. Chapter 2 reflects upon the background of the work which involves the information related to previous research work which are relevant to the work conducted in thesis along with a brief mathematical representation of CA problem and why mixed optimization CA method is seeked for inspiration of the other developed CA methods, for the development of CA method proposed in this thesis. Chapter 3 covers the methodology followed to develop the energy efficient CA optimization problem for path tracking. This chapter also explains the two different approach of employing the developed energy efficient CA optimization problem i.e. in an instantaneous manner and in a predictive manner which uses NMPC. Chapter 4 highlights the simulation results obtained in this thesis work, followed by the discussion on the results in Chapter 5 and concluding the work in Chapter 6.

2 Background

2.1 Literature Survey

The onset of research on electric vehicle's (EV's) motion control has provided a good amount of intuition into how the controllers can be designed to achieve the desired attitude of the vehicle by controlling the individual in-wheel motors accordingly.The literature survey was done initially to understand various vehicle control techniques and then address the work that used the advantage of over-actuation for controlling the vehicle.

Sakai et al. [3] introduced a novel "robust model matching controller" (R-MMC) which is a dynamic yaw-moment controller (DYC) that generates the desired yaw input by changing the torque distribution ratio between the left and right wheels of the EV that has independent driven in-wheel motors. For over-actuation which is also encountered in ships and underwater vehicles, Fossen et al. [10] made a survey on various methods of control allocation for over actuated marine vessels. The methods involved both online and offline optimization methods. The objective was to minimize the control effort subject to actuator rate, power constraints along with other operational constraints. A very clear and illustrative discussion on various methods of control allocation has been made by Bodson [8] and Oppenheimer [9] for flight control system. It shows how a Mixed Optimization method which is a combination of control and error minimization, can be solved as a standard linear program and was found to be a viable method for solving the online control allocation problem. Johansen et al. [11] emphasises on optimizing a non-linear control allocation using a Lyapunov design approach for improved computation efficiency. Tjønnås et al. [12] extended the work in [11] to construct update-laws that represent an asymptotically optimal allocation search and adaptation by using Lyapunov analysis. Torinsson [13][7] has an extensive work and findings on developing an analytical approach for CA using quadratic programming in electric cars as well as the impact on energy efficiency for having CA with different transmission ratios in the wheel motors along with coupling and decoupling the motors. M. Vignati and others [14] showed the ability of having different torque vectoring control strategies in electric vehicles with individual in-wheel motors and their respective impact on vehicle's performance while drifting.

The CA problem for overactuation in BEV is posed as an optimization control problem (OCP) in this work. Most of the optimization problem are solved by posing it in its equivalent nonlinear programming (NLP) form. The use of NLP approach for solving OCP has been a potential research topic in the field of numerical optimization methods. The motivation for NLP approach is its ability reduce the variable space by mapping the formulated cost function into a scalar value which is minimized using NLP algorithms. L. Kek [15] has summarized the research done on formulating various methods of converting an OCP to its equivalent NLP. The work shows the optimal solution of the formulated NLP model using control parametrization approximates closely to the true solution of the original OCP. P. Kelly [16] gave a thorough overview on two major classes of methods to convert OCP to NLP using simultaneous and sequential transcription methods and a detailed analysis between two simulataneous transcription namely Single-shooting method and Multiple-shooting method.Two most renowned methods for solving NLP problems are Sequential Quadratic Programming (SQP) and Interior Point (IP) methods. Both are classified under Newton Optimization methods with their respective dfferences. M. Diehl et al. [17] has covered a good overview on these two familes of Newton type numerical methods to solve real time optimal control problem in terms of NLP. MATLAB's documentation also has an extensive information regarding how these two NLP algorithms differ from each other and when are they recommended as the solver of choice based on the type of optimal control problem to solve.

Since the work requires making the developed CA algorithm allocate controls based on the requested vehicle dynamics corresponding to the path to be followed, the literature survey was also led in the direction of finding some of the path tracking techniques employed in the ground vehicles. M. Hoffman et al [18] had an approach of using the orientation of the front wheels of the vehicle instead of the orientation of the entire vehicle body using the equations of kinematics. A PI controller based steering law was used in developing this autonomous automobile tracking for off road driving and the root mean square of cross track error was under 0.1 m. Many of them have emphasized the use of nonlinear model predictive control (NMPC) method if the path can be profiled with the states required to follow the path. G. Bai et al. [19] used NMPC approach for controlling unmanned mining vehicle. The displacement and the heading error were adequately minimum. G.L Campagne [20] used NMPC approach for path planning and path tracking for developing an evasive manoeuvre assist function. The impact of SQP and IP method based solvers for the developed NMPC based approach is also highlighted in this work. [21] Derong Yang et al. is a study on brake based path control using a quasi-linear optimal path controller which provides insights into modelling of the vehicle and a simplified magic formula tire model. The work in [22][23] provides an detailed explanation of the steps involved in implementing path tracking using pure pursuit algorithm. The work also examines the effect of various parameters on the performance of the algorithm in various driving scenarios and states the drawbacks of the algorithm.

Therefore, to the farthest knowledge of the authors of this thesis, none of the previous work has involved implementing control allocation combined with a tracking algorithm. The work also involves using NMPC approach to compliment the path tracking along with the developed control allocation method.

2.2 Over Actuation and Control Allocation

In order to understand the approach used in this thesis work, this section covers a comprehensive understanding of what over-actuation means within the domain of control systems and different methods of implementing control allocation as a solution for solving the problem of over-actuation based on the previous research work and literature survey.

2.2.1 Over-Actuation - A cluster of control solutions

Over-actuation can be imagined as a situation where one knows multiple methods of solving a particular problem. Where every method has a caveat. Some solve the problem with high accuracy, some might be easy to implement, some might take longer time etc. but all of them can solve the problem, eventually. Based on one's preference, a particular method can be chosen to solve the problem out of the multiple available methods. This situation of knowing multiple methods for how to solve a particular problem is same as having multiple control combination in a system to achieve the desired output or behaviour of the system.

Over-actuation is a common occurrence in many control architectures such as flight control systems, ships and submarines [9][10], and the one which is of the prime interest in this thesis - an all-wheel driven BEV [5]. With a system being over-actuated, there exists multiple control solutions for how the controls can be allocated to achieve the desired behaviour of the system. The effect of each solution may or may not be unique but given a preferred control effect, a control solution can be chosen from the multiple available control solutions.

In an all-wheel driven BEV with in-wheel motors, over-actuation provides many different solutions for how the torques and steering angles are distributed between the wheels to achieve the desired dynamics. This consequence of over-actuation can be exploited by choosing a ¨preferred control solution¨ that not only provides the desired dynamics but also is energy efficient in nature. This problem of finding a ¨preferred control solution¨ in an over-actuated system is known as Control Allocation problem which is discussed briefly in the following subsection.

2.2.2 Mathematical representation of Control Allocation

The principle of over-actuation and CA is easy to understand with a continuous linear state space model and a model reference control law [24]. Consider a continuous linear state space model as shown in Equations 2.1

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{2.1}$$

where $\mathbf{x} \in \mathbb{R}^{n\times 1}$ represents the state vector with n degrees of freedom (DOF) ($n = dim(x)$) and $\mathbf{u} \in \mathbb{R}^{m\times 1}$ represents the control vector with m number of controls/actuator signals i.e $m = dim(u)$. Matrix A $\in \mathbb{R}^{n\times n}$ is the system matrix, B $\in \mathbb{R}^{n\times m}$ is the control matrix, it is assumed that the dimension of output vector $\mathbf{y}$ of the system is equal to the dimension of state vector $\mathbf{x}$ i.e $dim(y) = n$, hence the output matrix C $\in \mathbb{R}^{n\times n}$. If the equations in 2.1 is assumed to represent the linear two-track vehicle dynamics model which will be briefly discussed in the Vehicle model section of this report, then the state vector has 3 DOF, i.e $n = 3$ with the states - lateral velocity v_y, longitudinal velocity v_x and the yaw rate r. The control vector has 5 control inputs

$(m = 5)$ i.e. four individual motor torques T_{m_i} $(i = 1, 2, 3, 4)$ and a front steering angle (δ_f). As $m > n$, the matrix B is non-invertible matrix.

If the desired behaviour of the system can be represented using a reference model as shown in Equations 2.2,

$$\dot{y}_{des} = A_m x_m + B_m u_m \tag{2.2}$$

where $_{des}$ represents the desired dynamics of the closed-loop system and u_m is the applied input command, the objective of allocating the controls as per the desired dynamics given by 2.2 can be achieved by equating the derivative of the output vector in 2.1 to the Equation in 2.2 as shown below

$$\begin{aligned} \dot{y} &= C\dot{x} \\ \dot{y} &= CAx + CBu \\ \dot{y} &= \dot{y}_{des} \\ \implies CAx + CBu &= A_m x_m + B_m u_m \\ CBu &= \dot{y}_{des} - CAx \end{aligned} \tag{2.3}$$

Where the matrix product CB is termed as 'Control Effectiveness Matrix' $\mathbf{B_f} \in \mathbb{R}^{n \times m}$ and the term on the RHS of 2.3 $(y_{des} - CAx)$ is the desired control effect known as the virtual control input $\mathbf{v}$. Hence, the equation in 2.3 can be rewritten to get the solution $\mathbf{u^*}$ which gives the desired dynamics y_{des} as shown in 2.4

$$\mathbf{u^*} = (\mathbf{B_f})^{-1}\mathbf{v} \tag{2.4}$$

$$\begin{aligned} \mathbf{u_{min}} \leq \mathbf{u^*} \leq \mathbf{u_{max}} \\ \dot{\mathbf{u}}_{\mathbf{min}} \leq \dot{\mathbf{u}}^* \leq \dot{\mathbf{u}}_{\mathbf{max}} \end{aligned} \tag{2.5}$$

subjected to constraints defined by the maximum and minimum actuation effort and rate of actuation as shown in 2.5. With $m \geq n$ for over-actuated system, the control effectiveness matrix $\mathbf{B_f}$ is a non-square and a non-invertible matrix, furthermore, solving for the solution $\mathbf{u^*}$ with the non-square $\mathbf{B_f}$ matrix involves more unknowns than equations, creating an under-determined system, leading to the uncertainties of getting a unique solution based on the rank of the $\mathbf{B_f}$ matrix. This process of finding the solution vector $\mathbf{u^*}$ for the over-actuated system and the reference/request model posed in Equations 2.1 - 2.3 is known as **Control Allocation** according to [8].

The aforementioned approach of control allocation can also be extended to a non-linear system of equations in 2.6 [25][7]

$$\dot{\mathbf{x}} = \mathbf{f(x)} + \mathbf{g(u)} \tag{2.6}$$

The approach involves linearizing the system around a stationary control value and the rest of the approach and the thoughts around the definition of CA remains the same as shown in this section leading to the result in 2.4

The following section in this chapter attempts to brief about different methods of solving CA problem,

followed by a section where the use of optimization based CA method is motivated for the problem meant to be solved in this thesis work.

2.3 Different methods to solve the Control Allocation problem

As mentioned in the previous section, both linear and non-linear over-actuated systems can be can be solved by solving the linear control allocation problem mentioned in 2.4. The problem of having a non-invertible control effectiveness matrix $\mathbf{B_f}$ can be solved either by methods of converting the non-square matrix $\mathbf{B_f}$ into a square matrix or by directly solving the CA in 2.4.

- **Explicit Ganging**:

  Ganging in terms of control theory is a method of combining more than one control signals in the control vector $\mathbf{u}$ with a prior knowledge of the actuation effect of the combination. The resulting vector of this combination of controls is known as *pseudo control* vector u_{pseudo}, the name pseudo comes from the fact that the elements in the u_{pseudo} do not exist in physical control space but rather they reflect the combination of controls. The actual control vector and the pseudo control vector can be related using a ganging matrix $\mathbf{G}$ that defines the transformation between the two vectors as shown in Equation 2.7

  $$\mathbf{u} = \mathbf{G}\mathbf{u_{pseudo}} \tag{2.7}$$

  If the matrix $\mathbf{G}$ can be found such that the substitution of 2.7 in 2.4 results in a square control effectiveness matrix then the problem of CA is easily solved when $\mathbf{G} \in \mathbb{R}^{m \times n}$. Although to implemet this method, a priori of relation between the pseudo and actual controls is required.

  $$\mathbf{B_f G u_{pseudo}} = \mathbf{v} \tag{2.8}$$

- **Pseudo Inverse and Redistributed Pseudo Inverse methods**:

  This is a constrained optimization method which involves finding the vector $\mathbf{u}$ by solving the following constrained optimization problem with CA problem as the constraint and the cost function as shown in 2.9

  $$\begin{aligned} \min_{\mathbf{u}} \quad & \mathbf{J} = \min_{\mathbf{u}} \frac{1}{2}(\mathbf{u}+c)^T\mathbf{W}(\mathbf{u}+c) \\ \mathbf{s.t.} \quad & \mathbf{B_f u} = \mathbf{v} \end{aligned} \tag{2.9}$$

  where $\mathbf{c} \in \mathbb{R}^{m \times 1}$ is an off-nominal condition on control vector $\mathbf{u}$ which indicates the offset in the desired controls and $\mathbf{W} \in \mathbb{R}^{m \times m}$ is a weighing matrix. The optimization problem has an analytical solution as shown in 2.10

  $$\begin{aligned} \mathbf{u} &= -\mathbf{c} + \underbrace{\mathbf{W}^{-1}\mathbf{B_f}^T(\mathbf{B_f}\mathbf{W}^{-1}\mathbf{B_f}^T)^{-1}}_{B^{\#}}[\mathbf{v} + \mathbf{B_f c}] \\ \mathbf{u} &= -\mathbf{c} + \mathbf{B}^{\#}[\mathbf{v} + \mathbf{B_f c}] \end{aligned} \tag{2.10}$$

  where the term $\mathbf{W}^{-1}\mathbf{B_f}^T(\mathbf{B_f}\mathbf{W}^{-1}\mathbf{B_f}^T)^{-1}$ gives the pseudo inverse of the non-square matrix $\mathbf{B_f}$ for the solution in 2.10. This method does not guarantee the controls respecting the actuation rate constraints in

2.5 but is helpful in generating preference vectors in a more complex optimization based methods for robust analysis.

The probability of not obeying the actuator constraints using Pseudo Inverse method is resolved in its iterative dual problem called as *Redistributed Pseudo Inverse* which works in the similar manner as Pseudo Inverse but in an iterative fashion, where the iterations stops if no controls given by the solution exceeds minimum or maximum actuator constraints in 2.5. If a particular control signal in the solution vector $\mathbf{u}$ saturates or exceeds the limit then iteration continues by zeroing out the the rows in the B matrix and adding a negative offset value at the respective position in the off-nominal vector $\mathbf{c}$.

- **Daisy Chaining**

  This method assumes a hierarchy of controllers. If one controller on top level cannot produce the desired control effect $\mathbf{v}$, then the residual of the desired effect is requested to be fulfilled by the following controllers in the sequence of the hierarchy

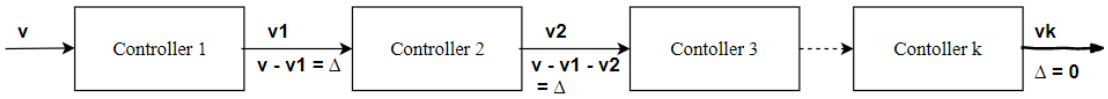


Figure 2.1: *CA requests being satisfied through a hierarchy of controllers*

In the Figure 2.1, the Controller 1 is the controller at the top of the hierarchy which is requested to achieve the desired control effect v. In the event of Controller 1 not completely satisfying the desired control effect, the residual of the desired control effect as Δ is then passed on to the successive controllers in the hierarchy of the controllers until this residual of desired control effect nullifies.

- **Direct Allocation method**

  Direct allocation method is again a constrained optimization problem which involves finding a real number ρ in the following optimization problem 2.11

$$\begin{aligned} & \max \mathbf{J} = \max \rho \\ \textbf{s.t.} \quad & \mathbf{B_f u}_1 = \mathbf{v} \\ & \mathbf{u_{min}} \leq \mathbf{u} \leq \mathbf{u_{max}} \end{aligned} \tag{2.11}$$

where $\mathbf{u} = \frac{1}{\rho}\mathbf{u}_1$ if $\rho > 1$ and $\mathbf{u} = \mathbf{u}_1$ if $\rho \leq 1$. This method was proposed by authors in [25] and works on the principle of attainable moment set (AMS) which is a convex hull of planar control surfaces. The term *moment* within the frame of the problem in 2.4 is the control effect or 'moment' produced by the product $\mathbf{B_f}u$.

In case of the problem in 2.11, the convex hull of planar surface represents the attainable control effect or the demand from the system as a function of control vector $\mathbf{u}$ and its constraints in 2.5 represented by the boundaries of the AMS. If $\rho \leq 1$ then the demand is within the AMS, i.e. a feasible solution respecting the control constraints is achieved. If $\rho > 1$ then the demand makes the controls exceed the control constraints, i.e the demand lies outside the convex hull of the AMS.

This method has been proven to be promising as it is easily solvable using linear programming and simplex algorithm [8]. However, the construction of AMS for a non-linear problem of the form $\mathbf{f}(\mathbf{u}) = \mathbf{v}$ is difficult [9].

- **Mixed Optimization method**

  Given a control effectiveness matrix $\mathbf{B_f}$ and a preferred control vector $\mathbf{u}_p$, the mixed optimization method tries to find the control vector by minimizing the following cost function in 2.12.

$$\begin{aligned} \min_u \mathbf{J} &= \min_u \|\mathbf{B_f u} - \mathbf{v}\| + \gamma \|\mathbf{u} - \mathbf{u}_p\| \\ \textbf{s.t.} \quad & \mathbf{u_{min}} \leq \mathbf{u} \leq \mathbf{u_{max}} \end{aligned} \tag{2.12}$$

  Where the first term in the cost function $\mathbf{J}$ is the error in the desired control effect and the second term is the error in control supply. γ is the weight constant that weighs the relative importance of error and control minimization. If γ is small in value then error minimization (first term in $\mathbf{J}$) is prioritized over control minimization (second term in $\mathbf{J}$).

  This method is inspired from the Error and Control Minimization method [9] which offers a good degree of flexibility in achieving control allocation based on the task to be performed by the system. Depending on the complexity of the system, the optimization problem in 2.12 can be transformed into linear or non-linear programming and solved.

2.4 Motivation for using Optimization based Control Allocation

Of the discussed methods of CA in the previous section, the ones that compliments the research work pertaining to this thesis seems to be well supported by the aids of constrained optimization approach used in *Mixed Optimization method*, as it allows to introduce an additional term of desired objective along with CA error represented by the first term in J. In an event where no unique solution can be found for distributing the controls to the actuators, the solution that minimizes the control error i.e. the second term of the cost function is chosen as the optimal solution to the problem.

However, in the optimization CA problem defined in 2.12, the first term is minimizing the control allocation error and the second term is penalizing the difference in the amplitudes of the actuators relative to a preferred actuator signal. Previous research work [5][6] have shown that the 'minimal actuator outputs does not necessarily lead to minimal power or energy consumption'.

Therefore, seeking inspiration from the approach of Mixed Optimization method, a modified version of this method is used within the framework of this thesis for improving the energy efficiency of BEVs, by introducing a loss model as the term to be penalized which replaces the second term of the mixed optimization method in 2.12. This modified implementation of Mixed Optimization based CA and the formulation of loss models will be briefly discussed in the Methodology section.

3 Methodology

The three main ingredients in formulating an energy efficient control allocation wheel motor torques and front axle steering angle in BEV for path tracking are represented in the figure 3.1. Once the framework of energy efficient CA algorithm represented in the Figure 3.1 is successfully developed, the algorithm is evaluated by running simulations in Simulink and IPG CarMaker environment to validate the research work of this thesis.

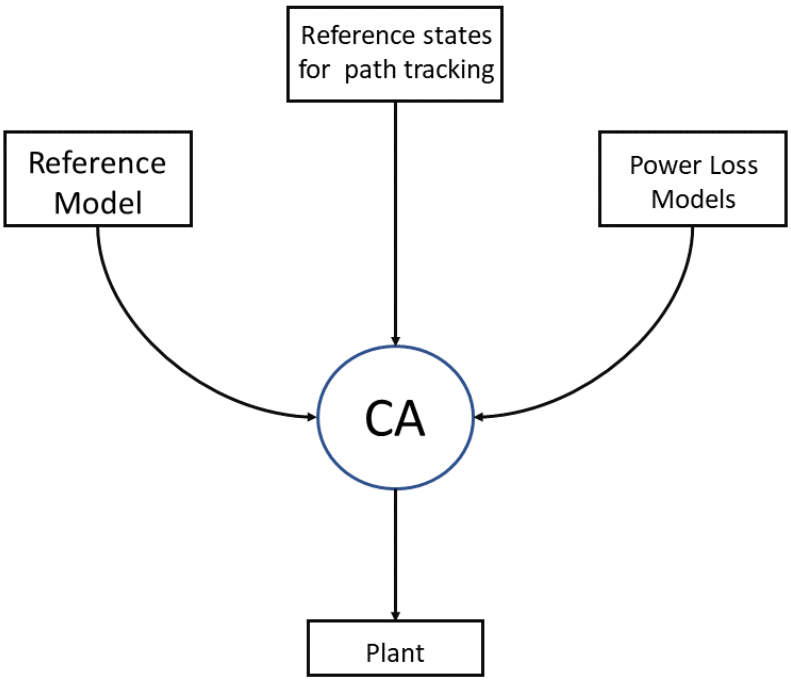


Figure 3.1: *Three main ingredients for formulating the energy efficient CA for a BEV*

1. **Reference Vehicle Model** : A two-track vehicle model is used as a reference model which helps in making the control allocator understand how the states evolve for the controls which it decides to allocate.

2. **Reference states for path tracking** : The path tracking is done by using global coordinates profile and the velocity profile of a given path. Based on the current position of the vehicle, the required longitudinal velocity and yaw angle for path tracking are computed online and given as the desired states to the control allocator.

3. **Power Loss Model** : As seeked from the literature survey, just a mere reduction in the amplitude of actuation post control allocation does not necessarily leads to reduced energy consumption in the system. To target the power losses arising due to the actuation of wheel torques and angles specifically, the control allocator should be provided with a loss model to know the impact on losses for the controls it decides to allocate.

4. **Control Allocator** : After a thorough literature study on the previously conducted research work, the method of solving CA problem using mixed optimization suits well to solve the non-linear control

allocation problem in hand, as it allows to inculcate the control allocator the idea of power losses and actuator constraints simultaneously while solving the optimization problem. This CA optimization problem problem is eventually solved as a non-linear programming (NLP) problem in an instantaneous and predictive manner. Two different Newton method type NLP solvers viz. IPOPT - an interior point method based solver and SQP - an iterative quadratic programming solver [17] are used and the respective results are analyzed in terms of the path tracking performance, energy efficiency computational cost efficiency, and robustness. This work uses toolbox such as fmincon - an optimization toolbox offered in MATLAB environment and an open-source tool for non-linear optimization known as 'CasAdi' which provides rapid and efficient implementation of different methods for numerical optimal control for solving the formulated optimization based CA problem.

5. **Vehicle Plant**: Two types of vehicle plant are used in this work. A low-fidelity, two-track vehicle model based plant for the simulations during the development of the algorithm. A high fidelity vehicle model for testing and validating the developed algorithm using IPG CarMaker environment.

A section wise breakdown of developing the three main aforementioned ingredients required for developing an energy efficient control allocator and the numerical methods involved in solving the formulated optimization based control allocation are discussed in the following parts of this chapter.

3.1 Vehicle Model

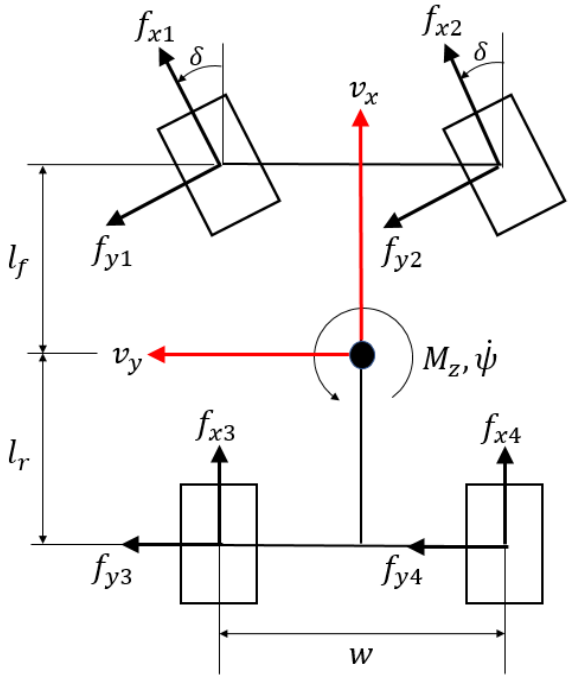


Figure 3.2: *3-DOF Two track vehicle model*

A 3 DOF two track model consisting of longitudinal, lateral and yaw motion of the vehicle is used to model the dynamics of vehicle. The equations of motion for the model are stated below from (3.1) to (3.6). Where F_{xi} and F_{yi} are longitudinal and lateral forces in the vehicle co-ordinates. Longitudinal, lateral and yaw accelerations are denoted by $\dot{v_x}$, $\dot{v_y}$ and $\ddot{\psi}$. X_g and Y_g are the co-ordinates of the vehicle position in the global

co-ordinate system. To keep the complexity of the model low, roll dynamics and weight transfer have not been considered.

$$m\dot{v_x} = mv_y\dot{\psi} + F_{x1} + F_{x2} + F_{x3} + F_{x4} \tag{3.1}$$

$$m\dot{v_y} = -mv_x\dot{\psi} + F_{y1} + F_{y2} + F_{y3} + F_{y4} \tag{3.2}$$

$$I_{zz}\ddot{\psi} = l_f(F_{y1} + F_{y2}) - l_r(F_{y3} + F_{y4}) + \frac{w}{2}(-F_{x1} + F_{x2} - F_{x3} + F_{x4}) \tag{3.3}$$

$$\dot{X}_g = \dot{x}\cos\psi - \dot{y}\sin\psi \tag{3.4}$$

$$\dot{Y}_g = \dot{x}\sin\psi + \dot{y}\cos\psi \tag{3.5}$$

$$\dot{\psi} = r \tag{3.6}$$

The relation between forces in vehicle co-ordinates and wheel co-ordinates are as given in equations (3.7) and (3.8) where f_{xwi} and f_{ywi} are wheel forces in wheel co-ordinates and δ_i is the wheel angle for the i^{th} wheel.

$$F_{xi} = f_{xwi}\cos\delta_i - f_{ywi}\sin\delta_i \tag{3.7}$$

$$F_{yi} = f_{xwi}\sin\delta_i + f_{ywi}\cos\delta_i \tag{3.8}$$

3.1.1 Tire Model

The tire model used in this thesis is the Magic Formula tire model[26]. It is a semi emperical tire model which is expressed as a function of the normal force on the tire and the slip angle as shown in equation(3.9). The coefficients B, C, D and E are used to fit the curve that resembles the curve from the experimental data. The coefficient D is the value of the peak force on the tire, BCD is the slope of the curve and E is the position of the peak along the horizontal axis. The values of the tire parameters are calculated following the method in [21].

The longitudinal wheel forces are related to wheel torques as given in equations (3.10) where τ_{mi} is the motor torque, r_e is the loaded wheel radius and n_t is the transmission ratio. The maximum lateral force available $f_{yi_{max}}$ is determined using the friction circle which is given in equation (3.11)[21].

$$f_{ywi} = -Dsin(C_i\tan^{-1}(B_i\alpha_i - E_i(B_i\alpha_i - \tan^{-1}(B_i\alpha_i)))) \tag{3.9}$$

$$f_{xwi} = \frac{\tau_{mi}n_t}{r_e} \tag{3.10}$$

$$D = f_{ywi_{max}} = \sqrt{(\mu_i F_{zi})^2 - f_{xwi}^2} \tag{3.11}$$

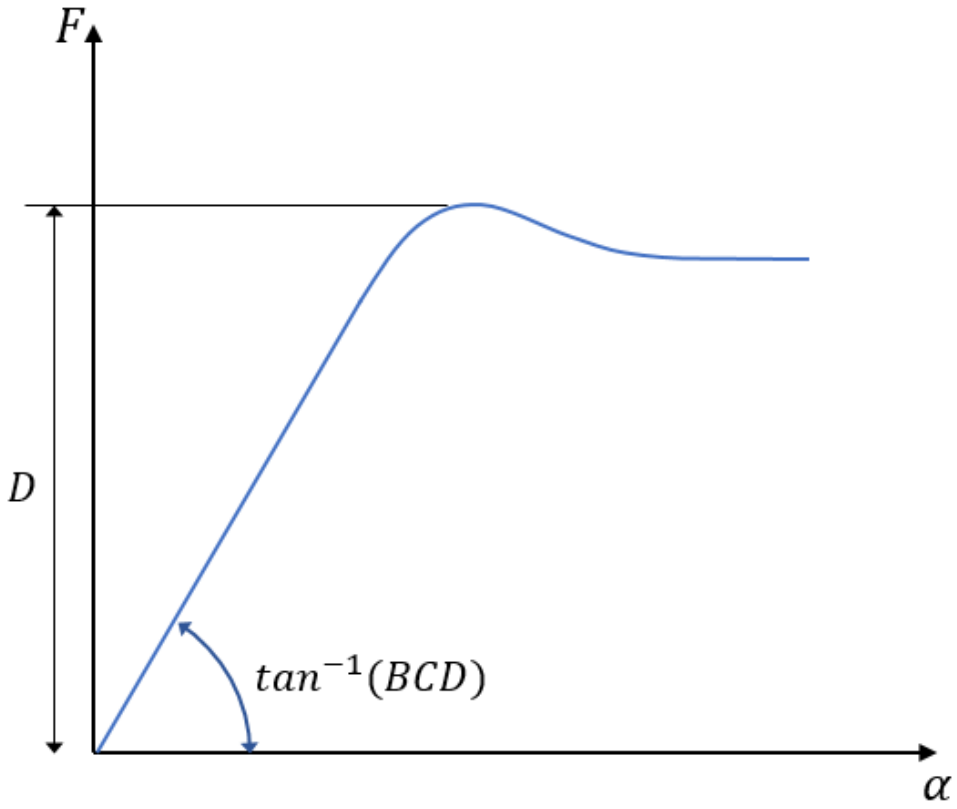


Figure 3.3: *Magic Formula Tire Model*

In equation(3.9), the slip angle α_i is defined as the difference in angle between the orientation of the wheel and the direction of travel of the said wheel. Figure(3.4) shows how the tire slip is calculated for a wheel that is steered at an angle of δ_i with respect to the x-direction in vehicle co-ordinate where the angle β_i where beta is the angle between the direction of travel of the wheel and the direction of heading of the vehicle. Equations (3.12) and (3.13) are used to compute the slip angles at each wheel where v_{xi} and v_{yi} are longitudinal and lateral corner velocities in vehicle co-ordinates which are computed using the equations (3.14)-(3.17)

$$\alpha_i = -\delta_i + \beta_i \tag{3.12}$$

$$\beta_i = \tan^- 1(\frac{v_{yi}}{v_{xi}}) \tag{3.13}$$

$$v_{x1} = v_{x3} = v_x - \frac{w}{2}\dot{\psi} \tag{3.14}$$

$$v_{x2} = v_{x4} = v_x + \frac{w}{2}\dot{\psi} \tag{3.15}$$

$$v_{y1} = v_{y2} = v_y + l_f\dot{\psi} \tag{3.16}$$

$$v_{y3} = v_{y4} = v_y - l_r\dot{\psi} \tag{3.17}$$

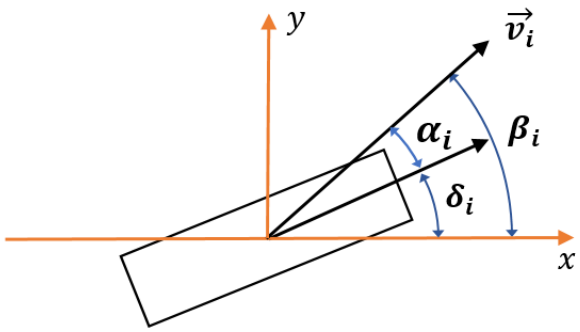


Figure 3.4: *Slip angle*

3.2 Modelling the Losses

During the operation of an electric vehicle, power losses occur in various sub systems of the vehicle such as tire losses, loss due to drag, transmission losses, battery losses and electric losses in motors and inverter. In this thesis, the losses that are chosen to be minimized are tire losses and electric losses in motors and inverter. The total power drawn from the battery P_{bat}, can expressed as the sum of power required to propel the vehicle P_{prop} and the electrical losses that occur when the requested torque is delivered to the wheels $P_{el,loss}$.

$$P_{bat} = P_{prop} + P_{el,loss} \tag{3.18}$$

The total power needed to propel the vehicle can be expressed as shown in equation(3.19). Where P_{im} is the input mechanical power to the electric motor from the battery. The modelling of these losses is shown in the following sections.

$$P_{prop} = P_{im} - P_{tire,loss} \tag{3.19}$$

3.2.1 Motor Electric losses

Measured power loss data for the entire operating region of the motor and inverter are used to model the electric losses. The surface fitting tool available on MATLAB is used to fit the electric loss data as a surface expressed in the form of a polynomial equation of order 2 for motor torque(T_m) and order 5 for motor speed(ω_m). Figure 3.5 shows the resulting curve-fitted surface for the electric losses as a function of torques and the motors speed

Using the information of the current motor speed(ω_m) from the vehicle plant,the polynomial expression can

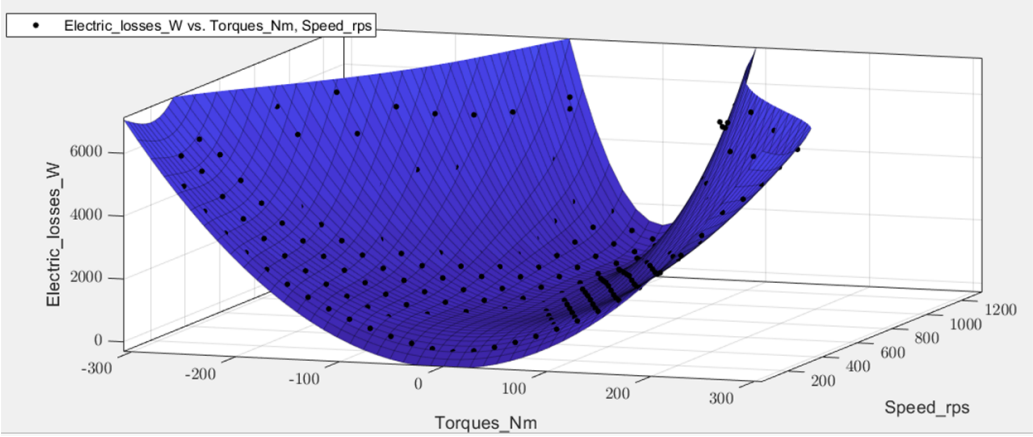


Figure 3.5: *Curve-fitting the electric losses in the individual motor as a function of torque and angular velocity*

be reduced to a quadratic equation in terms of motor torque as shown in equation (3.20) where p_0, p_1 and p_2 are the coefficients of the polynomials that are functions of motor speed ω_m.

$$P_{m,loss} = p_0(\omega_m) + p_1(\omega_m)\tau_m + p_2(\omega_m)\tau_m^2 \tag{3.20}$$

3.2.2 Tire-Losses

The tire losses are comprised of three kinds of losses viz. lateral slip loss, longitudinal slip loss and the rolling resistance slip loss. These losses mainly occur due to the difference in the dynamics between the vehicle body's frame and the wheel body's frame complimented by certain structural characteristics of the rubber material on the tires.

$$P_{tire,loss} = P_{sx} + P_{sy} + P_{rr} \tag{3.21}$$

The slip in the longitudinal and lateral direction of the vehicle arises due to the rubber element of the tires. The contact patch of the tire, i.e. the surface of the tire in contact with the ground follows the global vehicle longitudinal velocity vector whereas the dynamics within the local frame of the wheel makes the part of the rubber element just above the contact patch follow the rotational velocity of the wheel. This is known as longitudinal tire slip and is necessary for the forces to develop in order to achieve the desired traction. For the supplied motor torque, if the angular velocity of the wheel differs from the vehicle's longitudinal velocity significantly, there is excess tire slip which leads to power loss and is referred as longitudinal slip loss as shown in Equation 3.22. The control allocation algorithm must be acknowledged with this concept such that the torque allocation to different wheels does not generate excessive longitudinal slip.

$$P_{sx} = \sum_{i=1}^{4} f_{xi}(r_e\omega_{wi} - v_{xi}) \tag{3.22}$$

where f_{xi} is the longitudinal force acting on the wheel i, r_e is the nominal wheel radius, ω_{wi} and v_{xi} are the angular and longitudinal velocity of the wheel respectively.

Similarly, lateral slip losses occur when the steering angle of the wheel i.e. the orientation of the wheel is different from the orientation in which the wheel is moving. One cannot avoid steering the wheels to achieve the desired lateral force but this difference in the orientations of the wheel heading and the direction of the wheel's travel generates a force working against the direction of motion of the vehicle which results in some losses at the wheel's end which is referred to as lateral slip losses which is expressed as shown in 3.23

$$P_{sy} = -\sum_{i=1}^{4} F_{yi} v_{ywi} \tag{3.23}$$

where F_{yi} is the lateral force and the v_{ywi} is the lateral velocity in the wheel coordinate system which can be written as shown in 3.24

$$v_{ywi} = v_{yi}\cos(\delta_i) - v_{xi}\sin(\delta_i) \tag{3.24}$$

The rolling resistance losses on the other hand arise as a consequence of 'Law of conservation of energy'. As mentioned above, the rubber element of tire deflects and compresses at the contact patch of the tire and expands again to regain its shape when it leaves the contact patch, this continuous process of damping and regaining the structure is constantly resulting in conversion of potential energy back to heat energy. This is one of the form of rolling resistance losses known as hysteresis loss apart from the permanent plastic deformation of the the object or the surface for e.g. driving on soil. Only hysteresis losses are considered as a part of the rolling resistance losses. This loss of potential energy being dissipated as heat in the tires contributes to the total power loss and is referred to as rolling resistance losses which is expressed as shown in 3.25

$$P_{rr} = \sum_{i=1}^{4} -M_{yi}\omega_{wi} \tag{3.25}$$

where M_{yi} is the rolling resistance moment. Based on experimental tire measurements, the rolling resistance moment can be expressed as

$$M_{yi} = -F_{zi} r_0 \left\{ q_{sy1} + q_{sy2}\frac{F_{xi}}{F_{z0}} + q_{sy3}\left\|\frac{v_{xi}}{v_{ref}}\right\| + q_{sy4}\left(\frac{v_{xi}}{v_{ref}}\right)^4 \right\} \tag{3.26}$$

where F_{zi} is the vertical tire load of the tire , v_{xi} is the longitudinal velocity of the wheel center and the parameters q_{sy1}, q_{sy2}, q_{sy1} and q_{sy4} are from the experimental tire measurements obtained in [7]

3.3 Generating the references

For allocation of motor torques and wheel angle, the control allocator, now with the knowledge of the system model dynamics, should be fed with the reference states which defines the desired trajectory of the vehicle. These reference states are termed as 'Dynamics request'. It is for these desired states the control allocator tries to find a solution for the CA problem.

Having the information of the path to be followed as X-Y coordinates and the velocity at the respective waypoint in the global frame, the reference yaw angle and reference velocity are calculated using a method inspired by the work done on Pure Pursuit Path Tracking in [22]-[23].

The steps to calculate the reference longitudinal velocity v_{xref} and the reference yaw ψ_{ref} using the position

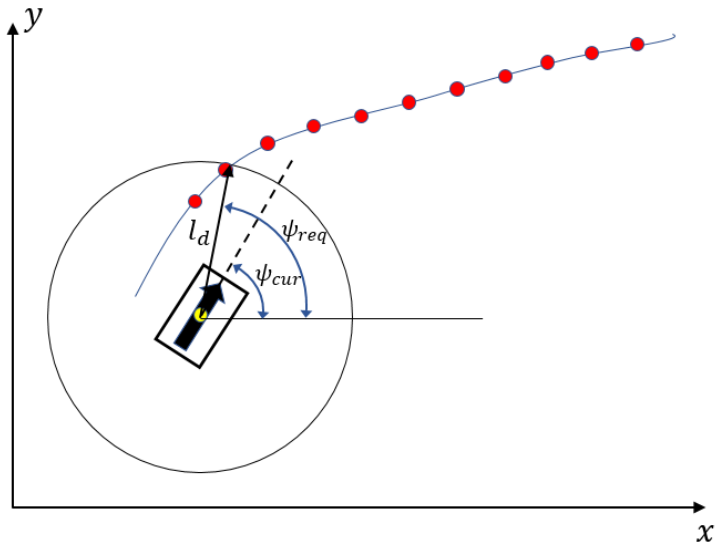


Figure 3.6: *Illustration of look-ahead distance and generating reference yaw angle as a request to the CA*

and velocity profile of the path are as follows:

1. The path data is created as a set of waypoints $[X\ Y\ v_x]$ seperated by a distance d. An index is assigned to every waypoint in the data.

2. The purpose of the index is to ensure that the car passes through every waypoint in the data. A log of the indices the car has already passed is recorded and only 100 waypoints ahead of the passed index are used while generating the references.

3. The algorithm searches for the waypoint nearest to the car and then climbs up along the path to find the waypoint closest to the look-ahead distance l_d.

4. The look-ahead distance l_d is a function or rather a multiple of the longitudinal velocity of the vehicle v_x as shown in Equation 3.27 and is a tunable parameter. This makes the look-ahead circle shown in 3.6 dynamic in nature i.e. at higher velocity, the look-ahead circle is bigger and vice-versa. The look-ahead distance has an impact on the aggressiveness of the path tracking as illustrated in figure(3.7). A smaller look-ahead distance causes the car to maneuver aggressively while a large look-ahead distance results in a smoother maneuvering of the vehicle but with a larger cross track error.

$$l_d = v_x n_l \qquad 0 \leq n_l \leq 1 \tag{3.27}$$

5. Yaw reference: The reference yaw angle ψ_{ref} is the angle made by line drawn from the CoG of the vehicle to the waypoint selected by the search algorithm. The reference yaw angle is calculated using the current position (X_{cur}, Y_{cur}) and the selected waypoint (X_{sel}, Y_{sel}) as shown in equation(3.28) where the function `atan2` gives the polar angle of the selected waypoint.

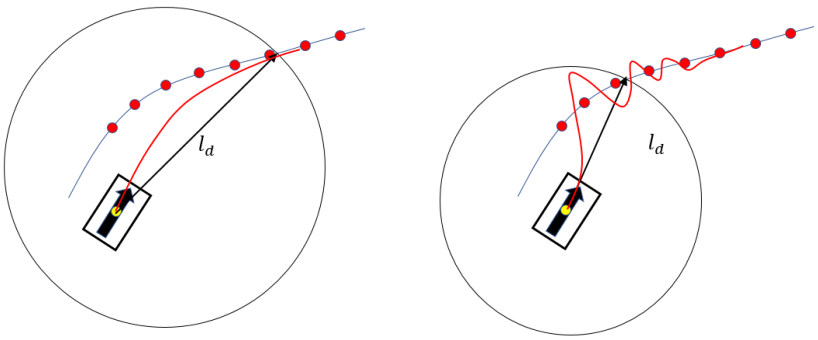
$$\psi_{ref} = \texttt{atan2}((Y_{sel} - Y_{cur}), (X_{sel} - X_{cur})) \tag{3.28}$$

6. Velocity reference: The target velocity v_{targ} for the vehicle is the velocity at the first waypoint ahead of the car. Using the information of the current velocity v_{cur} of the car and the distance to the target

velocity point d_{targ}, the acceleration required to reach the target velocity in time interval T_s is calculated. Thus the velocity reference for current time instance is given by

$$a_{req} = \frac{v_{targ}^2 - v_{cur}^2}{2d_{targ}} \tag{3.29}$$

$$v_{ref} = v_{cur} + a_{req}T_s \tag{3.30}$$



(a) *Large lookahead distance* (b) *Small lookahead distance*

Figure 3.7: *Effect of lookahead distance on path tracking*

3.4 Formulation of Control Allocation problem

The vehicle is made to follow a given path by providing it with reference velocity and reference yaw generated online as discussed in the 'Generating References' section. At every instance of the motion, these generated reference profile which consists of the reference yaw ψ_{ref} and reference longitudinal velocity v_{xref} plays the role of the dynamics request v_d. The corresponding states of the vehicle dynamics model must reach these reference states in order to follow the desired path. This is achieved by integrating the longitudinal acceleration and yaw rate equations in 3.1 and 3.6 and making them equal to the generated reference longitudinal velocity and yaw angle using a preferred integration method. Therefore the path tracking error is formulated as shown in 3.31.

$$\begin{aligned} &\int_{t_k}^{t_{k+1}} f(\hat{X}, u) = v_d \\ &\text{where,} \quad \hat{X} = \begin{bmatrix} \psi \\ v_x \end{bmatrix} \quad v_d = \begin{bmatrix} \psi_{ref} \\ v_{xref} \end{bmatrix} \end{aligned} \tag{3.31}$$

and f is the function that encapsulates the left-hand-side of the continuous non-linear vehicle dynamics equation for $\dot{v}_x$ in 3.1 and yaw rate $\dot{\psi}$ in 3.6 which are related to wheel motor torques and wheel steering angles using Equations 3.7 to 3.10, therefore making it a non-linear function of states v_x, ψ, and the controls u i.e.

the motor torques $\tau = [\tau_{m1} \quad \tau_{m2} \quad \tau_{m3} \quad \tau_{m4}]$ and the front axle steering angle δ_f.

$$u = \begin{bmatrix} \tau_{m1} \\ \tau_{m2} \\ \tau_{m3} \\ \tau_{m4} \\ \delta_f \end{bmatrix} \tag{3.32}$$

3.4.1 Preferred method for discretizing the model dynamics

Discretizing a continuous non-linear function in 3.31 gives a discrete sequence of $\dot{x}$ and ψ states of the vehicle dynamics for formulating the path tracking error as a part of the CA cost function. It is essential for the discretization method to approximate this discrete sequence accurately enough for the accurate operation of the control allocator. A very simple method of discretizing the continuous system of equation is the Explicit Euler method of integration [27]. The Euler method uses the rule described in Equation 3.33, where, Δt is the time step for the generated discrete sequence

$$x_{k+1} = x_k + \Delta t f(x_k, u_k) \qquad k = 0, 1, ...N \tag{3.33}$$

A single Euler step of discretization is illustrated in Figure 3.8 which shows that the method is a linear extrapolation of the model dynamics $f(x_k, u_k)$ in order to build the next discrete state x_{k+1} from the current state x_k. The linear extrapolation represented by the straight dotted line in the figure compromises the curvature of the function throughout the Euler step, hence longer the step, the more curvature is ignored leading to more discrepancies between the discretized state tajectory x_k and the true model trajectory $x(t)$. Therefore the Euler method of discretization can be expensive as it would take more number of computations to reach a given accuracy. Owing to this fact, it is ought to resort to a more accurately approximating integration method.

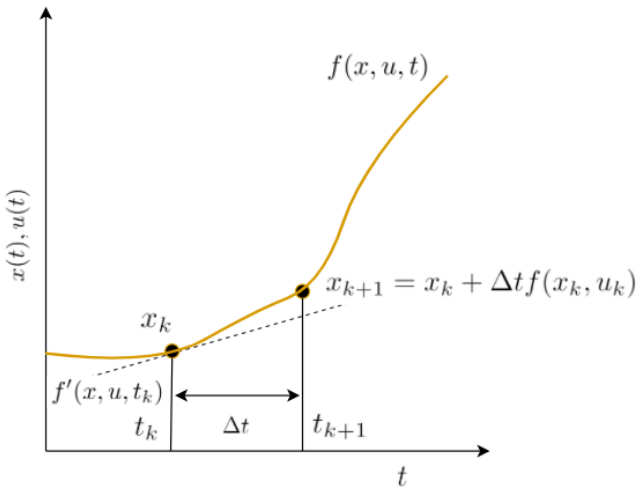


Figure 3.8: *Euler's Method*

The explicit Runge-Kutta (RK) methods of integration are the improved version of the explicit Euler method which extends the process of extrapolation by making it more trustworthy using multiple function differentiation and divided time steps. Instead of understanding the evolution of the true trajectory of the model dynamics based on a single derivative of the function information as done in Euler method, the RK method uses the information of multiple derivatives of the function at different points within the time-step to build the next discrete state. A 4^{th} order RK method also known as $RK4$ method uses the information of 4 slopes of the non-linear model dynamics function (hence the name) as illustrated in the figure 3.9.

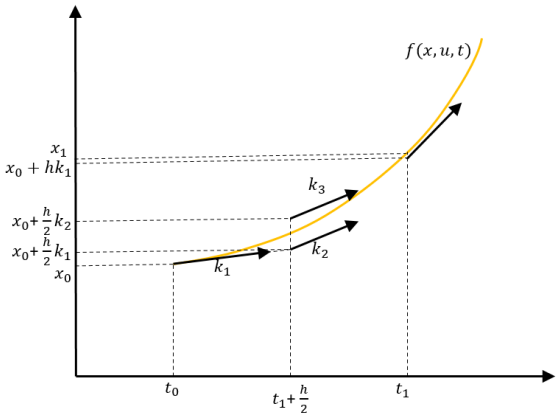


Figure 3.9: *Runge-Kutta* 4^{th} *order Method*

$$\begin{aligned} k_1 &= f(x_k, u_k) \\ k_2 &= f(x_k + \frac{\Delta t}{2} k_1, u_k) \\ k_3 &= f(x_k + \frac{\Delta t}{2} k_2, u_k) \\ k_4 &= f(x_k + \Delta t k_3, u_k) \end{aligned} \tag{3.34}$$

The 4 slopes viz. k_1, k_2, k_3, k_4 are computed as per the Equations in 3.34, where the first slope is found at the time instance k, the second and the third slopes are at time instance $k + \frac{\Delta t}{2}$, of which the slope k_3 is based on the slope k_2. The fourth slope k_4 is computed at time instance $t_k + \Delta t$, ultimately leading to the next discrete step given by the Equation 3.35

$$x_{k+1} = x_k + \Delta t \left(\frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} \right) \tag{3.35}$$

With the average knowledge of 4 slopes around the curvature of the model dynamics $f(x(t), u(t))$, the RK4 method is more reliable method of discretizing the non-linear vehicle dynamics model to derive the discrete velocity states from Equations 3.1 to 3.3 and therefore is the preferred method of discretization in this thesis.

3.4.2 Optimal control problem for energy efficient control allocation

For simplification of notation, consider the path tracking error vector $e = \int_{t_k}^{t_{k+1}} f(\hat{X}, u) - v_d$ and $e \in \mathbb{R}^{2\times 1}$. The solution for the Equation in 3.31 can therefore be solved as a constrained non-linear optimization problem as shown in Equations 3.36, where the cost function constitute the path tracking error in 3.31 subjected to

non-linear equality and inequality constraints. $Q \in \mathbb{R}^{2\times 2}$ is a weighing matrix to provide flexibility in tuning the path tracking performance. The non-linear equality constraint corresponds to the evolution of states in the vehicle dynamics system and the inequality constraints define the operational range of the torques and the steering angle as well as their respective actuation rate.

$$\begin{aligned} \min_{u} \; & J = e^T Q e \\ \text{s.t.} \;\; & \dot{X} = f(X, u) \\ & u_{min} \leq u \leq u_{max} \\ & \dot{u}_{min} \leq \dot{u} \leq \dot{u}_{max} \end{aligned} \tag{3.36}$$

where the state vector X is vector of 6 states and the Equations in 3.1 to 3.6 representing the system dynamics is formulated as a continuous function $\dot{X} = f(X, u)$. u is a control vector of 5 control signals and v_d is the vector of two reference states i.e. v_{xref} and ψ_{ref}. Using the RK4 approximation scheme, the yaw rate $\dot{\psi}$ and the longitudinal acceleration $\dot{v}_x$ are integrated to derive the states ψ and v_x in the system dynamics function to generate the path tracking error.

$$X = \begin{bmatrix} \psi \\ v_x \\ v_y \\ r \\ X_g \\ Y_g \end{bmatrix} \qquad u = \begin{bmatrix} \tau_{m1} \\ \tau_{m2} \\ \tau_{m3} \\ \tau_{m4} \\ \delta_f \end{bmatrix} \qquad v_d = \begin{bmatrix} \psi_{ref} \\ v_{xref} \end{bmatrix} \tag{3.37}$$

A control solution obtained by solving CA optimization problem in 3.36 should allocate the wheel motor torques and the front axle wheel angle such that the BEV follows the desired path. However, the allocation of the controls needs to be done in an energy efficient manner in order to complete the objective of this thesis i.e. the control solution chosen by solving CA optimization problem in 3.36 should also have minimal electric and tire losses. An event of over-actuation in a BEV takes place when the vehicle has to develop sufficient yaw moment in order to follow the curvature of the path, which can be achieved either by allocating torques for torque vectoring or by steering the front wheels, both have a consequence on the power consumption of BEV and the optimizer solving the optimization problem in 3.36 must be made aware of this consequence. This is done by introducing the power loss models to the cost function by modifying the mixed optimization method for control allocation in 2.12, where the error in control supply is replaced by the power loss model. The power loss model, as discussed in the previous section, are a function of motor torques and wheel angles that governs the energy losses arising in terms of electrical losses, lateral and longitudinal slip losses, and rolling resistance losses and can be expressed as

$$P_{losses}(u) = P_{m,loss}(\tau_m) + P_{tire,loss}(\tau_m, \delta_f) \tag{3.38}$$

and the resulting constrained non-linear optimal control problem for control allocation is as shown in the

Equation 3.39.

$$\begin{aligned} \min_{u} \mathbf{J}(\mathbf{X}, \mathbf{u}) &= e^T Q e + P_{losses}(\mathbf{u}) R \\ \text{s.t.} \quad \dot{\mathbf{X}} &= f(\mathbf{X}, \mathbf{u}) \\ \mathbf{u}_{min} &\leq \mathbf{u} \leq \mathbf{u}_{max} \\ \dot{\mathbf{u}}_{min} &\leq \dot{\mathbf{u}} \leq \dot{\mathbf{u}}_{max} \end{aligned} \tag{3.39}$$

Assuming that the control signals u is constant over a small constant sampling interval, the exact representation of the above system in 3.39 can be represented in its discretized format by sampling the states of the model dynamics at every time instant $k \geq 0$ for a small sampling time interval of Δt using the RK4 method of integration. The continuous constrained OCP in 3.39 now takes the discretized form as shown in 3.40, where k represents the current time instance, x_0, u_0 are the initial states and controls of the system. The path tracking error e_k now becomes the difference between the resulting vehicle states for the optimization control solution and the desired state of the system at time instance $k+1$ for the optimized control solution u at time instance k.

$$\begin{aligned} \min_{u} \mathbf{J}_k(\mathbf{X}_0, \mathbf{u}_k) &= \mathbf{e}_k^T Q \mathbf{e}_k + P_{losses}(\mathbf{u}_k) R \\ \text{s.t.} \quad \mathbf{e}_k &= \hat{\mathbf{X}}_u(k+1) - v_d(k+1) \\ \mathbf{X}(k+1) &= f(\mathbf{X}_u(k), \mathbf{u}(k)) \\ \mathbf{X}_u(0) &= \mathbf{X}_0 \\ \mathbf{u}_{min} &\leq \mathbf{u}(k) \leq \mathbf{u}_{max} \\ \dot{\mathbf{u}}_{min} &\leq \mathbf{u}(k) - \mathbf{u}(0) \leq \dot{\mathbf{u}}_{max} \end{aligned} \tag{3.40}$$

The CA problem formulated in 3.40 is the discretized non-linear optimal control problem (OCP) to be solved online and is the core objective of this thesis. Due to its non-linearity it an analytical method cannot be used since an analytical solution usually exists for linear-optimal control problem. Therefore the OCP in 3.40 is solved using numerical methods which is an iterative process, where an OCP solving algorithm is deployed iteratively until a local or global convergence is achieved.

3.5 Two different approaches for implementing Control Allocation in BEV

3.5.1 Instantaneous Control Allocation

When the OCP in 3.40 is being solved online at every instance k, the CA is performed such that the allocated energy-efficient actuation effort in the powertrain of the BEV is able to satisfy the online generated vehicle velocity and orientation request only for 'that' instance of time. This instantaneous method of optimization based CA is referred to as 'Instantaneous Optimization method' in this thesis. The control architecture for Instantaneous Optimization based CA is represented in the Figure 3.10.

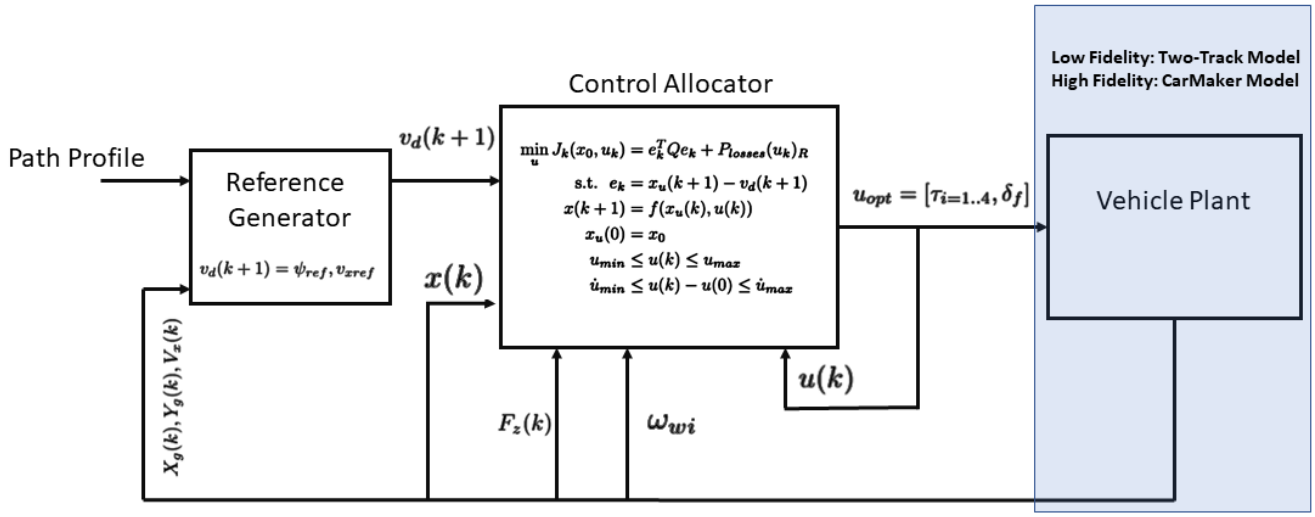


Figure 3.10: *Control Architecture for Instantaneous CA*

The CA using instantaneous optimization method allocates the torques and front axle wheel angle for the requests being fed to it at every instant in a piecewise manner. In other words, the BEV is being driven by looking at only one waypoint ahead of it at every instant. Therefore it is a point of interest to see how the control allocation takes place by not just looking at one waypoint ahead of the BEV but instead looking at the following waypoints as well and plan the controls to reach them after the first waypoint is reached. This is usually how a driver would drive his car.

Loosely speaking, the allocated control signals are not planned like how a driver would plan, who is aware of how the road ahead of the vehicle looks and can estimate what control signal at this instance of time will effectively manoeuvre the vehicle to follow a small segment of the path ahead in future. A control engineer would immediately recognize this method of controlling a system in a predictive manner using the technique of Model Predictive Control and therefore is also the motivation behind investigating the consequences of implementing the second approach for implementing the CA using a Non-linear Model Predictive Control (NMPC) system.

The following subsection will cover the working principle of MPC and how the optimization based CA in 3.39 is reformulated accordingly.

3.5.2 Non-linear Model Predictive Control based Control Allocation

MPC by itself is an optimal control problem which involves a cost to be optimized and an optimization algorithm to optimize the cost. Given the current time instance, and the state of the system, the MPC controller looks at a trajectory of the reference states to be followed by the system for N number of time steps in future known as the 'prediction horizon length'. It then proceeds to find a sequence of controls which when applied to the system takes the next state of the system to the reference state at their respective time step in the prediction horizon. Once a sequence of such a control solution is found, the first element of the control sequence is applied to the system at the current time instance and the process reiterates. The reference tracking offset and the actuation effort required to track the reference are quantified as a cost function which the NMPC optimizer tries to minimize during every iteration. When the cost function is non-linear, MPC is termed as a Non-linear

MPC (NMPC).

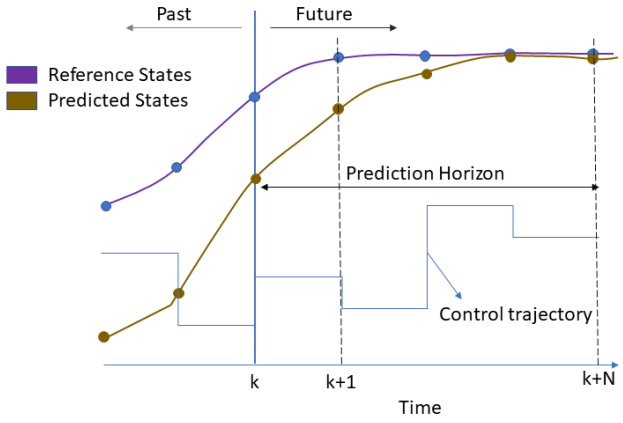


Figure 3.11: *Illustration of receding horizon control in MPC*

The degree of offset between the predicted states and the reference states depends on the objective cost function provided to the solver in NMPC, especially the weights in the objective cost function. A general formulation of an NMPC optimization problem can be written as shown in 3.41

$$\begin{aligned} \min_{x(k),u(k)} \quad & J_N(x(k),u(k)) \\ \text{s.t.} \quad & x(k) \in \mathcal{X}, \quad u(k) \in \mathcal{U} \\ & x(k+1) = f(x(k),u(k)) \end{aligned} \tag{3.41}$$

where $J(x(k),u(k))$ is the sum of running stage costs over the length of the prediction horizon. A stage cost is an $l2$ norm of the difference between the predicted state and the reference state at a given instance of time in the prediction horizon. A last stage cost, as the name suggests, is the difference between the predicted state and the reference state at the end point of the prediction horizon i.e $t = N$, therefore the general cost function of the MPC is as follows

$$J_N(x(k),u(k)) = ||x(N) - x_r(N)||_P^2 + \sum_{k=1}^{N} ||x(t+k) - x_r(t+k)||_Q^2 + ||u(t+k) - u_r(t+k)||_R^2 \tag{3.42}$$

where t is the current time step, x_r and u_r are the reference state vector and reference control vector, Q and R are the diagonal weight matrices for state tracking and penalizing the actuation effort.

Based on the above working principle and the optimization problem formulation for NMPC, a CA optimization

problem in 3.40 can be reformulated to fit into the frame of NMPC as shown in 3.43

$$\begin{aligned} \min_{u} \mathbf{J}(\mathbf{X}_0, \mathbf{u}_k) = & \sum_{k=1}^{N} ||\hat{\mathbf{X}}(t+k) - v_d(t+k)||_Q^2 + ||P_{losses(\mathbf{u}_k)}||_R^2 \\ \text{s.t.} \quad & \mathbf{X}(k+1) = f(\mathbf{X}_u(k), \mathbf{u}(k)) \\ & \mathbf{X}_u(0) = \mathbf{X}_0 \\ & \mathbf{u}(k) \in \mathcal{U}, \quad \forall k \in [0, N-1] \\ & \mathbf{X}_u(k) \in \mathcal{X}, \quad \forall k \in [0, N] \end{aligned} \tag{3.43}$$

where the first l_2 norm term indicates the error in the path tracking performance of the BEV for the given reference trajectory of longitudinal velocity and yaw angle throughout the length of the prediction horizon, and the second l_2 norm is the scalar value of losses appearing for the computed control trajectory. A control architecture for NMPC based CA is illustrated in Figure 3.12 which is seemingly similar to the control architecture of Instantaneous CA method except for the references used by the control allocator i.e. a sequence of references pertaining to the length of the prediction horizon.

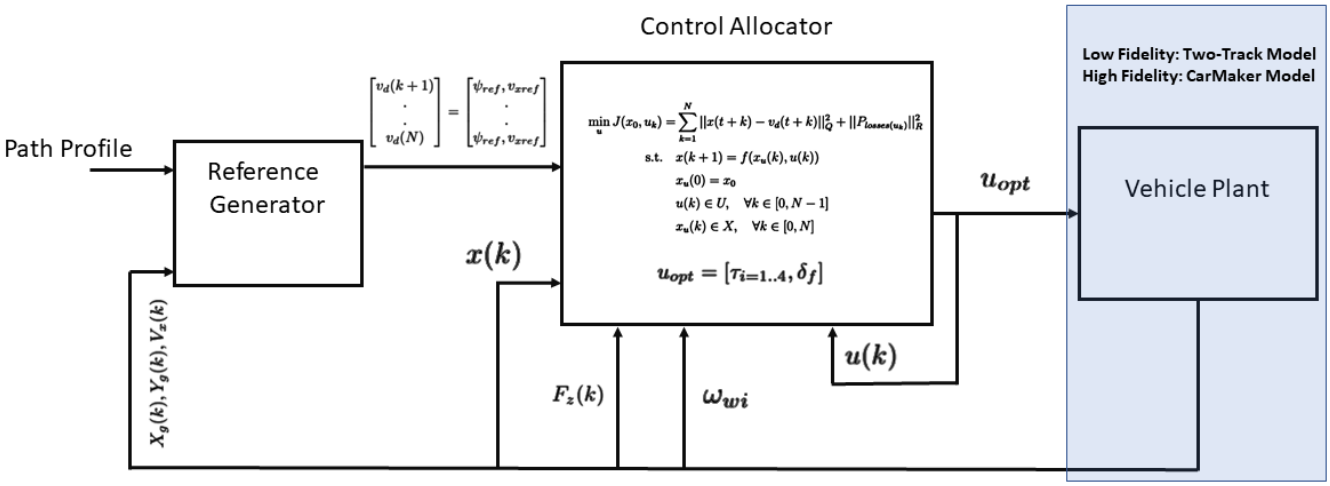


Figure 3.12: *Control Architecture for NMPC based CA*

3.6 Solving the Optimal Control Problem using Numerical Method

As the optimization based CA in this thesis branches out into two different OCP, i.e. the OCP in 3.40 pertaining to the Instantaneous CA problem and the OCP in 3.43 for NMPC based CA, both needs to enter the realm of NLP to be solved. NLP solves OCP by compressing the size of the problem which helps in faster computation 3.13. Although the conversion of these two OCPs into their respective NLP format are treated differently owing to the difference in the problem structure and decision variable size between the two.

3.6.1 Non-Linear Programming

The formulated discretized non-linear OCP in 3.40 and 3.43 are posed as an NLP problem using a 'transcription method' where the decision variables are considered as a set of real numbers z, the cost functions in both ,3.40 and 3.43 are transformed to a real number as function of decision variable z as $\Phi(z)$. The equality and inequality constraints are also mapped as a function of decision variables as $g(z)$ and $h(z)$ respectively. The dual problem of OCP in 3.40 and 3.43, both as an NLP takes the form as shown in 3.44, respectively, which helps in compressing the size of the problem.

$$\begin{aligned} &\min_{z} \Phi(z) \\ \text{s.t.} \quad &g(z) = 0 \\ &h(z) \leq 0 \end{aligned} \tag{3.44}$$

3.6.2 Transcription Methods for Instantaneous and Predictive Optimization technique

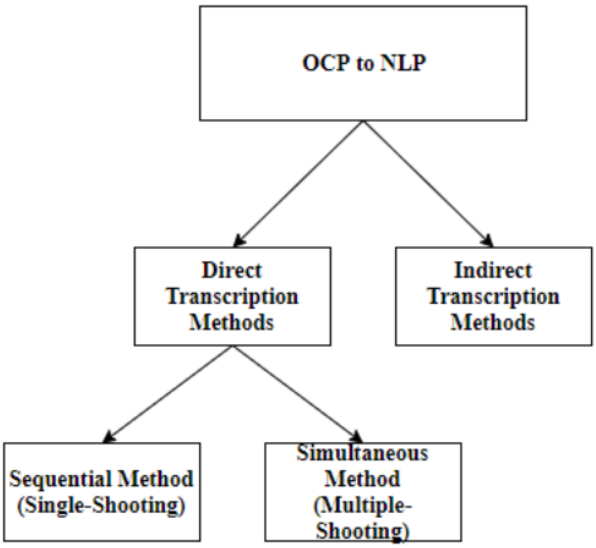


Figure 3.13: *Classification of the transcription method*

There are many efficient transcription methods developed for converting the discretized OCP in 3.40 to its equivalent NLP problem in 3.44 such as control parametrization [28] [29], collocation methods [28] [29], model reality method [30] [31], single-shooting and multiple shooting methods [32] [15], etc. These algorithms can be generally classified into two distinct classes viz. Direct method and Indirect method. In Direct method, the optimal control problem is converted into non-linear programming problem (NLP) where the state and control variables are approximated by a piece wise constant parametrization and then solved for optimality. Whereas, in Indirect method the Hamiltonion function should be constructed which results in a boundary-value problem which gives the trajectory of discretized states and controls such that the trajectory leads to the optimal condition. This boundary-value problem is again in the form of an NLP, which is based on a very accurate guess of the starting point within the feasibility region. Loosely speaking, the condition for optimality

of the OCP is figured out first and then the OCP is discretized and solved [15].

A classification of these transcription methods is illustrated in the Figure 3.13. The preferred transcription method in this thesis is the Direct transcription method since the Indirect transcription method relies on a very accurate initial guess to begin the iterations and are hard to solve in practice [20]. The Direct Methods can further be classified into sequential and simultaneous methods.

In a sequential method, only the controls are discretized using piecewise parametrization and enter the NLP as the decision variables. During every iteration of finding the best solution of controls, the states of the model and the cost function value are checked if they have hit the desired target and is repeated until the best solution satisfying all the optimization demands is found. Single-shooting method works according to this principle. In simultaneous method, both the controls and the states of the model are discretized and enter the NLP as the decision variables. The NLP is then solved by checking if the resulting states reach the target states for every control solution. Multiple-shooting method works according to this principle. Both these methods differ when it comes to their respective computation time and the propagation of non-linearity of the function with every iteration.

In Single shooting method, applied on a non linear OCP, if the stages involved in computing an optimal control trajectory are more than one, i.e. a scenario of multiple stage OCP such as NMPC, there are no simultaneous check on the non linear discrepancy that are developed between the predicted state and the desired future state for the estimated control trajectory. This results in increased number of iterations to achieve better convergence. Owing to this fact, the sequential method is beneficial for Instantaneous optimization technique which involves a single stage mapping between the current control $u(k)$ to its resulting state $x(k+1)$ which has reduced degree of non-linearity propagation. Whereas, for predictive optimization technique where NMPC involves a multiple stage mapping between the control trajectory $u(k)$ to $u(k+N)$ and its respective predicted state trajectory $x(k+1)$ to $x(N+1)$, the non-linearity propagates relatively at higher degree compared to the single stage mapping in Instantaneous optimization technique, and sequential methods become logically expensive in terms of computation time. Therefore, for faster computation using NMPC, the target check needs to be done simultaneously which is achieved by implementing the Multiple shooting transcription method. Although, including both the state and control variables as the decision variables z in the NLP problem increases the size of the problem and gives the problem a sparser structure, it seems to be a good trade-off for avoiding non-linearities in the solution and faster computation.

3.7 NLP solvers - Sequential Quadratic Programming and Interior Point Methods

The NLP in 3.44 where $\Phi(z)$, $g(z)$ and $h(z)$ are all real valued functions, it is essentially important to search for the optimal solution z^* within the feasible region defined by the equality and inequality constraints, $g(z)$ and $h(z)$ respectively. The method of Lagrange multipliers is the go-to approach for the constrained optimization problem, where the optima of the function is computed without breaking the rules of the defined constraints. The Lagrange function for the NLP problem in 3.44 is defined as where λ and μ are known as the Lagrange multipliers.

$$\mathcal{L}(z, \lambda, \mu) = f(z) + \lambda g(z) + \mu h(z) \tag{3.45}$$

In the event of an NLP problem without any sorts of inequality constraints $h(z)$, the ordeal of finding the optimal solution z^* becomes rather much simpler as it boils down to the rule of finding a point z at which the gradient of the cost function $\nabla f(z)$ vector should be parallel to the gradient of the equality function $\nabla g(z)$ vector as shown in Equation 3.46, which is equivalent to finding the point at which the gradient of the Lagrange function in 3.45 (without the inequality constraint function) vanishes where the negative sign indicates the direction of feasibility.

$$\nabla f(z) = -\lambda \nabla g(z) \tag{3.46}$$

The method takes a deviation when the inequality constraints $h(z)$ enters the problem during the event of which, the so called 'Karush-Kuhn-Tucker' (KKT) conditions provide necessary conditions for a local optimum (in the general case), as well as necessary and sufficient conditions for a global optimum (in the case of convex optimization case). The conditions dictated by KKT is as follows : ¨For any locally optimum point z^* within the feasible region, there exists multipliers λ^* and μ^* such that the following equations hold:

$$\begin{aligned} \nabla_z \mathcal{L}(z^*, \lambda^*, \mu^*) &= 0 \\ g(z^*) &= 0 \\ h(z^*) \leq 0 \quad \perp \quad \mu^* &\geq 0 \end{aligned} \tag{3.47}$$

The third condition in 3.47 involves a complementary condition represented by $\perp$ symbol between the two vectored valued inequalities which is known as the complemetary slackness defined by the following Equation 3.48.

$$h(z^*)\mu^* = 0 \tag{3.48}$$

All Newton type optimization methods try to find a point satisfying these above conditions 3.47 by successive linearization of the problem functions. But the family of Newton optimization methods begin to branch out when it comes to solving the third condition in 3.47 which is non-smooth in nature and this where the dawn of the difference between two big families of Newton type optimization method viz. Sequential Quadratic Programming (SQP) methods and Interior Point (IP) method begins [17]. In other words, the two methods differ in the way they treat the inequality constraint function $h(z)$.

3.7.1 Sequential Quadratic Programming (SQP)

SQP was the first variant of Newton optimization method to solve the KKT conditions by linearizing all the non-linear functions appearing in 3.47. When the the third complementarity condition in 3.47 is linearized, it turns out that the resulting linear system can be interpreted as the KKT conditions of a quadratic program (QP) as shown in 3.49

$$\begin{aligned} \min_z f^k_{QP}(z) = \nabla f(z^k)^T z + \frac{1}{2}(z - z^k)^T \nabla^2_z \mathcal{L}(z^k, \lambda^k, \mu^k)(z - z^k) \\ \text{s.t.} \quad g(z^k) + \nabla g(z^k)^T (z - z^k) = 0 \\ h(z^k) + \nabla h(z^k)^T (z - z^k) \leq 0 \end{aligned} \tag{3.49}$$

The process of computing the Hessian matrix $\nabla^2_z \mathcal{L}(z^k, \lambda^k, \mu^k)$ is computationally expensive especially when the problem size is too large and for very non-linear inequality constraint functions.

Therefore for larger problem size and very non-linear inequality constraints the QP in 3.49 resorts to using an approximated hessian matrix instead of computing the exact hessian matrix. This general approach to address the non-linear optimization problem is known as SQP method and NLP solvers like qpOASES work on this principle.

3.7.2 Interior Point method (IP)

The IP method uses an alternative approach for tackling the last complementarity condition in 3.47 by introducing a smooth non-linear condition for $h(z^*)\mu^* = 0$ with $h(z^*)\mu^* = \tau$, (not to be mistaken with the torque vector τ) as shown in 3.50.

$$\begin{aligned} \nabla_z \mathcal{L}(z^*, \lambda^*, \mu^*) &= 0 \\ g(z^*) &= 0 \\ h(z^*)\mu^* &= \tau \end{aligned} \tag{3.50}$$

The above system is then solved using the Newton's method. The obtained solutions is not a solution for the original problem in 3.44 but to the problem in 3.51

$$\begin{aligned} &\min_z f(z) - \tau log(-h(z)) \\ \text{s.t.} \quad &g(z) = 0 \end{aligned} \tag{3.51}$$

Therefore the obtained solution is not exactly on the boundary defined by the in-equality constraints but somewhere inside the feasible region close to the boundary known as 'Interior Point'. Once a solution for a given τ is found, the parameter τ is reduced iteratively by a constant factor without jeopardizing the convergence of Newton's method. After only a limited amout of iterations, the IP method achieves a quite accurate solution of the original NLP problem in 3.44 [17]. One of the widely used nonlinear Interior Point methods based NLP solver is the open source IPOPT solver.

Owing to the hessian approximation approach in SQP method, the SQP based solvers are known as 'Medium-scale algorithm' solvers which internally create full matrices worth memory and use dense linear algebra. If a problem is sufficiently large, the jacobian and hessian matrices take up significant amount of memory and the dense linear algebra takes longer-time to execute. On contrast to SQP solvers, the IP method based solvers are known as 'Large-Scale algorithm' solvers which can handle large sparse matrices.

3.8 Implementation

The implementation phase begins with the development of the proposed framework in a Simulink environment on a low fidelity model. A small segment of the reference path to be followed is given as dynamics request and the corresponding path tracking performance, the torque and steering angle allocation for both instantaneous and predictive based CA methods will be analyzed. For solving the OCP in both the methods, MATLAB's NLP solver 'fmincon' and 'CasADi', an open source tool for dynamic nonlinear optimization, will be used. Both IP and SQP solvers will be implemented to look at their impact on the CA.

However, not much of analysis can be done on power consumption, as the low fidelity model does not cover all the parameters required for calculating the power consumption and losses in a BEV.

Once the proposed CA methods are analyzed using the low fidelity model in terms of their path tracking performance and allocation of controls, the approach will then be applied on a high fidelity vehicle model in IPG CarMaker. The vehicle plant model in IPG CarMaker has an abundance of data pertaining to most of the parameters of the vehicle. This gives an opportunity for analyzing different losses as mentioned in the methodology and the overall energy consumption of the BEV.

Apart from comparing the results between the two methods, the analysis also involves comparing the energy efficiency for path tracking using NMPC based control allocation with path tracking using equal torque distribution (ETD) on all 4 wheels as a benchmark. In the over-actuated control space of BEV, of the multiple ways of allocating wheel motor torques, allocating equal torques is also one of them. This means no scenario of different torques on either sides of the axle, hence no torque vectoring. Torques in ETD strictly contribute to the longitudinal dynamics and the lateral dynamics is strictly achieved by front axles steering angle. Therefore, it is of interest to see how energy efficient is the path tracking using NMPC based CA compared to path tracking with ETD. This is achieved by using the velocity profile obtained by running NMPC based control allocation simulations for GCC as the reference velocity profile v_{xref}. The longitudinal velocity state v_x of the vehicle plant in IPG CarMaker is then made to follow the velocity profile obtained from the simulations of NMPC based control allocation which gives the required longitudinal acceleration $\dot{v}_x$. In order to catch up with the reference velocity, velocity tracking difference is tuned with a gain constant G. The required longitudinal force is then distributed equally to all 4 wheels and the respective torque at each wheel is mapped to the calculated wheel's longitudinal force as shown in Equations 3.52, where r_w is the wheel radius and n_t is the transmission ratio.

$$\begin{aligned} a_x &= (v_{xref} - v_x)G \\ f_{xi} &= \frac{ma_x}{4} \\ \tau &= \frac{f_{xi} r_w}{n_t} \end{aligned} \tag{3.52}$$

While this takes care of the longitudinal dynamics control, the lateral dynamics is controlled by using in built IPG CarMaker's driver model. The control architecture of ETD is illustrated in the Figure 3.14.

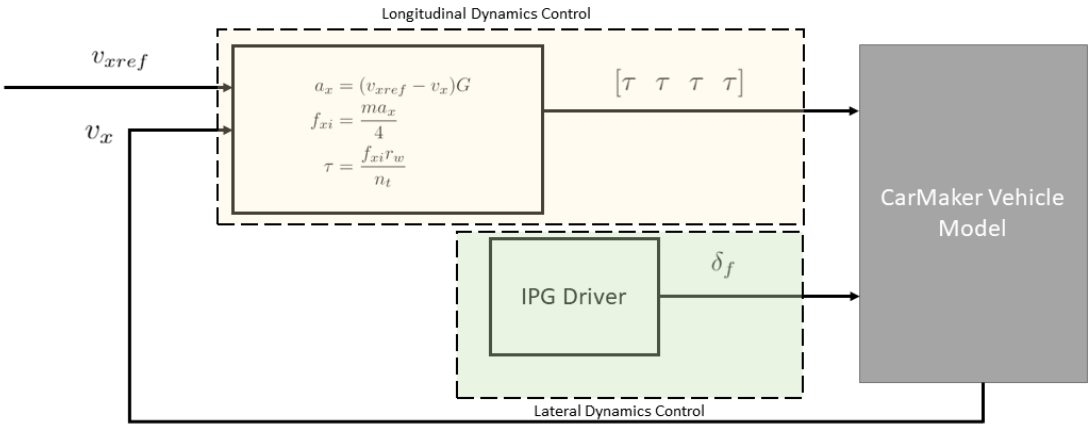


Figure 3.14: *Control architecture for equal torque distribution driving scenario*

The following part of the report will cover the results obtained from low fidelity model during the development of the two proposed methods of CA in this work, followed by the results obtained by implementing the two methods of CA on a high fidelity vehicle plant model.

4 Results

This chapter covers the results obtained after developing and implementing the the two methods of CA on low-fidelity vehicle model followed by the results obtained on implementing the proposed CA methods on a high fidelity vehicle model. The path profile required for path tracking is constructed using the data of Gothenburg City Cycle (GCC).

Gothenburg City Cycle (GCC) is a 66.25 Kms long drive cycle which involves the everyday driving scenario on the roads of Gothenburg city, including residential areas as well as the highways. Therefore it is a moderate driving scenario with a maximum velocity reaching 22 m/s. This drive cycle was developed at Volvo Cars [7] for the analysis of everyday city driving behaviour's impact on different characteristics of the vehicle. The GCC profile comes with the global X-Y coordinates. A driver model in CarMaker was made to follow this predefined position profile of GCC with the lateral and longitudinal accelerations limited as following

$$\begin{aligned} a_x &\in [-4m/s^2, 3m/s^2] \\ a_y &\in [-4m/s^2, 4m/s^2] \end{aligned} \tag{4.1}$$

The logged velocity data obtained on running the simulations using the CarMaker's driver model was used as the velocity profile of GCC required for path tracking.

4.1 Control Allocation for path tracking with low-fidelity plant model

The development of the proposed framework for both instantaneous and NMPC based CA methods in this work was initially implemented and tested by running simulations using a two-track vehicle model as the plant model. The feedback control architectures as illustrated in 3.10 and 3.12 were developed in MATLAB Simulink environment. Three different approaches of implementing CA for the requested dynamics were developed and tested on this low fidelity model :

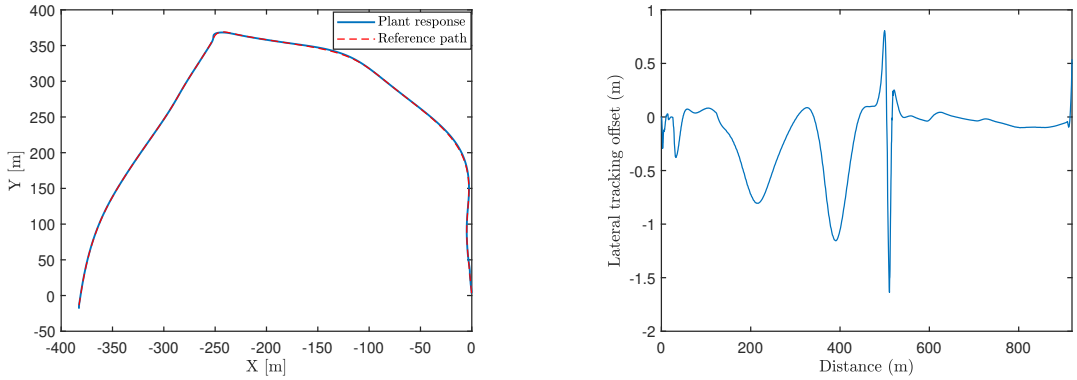
- Instantaneous CA - using MATLAB's fmincon toolbox (SQP solver)
- Instantaneous CA - using CasADi (IPOPT solver)
- NMPC based CA - using CasADi (IPOPT solver)

The intention behind testing the developed CA methods for path tracking on a low-fidelity model was to know which of these approaches can well handle the path tracking requests with sensible control allocation. The following part of this section will cover the simulation results for a simulation time of 100 seconds for the above mentioned three different approaches of CA for tracking the beginning sector of GCC. The results highlight the individual CA approach's potential in terms of path tracking performance, allocation of torques and wheel angle.

4.1.1 Instantaneous CA - fmincon (SQP solver)

Figure 4.1 shows the path tracking performance in terms of the path traced based on the given reference path and the corresponding lateral deviation during path tracking using SQP solver for instantaneous CA method.

For the executed simulation for 100 seconds, the vehicle covered a distance of 919.36 m of the given reference path. The lateral path tracking offset for this travel was within 1.5 m.



(a) *Path followed by the low fidelity vehicle model for the given dynamics request*

(b) *Lateral deviation from reference path*

Figure 4.1: *Path tracking and lateral deviation for fmincon's SQP solver*

In case of fmincon's SQP, the reference states for tracking were not only the longitudinal velocity v_x and yaw angle ψ, but yaw rate r as a reference state had to be additionally added to the reference state vector v_d. The reason for this additional state as reference along with v_x and ψ will be discussed in detail in the Discussion chapter. The instantaneous method using the fmincon's SQP solver had poor tracking of the reference velocity state although the reference yaw angle and yaw rate were being tracked well as seen in 4.2.

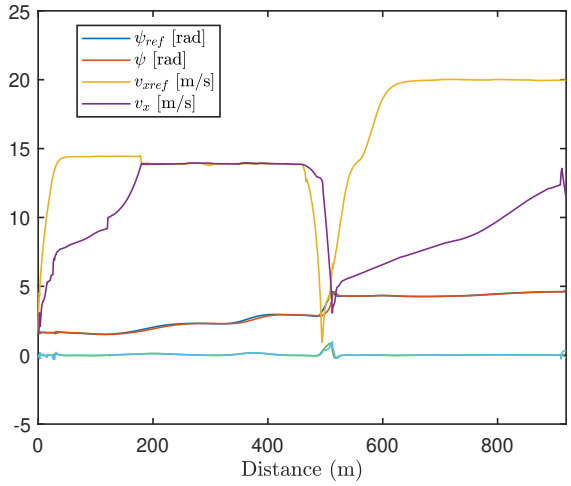
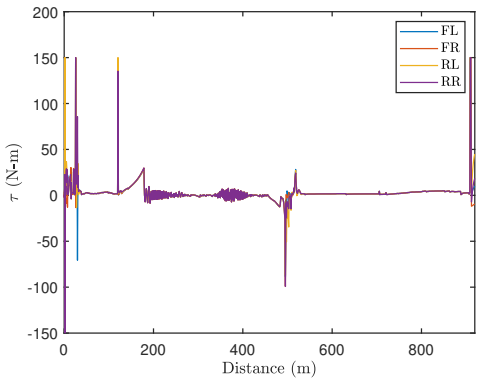
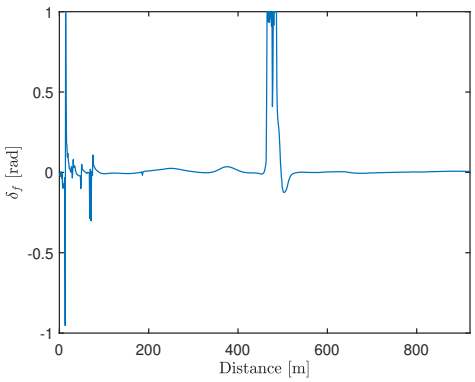


Figure 4.2: *Reference states tracking for fmincon using SQP solver*

The corresponding torques and steering angle allocated by SQP solver in fmincon are as shown in Figure 4.3



(a) *Torque allocation using SQP solver of fmincon*

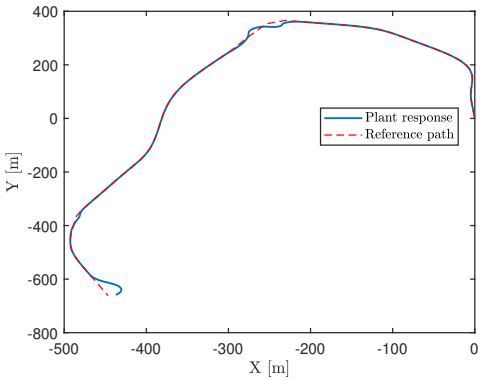


(b) *Steering angle allocation using SQP solver of fmincon*

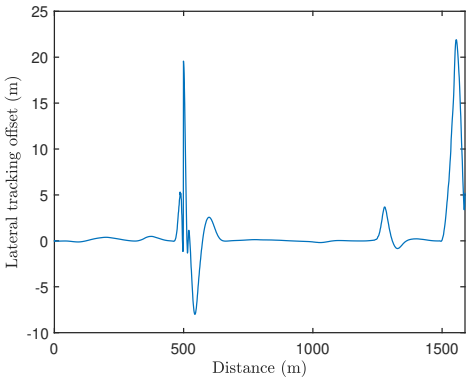
Figure 4.3: *Controls allocated by fmincon's SQP solver*

4.1.2 Instantaneous CA - CasADi (IPOPT solver)

For the given set of reference velocity and the yaw required to follow the similar path as in instantaneous method using fmincon's SQP solver, the vehicle travelled for 1588.6 m for 100 seconds of simulation time. The path tracking offset for this travel was minimal except at some points of the travel where the lateral deviation goes to around 20 m as shown in Figures 4.4. For instance, the left turning at the coordinate [-245.2, 381] in 4.4 (a) , the vehicle can be seen to go off the desired path but gets back to the path eventually.



(a) *Path followed by the low-fidelity vehicle model for the given dynamics request*



(b) *Lateral offset from reference path*

Figure 4.4: *Path tracking performance for Instantaneous method using CasADi IPOPT solver*

The reference state tracking by IPOPT solver using CasADi is as shown in Figure 4.5. The longitudinal velocity references tracked better than the fmincon's SQP solver, causing the vehicle to travel longer distance. Although the tracking of reference yaw angle gets disturbed at instances wherever the lateral deviation from the reference track is high. It can be observed that at such instances of deviation from the reference track such as the 20 m lateral deviation at 500 m of the travel, the vehicle turned left and the path tracking algorithm addressed this deviation and gives the yaw angle reference to make the vehicle turn right and get back on desired path. Hence, this shows that the path tracking algorithm is doing what is expected of it.

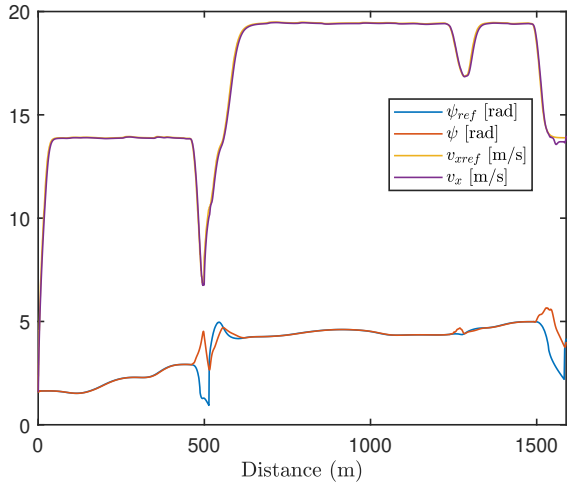
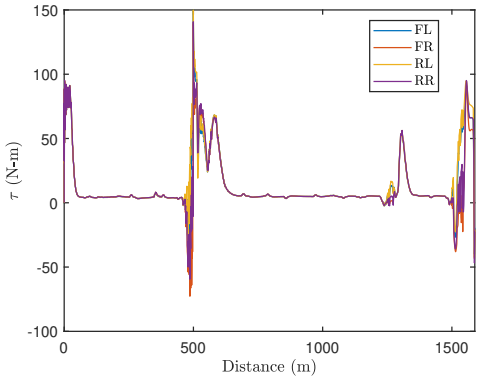
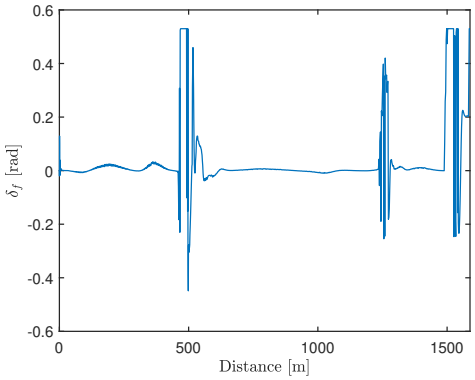


Figure 4.5: *Dynamics request tracking performance for IPOPT solver in CasADi*

The torques and steering angles allocated in order to track the requested dynamics are as shown in the Figure 4.6. The torques allocated using IPOPT solver for instantaneous method appear to be higher compared to the ones allocated using SQP solver. This is reasonable since the vehicle in this case is made to follow the velocity profile which requires higher torques than the ones allocated using SQP solver. Although steering angles are noisy whenever the later deviation goes higher than 1 m.



(a) *Torque allocation using IPOPT solver of CasADi*

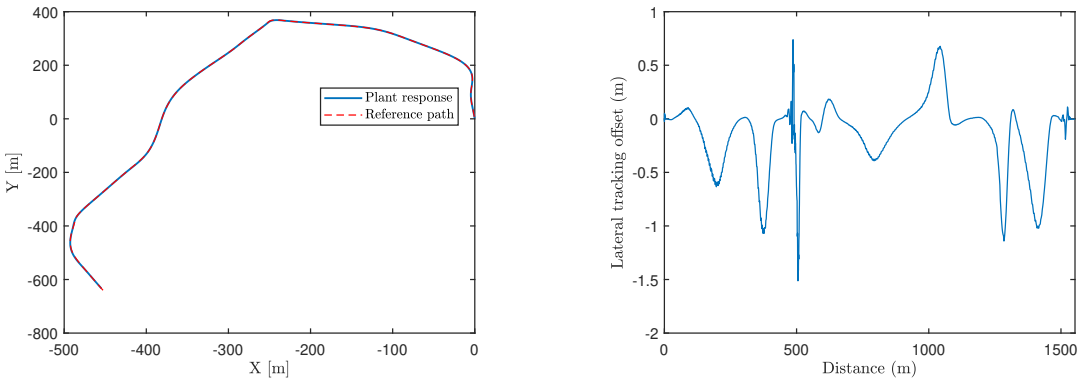


(b) *Steering angle allocation using IPOPT solver of CasADi*

Figure 4.6: *Controls allocated by CasADi's IPOPT solver*

4.1.3 NMPC based CA - using CasADi (IPOPT solver)

The NMPC based CA using CasADi's IPOPT solver had the vehicle travel the reference path for 1553.1 m for the same simulation time of 100 seconds with a lateral deviation within a range of 1.5 m as illustrated in Figures 4.7.



(a) *Path followed by the low-fidelity vehicle model for the given dynamics request*

(b) *Lateral offset from reference path*

Figure 4.7: *Controls allocated by CasADi's IPOPT solver*

The longitudinal velocity and the yaw requests generated to follow the desired path had very minimal discrepancy as seen in Figure 4.8, without any deviation fro the reference yaw angle at the turning as seen in reference state tracking for instantaneous CA method using IPOPT solver.

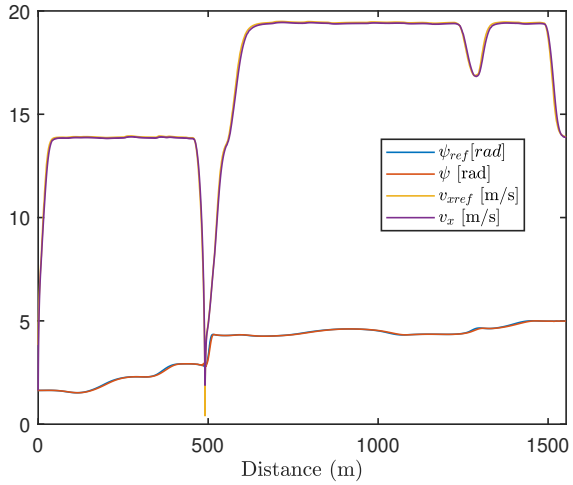
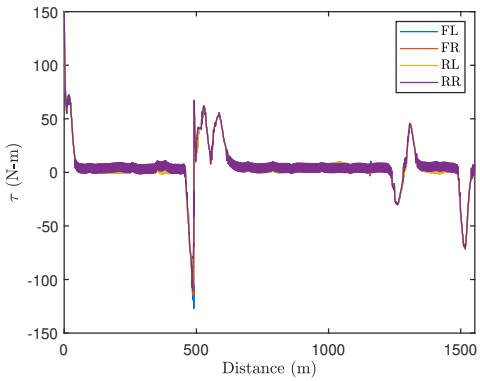
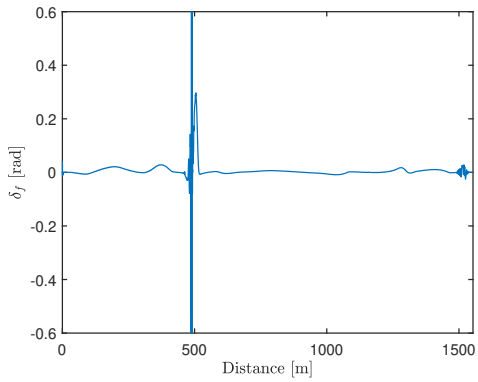


Figure 4.8: *Dynamics request tracking performance for NMPC based CA using IPOPT solver in CasADi*

To achieve the desired behaviour of the car to follow the reference path and the dynamics request, the NMPC based CA allocated torques and front axle angles as see in Figures 4.9. The steering angle at the turning at around 500 m of the travel becomes noisy, this is again at the point where the lateral deviation is maximum as shown in the Figure 4.8.



(a) *Torque allocation using IPOPT solver of CasADi for NMPC based CA*



(b) *Steering angle allocation using IPOPT solver of CasADi for NMPC based CA*

Figure 4.9: *Controls allocated by CasADi's IPOPT solver*

4.2 Simulation results for high-fidelity plant model

The two control allocation methods developed in the previous chapters are implemented in a high-fidelity simulation environment, IPG CarMaker 7.0.2. The vehicle model used for the simulation is a Volvo V60 with four identical in-wheel motors and front wheel steering. The dimensions and specifications of the vehicle are tabulated in table (4.1). The rest of the part of this subsection will reflect on the results obtained for implementing the Instantaneous CA and NMPC based CA method for path tracking of entire GCC. To avoid the verbosity in terms of graphs for the results on the entire GCC, the results shown in this subsection corresponds to a particular 900 m long sector of GCC.

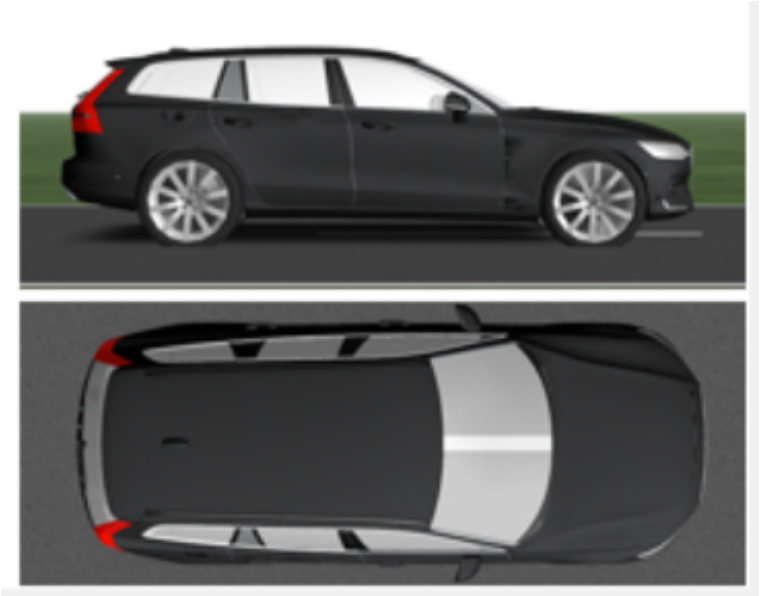


Figure 4.10: *Volvo V60*

Parameter	*Value*
m	1988 kg
I_{zz}	4300 kgm^2
l_f	1.258 m
l_r	1.615 m
w	1.6 m
n_t	10
C	1.38
E	-0.99

Table 4.1: *Vehicle parameters*

4.2.1 Instantaneous Optimization - IPOPT (CasAdi)

For instantaneous optimization method, the sample time for the IPOPT solver was set to be $0.1s$. The objective function for the optimization includes the path tracking offset and power loss term as show in equation 4.2. The steer rate and torque rates were constrained with an upper bound of 2.5 rad/s and 1500 Nm/s. The vehicle's behaviour in the simulation was observed by tuning the cost function using different weights and appropriate

weights to each term in the objective function such that the vehicle follows the reference trajectory at the requested longitudinal velocity while minimizing the power losses. It was observed that the mean computation time for the solver was 0.03 s.

$$J(X_k, u_k) = ||\hat{X}(t+k) - v_d(t+k)||^2_Q + ||P_{losses(u_k)}||^2_R + ||u(k+1) - u(k) - \epsilon||_{Rrate} \tag{4.2}$$

The power loss term in Equation 4.2 is as shown in 4.3 where each loss model is tuned with its own weight.

$$||P_{losses(u_k)}||^2_R = ||P_{m,loss}(u_k)||^2_{R_{el}} + ||P_{sx}(u_k)||^2_{R_{sx}} + ||P_{sy}(u_k)||^2_{R_{sy}} + ||P_{rr}(u_k)||^2_{R_{rr}} \tag{4.3}$$

The actuation rate term in the cost function is as shown in Equation 4.4.

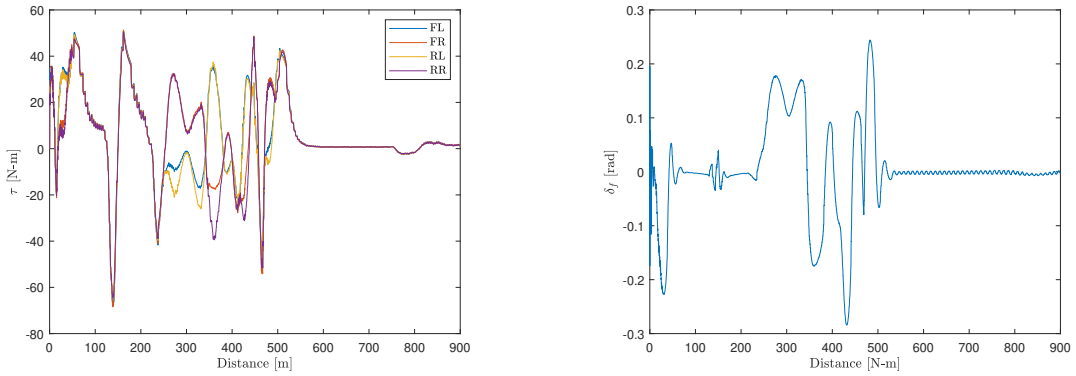
$$||u(k+1) - u(k) - \epsilon||_{R_{rate}} = ||\tau(k+1) - \tau(k) - \epsilon_T||_{R_\tau} + ||\delta_f(k+1) - \delta_f(k) - \epsilon_\delta||_{R_\delta} \tag{4.4}$$

The respective weights used in the cost function are as shown in the Table **??**

Q	R_{el}	R_{sy}	R_{sx}	R_{rr}	R_τ	R_δ
diag(7e12; 1e12)	**1.00E+05**	**5.00E+03**	**1.00E+02**	**1.00E-04**	**1.00E+04**	**1.00E+03**

Table 4.2: *Weights for path tracking error term and energy losses terms in the cost function of Instantaneous optimization problem*

The motor torques T_m and front wheel angle δ for the $900m$ section of GCC is shown in figure(4.11).



(a) *Motor torque Allocation* (b) *Steering angle allocation*

Figure 4.11: *Steering angle and motor torque allocation for instantaneous method*

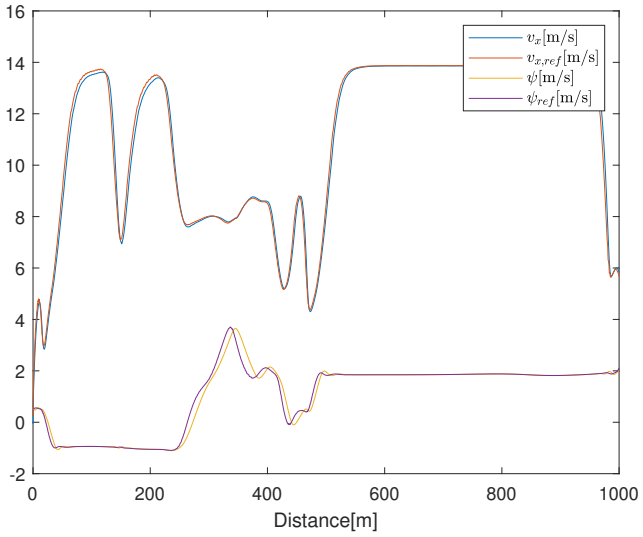


Figure 4.12: *Reference tracking for Instantaneous method*

The lateral deviation is within 0.5 m for sections of the road with large curvature. But at sections with small curvature, for example a sharp 90 degree turn had a lateral deviation of up to 3.5 m. Average lateral deviation was found to be 0.36m.

Figure(4.13) shows gg-diagram for the complete GCC. This diagram shows the combination of longitudinal and lateral acceleration occurring during the drive. Ideally the plots should be dense around the 0 g of both axes of the diagram for a comfortable drive.

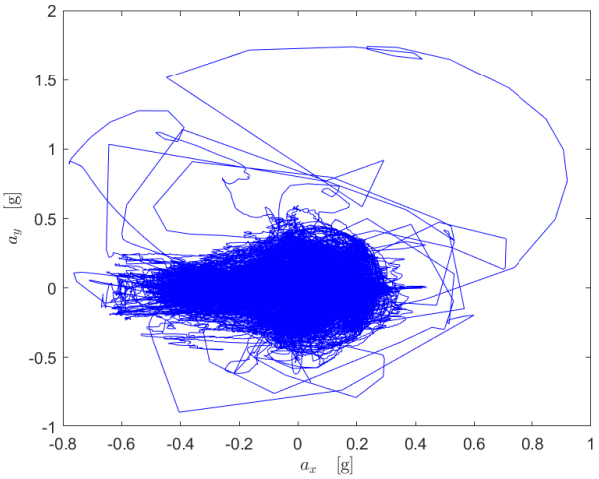
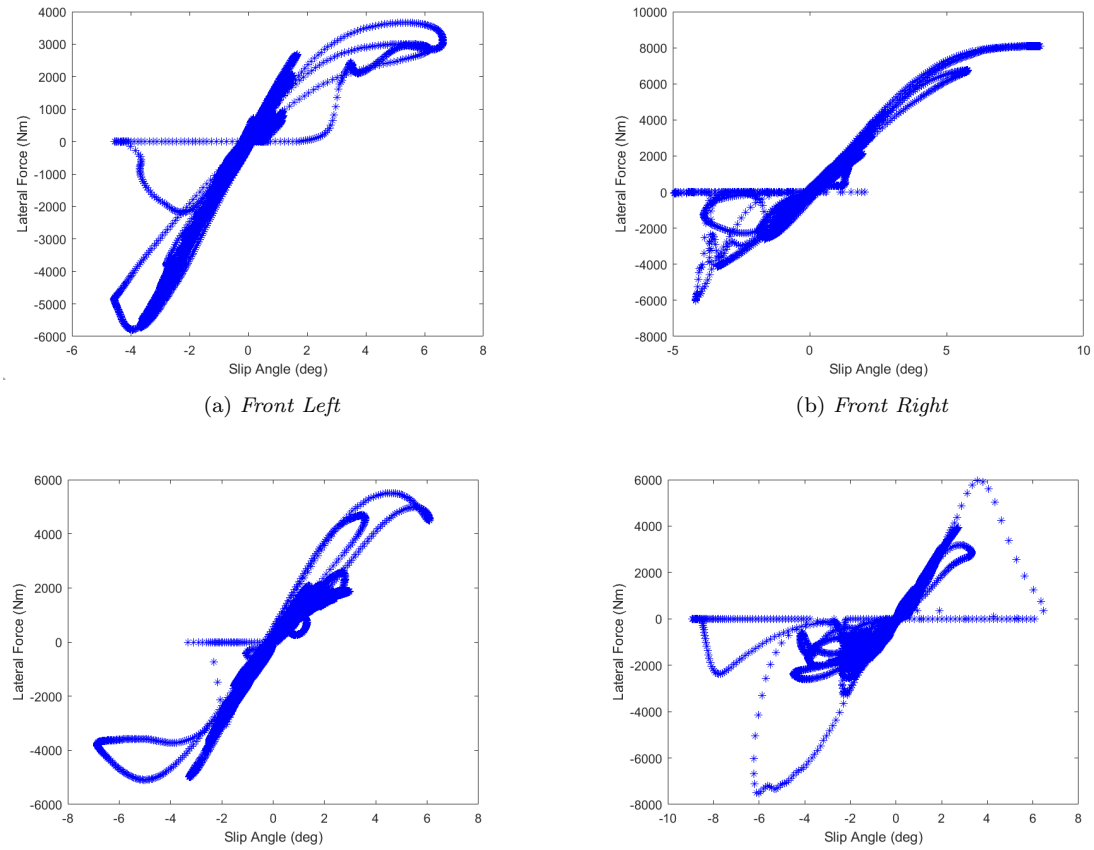


Figure 4.13: *gg-diagram for Instantaneous method*

The motivation behind using the Pacejka tire model and NLP was to capture non-linear dynamics of the vehicle in detail. Figure(4.14) shows the lateral force F_y and slip angle α obtained from the CarMaker environment. It can be seen that all four tires operate in the non-linear region.



(a) *Front Left* (b) *Front Right*

(c) *Rear Left* (d) *Rear Right*

Figure 4.14: *Friction utilisation of tires for Instantaneous Method*

The energy consumption and individual losses occurred for GCC using instantaneous method is shown in the table (4.3). The time taken for the vehicle to complete the cycle was 93 minutes and 32 seconds.

Overall energy consumption and losses for GCC	
Total energy consumed	50.98 MJ
Electric losses	11.09 MJ
Longitudinal slip losses	0.62 MJ
Lateral slip losses	1.69 MJ
Rolling resistance losses	16.66 MJ

Table 4.3: *Energy consumption for 66.25 kms long drive cycle of GCC using Instantaneous control allocation*

4.2.2 NMPC based CA - IPOPT (CasAdi)

For the NMPC based control allocation, the length of the prediction horizon N is kept as 10 with each time step T_s in the prediction horizon being 0.1 s. Therefore the vehicle which is being controlled by the NMPC based control allocator is able to plan its controls by looking at the reference states which it needs to follow for the upcoming $N \times T_s$ s i.e. 1 second.

The cost function defined in 3.43 is reformulated by making the inequality constraint of actuation rate as a

part of the cost function as shown in Equation 4.5

$$J(X_0, u_k) = \sum_{k=1}^{N} ||\hat{X}(t+k) - v_d(t+k)||_Q^2 + ||P_{losses(u_k)}||_R^2 + ||u(k+1) - u(k) - \epsilon||_{Rrate} \tag{4.5}$$

The respective weights in the above cost function are as shown in the Table 4.4

Q	R_{el}	R_{sy}	R_{sx}	R_{rr}	R_{τ}	R_{δ}
diag(7e12; 1e12)	**1.00E+05**	**1.00E+04**	**1.00E+02**	**1.00E+02**	**1.00E+04**	**1.00E+03**

Table 4.4: *Weights for path tracking error term and energy losses terms in the cost function of NMPC based CA problem*

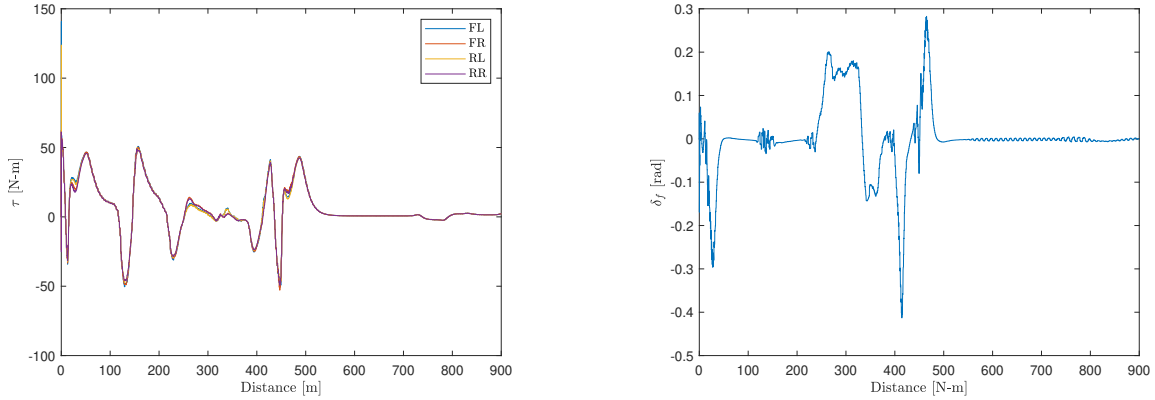
Similar to Instantaneous CA simulations in the previous section, the sampling time for CA was maintained at 0.1 s while the high fidelity model is running at 10^{-3} s as the sample time for simulation, making the CA allocate controls for every 0.1 seconds of the simulation. The slack parameters on actuation rate for torques and steering angle are also the same as in instantaneous based CA. The mean time per control allocation was found to be 0.04 s.

The total time taken for the the vehicle to complete the cycle using NMPC based control allocation was 94 minutes 23 seconds. The energy consumption and losses for the entire stretch of GCC can be summarized as shown in the Table 4.5

Overall energy consumption and losses for GCC	
Total energy consumed	49.087 MJ
Electric losses	10.79 MJ
Longitudinal slip losses	0.51 MJ
Lateral slip losses	1.36 MJ
Rolling resistance losses	16.41 MJ

Table 4.5: *Energy consumption for 66.25 kms long drive cycle of GCC using NMPC based control allocation*

The wheel motor torques and steering angle allocated for tracking same 900 m path sector of GCC for the NMPC based control allocation method are as shown in the figure 4.15



(a) *Motor torque allocation* (b) *Steering angle allocation*

Figure 4.15: *Motor torques and steering angle allocation in high fidelity CarMaker's vehicle model using NMPC based CA in the high fidelity plant model*

The given reference states of longitudinal velocity v_{xref} and yaw angle ψ_{ref} generated by the path tracking algorithm was found to be well tracked for these allocated controls as shown in Figure 4.16.

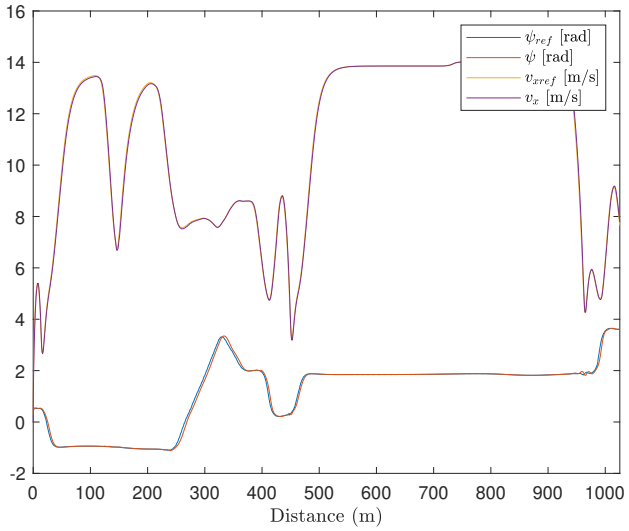


Figure 4.16: *Tracking of the reference states generated by path tacking algorithm using NMPC based CA in the high fidelity plant model*

The lateral deviation is within 0.36 m for sections of the road with large curvature. But at sections with small curvature, for example a sharp 90 degree turn had a lateral deviation of up to 0.6 m. The gg-digram for entire GCC using NMPC based CA is as shown in the Figure 4.17

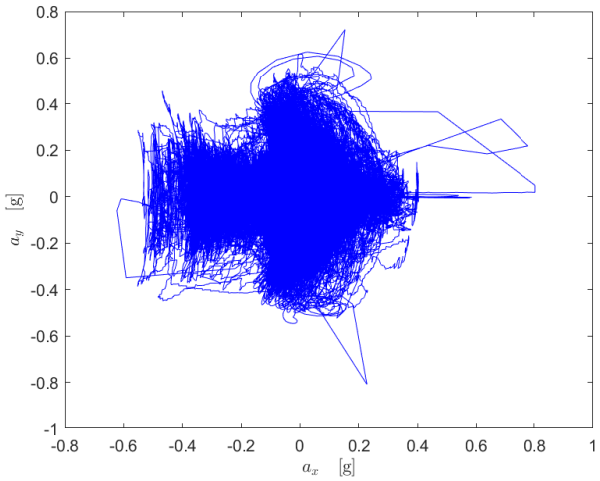
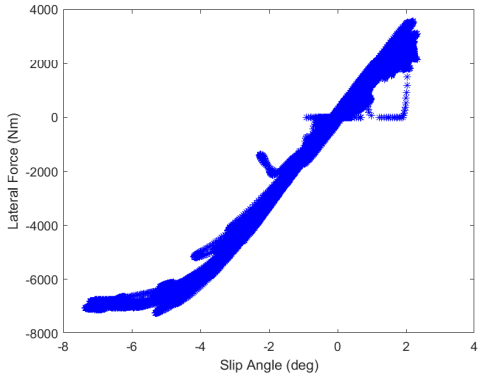
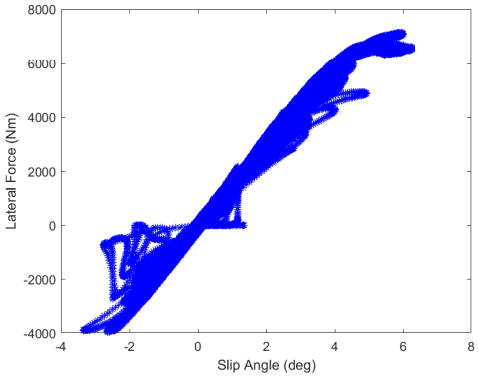


Figure 4.17: *gg-diagram for NMPC based control allocation*

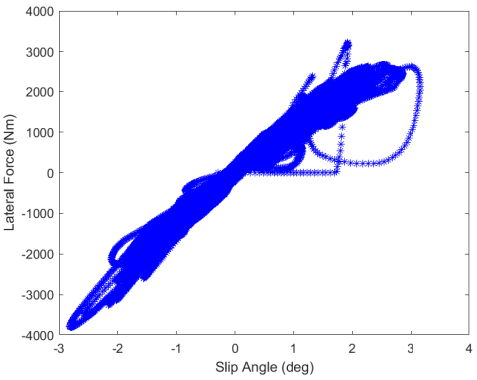
The friction utilization graphs for all 4 wheels of the car shows that most of the lateral force lie on the linear region of tire model using NMPC based CA.



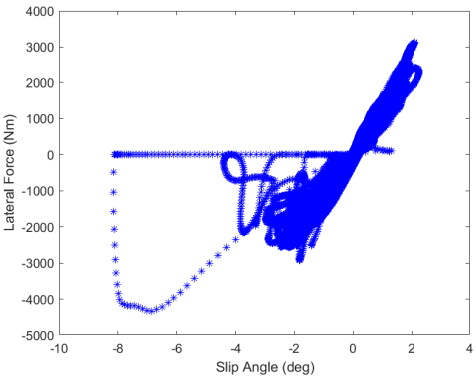
(a) *Front Left*



(b) *Front Right*



(c) *Rear Left*



(d) *Rear Right*

Figure 4.18: *Friction utilisation of tires for NMPC based CA*

4.2.3 Simulation results for ETD

The energy consumption and different losses obtained for ETD driving scenario using the velocity profile of the vehicle simulated using NMPC based control allocation in IPG CarMaker are as shown in Table 4.6. The time taken for simulation for the entire GCC using ETD in IPG CarMaker environment was 94 minutes 21 seconds.

Overall energy consumption and losses for GCC	
Total energy consumed	48.61 MJ
Motor electric losses	10.80 MJ
Longitudinal slip losses	0.52 MJ
Lateral slip losses	0.88 MJ
Rolling resistance losses	16.38 MJ

Table 4.6: *Energy consumption using ETD driving scenario for 66.25 kms long drive cycle of GCC*

As intended, the torques were equally distributed and the developed control architecture for ETD driving scenario was able to track the reference velocity profile obtained from simulating NMPC based control allocation and ultimately tracking the position profile of the path as well. Figure 4.19 illustrates the ETD control architecture's velocity profile tracking performance for Part 1 of GCC. As can be seen, there is no occurrence of significant degree of offset in tracking the reference velocity profile.

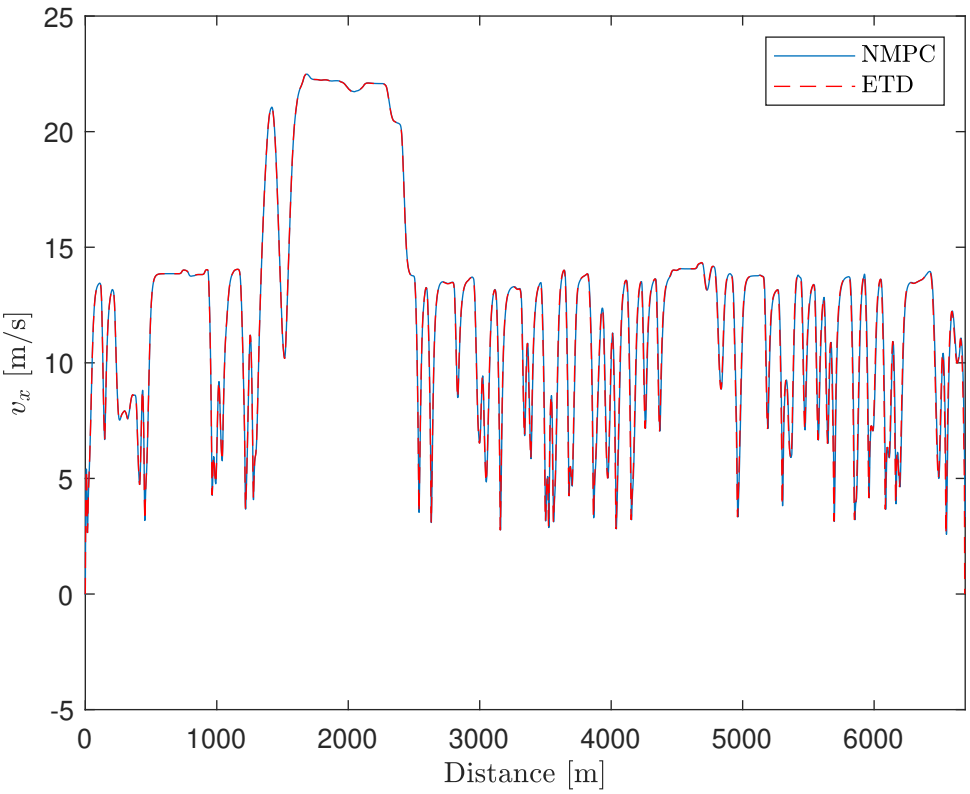


Figure 4.19: *Reference velocity profile tracking by ETD longitudinal dynamics control*

5 Discussion

This section covers the discussion and analysis on the findings from the results described in the previous chapter. Beginning with the findings on solver compatibility between SQP and IPOPT solvers to solve the formulated NLP problem for CA using low-fidelity model.

While developing and implementing the proposed methods for instantaneous CA and NMPC based CA, it was found that SQP solver was not compatible for the formulated CA cost function. Trying to execute the instantaneous CA in CasADi using SQP solver in a way similar to how instantaneous CA in CasADi using IPOPT solver was done led to incorrect control allocation i.e. the requests were not satisfied, the given reference velocity profile was being followed but the given reference yaw angle was being compromised after initial few iterations. This ultimately also led to not following the desired path.

In the very beginning phase of developing the algorithm for instantaneous CA, a 'black box optimization' method was used where the solver to solve the NLP was not specified and the problem was made to be solved as a 'Problem based optimization' instead of 'Solver based optimization' using fmincon toolbox of MATLAB for just 1 iteration of optimization. The intention behind doing this was to let fmincon toolbox recommend what type of NLP algorithm should be used for the type of problem in hand. This was done using 4 reference states i.e. v_x, r,Y_g and ψ, required to follow the desired path. The solver recommended by fmincon's toolbox was SQP algorithm. Therefore with 4 out of 6 states of the plant model entering the nonlinear constraints, fmincon could be used with SQP solver. Eventually the problem was redefined to be solved using only 2 references ψ and v_x for following the path. This led to fmincon not performing as well as it did with 4 states as references. When the same problem was tried to be solved using CasADi, the solver complained about the sparsity of constraint jacobian. Having lesser constraints than the decision variables resulted in a sparser constraint jacobian matrix, making the optimization algorithm a large scale algorithm. Since SQP is not classified as a large scale algorithm, this explains the infeasible solution given by the SQP method. However, the availability of the yaw angle reference generated online could be used to generate the reference yaw rate r by using the derivative of the generated yaw angle reference, which was used as an additional reference for instantaneous CA using fmincon toolbox and SQP algorithm as shown in the Results chapter. The control allocation and the dynamics request tracking had higher discrepancies compared to instantaneous CA using IPOPT solver and NMPC based CA which is again using IPOPT solver. This is reasonable as IPOPT is classified as large scale algorithm that can handle sparser matix during its operations. The motivation to choose IPOPT over SQP solvers such as qpOASES for implementing NMPC based CA also follows the same reasoning. The NMPC based CA uses Multiple-Shooting as the transcription method, which involves both the 6 state space variables of the formulated reference two-track vehicle model and the 5 controls for the entire stretch of prediction horizon of length 10. This creates a constraint jacobian matrix of size 66×5 which is sparser in nature. Therefore the implementation of both instantaneous and NMPC based CA on a high fidelity vehicle model of IPG CarMaker was done using IPOPT solver.

5.1 Comparison between Instantaneous and NMPC based control allocation methods

The energy consumption beteen NMPC based CA and Instantaneous CA methods are compared in Table 5.1 The instantaneous based control allocation with a total energy consumption of 50.984 MJ was found to be 3.9 % more than the total energy consumed from NMPC based control allocation method which had a total energy consumption of 49.074 MJ. The different losses that occurred during the simulation for the entire GCC can be visually summarized as illustrated in the Figure 5.1

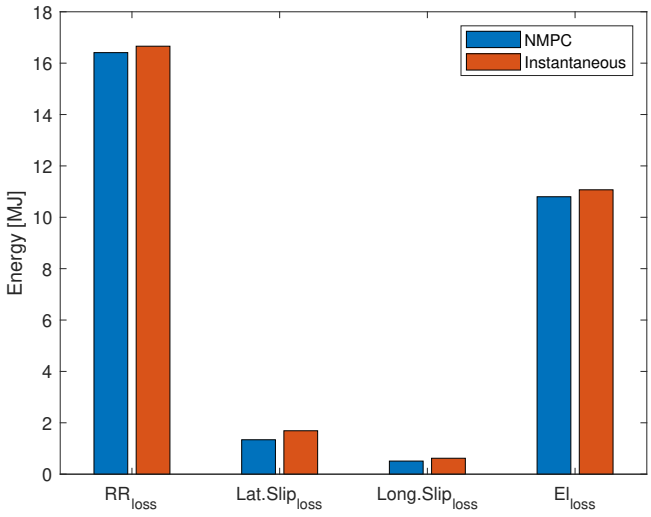


Figure 5.1: *Electric and Tire losses for Instantaneous and NMPC based control allocation*

CA Method	Energy (MJ)
NMPC	49.074
Instantaneous	50.984
% Increase in Instantaneous over NMPC	3.89 %

Table 5.1: *Difference in energy consumption between NMPC based CA and Instantaneous CA*

Total losses in NMPC CA (MJ)	29.05
Total losses in instantaneous CA (MJ)	30.04
Excess of losses in instantaneous CA (MJ)	0.991
% Excess of losses in instantaneous over NMPC	3.41 %
% Proportion of different losses contributing to overall excess in loss	
Rolling resistance loss	25.94%
Lateral slip loss	35.33%
Longitudinal slip loss	10.61%
Electric loss	28.17%

Table 5.2: *Excess losses in instantaneous compared to NMPC and % proportion of each loss in the excess of losses*

The weights for losses in the cost function for both the methods were increased as much as possible such that a stable path and velocity profile tracking is achieved. For all the losses involved, there was an excess of loss in instantaneous CA by 3.4 % compared to NMPC based CA. This excess was majorly contributed due to the excess arising in lateral slip loss, followed by electric and rolling resistance loss respectively. It was observed that at the exit of sharp corners, the vehicle exhibited lateral oscillations about the desired path for a certain distance which could be the cause of higher lateral slip loss. These oscillations can cause the vehicle to slow down, hence requiring higher torques to propel the vehicle forward leading to more electric loss.

Rolling resistance losses and electrical losses form the major composition of the overall energy losses. Figure 5.2 shows a breakdown of different losses constituting the overall losses occurring on implementing the Instantaneous and CA methods.

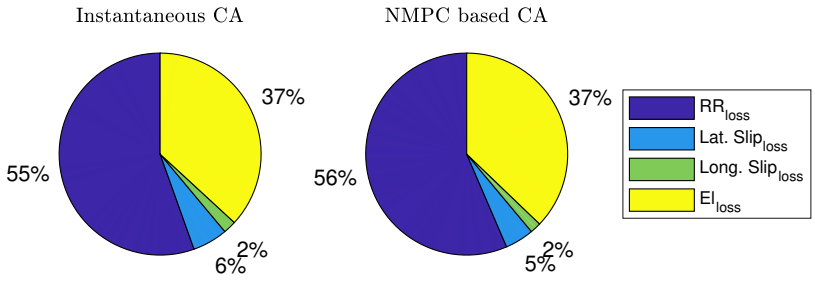


Figure 5.2: *Composition of overall losses occurred in Instantaneous CA and NMPC based CA*

A significant amount of time was spent on tuning the cost function with the right set of weights for path tracking offset and losses in both the methods. Having lesser weights on path tracking offset resulted in the vehicle going off track in both the methods. Therefore it was necessary to anchor the path tracking offset in the cost function with very large weights. For these weights, the path tracking performance using NMPC based CA had lesser lateral deviation within a range of 1 m whereas the instantaneous CA had lateral deviation measuring above 2.5 m at sharp instances like sharp 90° cornering. This is an advantage of tracking the path by allocating controls which are planned by looking at references that are 10 steps ahead than just allocating the controls for the references requested instantaneously.

There is also a very significant difference in the gg-diagram of these two methods. Having a combination of high lateral and longitudinal acceleration results in discomforted ride. As it can be seen in Figures 4.17 and 4.13 , the acceleration plots are mostly dense around 0 g in both the methods, except for the instantaneous based CA which has anomalous instances where the lateral acceleration goes to 1.5 g unlike NMPC which is always within the bounds of 0.6 g to -0.6 g of longitudinal acceleration and 0.5 g to -0.5 g of lateral acceleration.

While decelerating, the NMPC based CA actuates its steering angles, this explains the appearance of some lateral forces on the negative scales of longitudinal acceleration in its gg-diagram.

Despite increasing the weights on rolling resistance losses, there was no significant reduction in rolling resistance loss. It is almost unaffected in both the methods. Where as the electric losses could be minimized by increasing weight only until a point where the control allocator failed to give the torques required for propelling the car. Such an instance was observed while running the simulations for a particular segment of GCC using NMPC based CA, where the starting point is more inclined compared to other parts of GCC. Without having any initial inertia, the car required more wheel torques to climb the inclined road. This led to higher degree of torque vectoring as shown in 5.3 compared to the torque allocation shown in Figure 4.15 where the car starts on a flat road.

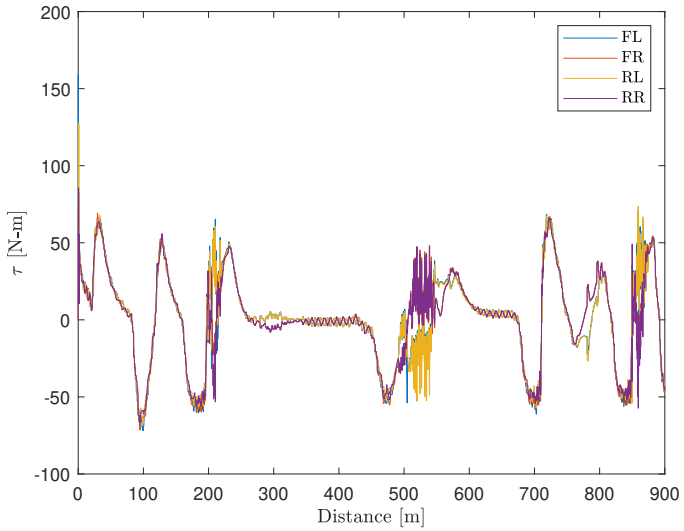


Figure 5.3: *Torques allocation for the vehicle starting on a road with large inclination and with the reduced weights on the electric losses.*

This reflects the impact of weights on electric losses in the formulated cost function, where higher weights reduced the torque vectoring for generating required yaw moment but increasing it further above a value of 10^5 resulted in unstable behaviour or no movement of the car for the requested dynamics.

The wheel steering angle controls given by the Instantaneous CA were not as smooth as seen in the NMPC based CA steering angle control solution shown in 4.15. They appeared to have a transient behaviour which was causing the car to wobble even while following a straight path. Therefore the steering angle allocation for instantaneous CA in 4.11 are obtained after passing the steering angles given by the CA through a first order low pass filter with a transfer function as

$$G(s) = \frac{K}{\tau s + 1} \tag{5.1}$$

where τ is a time constant which is a function of the frequency of the signal passing through it and K is the gain constant of the low pass filter. These constants had to be tuned accordingly until the car was following the path without wobbling much on the straight path.

Lateral slip loss could also be reduced up to a certain extent above which the car would not successfully turn

at the sharp corners. Having no weights on lateral slip loss resulted in unstable driving for both the methods. In NMPC based CA, longitudinal slip losses seemed to be affected with increasing weights on lateral slip losses although not much of longitudinal slip loss was reduced by increasing the weights on itself.

There was a difference in how the torques and steering angles were allocated to the wheels in both the method. At around a point between 200 to 300 m of the travel, the car has to follow a sharp left turning. To achieve this behaviour, instantaneous CA gave a control solution as turning the front wheels to the left as well as some significant torque vectoring. Whereas, the CA using NMPC gave a control solution with minimal torque vectoring and turning the front wheels to the left in order to follow the same turning. This is due to having higher weights on minimizing the difference between requested longitudinal velocity and yaw angle states and the actual state of the plant, relative to the weights on the losses. Weights applied on a particular term in the cost function amplifies the value that term in the total cost value. Therefore a majority of the cost function's value is amplified if the states of the plant do not reach the requested states. This makes the solver to minimize the path tracking offset with a higher priority than minimizing the losses. NMPC on the other hand also has a similar ratio of weights for path tracking offset and losses but it comes with an advantage of planning for N steps of prediction horizon. As the solver in NMPC minimizes a cost function which computes the tracking offset and the losses for 10 steps ahead in future instead of considering the offset and the losses for just 1 step ahead. In a frame of just 1 step optimization, the path tracking offset in the cost function emerges out to be more dominant than the power losses, owing to the higher weights on path tracking offset. The solution given by the solver is intended to minimize the dominant part of the cost function and this causes the instantaneous CA to prioritize the tracking performance more whenever there is significant change in states compared to the requested states, leading to a not very energy efficient control solution but good enough to follow the path. But in a frame of 10 steps (length of prediction horizon) of optimization, in an event of energy inefficient control trajectory as a candidate for the solution, the cost function value is piled up with more power loss over the length of the horizon, which makes the solver to look for other control trajectories that can considerably reduce the power losses along with minimal path tracking offset.

During the iterations of instantaneous CA optimization using IPOPT solver in CasADi, there were instances where the solver indicates the appearance of NaN entities in the jacobian of the cost function. In NMPC which uses multiple shooting method of transcription where the equality constraints involves model state evolution equations using RK4 method, rows in the jacobian of constraints corresponding to v_x and r also had NaN entities. NaN entities occur when an element is being divided by 0. This is usually harmless while using the NLP solver such as IPOPT in CasADi since the iteration continues by simply stepping outside the constrained boundary and in the next iteration backtracks the previous linesearch and takes a step that does not fall outside the constrained boundary. Such instances of having NaN in jacobian matrix of cost and constraints with respect to the controls were prominent in the beginning of following the path of every part of GCC where the car begins from initial longitudinal velocity 0 ms^{-1}. This led to investigating the jacobian of the formulated cost function symbolically. On computing the jacobian of the formulated cost function, and symbolically substituting the initial states of longitudinal velocity v_x as 0 and keeping other states ψ and r as non-zero, there was no division by 0 in the jacobian matrix. But when the initial guess for v_x and r together were given as 0 for computing the jacobian , there is a division of 0 taking place in the jacobian matrix. Same is the case with the hessian matrix of the cost function with respect to controls where NaN occurs for a combination of v_x and r as 0. This leads to the solver giving a control solution in terms of torques and steering angle without respecting the law of evolution for v_x and r given by the Equations 3.1 - 3.6 and can be accountable for the initial unstable behaviour of the car causing it to laterally deviate from the path for a very few seconds of the simulation in the

beginning when the initial states v_x and r of the car are 0. However, this behaviour is stabilized eventually, as the following iterations begin to respect the pre-defined constraint when the v_x and r states of the car become non-zero. This initial sub-optimal solution is also accountable for unexpected high torque allocation which contributes to the total electric loss of the drive.

5.2 Comparison between ETD and NMPC based control allocation

This section covers the comparison of results for energy consumption and losses between two driving scenarios where in one the torques are allocated using NMPC based CA and the other where the same velocity profile is tracked with having equal torque allocation on all 4 wheels which is referred as ETD driving scenario.

First off, the velocity profile given to ETD control architecture as a reference to be tracked, which was originally the velocity profile obtained by running the simulations for entire GCC using NMPC based CA, was well tracked to a considerable extent using ETD strategy for longitudinal dynamics control and IPG driver model for lateral dynamics control as it can be seen in Figure 4.19. The ETD simulation was completed 2 seconds faster than the time taken for the simulation using NMPC based CA in CarMaker.

Energy consumed for ETD	48.095
Energy consumed for NMPC	49.074
Excess energy consumed	0.9792
% Increase in NMPC over ETD	2.03%

Table 5.3: *Total energy consumption for ETD and NMPC based torque allocation*

The ETD torque allocation strategy consumed a total energy of 48.095 MJ for all 6 parts of GCC where as NMPC based CA had consumed 49.074 MJ for following the same velocity profile. Therefore the energy consumption using NMPC based CA was 0.97 MJ more than ETD driving scenario. Unlike ETD where the driving scenario involved controlling the lateral dynamics using CarMaker's driver model, NMPC based CA controlled both the longitudinal as well as the lateral dynamics by allocating both torques and wheel angles for steering. A comparison of different losses between these two torque allocation strategies is illustrated in the bar graph as shown in Figure 5.4

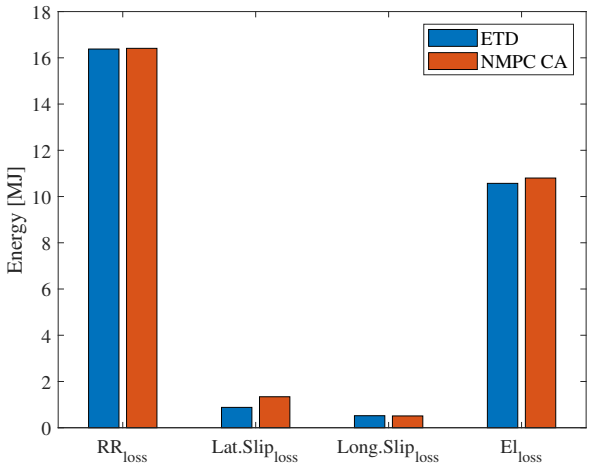


Figure 5.4: *Electric and Tire losses for ETD and NMPC based CA driving scenario*

Losses (MJ)	ETD	NMPC CA	Difference in losses (MJ)
Electric losses	10.574	10.795	0.222
Rolling resistance losses	16.379	16.406	0.027
Lateral slip losses	0.879	1.338	0.458
Longitudinal slip losses	0.516	0.513	-0.003
Total losses	28.35	29.05	0.704

Table 5.4: *Individual category losses for ETD and NMPC based CA and the difference between them*

The table in 5.4 shows by how much the losses in NMPC based CA is greater than losses occuring for ETD driving scenario except for the longitudinal slip loss which is slightly lesser than the longitudinal slip loss in ETD. As can be seen, overall losses in NMPC based CA is 0.704 MJ more than that occurring in ETD driving scenario. Around 65 % of this difference in losses comes from higher lateral slip loss in NMPC based CA compared to ETD, and around 21 % of this difference comes from higher electric electric loss in NMPC based CA. The difference in steering control between two driving scenario can be held accountable for this major difference in lateral slip loss. Figure 5.5 shows the steering controls given by IPG driver model and NMPC based CA.

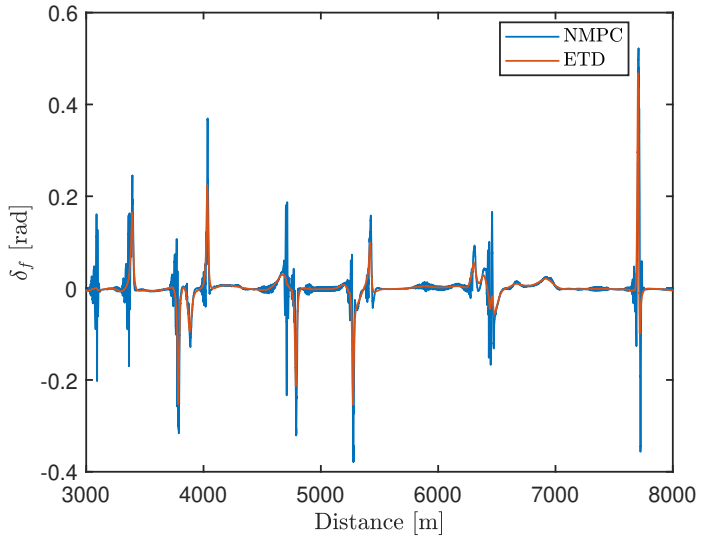


Figure 5.5: *Difference in steering control given by NMPC based CA and IPG driver*

As it can be observed in Figure 5.5, the steering angle allocated using NMPC based CA is transient in nature compared to that given by IPG driver. Before every turning, where the car needs to slow down, the NMPC based CA seemed to actuate the steering angle δ_f while decelerating. This leads to excess lateral slip loss. This also slows down the vehicle as the lateral slip loss results in dissipating the longitudinal inertia laterally, causing the NMPC based CA to allocate more torques to keep up with the reference velocity profile, therefore leading to excess electric loss as well.

On further investigating on what is causing the transient steering angles allocation in NMPC based CA, a correlation was observed between the transient steering angles, longitudinal deceleration and the lateral offset, all occurring at the same instance. Figure attempts to illustrate this correlation.

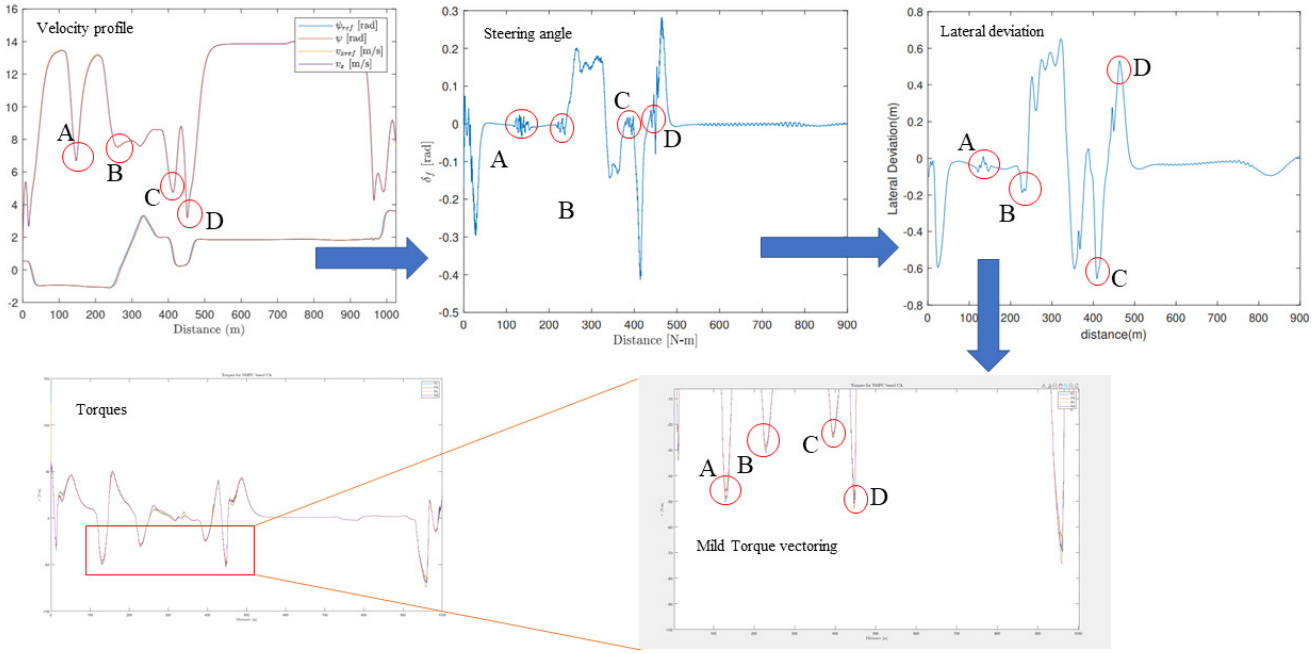


Figure 5.6: *Correlation between deceleration, lateral offset and transient steering angles allocated by NMPC based CA*

As it can be observed in the above figure, in each of the highlighted points during path tracking using NMPC based CA, tagged as A, B, C and D, it can be deduced that at the end of deceleration highlighted by the tags in the velocity profile graph, the optimizer allocates noisy steering angles which causes a sudden lateral offset and at the same instance of noisy steering angles allocation, there is mild torque vectoring allocated. Two speculations can be formulated based on these observations

1. The reference two-track vehicle model used for path tracking might not be reliable while operating in the lower velocity range, which leads to an increase in the lateral offset. Since the path tracking offset has higher weights than the power losses in the cost function, the solver gives a higher priority to minimize the lateral deviation over minimizing the losses such as lateral slip loss, eventually allocating the transient steering angles to have minimum lateral deviation.

2. Among the weights assigned to different losses, the weight on the electric losses is the largest compared to the weights assigned to other losses. Also on observing the composition of losses illustrated in 5.2, the electric losses value are larger than the lateral and longitudinal slip losses. Therefore the NMPC based CA might be trying to achieve the required deceleration by allocating transient steering angles which dissipates the longitudinal inertia laterally in the form of lateral slip loss, rather than using the motor torques which has a higher impact on the power loss cost. This behaviour of NMPC based CA i.e. prioritizing the actuation of steering angle rather than the motor torques to reduce the power losses was also evident during turning around a corner, where the required yaw moment is achieved majorly by actuating the steering angles rather than employing torque vectoring as shown in Figure **??**. And as mentioned in the first speculation, the lateral deviation has the highest weights of all the terms in the cost function shown in Equation 4.5. It is only during an event of such lateral deviation due to undesired yaw moment generated due to noisy steering angles is when the torque vectoring is employed in order to

oppose this undesired yaw moment. This contributes to both, increase in lateral slip losses and electric losses.

6 Conclusion

This work involved developing an online optimization based energy efficient control allocation problem to allocate the motor torques to the individual wheels and front axle wheel steering angles in an over actuated powertrain system of the BEV while following the desired path. The developed path tracking algorithm which dictates the desired behavior of the BEV was inspired from Pure Pursuit Path tracking approach. The longitudinal velocity v_{xref} and yaw angle ψ_{ref} as the reference states were sufficient to trigger the reference two track kinematics vehicle model to give the desired accelerations in order to follow the desired path. The formulated CA problem was inspired from the mixed optimization CA method where the secondary objective function comprised the power loss models for electric loss in individual in wheel motors, the tire losses in terms of rolling resistance loss, lateral and longitudinal slip loss. This optimization based CA method helps in not only distributing the control requirements to the actuators but also to distribute them in an energy efficient manner due to its ability to afford minimizing a secondary objective in its cost function and also allowing to tune the optimization process using weights. In order to compliment faster convergence, reducing the degree of non-linearity of the cost function is advisable. The non-linearity arises at two places in the cost function, one in the formulated system model pertaining to tire dynamics and the other in the formulated loss models. There is not much freedom in the reference system model to reduce the non-linearity as one cannot avoid the reference kinematics model entering the non-linear operating range of the tire dynamics governed by the Pacejka tire model. Also, it is intended that the formulated reference vehicle model should capture close to true behaviour of the vehicle plant model to promote the accuracy of the control allocator. Therefore, the approach resorts to reducing the non-linearity in the formulated loss model, specifically the electric losses. The loss model for electric losses in motor and inverter was formulated by fitting a polynomial surface for power loss data obtained from measurements. The process of fitting a surface involves choosing an appropriate degree of the polynomial which provides the least residual fitting error. A higher degree may yield low residual error but the complexity of objective function significantly increases as the degree of the polynomial increases. The accuracy of the electric loss model is compromised as a consequence of limiting the degree of the objective function to be quadratic. However, to what extent is the accuracy of the electric loss model is being compromised has not been evaluated. This formulated cost function for optimization using IP NLP solver such as IPOPT solver resulted in good convergence, i.e. the reference states for path tracking were being followed accurately and the losses were converging to their lower value with increasing the weights. Although, the obtained energy optimal control solution are globally optimum or not was not analyzed. This is due to the unknown information about the landscape of the combined cost function itself. Of the two different approximation schemes namely, Euler's method and RK4 method, the latter has a better caliber in capturing more non-linear curvature of the cost function and was the preferred approximation scheme for discretizing the OCP. Despite it being computationally expensive than Euler's method, it was found to be feasible to compliment the optimizer which is operating at a frequency of 10 Hz.

Two different approaches to implement the optimization based CA were developed, namely instantaneous optimization based CA and NMPC based CA. The instantaneous optimization based CA had 3.9 % higher energy consumption than the NMPC based CA, whereas the total losses were 3.4 % higher than the losses occurred using NMPC based CA. The average time taken per control allocation using Instantaneous CA method was 0.03 seconds where as for NMPC based CA had a computation time of 0.04 seconds per control allocation. The controls allocated using instantaneous CA seemed to have higher degree of over-actuation than the NMPC based CA such as having both torque vectoring and actuation of front axle wheel steering angle to turn around the corner. Whereas turning around the same corner was performed by having very minimal torque vectoring

and actuating the front axle wheel steering angle for the controls given by NMPC based CA.

The planning behaviour of NMPC based CA seemed to have a positive effect on energy efficiency. A comparison between the energy consumption for ETD using IPG driver model and for the torque allocation given by NMPC based control allocator was done. The energy consumption and losses, both were higher in NMPC based CA than the ETD torque allocation driving scenario with IPG driver model giving the steering control. Around 65 % of this excess loss was due to the lateral slip loss which highlighted the impact of transient steering control allocated by the NMPC based CA while decelerating before turning around the corner. Either the NMPC based CA tried to decelerate the vehicle using front axle steering with an intention to reduce electric losses by not actuating the wheel motor torques instead, or the reference kinematics model used for path tracking in the NMPC based CA optimization which governs the evolution of the vehicle dynamic states may not be reliable for small longitudinal velocity and acceleration. An in depth investigation into this problem needs to be done. This transient steering control allocation behaviour was more prominent in instantaneous based CA even while following a straight path. Adding a low pass filter to the steering control given by the optimizer regularized the transient steering controls to a considerable extent. Similarly, having a low pass filter for the steering controls given by NMPC based CA could also possibly help in getting rid of the swiveling behaviour of the car before turning around the corner and ultimately reducing the lateral slip loss. However, the transfer function of the low pass filter needs further tuning effort to avoid a delay in the vehicle's response to the wheel angles input fed through this low pass filter.

Two numerical optimization algorithms namely Interior Point (IP) and Sequential Quadratic Programming (SQP) were used for solving the NLP problem. SQP algorithm was found to be incompatible for the formulated CA optimization problem whereas the IP based solver such as IPOPT could solve the NLP problem very well. Although the caveat of having IPOPT as the NLP solver is its tendency to give solutions out of the constraint boundaries whenever the jacobian matrix of constraints with respect to controls involves division by 0. This occurred whenever the longitudinal velocity and the yaw rate both had their state value as 0, which is the case when the car is requested to move from its rest position. Therefore in the beginning of every simulation, the car starts to move with a sudden thrust of torque allocations to the wheel motors with a slight swiveling behaviour but eventually stabilizes itself and follows the path without much lateral offset and lesser torques. Therefore, the solution was sub-optimal in the beginning of every simulation. This can be regularized externally to the optimizer by limiting the allocated torques whenever the car starts to travel from its rest position. In fact any form of anomalous controls given by the NMPC based CA can be externally processed as desired before feeding them to the plant model. This can help in reducing the unstable driving behaviour of the car. The problem of having noisy front axle wheel steering angles using NMPC based control allocator might also be an indication of the formulated CA problem not being compatible to satisfy both longitudinal and lateral dynamics requirements of the BEV together. Having the control allocator allocate the torques only to satisfy the longitudinal dynamics requirements and letting the lateral dynamics requirements be taken care by the driver using the steering wheel can reduce the lateral slip loss since the vehicle now is not being fed with noisy front axle wheel steering angles from the NMPC based control allocator. Overall, the complete potential of the developed NMPC based control allocator could not be evaluated thoroughly within the given time frame of this thesis work. Further development and observations of this work by tuning the cost function with different set of weights, assigning different set of slack parameters to individual terms of the cost function, trying different solvers suitable for large and sparse NLPs such as KNITRO, WORHP, SNOPT can be some ways of carrying forward this work for achieving better results. Extending this work in the direction of solving the formulated OCP with different optimizers can also be done by looking at a new state of art nonlinear optimizer namely

Proximal Averaged Newton-type method for Optimal Control (PANOC) [33] which has a clever combination of proximal operator with quasi-newton method and is being extensively used in implementing NMPC based optimal control for obstacle avoidance problems, which also claims to have a faster convergence rate that meets the tight runtime requirements of fast dynamical systems, especially when the aim is to have computations to run in the range of 100 Hz - 1000 Hz.

References

[1] N.L.Azad B. Sakhdari. "An Optimal Energy Management System for Battery Electric Vehicles". In: *4th IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling E-COSM 2015* (2015).

[2] Y. Ding et. al. "Automotive Li-Ion Batteries: Current Status and Future Perspectives". In: *Electrochem. Energ. Rev. 2, 1–28 (2019).* (2019).

[3] Shin ichiro Sakai et al. "Motion Control in an Electric Vehicle with Four Independently Driven In-Wheel Motors". In: *IEEE/ASME Transactions on Mechatronics* (1999).

[4] Lance Cleveland D.L. Margolis. "All Wheel Independent Torque Control". In: *SAE Transactions Vol. 97, Section 4: JOURNAL OF PASSENGER CARS (1988)* (1988).

[5] Y Chen et al. "Energy-efficient control allocation for over-actuated systems with electric vehicle applications". In: *ASME 2010 Dynamic Systems and Control Conference* (2010).

[6] Y Chen et al. "An Adaptive Energy-efficient control allocation on Planar Motion Control of Electric Ground Vehicles". In: *IEEE Transactions on Control Systems Technology* (2011).

[7] J. Torinsson et. al. "Energy reduction by power loss minimisation through wheel torque allocation in electric vehicle: a simulation based approach". In: *International Journal of Vehicle Mechanics and Mobility* (2020).

[8] M Bodson. "Evaluation of optimization methods for control allocation". In: *Journal of guidance, control and dynamics. Vol. 25, No. 4* (2002).

[9] M.W. Oppenheimer et al. "Control Allocation for Over Actuated Systems". In: *Mediterranean Conference on Control and Automation (MED)* (2006).

[10] T A Johansen T I Fossen. "A Survey of Control Allocation Methods for Ships and Underwater Vehicles". In: *Mediterranean Conference on Control and Automation (MED)* (2006).

[11] T. A. Johansen. "Optimizing Nonlinear Control Allocation". In: (2004).

[12] J Tjønnås et al. "Adaptive Control Allocation". In: *Automatica, Volume 44, Issue 11* (2008).

[13] J. Torinsson. "Power loss minimization in electric cars by wheel force allocation". In: (2020).

[14] M. Vignati. "A Torque Vectoring Control for Enhancing Vehicle Performance in Drifting". In: *Electronics, Volume 7, Issue 12 (December 2018)* (2018).

[15] Sie-Long Kek. "Nonlinear Programming Approach For Optimal Control Problems". In: *Proceedings The 2nd International Conference On Global Optimization and Its Applications 2013* (2013).

[16] M.P. Kelly. "Transcription Methods for Trajectory Optimization A beginners tutorial". In: (2017).

[17] M. Diehl et.al. "Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation". In: *Lecture Notes in Control and Information Sciences book series (LNCIS, volume 384)* (2008).

[18] C.J. Tomlin G.M. Hoffman. "Autonomous Automobile Trajectory Tracking for Off-Road Driving: Controller Design, Experimental Validation and Racing†". In: *American Control Conference (ACC)* (2007).

[19] Li Liu Guoxing Bai. "Path Tracking of Mining Vehicles Based on NonlinearModel Predictive Control". In: *Appl. Sci., Volume 9, Issue 7 (April-1 2019)* (2019).

[20] G.L. Campagne et al. "A Nonlinear Model Predictive Control based Evasive Manoeuvre Assist Function". In: *TU Delft masters student theses repository* (2019).

[21] D. Yang et al. "Quasi-Linear Optimal Path Controller Applied to Post Impact Vehicle Dynamics". In: *IEEE Transactions on Intelligent Transportation Systems, 13 (4), pp. 1586-1598* (2012).

[22] R. Craig Conlter. "Implementation of the Pure Pursuit Path tracking Algorithm". In: (1992).

[23] Stefan F. Campbell. "Steering Control of an Autonomous Ground Vehicle with Application to the DARPA Urban Challenge". In: (2007).

[24] L. Ljung T. Glad. *Control Theory - Multivariable and Nonlinear Methods.* CRC press, 2011.

[25] W.C. Durham. “Constrained control allocation - Three-moment problem”. In: *journal of guidance, control and dynamics* (1994).

[26] L. Lidner. “Experience with the magic formula tyre model”. In: *International Journal of Vehicle Mechanics and Mobility* (1992).

[27] S. Gros. *Control Theory - Multivariable and Nonlinear Methods*. CRC press Taylor Francis group, 2011.

[28] C.J . Goh. “Control parameterization: A unified approach to optimal control problems with general constraints”. In: *Automatica, Volume 24, Issue 1* (1988).

[29] K.L . Teo et al. “A Unified Computational Approach to Optimal Control Problems”. In: (1991).

[30] V.M. Becerra et al. “Dynamic integrated system optimization and parameter estimation for discrete time optimal control of nonlinear systems”. In: *International Journal of Control* (1996).

[31] V.M. Becerra et al. “Application of a novel optimal control algorithm to a benchmark fed-batch fermentation process”. In: *Transactions of the Institute of Measurement and Control* (1998).

[32] M.W. Mehrez et al. “Stabilizing NMPC of wheeled mobile robots using open-source real-time software”. In: *International Conference on Advanced Robotics (ICAR)* (2013).

[33] A. Sathya et al. “Embedded nonlinear model predictive control for obstacle avoidance using PANOC”. In: *European Control Conference (ECC)* (2018).