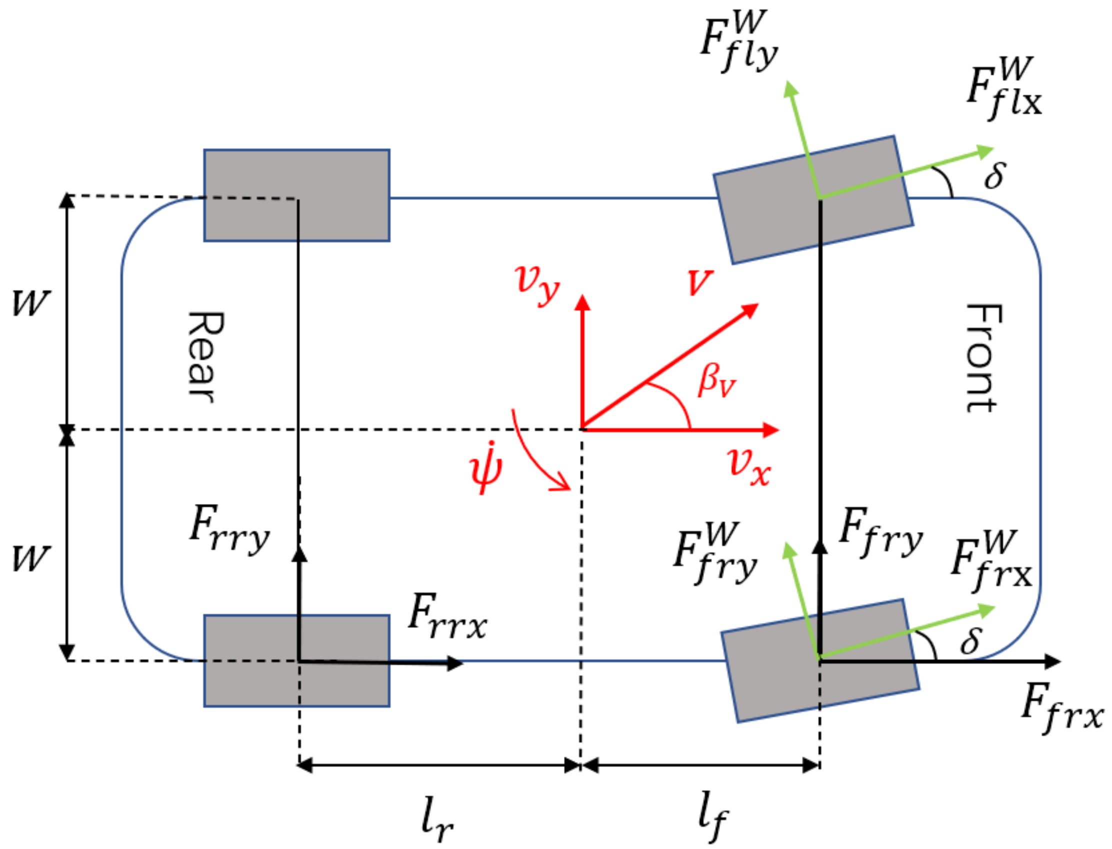


Numerical Strategies for Energy Optimization in Battery Electric Vehicles

Master's thesis in Systems, Control and Mechatronics

Jiapeng Wu, Lihe Chen

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

Numerical Strategies for Energy Optimization in Battery Electric Vehicles

Jiapeng Wu
Lihe Chen

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems, Control and Mechatronics
Chalmers University of Technology
Gothenburg, Sweden 2022

Numerical Strategies for Energy Optimization in Battery Electric Vehicles
Jiapeng Wu
Lihe Chen

Supervisor: Anand Ganesan, Volvo Cars; Derong Yang, Volvo Cars
Examiner: Nikolce Murgovski, Department of Electrical Engineering, Chalmers University of Technology

Master's Thesis 2022
Department of Electrical Engineering
Division of Systems, Control and Mechatronics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Numerical Strategies for Energy Optimization in Battery Electric Vehicles
Jiapeng Wu
Lihe Chen
Department of Electrical Engineering
Chalmers University of Technology

Abstract

For a vehicle with in-wheel electric machines, which is an over-actuated system, a desired output can be achieved by different sets of control inputs. Based on different criteria, those control signals satisfying the control requirements can be further evaluated. Therefore, the controller of an over-actuated system can be designed from a particular perspective such as energy consumption. This thesis report presents an energy-efficient torque vectoring (TV) control method for front-axle-steering electric vehicles with four in-wheel electric machines. The goal is to minimize energy losses, while following a predefined trajectory with speed reference. Different energy loss models for both tire losses such as lateral slip loss and rolling resistance loss and loss in wheel motors and inverters are established in this work. A spatial vehicle dynamics model is established and used as the control model. The proposed spatial model considers the relationship between the vehicle and the path in order to track the path, and uses traveled distance along the path rather than time as the independent variable. Combining the vehicle dynamics and loss models, the optimal control problem is formulated. The cost function (i.e., the energy losses) and system dynamics of the optimal control problem are nonlinear, which makes it a nonlinear programming problem (NLP). A Newton type optimization method called sequential quadratic programming (SQP) is then used to solve the formulated NLP program.

The performance of the proposed torque vectoring method is verified through simulation in different drive cycles. In each scenario, the energy losses of the proposed TV method and an equal torque distribution method is compared. The results show that in the uniform circular motion scenario, the TV method saves up to 3.85% of the total energy losses compared to the equal torque distribution method. For another drive cycle which is the handing track 1 of the Volvo Hällered Proving Ground, the proposed TV method also performs better by saving 3.6% of the total energy loss.

Keywords: All-Wheel-Drive Electric Vehicle, Model-Based Control, Torque Vectoring, Energy Optimization, Numerical Optimization Methods.

Acknowledgements

We would like to thank our supervisors Anand Ganesan and Dr. Derong Yang for their constant encouragement and guidance. Without their help this thesis work would not have been accomplished. We also feel grateful to Manish Suvarna and Juliette Torinsson from Volvo Car Corporation for their various suggestions and instructions. Special thanks to our team manager Ole-Fredrik Dunderberg, who believes us and gave us the opportunity to work on this interesting and challenging topic in Volvo Cars. We would also like to express our sincere gratitude to our academic examiner Prof. Nikolce Murgovski, for all the help and trust.

Jiapeng Wu and Lihe Chen, Gothenburg, June 2022

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AWD	All-Wheel-Drive
BEV	Battery Electric Vehicle
CoG	Center of Gravity
DAE	Differential-Algebraic Equations
DoF	Degree(s) of Freedom
MPC	Model Predictive Control
NLP	Nonlinear Programming Problem
ODE	Ordinary Differential Equations
PMSM	Permanent Magnet Synchronous Motor
QP	Quadratic Programming
SQP	Sequential Quadratic Programming
TV	Torque Vectoring

Nomenclature

Below is the nomenclature of indices, parameters, and variables that have been used throughout this thesis.

Indices

$\star$	Index for wheel location, can be front(f) or rear(r)
$\bullet$	Index for wheel location, can be left(l) or right(r)
$(...)^W$	Variables in the wheel frame
i	Index for i-th wheel, see Fig 1.1

Parameters

$\mathbf{x}_{s,0}$	Initial state vector of the path
S	Length of the entire path
$x_{s,0}$	Initial x coordinate of the path
$y_{s,0}$	Initial y coordinate of the path
$\psi_{s,0}$	Initial yaw angle of the path
m	Mass of the vehicle
I_{zz}	Rotational inertia of the vehicle around z axis
l_f	Distance from vehicle's CoG to front axle
l_r	Distance from vehicle's CoG to rear axle
w	Half track width
h	Vertical distance from vehicle's CoG to the ground
r	Wheel radius
R_{tr}	Transmission ratio between the motor and the wheel
C_α	Cornering stiffness of the wheel
g	Gravitational constant

r_0	Unloaded tire radius

Variables

κ	Curvature of the path
s	Arc length, represents traveled distance along the path
t	Time
d	Distance from vehicle's CoG to path reference
v_x	Longitudinal velocity in vehicle frame
v_y	Lateral velocity in vehicle frame
a_x	Longitudinal acceleration in vehicle frame
a_y	Lateral acceleration in vehicle frame
ψ	Yaw angle of the vehicle
$\dot{\psi}$	Yaw rate of the vehicle
$F_{\star\bullet x}$	Longitudinal force provided by one tire in vehicle frame
$F_{\star\bullet y}$	Lateral force provided by one tire in vehicle frame
F_d	Summation of dissipative forces
β_V	Tangent angle of the vehicle velocity
δ	Steering angle
τ	Torque applied on the wheel
τ_m	Torque output from the motor
ω	Angular velocity of the wheel
α	Slip angle of the wheel
$P_{energy,loss}$	Total energy loss
$P_{drivetrain,loss}$	Drivetrain energy loss
$P_{tire,loss}$	Tire loss
P_{sx}	Longitudinal slip loss
P_{sy}	Lateral slip loss
$E_{velocity,error}$	Velocity tracking error

Contents

List of Figures

1

Introduction

1.1 Background

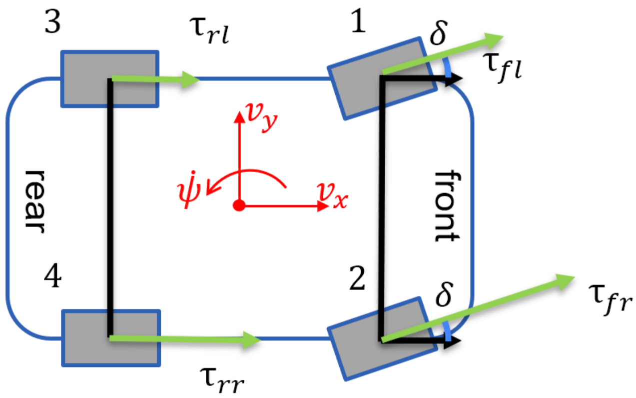


Figure 1.1: An all-wheel-drive vehicle with front steering and four independent motors.

Transportation makes a great contribution to CO_2 emissions, which is one of the critical reasons leading to extreme weather events today. Traditional vehicles using internal combustion engines consume fossil energy. The fossil energy is non-renewable, and it is also responsible for growing CO_2 emissions. These drawbacks have prompted people to continuously seek cleaner source of energy for transportation. Among them, electric vehicles have become the focus of research in recent decades.

Among various electric vehicle architectures, all-wheel-drive(AWD) vehicles have great potential in energy optimization, dynamic performance and safety due to their high degrees of actuation freedom. To reach the full potential of AWD cars, new control strategies will be needed.

As shown in the Figure 1.1, with front steering and four independent in-wheel motors, the AWD vehicle is an over-actuated system. This means that for a certain control target, there might be more than one alternative control signal to achieve the desired behavior of the system. For example, the yaw motion of the car can be achieved either by selecting a certain steering angle, or by generating different distribution of the torque of the two front wheels.

The capability to distribute wheel torques for the four independent electric motors makes the AWD vehicle an over-actuated system, and it needs more advanced control strategy to find the energy efficient solution among multiple alternatives for a certain control target. Therefore, while over-actuation brings the potential for energy optimization[14], it also has higher computational demands.

To address such problems, many methods have been suggested. For example, the authors in [13] propose an optimal torque split strategy for BEV powertrain considering thermal effects, which gives the desired longitudinal velocity while taking thermal performance and thermal-dependent power loss into account.

In [14], a method that allocates wheel torques and steering angles for all wheel drive battery electric vehicles is proposed, to make the vehicle follow a given set of desired longitudinal, lateral and yaw motion while minimizing the energy cost. It has achieved impressive results in reference tracking and energy optimization. The key to its success is the idea of the cost function containing both the error in the path tracking performance and power losses that occur in various sub-systems of the vehicle. However, this method suffers from transient steering angles while decelerating, which generates additional lateral slip loss. This can be further improved by regularizing the transient behavior of the allocated steering angles.

In [2], an energy-efficient torque vectoring controller is implemented based on the experimental assessment of the influence of the control yaw moment on the energy consumption for a wide range of lateral accelerations. The controller is divided into several levels and in the higher level where forces and yaw moment request are given, optimal control signals are decided based on the experimental assessment data of electric drivetrain power losses. This method gives the optimal solution under certain skid-pad tests, but might be unable to give a series of control inputs that can optimize energy consumption for a complicated drive cycle.

In this presented work, the authors propose an optimization-based torque vectoring method that tracks a predefined path reference and minimizes energy losses simultaneously. This approach increases the energy efficiency of AWD electric vehicles. Furthermore, the Acados software package was used to implement the optimizer whose c-code can be used further in the real-time system for future application development. Unlike the rule-based method proposed in [2] and [13], the optimization-based approach proposed in this work places more emphasis on investigating the maximal energy that can be saved in an ideal scenario. Thus, detailed dynamics of the system and energy loss models are needed in order to investigate the full potential of energy savings. And compared to [14], transient steering angles did not appear in this work. Difference in vehicle parameters might be the reason for the disappearance of transient steering angles but further investigation into this problem needs to be done.

In particular, model predictive control(MPC) is not implemented in this work. MPC is an iterative optimization method which gives the best solution as a combination of a sequence of sub-optimal solutions. The authors put emphasis on investigating the full potential of energy savings in this work, so the optimization is implemented for the entire path at once. For the real-time implementation, MPC can be a good choice and related discussions can be found in the subsection 'Future work'.

1.2 Scope

The controller implemented in this work is designed and analyzed using MATLAB. The energy losses focused on in this work contain only drivetrain losses from electric motors and inverters, as well as power losses from tires. The vehicle architecture studied in this work is front-steering vehicle with four identical electric motors and inverters. The optimization is done offline to understand the maximum energy saving possibility in an ideal scenario, i.e. the real-time capability is out of the consideration of this work. The controller model used in the controller design is assumed to be identical to the vehicle plant model, which means that the state feedback from the controller model is assumed to be close to the true values of the system states.

The main objectives of this work are as follows:

- How to model the non-linear dynamics of the AWD vehicle and the power losses as a function of the control variables, wheel torques and steering angle, with good enough numerical simplification such that it is sufficiently accurate and computationally tractable to solve the optimization problem.
- How to generate a path reference from the given X-Y coordinates, which enables simplification of problem formulation and the use of numerical solution approaches.
- How to formulate the path tracking problem, to make the vehicle follow the predefined path while optimizing the energy losses simultaneously.
- How to solve the formulated optimization problem efficiently by selecting proper numerical solution strategies, e.g., iterative algorithms and other problem decomposition methods.

1.3 Outline

This paper is structured as follows. Chapter 2 describes the methodology of formulating the problem of path tracking and energy loss optimization and also introduces the numerical method chosen to solve the mentioned optimization problems. Chapter 3 presents and discusses the simulation results and Chapter 4 summarizes the conclusions from this work.

2

Methods

This section introduces how the energy-efficient control solution is found by formulating and solving an optimization problem. It first introduces an approach to generate the curvature-based path reference, followed by the modelling of the spatial vehicle dynamics and the energy losses. Finally, the algorithms and tools used for solving the nonlinear optimization problem are introduced.

2.1 Simulation process

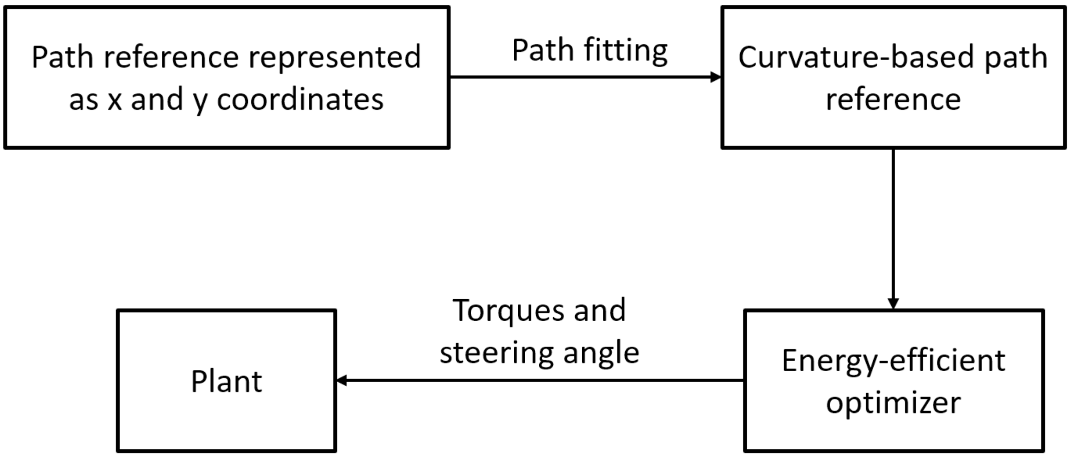


Figure 2.1: The simulation process proposed in this work.

Figure 2.1 shows the simulation process proposed in this work. There are two main components considered in this thesis to formulate an energy-efficient TV optimizer to track a given path reference: i) a reference generator, which can transform a path reference from the form of a sequence of x and y coordinates to its curvature-based representation, i.e, the reference is defined using a sequence of arc-length along itself and corresponding curvatures; ii) an optimizer accepting path reference as constraints of the optimal control problem, and solve it to minimize the energy losses of the vehicle. Besides, a two-track vehicle model is needed as the controller model and the plant model. More detailed descriptions of those components are as follows:

- **Curvature-based path reference**: The path reference represented with x and y coordinates is transformed to its curvature-based version. This makes it possible to achieve path tracking simply by limiting the deviation between path reference and the CoG of the vehicle.

- **Energy-efficient optimizer**:
  - **Vehicle dynamics model**: Spatial dynamics, including a two-track vehicle model, and the deviation from the path reference are used as the controller model to help the optimizer understand how system states respond to the control inputs.
  - **Energy loss model**: As mentioned in the first chapter, for an over-actuated system, there could be infinite control solutions. Energy loss models, consisting of a tire loss model and a drivetrain loss model for an electric machine, are used as the cost criterion for the optimizer to determine the energy losses for a certain set of control inputs.
  - **Optimization algorithms**: Once the optimal control problem is formulated, it is transcribed to a constrained nonlinear problem (NLP) and solved using off-the-shelf numerical solvers. In this work, an open-source toolbox called 'Acados' is used to solve the formulated problem.
- **Plant**: For simplification, the developed TV controller is validated on a plant model identical to the controller model used for controller design.

2.2 Curvature-based representation of the path

Rather than representing the path in $X-Y$ coordinates, in this work the curvature κ is used instead to represent the path. More specifically, the reference path can be described using the initial state,

$$\mathbf{x}_{s,0} = \begin{bmatrix} x_{s,0} \\ y_{s,0} \\ \psi_{s,0} \end{bmatrix} \tag{2.1}$$

and the curvature function,

$$\kappa = \kappa(s) \tag{2.2}$$

where $x_{s,0}$ and $y_{s,0}$ are the initial coordinates of the reference path, $\psi_{s,0}$ is the initial tangent angle of the path and s is the traveled distance along the path.
Based on the initial state and the curvature function, the coordinates and yaw angle of the path at any length s can be easily derived as

$$\begin{aligned} x_s(s) &= x_{s,0} + \int_0^s \cos\psi_s(\sigma)\mathrm{d}\sigma \\ y_s(s) &= y_{s,0} + \int_0^s \sin\psi_s(\sigma)\mathrm{d}\sigma \\ \psi_s(s) &= \psi_{s,0} + \int_0^s \kappa(\sigma)\mathrm{d}\sigma \end{aligned} \tag{2.3}$$

Based on the spatial representation of the path, one can derive the spatial vehicle dynamics using s as the independent variable. The motivation to use this representation is that it is easy to present obstacles (in this work, road boundaries are considered) in the space domain rather than in the time domain. In addition, in spatial representation, one avoids the problem of finding the correct reference point on the reference path to track. The kinematic relationship between the path and the vehicle is discussed in the following subsection. The possible downside of using the representation is that it is hard to describe the vehicle's stopping, since the vehicle dynamics according to this representation assumes that the velocity of the vehicle is not zero. This will be shown in Equation (2.22a) in the following section. One may use a very small velocity to handle this situation.

2.2.1 Transforming the path representation from $X - Y$ coordinates to curvature κ

Since most existing reference path, or drive cycle, is represented using $X - Y$ coordinates, a method need to be developed to transform the path representation from $X - Y$ coordinates to curvature κ. In this work, this is achieved by formulating and solving an optimization problem that estimates road angle and curvature by fitting existing measurements on positions, while having the possibility to impose constraints or penalize large deviations of the estimated curvature. The motivation for estimating the curvature using optimization instead of simple numerical calculation is to obtain a smooth curvature profile accurately enough and prevents numerical issues while solving the optimization problem later.
The state vector for the estimated path is defined as

$$\mathbf{x}_s = \begin{pmatrix} x_s \\ y_s \\ \psi_s \end{pmatrix} \tag{2.4}$$

where x_s, y_s are the coordinates of the path or the waypoints in global coordinate, and ψ_s is the yaw angle of the path in global coordinate.

The control signal is defined as

$$u_s = \kappa, \tag{2.5}$$

Therefore, the derivative of the state is

$$\frac{\mathrm{d}\mathbf{x}_s}{\mathrm{d}s} = \begin{pmatrix} \cos\psi_s \\ \sin\psi_s \\ \kappa \end{pmatrix} = f(\mathbf{x}_s, u_s) \tag{2.6}$$

To enable solving the optimization problem using numerical methods, discretization needs to be performed first. In this work, the Runge–Kutta method (RK4) is used to approximate the dynamics. Note that the variables used here are different and independent of the variables outside of this introduction section.
Define an initial value problem, expressed as

$$\frac{\mathrm{d}y}{\mathrm{d}t} = f(t, y), \quad y(t_0) = y_0 \tag{2.7}$$

where y is an unknown function of time t to be approximated. The derivative of the function y and its initial conditions (t_{0}, y_0) are given.

The RK4 method approximates the function y discretely as

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{6} h \left(k_1 + 2k_2 + 2k_3 + k_4\right) \\ t_{n+1} &= t_n + h \end{aligned} \tag{2.8}$$

for $n = 0, 1, 2, ...$, where h is the time step, using

$$\begin{aligned} k_1 &= f\left(t_n, y_n\right) \\ k_2 &= f\left(t_n + \frac{h}{2}, y_n + h\frac{k_1}{2}\right) \\ k_3 &= f\left(t_n + \frac{h}{2}, y_n + h\frac{k_2}{2}\right) \\ k_4 &= f\left(t_n + h, y_n + hk_3\right) \end{aligned} \tag{2.9}$$

Here y_{n+1} is the RK4 approximation of $y\left(t_{n+1}\right)$. The next value (y_{n+1}) is calculated by the present value (y_n) plus the weighted average of four increments, where each increment is the product of the size of the interval, h, and an estimated slope specified by the derivative function f. Using the RK4 method to discretize the system dynamics yields

$$\mathbf{x}_s(k+1) = g(\mathbf{x}_s(k), u_s(k)) \tag{2.10}$$

As mentioned previously, the path reference represented as the coordinates x and y needs to be transformed to a curvature-based path reference by solving an optimization problem

$$\begin{aligned} \min_{\mathbf{x}_s(k), u_s(k)} \quad & \sum_{k=1}^{N} (x_r(k) - x(k))^2 + (y_r(k) - y(k))^2 + Q \sum_{k=1}^{N-1} (\kappa(k+1) - \kappa(k))^2 \\ \text{s.t.} \quad & \mathbf{x}_s(k+1) = g(\mathbf{x}_s(k), u_s(k)) \\ & \mathbf{x}_s(0) = \mathbf{x}_{s,0} \end{aligned} \tag{2.11}$$

where x_r and y_r are the coordinates of the sample points, $Q \sum_{k=1}^{N-1} (\kappa(k+1) - \kappa(k))^2$ is the penalty term on the change rate of the curvature. The penalty term is intended to smooth the curvature. The constraint for initial position$(\mathbf{x}_s(0) = \mathbf{x}_{s,0})$ is not necessarily needed if, e.g., the measurement of the initial positions is uncertain.

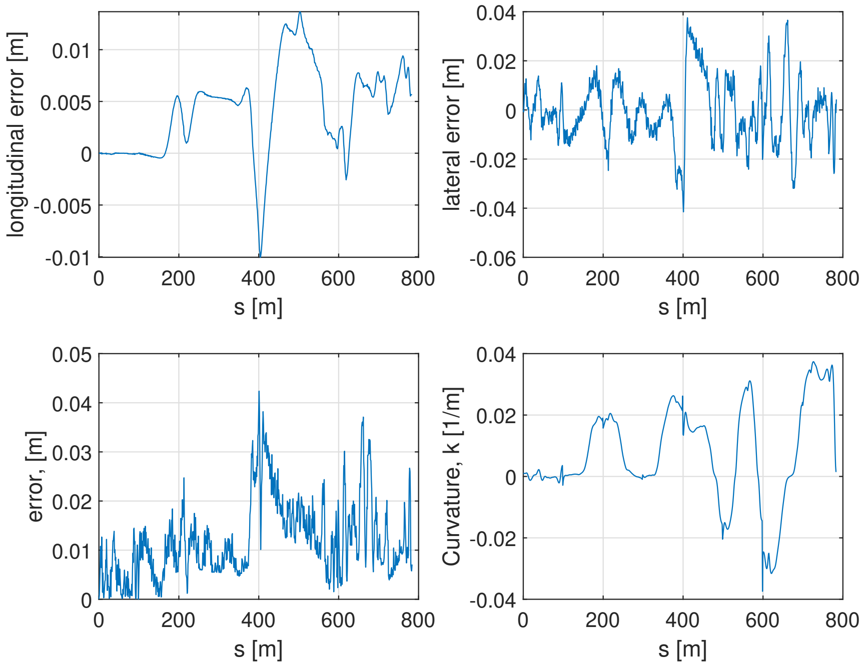


Figure 2.2: The figure shows the curvature of the path estimated by solving the optimization problem with the penalty term, and the spatial errors of the estimated path when compared with the reference path.

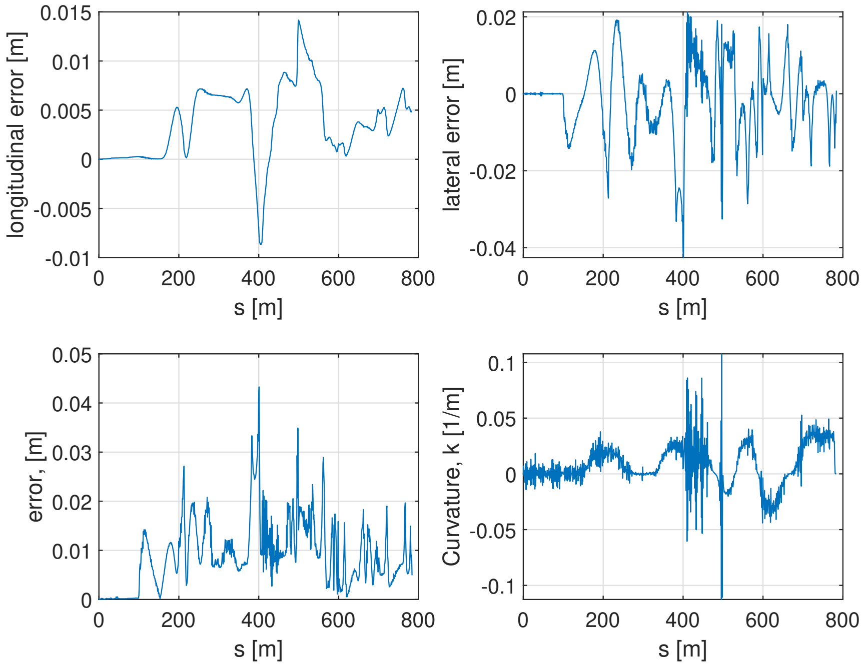


Figure 2.3: The figure shows the curvature of the path estimated by solving the optimization problem without the penalty term, and the spatial errors of the estimated path when compared with the reference path.

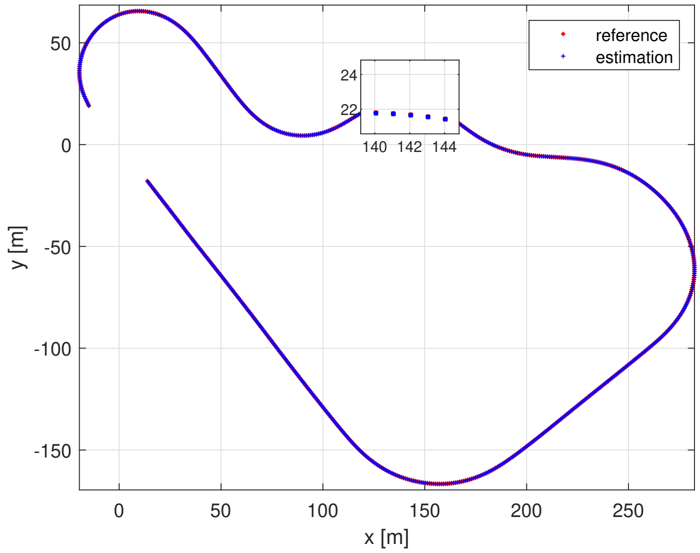


Figure 2.4: The curvature-based path reference (waypoints) estimated by solving the optimization problem with the penalty term.

Figures 2.2 and 2.3 show the curvature of the path estimated by solving the optimization problem without or with the penalty term, and the spatial errors of the estimated path when compared with the reference path, using an optimization toolbox CasADi. By comparison, it shows that the penalty term can greatly smoothen the curvature while maintaining the range of the distance error between the estimated points and the reference points. Figure 2.4 shows the waypoints generated by solving the optimization problem with the penalty term, to transform the representation of the path reference. The estimated waypoints are very close to the reference waypoints.

2.3 Spatial vehicle model

2.3.1 Two-track vehicle dynamics

In this study a two-track vehicle model is used that consists of longitudinal, lateral, and yaw motions to model the dynamics of the vehicle. The control inputs in this work are the steering angle of the front wheel δ and the four independent torques applied to each wheel τ_i:

$$\mathbf{u} = \begin{bmatrix} \delta \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix}. \tag{2.12}$$

As shown in Figure 2.5, v_x and v_y are the longitudinal and lateral velocities of the CoG of the vehicle, respectively, $\dot{\psi}$ is the yaw angular velocity of the vehicle. The motion of the vehicle is described by the dynamics of these three states (note

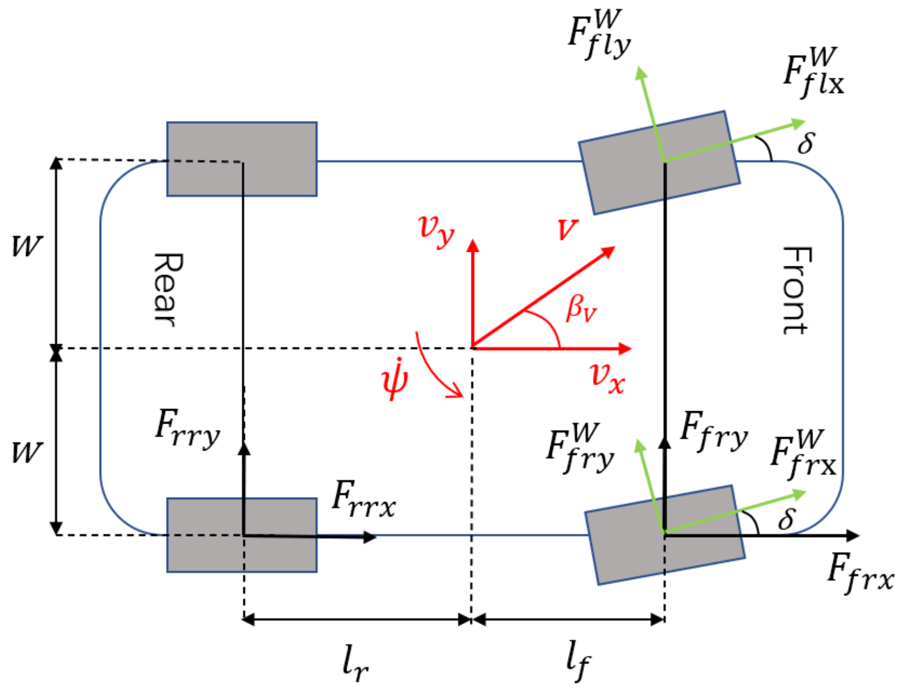


Figure 2.5: 3-DoF Two-Track Vehicle Model

that the state vector defined later includes more states, see Equations (2.21)):

$$\begin{aligned} \frac{\mathrm{d}v_x}{\mathrm{d}t} &= \frac{v_y\dot{\psi} + (F_{flx} + F_{frx} + F_{rlx} + F_{rrx} - F_d)}{m} \\ \frac{\mathrm{d}v_y}{\mathrm{d}t} &= \frac{-v_x\dot{\psi} + F_{fly} + F_{fry} + F_{rly} + F_{rry}}{m} \\ \frac{\mathrm{d}\dot{\psi}}{\mathrm{d}t} &= \frac{1}{I_{zz}} \cdot [l_f \cdot (F_{fly} + F_{fry}) - l_r\,(F_{rly} + F_{rry}) \\ &+ w \cdot (F_{frx} + F_{rrx} - F_{flx} - F_{rlx})] \end{aligned} \tag{2.13}$$

where $F_{\star\bullet x}$ and $F_{\star\bullet y}$ are longitudinal and lateral forces for each tire in vehicle-fixed coordinates (discussed in detail in Section 2.3.3), I_{zz} is the rotational inertia of the vehicle around the z axis, l_f and l_r are the length from CoG to front and rear axles, respectively, w is the half track width of the vehicle, F_d is the dissipative forces, i.e, aerodynamic drag and rolling resistance, expressed as follows [9]

$$F_d = C_D A \frac{\rho v_x^2}{2} + mgC_{rr} \tag{2.14}$$

where C_D is the drag coefficient, A is the frontal area, ρ is the air density, g is the gravity acceleration, C_{rr} is the rolling resistance coefficient. Note that the slope of the ground is assumed to be zero in this work.

Note that there are two types of coordinate reference systems used for analysis. One is the vehicle coordinate reference system fixed with the moving vehicle. The other is the coordinate reference system fixed with each wheel(with superscript W). The equations that describe the relation of tire forces represented in these two different coordinates are as follows:

$$\begin{aligned} F^W_{\star\bullet x} &= F_{\star\bullet x}\cos\delta + F_{\star\bullet y}\sin\delta \\ F^W_{\star\bullet y} &= -F_{\star\bullet x}\sin\delta + F_{\star\bullet y}\cos\delta \end{aligned} \tag{2.15}$$

2.3.2 Transforming temporal dynamics to spatial dynamics

Compared to time-dependent vehicle models, spatial models have proven to be superior in some areas. The authors in [4] show that spatial predictive control can improve the obstacle avoidance performance of the path planner while maintaining the real-time feasibility. This is mainly because representing a general-shaped obstacle with spatial models is more straightforward than representing it with time-dependent models. In [8], spatial model is shown to be useful in terms of improving driving safety and smoothness.

In this subsection, the two-track vehicle dynamics is converted to its corresponding spatial representation, which combines the vehicle dynamics and the stretch of the reference path using arc-length along the path s as an independent variable. Figure 2.6 shows the geometry variables related to the following derivation.

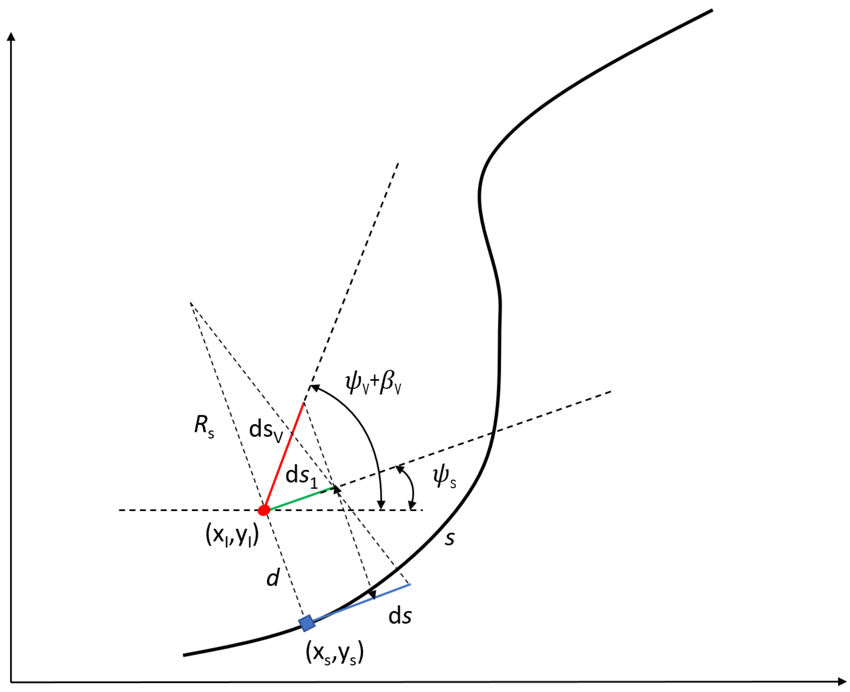


Figure 2.6: The initial position (x_I, y_I), which is represented as a circle, is at distance d from the corresponding projection along the path, (x_s, y_s), represented by the square. When the sample distance traveled along the path is ds, the corresponding real vehicle movement is ds_V, in the direction of $\psi_V + \beta_V$. The tangent angle of the path and its radius are denoted by ψ_s and R_s, respectively.

Based on temporal representation, a vehicle can be described as

$$\dot{\mathbf{x}}(t) = \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = f(\mathbf{x}(t), \mathbf{u}(t), t) \tag{2.16}$$

The state vector $\mathbf{x}$ will be defined later in Equation (2.21). Alternatively, one can sample along the path s instead of time and derive the spatial dynamics as follows:

$$\mathbf{x}'(s) = \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}s} = f(\mathbf{x}(s), \mathbf{u}(s), s)\frac{\mathrm{d}t}{\mathrm{d}s} \tag{2.17}$$

The term $\mathrm{d}t/\mathrm{d}s$ can be expanded to

$$\frac{\mathrm{d}t}{\mathrm{d}s} = \frac{\mathrm{d}t}{\mathrm{d}s_{\mathrm{V}}}\frac{\mathrm{d}s_{\mathrm{V}}}{\mathrm{d}s_1}\frac{\mathrm{d}s_1}{\mathrm{d}s'} = \frac{1}{v}\frac{\mathrm{d}s_{\mathrm{V}}}{\mathrm{d}s_1}\frac{\mathrm{d}s_1}{\mathrm{d}s} \tag{2.18}$$

According to the geometric relationship, the last two terms can be derived as

$$\begin{aligned} &\frac{\mathrm{d}s_{\mathrm{V}}}{\mathrm{d}s_1} = \frac{1}{\cos\left(\psi_{\mathrm{V}} + \beta_{\mathrm{V}} - \psi_{\mathrm{s}}\right)} \\ &\frac{\mathrm{d}s}{R_{\mathrm{s}}} = \frac{\mathrm{d}s_1}{R_{\mathrm{s}} - d}, \Rightarrow \frac{\mathrm{d}s_1}{\mathrm{d}s} = 1 - \kappa_{\mathrm{s}} d \end{aligned} \tag{2.19}$$

where R_s, $\kappa_s = 1/R_s$ and ψ_s are the radius, curvature, and tangent angle of the path s for the current position, respectively, d is the lateral deviation of the vehicle position from the reference path and $\mathrm{d}s$ is the path traveled by the vehicle in one infinitely small change $\mathrm{d}s_V$. This yields

$$\mathbf{x}'(s) = \frac{1 - \kappa_{\mathrm{s}} d}{v\cos\left(\psi_{\mathrm{V}} + \beta_{\mathrm{V}} - \psi_{\mathrm{s}}\right)} f(\mathbf{x}(s), \mathbf{u}(s), s) \tag{2.20}$$

The state vector $\mathbf{x}$ can be represented as

$$\mathbf{x} = \begin{bmatrix} t \\ d \\ v_x \\ v_y \\ \psi \\ \dot{\psi} \end{bmatrix} \tag{2.21}$$

where t is the time, v_x and v_y are the longitudinal and lateral velocities of the vehicle in the vehicle frame, ψ and $\dot{\psi}$ are the yaw angle and yaw rate of the vehicle, respectively, s is the length traveled along the reference path. The vehicle dynamics in the vehicle frame can be modeled as

$$
\begin{aligned}
\frac{\mathrm{d}t}{\mathrm{d}s} &= \frac{1-\kappa_s(s)\cdot d}{v\cos(\psi+\beta-\psi_s(s))} && \text{(2.22a)}\\
\frac{\mathrm{d}d}{\mathrm{d}s} &= (1-\kappa_s(s)\cdot d)\cdot\tan(\psi+\beta-\psi_s(s)) && \text{(2.22b)}\\
\frac{\mathrm{d}v_x}{\mathrm{d}s} &= (v_y\cdot\dot{\psi}+\frac{F_x-F_d}{m})\cdot\frac{\mathrm{d}t}{\mathrm{d}s} && \text{(2.22c)}\\
\frac{\mathrm{d}v_y}{\mathrm{d}s} &= (-v_x\cdot\dot{\psi}+\frac{F_y}{m})\cdot\frac{\mathrm{d}t}{\mathrm{d}s} && \text{(2.22d)}\\
\frac{\mathrm{d}\psi}{\mathrm{d}s} &= \dot{\psi}\cdot\frac{\mathrm{d}t}{\mathrm{d}s} && \text{(2.22e)}\\
\frac{\mathrm{d}\dot{\psi}}{\mathrm{d}s} &= \frac{1}{I_{zz}}\cdot[l_f\cdot(F_{fly}+F_{fry})-l_r\cdot(F_{rly}+F_{rry})+ \\
&\qquad w\cdot(F_{frx}+F_{rrx}-F_{flx}-F_{rlx})]\cdot\frac{\mathrm{d}t}{\mathrm{d}s} && \text{(2.22f)}
\end{aligned}
$$

where F_x is the sum of the longitudinal force for the four tires in the vehicle frame, expressed as

$$F_x = F_{flx} + F_{frx} + F_{rlx} + F_{rrx}$$

.

The forces applied on four tire are distinguished using three indexes. The first index can be f(front) or r(rear), the second index can be l(left) or r(right) and the third index can be x(longitudinal) or y(lateral). And F_y is the sum of the longitudinal force for the four tires in the vehicle frame, expressed as

$$F_y = F_{fly} + F_{fry} + F_{rly} + F_{rry}$$

and β_V is the tangent angle of the vehicle velocity, as shown in the Figure 2.6, expressed as

$$\beta_V = \arctan\frac{v_y}{v_x}$$

2.3.3 Linear tire model

To simplify the optimization problem, the vehicle dynamics in this work does not include as state variables the angular velocities of the four wheels. As a consequence, it is assumed that there is no longitudinal slip in the wheels. Therefore, the longitudinal tire force is simplified as

$$F^W_{\star\bullet x} = \tau_{\star\bullet}/r \tag{2.23}$$

where $\tau_{\star\bullet}$ is the torque applied to the wheel and r is the wheel radius.
the torque applied on the wheel $\tau_{\star\bullet}$ is determined by the torque produced by the motor $\tau_{\star\bullet m}$ and has the relation

$$\tau_{\star\bullet} = \tau_{\star\bullet m}\cdot R_{tr} \tag{2.24}$$

where R_{tr} is the transmission ratio between the motor and the wheel.

The lateral slip angle is defined as

$$\alpha_{\star\bullet} = \delta_{\star\bullet} - \arctan \frac{v_y}{v_x} \tag{2.25}$$

The lateral tire force

$$F^W_{\star\bullet y} = C_{\star\bullet\alpha} \tan \alpha_{\star\bullet} \tag{2.26}$$

where $C_{\star\bullet\alpha}$ is the cornering stiffness of the wheels. Note that Equation (2.26) states that the lateral tire force is the multiplication of a constant coefficient and the tangent of the slip angle, which guarantees the linearity of the lateral force model when the slip angle is small. There are more complicated nonlinear tire models based on experiments [9]. In this work, the linearized model is used mainly to decrease the computational load.

2.3.4 Weight transfer model

When a vehicle is driven with relatively large accelerations, the normal force distribution will be significantly affected by weight transfer. As a result, the rolling resistance moment in each tire needs to be updated according to the corresponding normal forces after considering the weight transfer. Therefore, it is important to implement an online weight transfer model to properly handle the rolling resistance moment. The authors in [11, 12] have investigated the contribution of weight transfer to energy losses. In both works, the weight transfer model is simplified under certain assumptions.

Assuming the vehicle is a rigid body in both longitudinal and lateral directions, the static weight transfer model can be described as follows:

$$\begin{aligned}
F_{z,f} &= m\frac{l_f g - h a_x}{L} \\
F_{z,r} &= m\frac{l_r g + h a_x}{L} \\
\Delta F_{z,f} &= \frac{m a_y h}{w} \cdot \frac{l_f}{L} \\
\Delta F_{z,r} &= \frac{m a_y h}{w} \cdot \frac{l_r}{L}
\end{aligned} \tag{2.27}$$

where h is the vertical distance from the CoG of the vehicle to the ground, and L is the sum of l_f and l_r. The normal forces distributed at each wheel can then be expressed as follows:

$$\begin{aligned}
F_{z,fl} &= \frac{1}{2}F_{z,f} - \Delta F_{z,f} \\
F_{z,fr} &= \frac{1}{2}F_{z,f} + \Delta F_{z,f} \\
F_{z,rl} &= \frac{1}{2}F_{z,r} - \Delta F_{z,r} \\
F_{z,rr} &= \frac{1}{2}F_{z,r} + \Delta F_{z,r}
\end{aligned} \tag{2.28}$$

This weight transfer model describes the torque equilibrium along the axis x and y, considering the effect of additional torques caused by longitudinal and lateral acceleration. Since normal forces are functions of acceleration, which are the derivatives of the system states, algebraic variable $\mathbf{z}$ is introduced to represent normal forces, and the system dynamics is described in an implicit form

$$\mathbf{f}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}, \mathbf{z}, \mathbf{p}) = 0 \tag{2.29}$$

where $\mathbf{p}$ is the related parameters.

2.4 Energy losses model

During the operation of an electric vehicle, there are different types of power loss occurring in different vehicle subsystems, for example, tire losses, electric losses in motors and inverters, loss due to aerodynamic drag, transmission losses and battery losses. In this thesis, the losses chosen to be minimized are tire losses and electric losses in motors and inverters.

2.4.1 Drivetrain losses

There are drivetrain or electrical losses that occur in motors and inverters during the operation of the BEV. To quantify them as a simple empirical model, the corresponding power loss measurement data are used for the positive operating region of the motor and the inverter. In this work, it is assumed that the motors and inverters for the four tires are all identical Permanent Magnet Synchronous Motor(PMSM). [2, 7, 15] managed to approximate the power loss of the drivetrain through polynomials that are functions of the drivetrain torques. Similarly, in this work the drivetrain loss is approximated using polynomials that are functions of both the drivetrain torque and the angular velocity of the electric motor. Implemented using the curve fitting tool in MATLAB, drivetrain losses are modeled as polynomials of 3rd order with respect to the angular velocity of the motor(ω_m) and of 1st order with respect to the squared output torque of the motor(τ_m^2), expressed as the equation

$$P_{\text{drivetrain,loss}}(\omega_m, \tau_m^2) = p_{10}\omega_m + p_{01}\tau_m^2 + p_{20}\omega_m^2 + p_{11}\omega_m\tau_m^2 + p_{30}\omega_m^3 + p_{21}\omega_m^2\tau_m^2, \tag{2.30}$$

where $p_{10}, p_{01}, p_{20}, p_{11}, p_{30}, p_{21}$ are the fitting coefficients. They are constrained to be positive so that the drivetrain loss function is monotonically increasing, thus more realistic. This model also satisfies $P_{\text{drivetrain,loss}}|_{\omega_m=0,\tau_m=0} = 0$, which also makes the model more realistic.

The motivation for using squared torque as an argument for the fitting function is that it can handle the negative torque during braking, assuming that the drivetrain loss function has the same value regardless of the sign of the torque.

2.4.2 Tire losses

The tire losses are modeled using the first principle method. In principle, tire losses comprise three types of losses, that is, longitudinal slip loss, lateral slip loss, and rolling resistance loss, expressed as

$$P_{\text{tire,loss}} = P_{sx} + P_{sy} + P_{rr} \tag{2.31}$$

To the knowledge of the authors, it is necessary to include individual tire dynamics if one would like to calculate longitudinal tire slip loss. In this work, longitudinal slip loss is not included. The main reason is that including individual tire dynamics makes the whole model too complex to calculate and also decreases the stability of the optimizer. In addition, compared to rolling resistance loss and drivetrain loss, longitudinal slip is much smaller[14].

When applying a steering angle to the wheel and turning the vehicle, lateral slip occurs [16]. The slip results in a lateral slip velocity that is defined as the lateral velocity of the wheel in the wheel frame. Slip velocity gives rise to a power loss characterized by the following formula

$$P_{sy} = -\sum_{i=1}^{4} F_{yi}^{W} v_{yi}^{W} \tag{2.32}$$

where v_{yi}^{W} is the lateral slip velocity of tire i in the wheel frame, defined as

$$v_{yi}^{W} = v_{yi}\cos(\delta_i) - v_{xi}\sin(\delta_i) \tag{2.33}$$

where v_{xi} and v_{yi} are the longitudinal and lateral velocity of the wheel i in the vehicle frame, and δ_i is the steering angle of the wheel i.

Rolling resistance arises due to the deflection of the carcass while rolling and hysteresis in the tire. When the tire rolls, its rubber parts deflect upon contacting the ground. The energy consumed during the deflection is not fully recovered when the rubber parts leave the contact patch due to the internal damping in the tire. This leads to a front-biased pressure distribution on the tire contact patch that gives rise to a moment in the opposite direction of the wheel rotation, i.e., the rolling resistance moment. The value of rolling resistance is affected by: tire design, tire pressure, operating conditions, normal force, and torque applied to the tire. In this thesis, the rolling resistance moment is defined as [9]

$$M_{yi} = -F_{zi} \cdot r_0 \cdot \left\{ q_{sy1} + q_{sy2} \cdot \frac{F_{xi}}{F_{z0}} + q_{sy3} \cdot \left| \frac{v_{xi}}{v_{ref}} \right| + q_{sy4} \cdot \left(\frac{v_{xi}}{v_{ref}} \right)^4 \right\} \tag{2.34}$$

where F_{zi} is the normal force on the tire i, r_0 is the unloaded tire radius during the tire measurement, $q_{sy1} - q_{sy4}$ are the fitting parameters for the tire, F_{z0} is the normal force during the tire measurement, v_{ref} is the translational velocity of the wheel center during the tire measurement. In this thesis, these parameters take the values from [15] as the study case.

To simplify the calculation of the problem using numerical methods, the optimizer removes the absolute operation in the third term of the Equation (2.34) during the calculation. This change will not affect the result since in the testing scenarios in this study, the vehicle always moves forward and $\frac{v_{xi}}{v_{ref}}$ is always positive.
The total power loss due to rolling resistance for the four tires is expressed as

$$P_{rr} = \sum_{i=1}^{4} -M_{yi} \cdot \omega_i \tag{2.35}$$

2.5 Formulation of the optimal control problem

The core objective of this thesis is to implement an optimizer for energy optimization while tracking the given reference path. As described in section 'Spatial Vehicle Dynamics', the distance from the CoG of the vehicle to the corresponding point on the reference path, denoted as d, is included as one of the state variables, and path tracking is achieved by placing a constraint on d. The energy loss models described in the section 'Energy Losses Model' will be used to formulate the cost function of the optimal control problem. The optimal control problem is formulated as

$$\min_{\mathbf{u}} \int_0^S (P_{\text{energy,loss}}(\mathbf{u},\mathbf{x}) + E_{\text{velocity,error}}(\mathbf{u},\mathbf{x}))\mathrm{d}s \tag{2.36a}$$

$$s.t. \quad \text{Equations (2.22)} \tag{2.36b}$$

$$\mathbf{x}(0) = \mathbf{x_0} \tag{2.36c}$$

$$\mathbf{g}(\mathbf{x},\mathbf{u},s) \leq 0 \tag{2.36d}$$

$$\mathbf{u}_{\text{min}} \leq \mathbf{u} \leq \mathbf{u}_{\text{max}} \tag{2.36e}$$

$$\mathbf{x}_{\text{min}} \leq \mathbf{x} \leq \mathbf{x}_{\text{max}} \tag{2.36f}$$

where the energy loss

$$P_{\text{energy,loss}}\,(\mathbf{u},\mathbf{x}) = P_{\text{drivetrain,loss}}(\mathbf{u},\mathbf{x}) + P_{\text{tire,loss}}(\mathbf{u},\mathbf{x}) \tag{2.37}$$

and the velocity tracking error is chosen to be

$$E_{\text{velocity,error}}\,(\mathbf{u},\mathbf{x}) = (v_x - v_{x,ref})^2 \tag{2.38}$$

The optimal control problem formulated as Equations (2.36) has the energy loss and velocity tracking error as a cost function, subject to system dynamics (2.36b), control constraints (2.36e) and state control constraints (2.36f). The state vector $\mathbf{x}$ is a vector of 6 states as described in Equations (2.21). Here, control constraints and state constraints define the reasonable range of control signals as well as the box constraints for both the state variables $\mathbf{x}$ and the algebraic variables $\mathbf{z}$.

As a benchmark, a scenario of no torque vectoring is used, i.e., equal driving torque for the four wheels. Under this scenario, the solutions are obtained also by solving the above optimal control problem with the additional constraints on the control variables such that the control variables satisfy the following relation:

$$\tau_1 = \tau_2 = \tau_3 = \tau_4 \tag{2.39}$$

Given constant reference velocity, the alternative to implement velocity tracking, other than using velocity tracking error in the cost function, as described in Equations (2.36), is to directly set the velocity constraints close enough to the constant reference velocity for the velocity to be tracked. Constraints are expressed as

$$v_s(1-\delta_1) \leq v_x \leq v_s(1+\delta_2) \tag{2.40}$$

where v_s is the constant reference velocity, and δ_1 and δ_2 are small positive numbers. The first method is suitable for scenarios with changing velocity reference, while the latter one simplifies the problem for the case where the velocity reference is constant.

2.6 Solving optimal control problem using numerical method

2.6.1 Transcription of OCP to NLP

Optimal control problem, as an infinite-dimensional problem, can be converted to a finite-dimensional approximation(NLP, namely) using a certain transcription method.[5, 10] In this thesis, the multiple-shooting method is chosen to convert the dynamic system into an NLP problem with finite variables. The multiple shooting method works by breaking up the entire control horizon into several segments and applying parameterized controls on these segments. Piecewise constant controls are used in this work. Apart from controls, intermediate state variables also work as decision variables in the multiple shooting method, which means that the transcription is simultaneously achieved, and one should pay extra attention to match the end of one segment to the start of the next.

There is an alternative transcription method called the single shooting method. The main difference between single shooting and multiple shooting is that single shooting works sequentially by optimizing a sequence of controls, while multiple shooting works simultaneously by optimizing states as well. Although multiple shooting increases the number of decision variables, it actually lowers down the nonlinearity because a set of control signals only affects a certain segment. Thus, multiple shooting tends to perform better than single shooting for large-scale problems and is used in this work. In the end, the general NLP problem can be formulated as:

$$\begin{aligned} & \min_{x} f(x) \\ \text{s.t.} \quad & g(x) \leq 0 \\ & h(x) = 0 \end{aligned} \tag{2.41}$$

In particular, in this work the NLP for energy optimization can be defined as

$$\min_{\mathbf{x}(0:N),\mathbf{u}(0:N-1)} \sum_{j=0}^{N-1} P_{\text{energy,loss}}\left(\mathbf{x}(j),\mathbf{u}(j)\right) + E_{\text{velocity,error}}\left(\mathbf{x}(j),\mathbf{u}(j)\right) \tag{2.42a}$$

$$\text{s.t.} \quad \mathbf{x}(j+1) = f(\mathbf{x}(j),\mathbf{u}(j)), \forall j \in [0, N-1] \tag{2.42b}$$

$$\mathbf{x}(0) = \mathbf{x_0} \tag{2.42c}$$

$$\mathbf{g}(\mathbf{x}(j),\mathbf{u}(j)) \leq 0, \forall j \in [0, N-1] \tag{2.42d}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}(j) \leq \mathbf{u}_{\max}, \forall j \in [0, N-1] \tag{2.42e}$$

$$\mathbf{x}_{\min} \leq \mathbf{x}(j) \leq \mathbf{x}_{\max}, \forall j \in [0, N] \tag{2.42f}$$

where Equation (2.42b) represents the discrete system dynamics, which can be obtained by applying the Runge-Kutta method to Equations (2.22).

2.6.2 Solving NLP using sequential quadratic programming

For a general optimization problem with both equality and inequality constraints, as Equations (2.41) show, a common approach called the Lagrange multiplier method[1] is exploited in this study.

The Lagrange function is defined as

$$\mathcal{L}(x, \mu, \lambda) = f(x) + \mu^\top g(x) + \lambda^\top h(x) \tag{2.43}$$

where μ and λ are Lagrange multipliers for inequality constraints and equality constraints, respectively. Once the Lagrange function is formulated, one can use the so-called KKT conditions[6] to find the potential optimum of the original optimization problem. The KKT conditions state that if $x^\star$ is a local minimum of the NLP problem represented by Equations (2.41), then there exist unique $\mu^\star$ and $\lambda^\star$ that satisfy following equations:

$$\begin{gathered} \nabla_x \mathcal{L}\left(x^*, \mu^*, \lambda^*\right) = 0 \\ \mu^* \geq 0 \\ g\left(x^*\right) \leq 0, \quad h\left(x^*\right) = 0 \\ \mu_i^* g_i\left(x^*\right) = 0, \quad i = 1, \ldots, m \end{gathered} \tag{2.44}$$

The KKT conditions are very useful for finding a local optimum and they are the basis for the optimization algorithm used in this thesis, namely the sequential quadratic programming method. It turns out that one can linearize the fourth condition in 2.44, and turn the KKT conditions into a corresponding quadratic problem. SQP[1] is an iterative method by solving a sequence of quadratic optimization subproblems. At each iteration, a quadratic subproblem like the following is defined to find the search direction:

$$\begin{aligned} \min_x f_{QP}^k(x) = & \nabla f\left(x^k\right)^T x + \frac{1}{2}\left(x - x^k\right)^T \nabla_x^2 \mathcal{L}\left(x^k, \lambda^k, \mu^k\right)\left(x - x^k\right) \\ \text{s.t. } & g\left(x^k\right) + \nabla g\left(x^k\right)^T\left(x - x^k\right) \leq 0 \\ & h\left(x^k\right) + \nabla h\left(x^k\right)^T\left(x - x^k\right) = 0 \end{aligned} \tag{2.45}$$

2.6.3 Acados

Acados is a software package used in this work that can efficiently solve the optimal control and estimation problems. It consists of the following implementations: the integration of ordinary differential equations (ODE) and differential-algebraic equations (DAE), nonlinear programming solvers with real-time iteration, and interfaces to a couple of state-of-the-art QP solvers, including qpOASES, qpDUNES, HPIPM, and OSQP. Acados is powered by the high-performance linear algebra package BLASFEO behind the scenes.

In this work, HPIPM is used in Acados to solve the QP subproblems. HPIPM is a high-performance framework for quadratic programming (QP), designed to solve model predictive control (MPC) problems robustly and efficiently. Numerical experiments show that HPIPM reliably solves complicated QPs and that it supersedes other state-of-the-art solvers in speed [3].

The motivation to select to use Acados in this work is that this work can make use of the modules provided by Acados to construct the optimization problem formulated and solve solutions. Furthermore, Acados is able to produce the code that can be used in the real-time system for future application development.

3
Results

To test the performance of the algorithm, this work performs simulations under two reference path, i.e., uniform circular motion and the track 1 of the Volvo Hällered Proving Ground, as shown in the Figure 3.1. The first path reference is intended to test the algorithm's performance when the system is in the steady state, and the second one is intended to test the algorithm's performance under more complicated and practical situations.

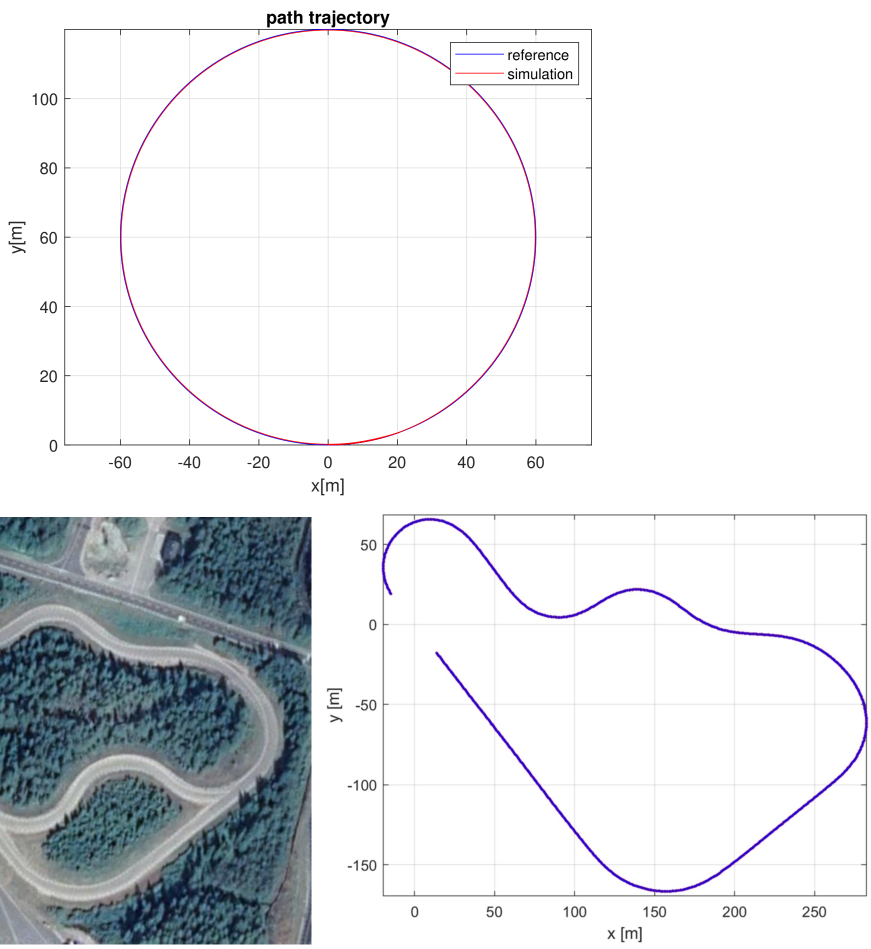


Figure 3.1: The two path references used for the simulation, i.e., a uniform circular motion track (top) and the track 1 of the Volvo Hällered Proving Ground (bottom left) whose waypoint representation (bottom right) is generated using optimization as explained in section 2.2.1.

As a benchmark, a scenario of no torque vectoring is used, i.e., equal driving torques for the four wheels. In this scenario, the solutions are also obtained by running the optimizer to minimize energy losses. To implement no torque vectoring, additional constraints for the control signal torques are set in the optimizer.

3.1 Uniform circular motion

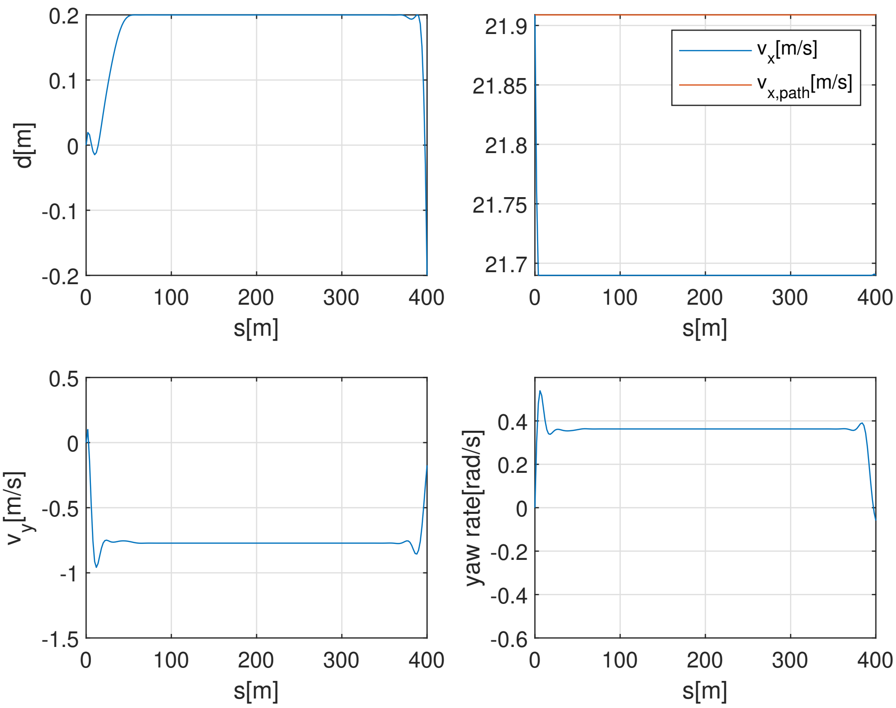
First as a simple scenario, a circle is selected as a path, and the reference speed is constant during driving. Different speeds lead to different lateral accelerations, which are used in this work to accurately describe the sharpness with which the vehicle turns.

Since in this scenario the reference speed is constant during driving, speed tracking is implemented by setting a constraint on the speed state, as explained in the section 'Formulation of an Optimal Control Problem'.

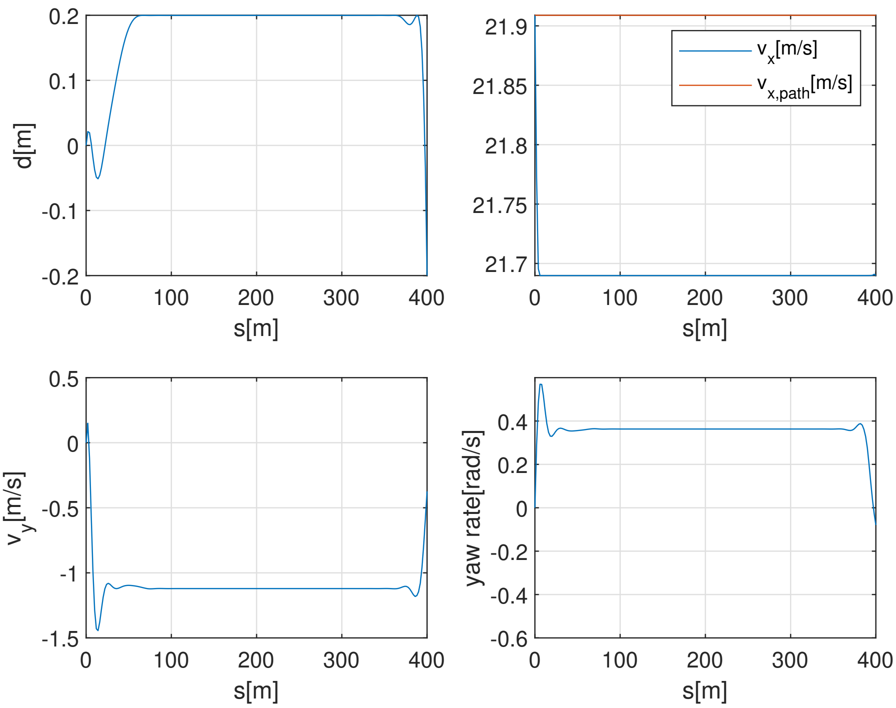
3.1.1 Scenario with lateral acceleration $a_y = 8$ m/s^2 and the path radius $r = 60$ m

Following are the results of the solution given by the optimizer, under the single scenario where the lateral acceleration $a_y = 8$ m/s^2 and the path radius $r = 60$ m. As shown in the Figure 3.1, the vehicle can drive smoothly along the circle without a large lateral deviation.

To investigate it more clearly, the state trajectory from the solution is checked. The Figure 3.2 shows that the deviation from the path (d) is always within the allowed range, set to ± 0.2 m in the optimizer. During most of the driving, the speed is almost constant at the value of the lower bound for the speed during the whole driving, and the yaw rate and the lateral speed of the vehicle are also very stable. Therefore, it can be considered that the system is in steady state during most of the driving.



(a) State trajectory when with torque vectoring



(b) State trajectory when without torque vectoring

Figure 3.2: State trajectory from the solution of the optimization problem.

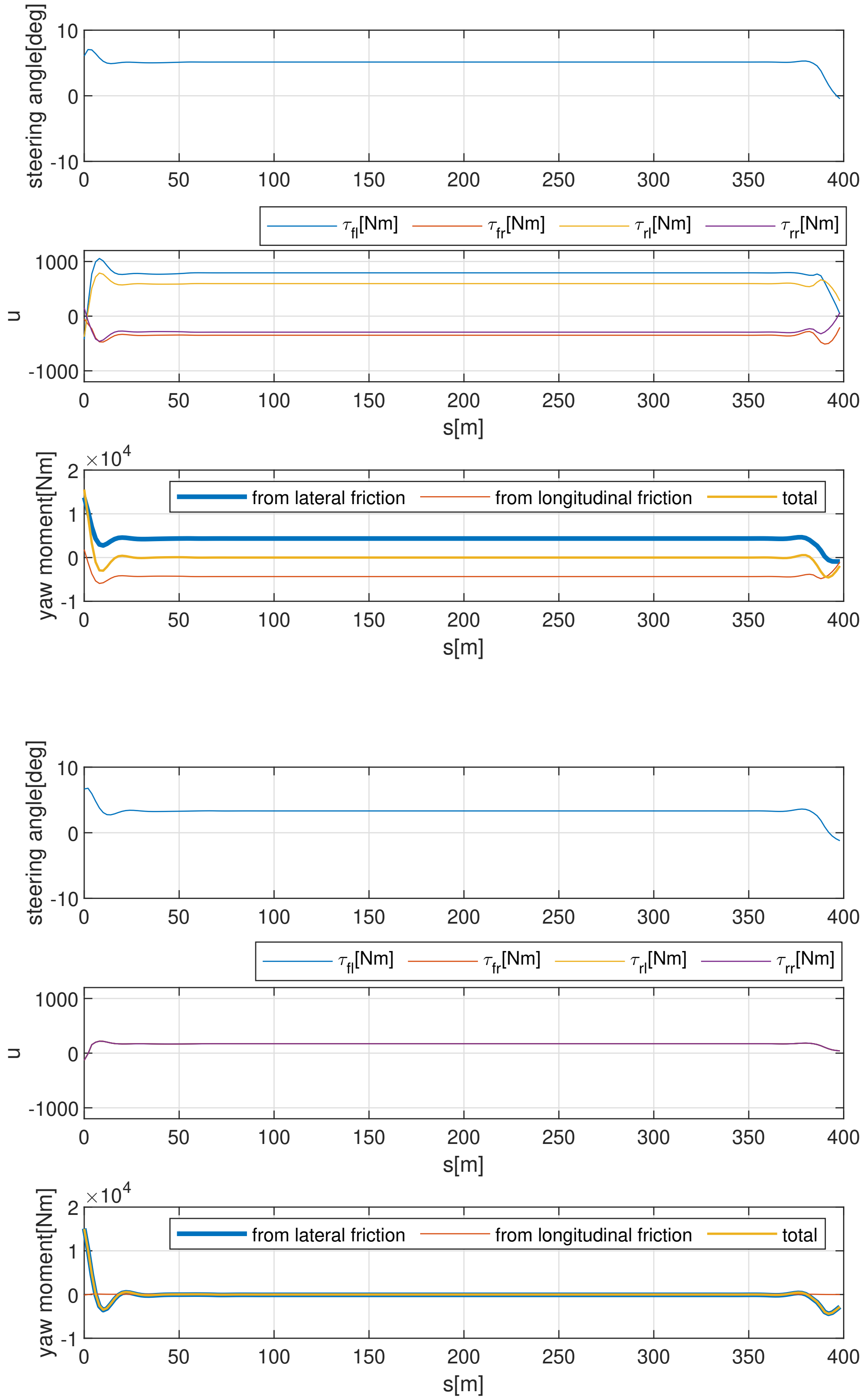


Figure 3.3: Control signal trajectory when with torque vectoring (top) and without torque vectoring (bottom).

Figure 3.3 shows the control signal trajectory when with and without torque vectoring. After around 60-70 m, the system starts to be in the steady state. The torques(M_{w1-4}) for the four wheels are different when torque vectoring is used, whereas the torques are the same without torque vectoring. Since torque vectoring gives rise to the difference in the longitudinal frictions for the four tires, there is an additional yaw moment on the vehicle and the steering angle is different in the two cases. Meanwhile, the difference in lateral friction in the four tires gives rise to another yaw moment, and cancels out the yaw moment of the longitudinal frictions. As a result, the total yaw moment is (almost) zero in the middle part of the simulation, so that the vehicle keeps a constant yaw rate and achieves uniform turning.

The energy loss during the driving simulation with and without torque vectoring is shown in the Figure 3.4. Note that the energy loss here is the sum of the losses from the four wheels. The loss becomes constant when the system goes into steady state. In the steady state, each type of energy loss for the two cases has a different level. Compared to the result without torque vectoring, torque vectoring gives rise to a higher drivetrain electric loss and slightly higher lateral slip loss, but a lower rolling resistance loss and a lower total loss, i.e., the sum of the three losses.

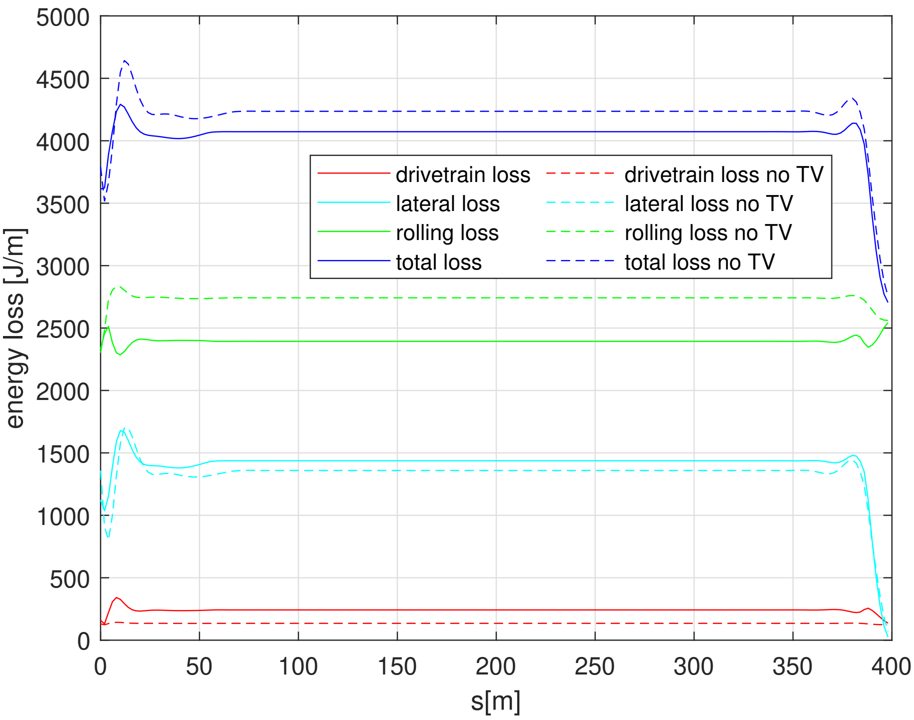


Figure 3.4: Energy loss during the driving simulation when with or without torque vectoring.

3.1.2 Scenarios with a range of lateral accelerations $a_y = 0 \sim 8\ \mathrm{m/s^2}$ and the path radius $r = 60\ \mathrm{m}$

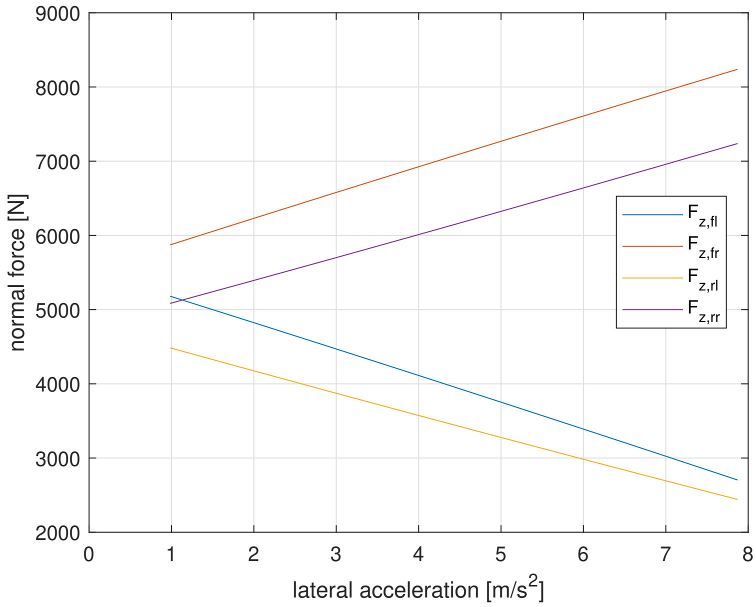


Figure 3.5: Normal forces on the four tires for different lateral accelerations.

To test the performance of the algorithm in different situations, the scenarios with a range of lateral accelerations a_y and a constant path radius $r = 60$ m are used for testing.

The Figure 3.5 shows the normal forces $F_{i,z}$ applied on the four tires for different lateral accelerations. When the lateral acceleration is higher, the difference between the four normal forces is larger. Here, the normal force $F_{i,z}$ at the outer wheels (i.e., the wheels on the right side since the vehicle is turning left) is larger than at the inner wheels. As shown in Equations (2.34) and (2.35), the rolling resistance loss is dependent on the normal forces. This gives room for the algorithm to optimize the rolling resistance loss according to different lateral accelerations.

The Figure 3.6 shows the result of the motor torque distribution when with torque vectoring(left) and without torque vectoring(right). As lateral acceleration increases, the motor torques on the outer wheels are smaller. That is, the absolute value of the negative yaw moment generated due to torque vectoring is larger.

The direction of turning is opposite to the direction of the torque vectoring yaw moment, which may not be intuitive. This can be explained as follows. According to the second term of the equation (2.34), the system can save energy by the torque distribution where the longitudinal force $F_{i,x}$ is smaller on the wheels with the higher normal force $F_{i,z}$. The other terms of the equation (2.34) are more constant and can be ignored for this scenario, since the longitudinal speed should track a constant reference speed.

This result may not be good for stability and other aspects of vehicle operation. But since this work is focused on energy optimization, this result is good in this regard.

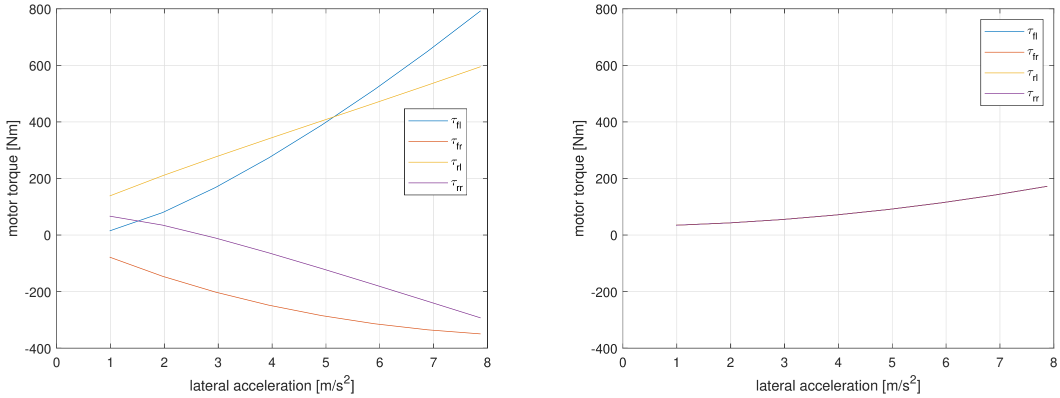


Figure 3.6: Motor torque distribution when with torque vectoring(left) and without torque vectoring(right).

Figure 3.7 shows the energy loss with torque vectoring and without torque vectoring for different lateral accelerations. When the lateral acceleration increases, the energy loss difference between with and without torque vectoring increases. The total energy loss is saved by reducing rolling resistance loss while compromising on lateral slip loss and drivetrain electric loss. Figure 3.8 shows the total energy saving during the entire driving by the algorithm's torque vectoring for different lateral accelerations. As lateral acceleration increases, torque vectoring can save energy, ranging from 0.22% to 3.85%.

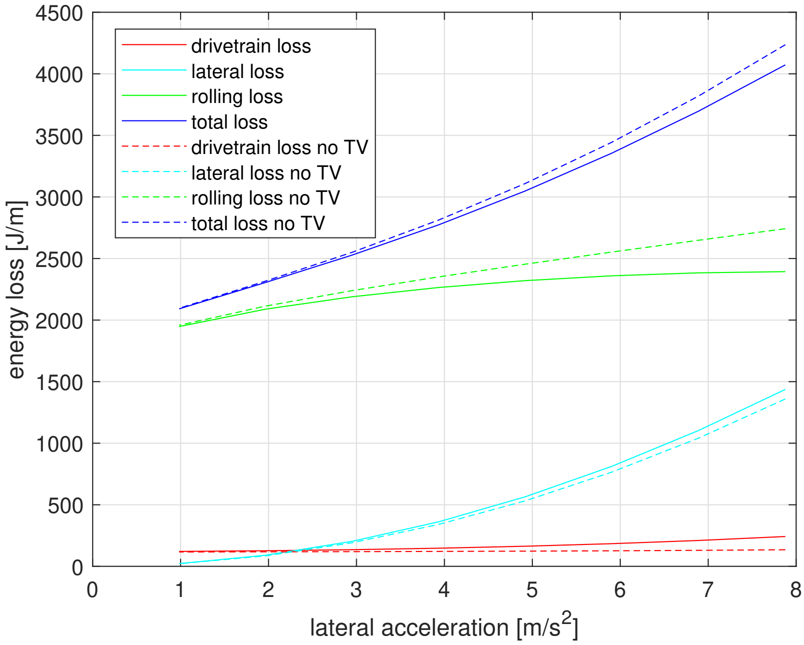


Figure 3.7: Energy loss when with torque vectoring and without torque vectoring for different lateral accelerations.

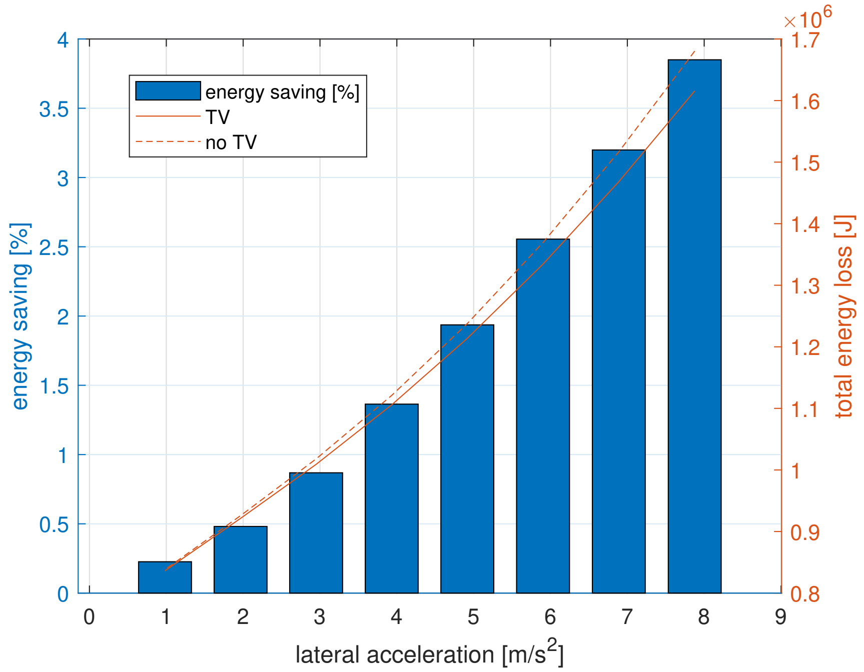


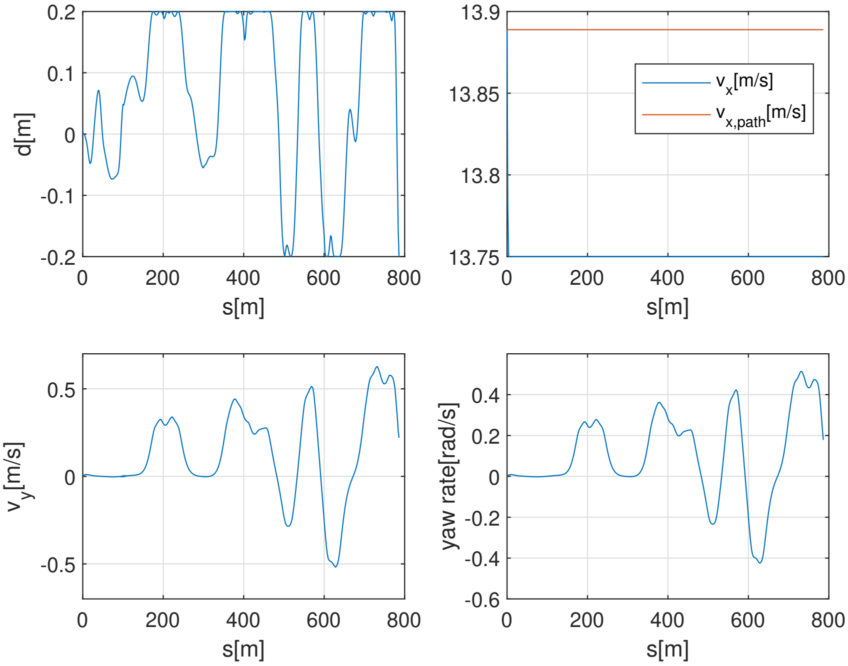
Figure 3.8: The total energy saving and energy loss during the entire driving by the algorithm's torque vectoring for different lateral accelerations.

3.2 Handling track 1 of the Volvo Hällered Proving Ground

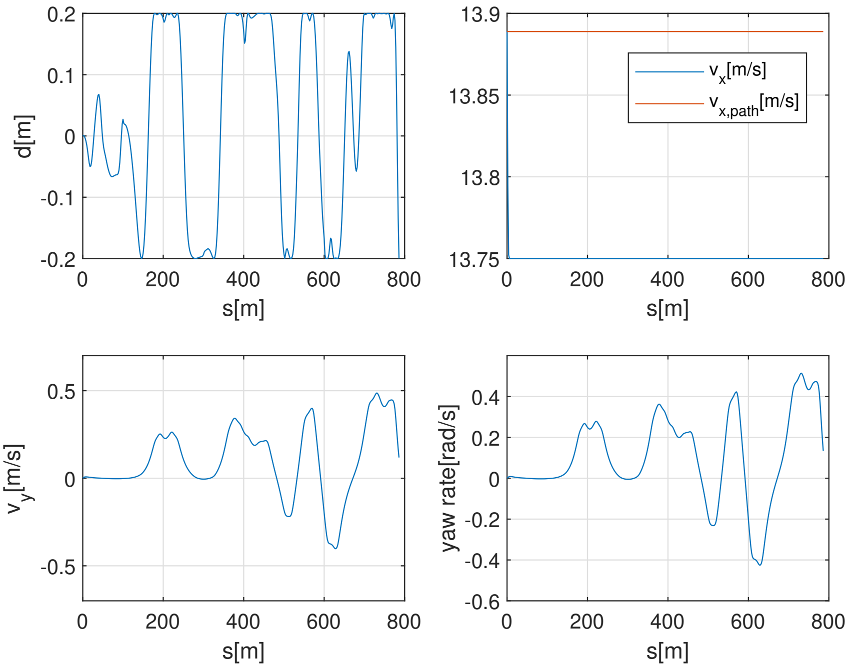
In order to test the algorithm's performance under more complicated and practical situations, handling track 1 of the Volvo Hällered Proving Ground is used as the second test scenario. The Hällered Proving Ground is a complete testing facility for vehicles located north of Borås, Sweden, owned and operated by the Volvo Car Group.

3.2.1 Simulation results

The simulation on the handling track with a constant velocity of 50 km/h is performed. Since in this scenario the reference speed is constant during driving, the speed tracking is implemented by setting a constraint on the speed state, as explained in the section 'Formulation of the optimal control problem'. Figure 3.9 shows the state trajectory of the simulation result given by the optimizer Acados. Similar to the uniform circular motion, in both cases with torque vectoring and without torque vectoring, the distance from the vehicle's CoG to the path reference d is always in the constraint range. In addition, the longitudinal velocity is always in the constraint range. Therefore, it shows that the path tracking and the velocity track are good. The yaw rate changes according to the path reference.



(a) State trajectory when with torque vectoring



(b) State trajectory when without torque vectoring

Figure 3.9: The state trajectory of the simulation result.

The Figure 3.10 shows the control trajectory of the simulation result. Similar to the uniform circular motion, the torque vectoring has a scattered torque distribution for the four tires when the vehicle is turning. This also gives rise to a yaw moment opposite to the turning direction.

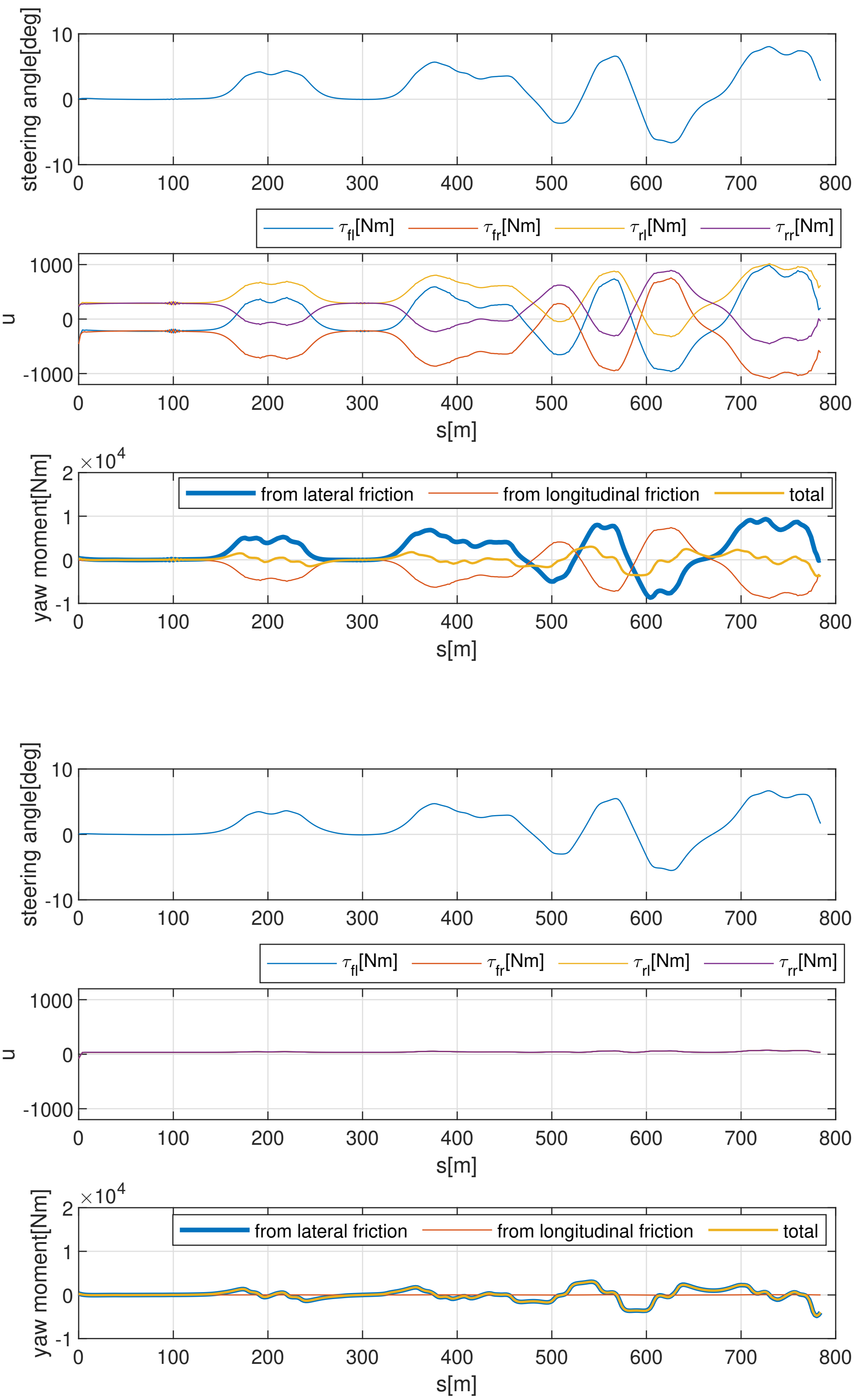


Figure 3.10: The control trajectory of the simulation result with torque vectoring (top) and without torque vectoring (bottom).

The Figure 3.11 shows the four types of energy losses in the vehicle during the simulation on the handling track with and without TV. Similar to the uniform circular motion, when the vehicle is turning, rolling resistance loss is largely saved compared to no torque vectoring, while compromising on lateral slip loss and drivetrain electric loss. As a result, the curve for the total energy loss is flattened for the turning section of the driving. The total energy loss is reduced compared to no torque vectoring.

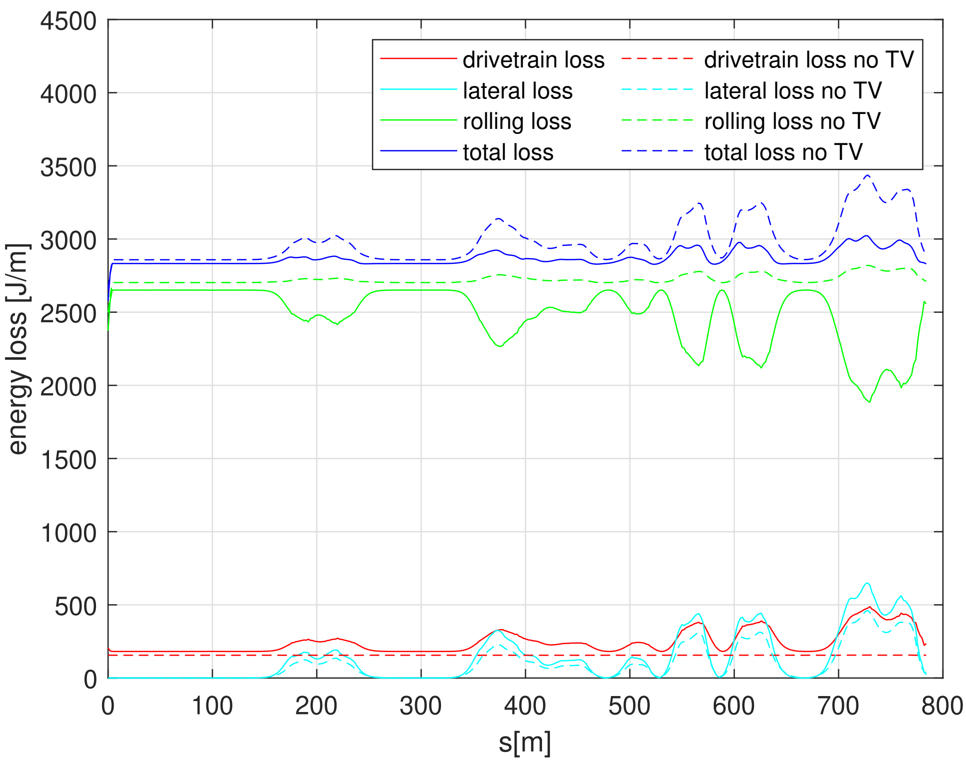


Figure 3.11: The four types of energy losses in the vehicle during the simulation on the handling track with or without TV.

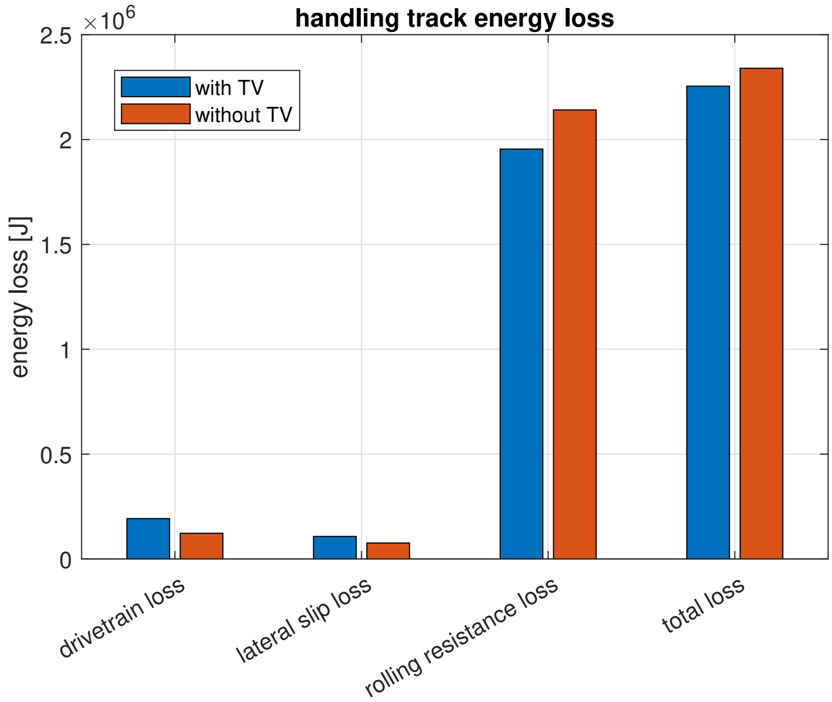


Figure 3.12: The accumulated energy loss during the whole driving when with torque vectoring and without torque vectoring.

The Figure 3.12 shows the accumulated energy loss during the entire driving when with torque vectoring and without torque vectoring. As shown previously, for torque vectoring, the drivetrain loss and the lateral slip loss is larger, while the rolling resistance loss is smaller, and the total energy loss is smaller by 3.6%.

4
Conclusion

4.1 Conclusion

In this work an offline energy-efficient controller is proposed to distribute the four independent wheel torques and front wheel steering angle of the AWD BEV while following the given path and velocity reference. The control strategy is designed based on solving an optimization problem offline in terms of power loss minimization. Path tracking is achieved by formulating the optimization problem in a constrained manner and velocity tracking can be achieved using the same method or by adding a velocity tracking error penalty term to the cost function. The latter method gives the flexibility to track velocity varying in a large range, but would somehow complicate the optimization target and thus leads to extra tuning effort to assign proper weights to different terms of the cost function in order to make the best trade-off between velocity tracking and power loss minimization.

To formulate the optimization problem, a curvature-based path reference is given first. Then the space-based nonlinear system dynamics of a two-track vehicle is derived to capture the relationship between the stretch of the path reference and the corresponding dynamics of the vehicle. The desired path tracking algorithm works by constraining the deviation from CoG of the vehicle to the path reference at all of the sample points. Furthermore, energy loss models are also derived in order to formulate the optimization problem. One can formulate the continuous optimal control problem once the dynamics of the system and the cost functions are accessed. In this work, multiple shooting method, which is a direct transcription method, is used to transfer the original OCP problem with infinite dimensions to a corresponding descritized NLP problem. RK4 method is used to approximate the nonlinear dynamics at each segment, based on which the end of one segment is enforced to match up with the start of the next. This formulated NLP problem is then solved using a software package called Acados, which internally calls SQP algorithm to solve NLP problem iteratively and use QP solvers like HPIPM to solve the subproblems introduced by SQP.

To test the performance of the proposed method, simulations are done on uniform circular motion and handing track 1 of the Volvo Hällered Proving Ground. Before the simulation, path fitting is performed to transform the path reference of the handling track represented as x and y coordinate to curvature-based path reference by formulating and solving an optimization problem. The result gives a smooth curvature and low error with the penalty term on the change rate of the curvature. The simulation result for the torque vectoring method shows that the path tracking and the velocity tracking is good, since the deviation from the vehicle to the path reference and the velocity is always within the constraint range. The torque vectoring is in effect and gives rise to a yaw moment, whose direction is opposite to the vehicle turning's direction. For the proposed method, the lateral slip loss and the drivetrain loss is larger, while the rolling resistance loss is reduced and the total energy loss is reduced. For the uniform circular motion with a lateral acceleration of 8 m/s^2 and path radius of 60 m, the total energy loss is saved by 3.85% compared to no torque vectoring. For the handling track, the the total energy loss is saved by 3.6% compared to no torque vectoring.

4.2 Future work

One of the assumptions in this work is that there is no longitudinal slip, which is not realistic. In order to capture the difference between the speed of the wheel surface and the speed of the axle with respect to the road surface, wheel dynamics need to be included. By doing so, angular velocities of all the wheels would be the four extra state variables, which complicates the system. Also, torques applied to the wheels can be directly transformed to the longitudinal tire frictions under current assumption, which need to be modified if independent wheel dynamics is included. One critical shortcoming of this work is that the evaluation of the developed controller is running with the same model as the plant model for controller design. Using a model with high fidelity from other software like IPG CarMaker, as the plant model to validate the proposed controller could be very interesting and valuable for future work.

For the real-time implementation, model predictive control is more suitable compared to the developed controller in this work. MPC could be implemented based on the current work in this thesis following two steps: the length of the control horizon should be shortened firstly, and at each horizon the control signal at the first sample point should be chosen. By iteratively moving the horizon and applying the control signal at the first sample point, model predictive control is achieved.

Bibliography

[1] Niclas Andréasson, Anton Evgrafov, and Michael Patriksson. *An introduction to continuous optimization: foundations and fundamental algorithms.* Courier Dover Publications, 2020.

[2] Giovanni De Filippis et al. “Energy-Efficient Torque-Vectoring Control of Electric Vehicles With Multiple Drivetrains”. In: *IEEE Transactions on Vehicular Technology* 67.6 (2018), pp. 4702–4715.

[3] Gianluca Frison and Moritz Diehl. *HPIPM: a high-performance quadratic programming framework for model predictive control.* 2020.

[4] Yiqi Gao et al. “Spatial Predictive Control for Agile Semi-Autonomous Ground Vehicles”. In: 2012.

[5] Matthew P Kelly. “Transcription methods for trajectory optimization: a beginners tutorial”. In: *arXiv preprint arXiv:1707.00284* (2017).

[6] H. W. Kuhn and A. W. Tucker. “Nonlinear programming”. In: *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 1950.* University of California Press, Berkeley-Los Angeles, Calif., 1951, pp. 481–492.

[7] B. Lenzo et al. “Torque Distribution Strategies for Energy-Efficient Electric Vehicles with Multiple Drivetrains”. English. In: *Journal of Dynamic Systems, Measurement and Control* 139.12 (Aug. 2017). ISSN: 0022-0434.

[8] Pedro F. Lima et al. “Spatial Model Predictive Control for Smooth and Accurate Steering of an Autonomous Truck”. In: *IEEE Transactions on Intelligent Vehicles* 2.4 (2017), pp. 238–250.

[9] Hans B. Pacejka. “Tire and Vehicle Dynamics”. In: 1982.

[10] Luıs Tiago de Freixo Ramos Paiva. “Numerical methodes for optimal control and model predictive control”. PhD thesis. Universidade do Porto (Portugal), 2014.

[11] Hyungchai Park. “Optimal tire force allocation for autonomous trajectory tracking”. PhD thesis. Stanford University, 2017.

[12] Hyungchai Park and J. Christian Gerdes. “Optimal tire force allocation for trajectory tracking with an over-actuated vehicle”. In: *2015 IEEE Intelligent Vehicles Symposium (IV).* 2015, pp. 1032–1037.

[13] Avanish Raj and Dhananjay Yadav. “Optimal torque split strategy for BEV power train considering thermal effects”. In: (2021).

[14] Manish Suvarna and Anand Prabhu. “Numerical optimization methods for improving energy efficiency in Battery Electric Vehicles using wheel torques and steering angles”. In: (2021).

[15] Juliette Torinsson et al. “Energy reduction by power loss minimisation through wheel torque allocation in electric vehicles: a simulation-based approach”. In: *Vehicle System Dynamics* 60.5 (2022), pp. 1488–1511.

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

AVANCEZ

1829

CHALMERS
UNIVERSITY OF TECHNOLOGY