Control of An Autonomous Motorbike

Tiago Azevedo Marcelo

Dissertation submitted for obtaining the master degree of

Electrical and Computer Engineering

Juri committee

President: Prof. Carlos Filipe Gomes Bispo
Advisor: Prof. João Fernando Cardoso Silva Sequeira
Vowel: Prof. Bertinho Manuel D’Andrade da Costa

December 2012
Acknowledgments

First of all, I would like to express my gratitude and thank to my supervisor in Instituto Superior Técnico, Professor João Silva Sequeira, for the support of my study and research, for his patience, motivation, enthusiasm and knowledge.

Besides my supervisor, I would like to thank to my colleagues José Sousa and Bruno Tibério for their support during the developing of this thesis. I would also thank to my colleague Tiago Milhano, of Space Engineering, for providing the Model Predictive Control Toolbox from Delft University of Technology.
Abstract

The work developed in this thesis aims at developing a control system for an autonomous motorbike using Model Predictive Control (MPC) and Proportional Integral Derivative (PID) control strategies.

The mathematical modelling of the motorbike is based on kinematics and dynamics and is inspired in the literature. These non linear models were linearized to use with a MPC toolbox developed in TU Delft. The linearized models were then tested to determine which of the model is more suitable to be implemented in a MPC approach.

The thesis also reviews the racing problem. The racing problem addresses the time it takes to do one lap of a closed circuit. Though the racing problem essence is that of a time optimal control problem, in this thesis the perspective is that of verifying the time it takes for a motorbike to lap the circuit when different models and references are used.

The proposed control system was validated by two simulations in a closed circuit, one using the kinematics of the motorbike as reference model and the other using the dynamics model. The results obtained are compatible with those of a real motorbike driven by a human rider.

Keywords

Autonomous Motorbike, Dynamics, Kinematics, Model Predictive Control, Racing Problem.
Resumo

O trabalho desenvolvido nesta tese visa desenvolver um sistema de controlo para uma mota autónoma baseado em estratégias de controlo como MPC e PID.

O modelo matemático da mota é inspirado na literatura e tem por base modelos cinemáticos e dinâmicos. Os modelos não lineares foram linearizados para poderem ser implementados numa MPC toolbox desenvolvida em TU Delft. Estes foram depois testados para determinar o modelo mais adequado a ser implementado no MPC.

Esta tese também analisa o racing problem. O racing problem aborda o tempo que se leva para fazer um determinado percurso, normalmente uma volta a um circuito fechado. Embora a essência do racing problem se resuma a um problema de controlo de tempo ótimo, a perspectiva nesta tese é verificar o tempo necessário para uma mota realizar uma volta à pista quando são usados diferentes modelos e referências.

O sistema de controlo proposto foi validado através de duas simulações, uma utilizando a cinemática da mota como modelo de referência e o outro usando o modelo da dinâmica. Os resultados obtidos são compatíveis com os obtidos com uma mota real num circuito fechado conhecido.

Palavras Chave

Cinemática, Dinâmica, Modelo de Controlo Preditivo, Mota Autónoma, Racing Problem.
Contents

1 Introduction 1
   1.1 Motivation .................................................. 2
   1.2 State of The Art ............................................. 3
   1.3 Objectives .................................................. 5
   1.4 Thesis Structure ............................................ 5
   1.5 Contribution of This Work ................................. 5

2 Model Predictive Control 7
   2.1 Process Model and Disturbance Model ..................... 8
   2.2 Performance Index .................. .......................... 9
   2.3 Prediction .................................................. 10
   2.4 Constraints ................................................ 10
   2.5 Optimization .............................................. 11
   2.6 Receding Horizon Principle ............................... 11
   2.7 Tuning ..................................................... 12
   2.8 The Standard Predictive Control Problem ................. 12
      2.8.1 Unconstrained standard predictive control problem .... 13
      2.8.2 Equality constrained standard predictive control problem .... 14
      2.8.3 Full constrained standard predictive control problem .... 14
   2.9 Application Example: MPC for an elevator .............. 16

3 Kinematics 19
   3.1 Linearization ............................................. 20
      3.1.1 Linearization of the kinematics model ................ 21
      3.1.2 Linear model for the motorbike trajectory tracking .... 21
   3.2 Model Validation of the Linearization .................... 23
      3.2.1 Linear kinematics model ................................ 24
      3.2.2 Motorbike trajectory tracking .......................... 25
   3.3 Representation of the Kinematic Model for MPC .......... 27
      3.3.1 Straight line trajectory ................................ 28
      3.3.2 Circular trajectory ..................................... 29
      3.3.3 S shaped trajectory ................................... 30
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Basic MPC design</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Receding Horizon Principle (RHP)</td>
<td>11</td>
</tr>
<tr>
<td>2.3</td>
<td>MPC elevator scheme</td>
<td>17</td>
</tr>
<tr>
<td>2.4</td>
<td>Simulation result of MPC elevator</td>
<td>17</td>
</tr>
<tr>
<td>3.1</td>
<td>Motorbike system geometry (adapted from [1])</td>
<td>20</td>
</tr>
<tr>
<td>3.2</td>
<td>Trajectory tracking</td>
<td>22</td>
</tr>
<tr>
<td>3.3</td>
<td>Simulink model of the motorbike kinematics</td>
<td>23</td>
</tr>
<tr>
<td>3.4</td>
<td>Kinematics Linearization - Orientation of the motorbike for a straight line trajectory</td>
<td>24</td>
</tr>
<tr>
<td>3.5</td>
<td>Kinematics Linearization - Orientation of the motorbike for a circular trajectory</td>
<td>25</td>
</tr>
<tr>
<td>3.6</td>
<td>Kinematics Linearization - Orientation of the motorbike for an S shaped trajectory</td>
<td>25</td>
</tr>
<tr>
<td>3.7</td>
<td>Trajectory Tracking Linearization - Orientation of the motorbike for a straight line trajectory</td>
<td>26</td>
</tr>
<tr>
<td>3.8</td>
<td>Trajectory Tracking Linearization - Orientation of the motorbike in a circular trajectory</td>
<td>26</td>
</tr>
<tr>
<td>3.9</td>
<td>Trajectory Tracking Linearization - Orientation of the motorbike for an S shaped trajectory</td>
<td>27</td>
</tr>
<tr>
<td>3.10</td>
<td>Schematic of the MPC implementation of the motorbike kinematics</td>
<td>28</td>
</tr>
<tr>
<td>3.11</td>
<td>Different initial conditions for a straight line trajectory</td>
<td>28</td>
</tr>
<tr>
<td>3.12</td>
<td>Different MPC tunings for a straight line trajectory</td>
<td>29</td>
</tr>
<tr>
<td>3.13</td>
<td>Different initial conditions for a circular trajectory</td>
<td>29</td>
</tr>
<tr>
<td>3.14</td>
<td>Different MPC tunings for a circular trajectory</td>
<td>30</td>
</tr>
<tr>
<td>3.15</td>
<td>Different initial conditions for an S shaped trajectory</td>
<td>30</td>
</tr>
<tr>
<td>3.16</td>
<td>Different MPC tunings for an S shaped trajectory</td>
<td>31</td>
</tr>
<tr>
<td>4.1</td>
<td>Right footpegs force scheme</td>
<td>36</td>
</tr>
<tr>
<td>4.2</td>
<td>Steering model validation for $v = 0$ km/h</td>
<td>37</td>
</tr>
<tr>
<td>4.3</td>
<td>Steering model validation for $v = 100$ km/h</td>
<td>37</td>
</tr>
<tr>
<td>4.4</td>
<td>Influence of the velocity in the roll dynamics</td>
<td>38</td>
</tr>
<tr>
<td>4.5</td>
<td>Longitudinal dynamic model validation for $v = 0km/h$</td>
<td>38</td>
</tr>
<tr>
<td>4.6</td>
<td>Longitudinal dynamic model validation for $v = 100km/h$</td>
<td>39</td>
</tr>
<tr>
<td>4.7</td>
<td>Influence of the torque engine in the longitudinal dynamics</td>
<td>39</td>
</tr>
<tr>
<td>4.8</td>
<td>Footpegs model validation for $v = 0km/h$</td>
<td>40</td>
</tr>
</tbody>
</table>
4.9 Footpegs model validation for \( v = 100 \text{km/h} \) .............................................. 40
4.10 Influence of the feet force in the roll dynamics. ...................................................... 41
4.11 Schematic of the MPC implementation of the motorbike dynamics. ...................... 41
4.12 MPC simulation for a constant leaning reference of 20° ........................................ 42
4.13 MPC simulation for a right turn with maximum inclination of 50° ......................... 43
4.14 MPC simulation for an exit of a right turn with \( \delta = -0.1° \) .......................... 44
4.15 MPC simulation for an exit of a right turn with \( \delta = -9.5° \) ......................... 44
4.16 MPC simulation for a constant leaning reference of 20° .................................. 45
4.17 MPC simulation for a right turn with maximum inclination of 50° .................. 46
4.18 MPC simulation for an exit of a right turn ..................................................... 46
4.19 Estoril circuit .................................................. 50
4.20 Image at scale of Estoril circuit ................................................................. 50
4.21 Processed image of the Estoril circuit .......................................................... 51
4.22 Estoril circuit sections .................................................. 51
5.1 Speed by time curve ................................................................. 52
5.2 Estoril racetrack for the kinematics simulation - Trajectories ............................. 53
5.3 Kinematic simulation results - Orientations .................................................... 53
5.4 Simulation diagram ................................................................. 54
5.5 Simulation diagram for the first routine ......................................................... 55
5.6 Simulation diagram for the second routine ..................................................... 55
5.7 Estoril racetrack for the dynamics simulation - Trajectories ............................. 56
5.8 Orientation results for the dynamics simulation - Orientations .......................... 56
5.9 Inclination results for the dynamics simulation - Inclinations ........................... 57
5.10 Optimizer PID gains ............................................................... 61
5.11 Optimizer PID gains ............................................................... 61
List of Tables

B.1 Kinematics simulation data. ............................................. B-2

C.1 Dynamics simulation data. ............................................. C-2
Acronyms

**MPC**  Model Predictive Control

**PID**  Proportional Integral Derivative

**GPC**  Generalized Predictive Control

**IO**  Input-Output models

**DIO**  Discrete Input-Output models

**IIO**  Increment Input-Output models

**ZMWN**  Zero Mean White Noise

**LQPC**  Linear Quadratic Predictive Control

**SPCP**  Standard Predictive Control Problem

**RHP**  Receding Horizon Principle
List of Symbols

Greek symbols

Γ - Diagonal selection matrix.
δ - Steering angle.
Δ(t) - Time step.
λ - Weighting on the control signal.
ρ - Atmospheric air density.
σ - Curvature of the path.
ϕ - Leaning inclination angle.
ψ - Orientation angle.
ψ_{ref} - Reference orientation angle.

Roman symbols

A_{vu} - Front projection area of the motorbike.
A_i, K_i, L_i, B_i, C_i, D_H, D_F - System matrices of IIIO model.
b - Distance between the mass center of the motorbike and the x origin.
c - Drag factor.
d_{dp} - Length of the footpegs.
F_e - Longitudinal force generated by the engine.
F_b - Longitudinal force generated by the brakes.
F_p - Feet force.
g - Gravitational acceleration.
gr - Transmission ratio of the motorbike gear.
h - Height from the center of mass to the contact point between the rear wheel and the floor.
K - Approximation gain of ϕ.
m - Mass of the motorbike.
N - Prediction horizon.
N_c - Control horizon.
N_e - Engine Torque.
N_m - Minimum cost horizon.
$P(q)$ - Tracking Filter.
$R_r$ - Rear wheel radius.
$t$ - Time instant.
$V$ - Constant velocity.
$V_{ref}$ - Constant reference velocity.
$v$ - Forward velocity.
$W_u,W_y$ - Weighting parameters.
$w$ - Wheelbase of the motorbike.
$x,y$ - Cartesian coordinates that indicates the motorbike position in the world frame.

**Subscripts**

$\dot{x}$ - Movement along $x$ axis.
$\dot{y}$ - Movement along $y$ axis.
$\dot{\psi}$ - Vertical rotation movement.
$\dot{v}$ - Velocity derivative.
$\dot{\phi}$ - Leaning velocity.
$\ddot{\phi}$ - Leaning acceleration.
Introduction

Contents

1.1 Motivation .................................................. 2
1.2 State of The Art ............................................. 3
1.3 Objectives .................................................. 5
1.4 Thesis Structure ............................................. 5
1.5 Contribution of This Work ................................. 5
1. Introduction

The study of autonomous two-wheel vehicles has been object of interest, from a view point of modeling and controllability, since the last years of the 19th century. From a theoretical perspective, the unstable equilibrium inherent to two-wheel vehicles makes the control more complex than four-wheel vehicles, and thus more challenging. The history of motorcycles began in the second half of the 19th century and soon became one of the most important ways of transportation. Motorbikes are one of the most approachable forms of motorized transport due to its agility and deployments, particularly for off-road environments such as deserts and mountains. Furthermore, the light weight of motorcycles, the high energy efficiency and the fast acceleration makes them unique for high performance competitions, such as racing.

In the literature there are multiple models that characterize the kinematics and the dynamics of a motorbike, some more complex than others (see for instance [1]).

The work in this thesis aims at developing a control strategy capable of reproducing the behavior of a racing driver while following a reference trajectory. The kinematic model used was developed in a previous MSc thesis [2]. In the literature, among the different proposals of control strategies, Model Predictive Control (MPC) is commonly referred.

1.1 Motivation

Control of a motorbike is a rich problem offering a number of considerable challenges of current research and interest in the areas of mechanism and robot control. The underlying idea to control a motorbike is to include more effects related to the kinematics and dynamics of the motorbike. This results in more information available to the controller, and so eventually is easier to achieve a control strategy for higher performance driving.

Under constant conditions, for instance, while moving in a straight line or in a curve of constant curvature, it is relatively simple to get a control strategy for a motorbike using for example a Proportional Integral Derivative (PID) controller. To control a motorbike that makes, for example, a transition from a right turn to left turn, the PID control strategy is in general not optimal due to the need to adjust consecutively the controller gains between situations.

The purpose of this thesis is to design a control system for an autonomous motorbike moving around a closed circuit. The control strategy will be based on MPC. In addition some linear control laws from robot control are used. The kinematic model was obtained from a basic bicycle structure moving on a horizontal plane, and the dynamic model was based on the roll dynamics of an inverted pendulum.
1.2 State of The Art

Motorbikes share many similar properties with bicycles. In fact, motorcycles come from the “safety bicycle”. During the early 20th century, several authors studied the problems of bicycle stability and steering, from the viewpoint of control. Different types of bicycles have different dynamics and these play a role in how a bicycle performs in given conditions and how the control is made. Åström et al, in [3] analyzes the dynamic of adapted bicycles from the perspective of control for education and research, using models of different complexities. The models considered are models that capture essential behaviors such as self-stabilization as well models that demonstrate difficulties with rear wheel steering. Other interesting properties are explored such as gyroscopic effects which have an important role in stabilization. In conclusion, this research shows that the front fork creates feedback, which under certain circumstances stabilizes a bicycle and that rear-steered bicycles are difficult to ride.

The physics involved in driving a bicycle are described in [4]. A motorbike, however, is more complicated because of suspensions, chain pull effect, tires profiles, damping system, and others. Unlike a four-wheel vehicle driver who focuses primarily on control of the steering system to make changes in direction, a motorbike driver can do these changes either by direct control of the steering system or by forces and torques applied by the body’s driver to the seat, footpegs or by body movement in order to lean the motorbike. A number of research works are focused on understanding the use made of these options in the control process of motorbikes. Weir was one of the first authors to study in some detail the control procedures of a motorbike (see for instance [5]).

There are a few studies in the literature on control of autonomous motorbikes using approaches like trajectory tracking and proportional integral controllers. Getz et al. [6] present the first controller allowing tracking of arbitrary trajectories while maintaining the equilibrium. The paper starts to derive a reduced set of equations of motion for the bicycle, disregarding the unstable roll-angle, to show how the bicycle can track a desired trajectory. Finally the authors show how to steer the kinematic bicycle, based on their knowledge on control, in order to construct a controller that allows a leaning bicycle to track trajectories. Despite the good results of the simulation, the practicality of the controller is judged on the realizability of the control gains. On the other hand, the choice of the desired trajectory affects the results due to the limitations of the model achieved, and the fact that the nonholonomic constraints are only an approximation to the actual tire/road interaction. The authors conclude that control of the bicycle is complicated by the nonholonomic constraints on the vehicle as well as the need to track a path while trying to keep the balance.

Jingang et al. [7] present a trajectory tracking control algorithm for an autonomous motorcycle. The dynamic model of the motorcycle is based on existing work in modeling a bicycle with additional extended features such as the steering effect on self-stabilization. The trajectory tracking and self-balancing controller is developed based on a non linear control approach. The desired reference trajectory is generated using global positioning system. The motion planning is achieved by the fused global positioning system and on-board computer vision system information. The simulation results show that the proposed autonomous motorcycle navigation and control system can work well on an
1. Introduction

Vittori \[2\] proposed to achieve a realistic motorbike model and develop an automatic control for an autonomous motorbike. The kinematic model started from a basic bike structure including some features such as the suspension model and the pneumatic effect of the tire. Also, other alternative structures were explored and studied such as double swing arm. The dynamical model was built using the classical free Lagrangian method based on total kinetic energy, and the external forces that constraint the free Lagrangian. The simulation of the model consists in following a particular path obtained from telemetry data of a real racing motorbike, and it must analyze if the control strategy is able to make the motorbike follow the racetrack path by actuating on the control variables (steering angle and torque). This work incorporates additional features in the kinematic model. The results obtained suggest that the model is valid since the motorbike follows the reference path even using a simple control strategy.

In the last decades, model predictive control has shown to be able to respond in an effective way in many practical process control applications. The ideas of receding horizon and model predictive control can be traced back to the 1960s (by the work of Garcia et al. \[8\]). Since 1970 various techniques have been developed for the design of model based control systems for robust multivariable control of industrial process, but the interest on this field started to surge only in 1980s. In 1987, the first comprehensive exposition of Generalized Predictive Control (GPC) was introduced by Clarke et al. \[9\].

The concept of autonomous motorcycles has been proposed by several researchers. In fact, there are several attempts to develop autopilots for bicycles and motorcycles. Some of these attempts are based on model predictive control, due to the fact that MPC techniques can be applied for multivariable constrained control problems such as the dynamics of a motorbike. Other reasons for the use of MPC as a control strategy to control an autonomous motorbike are stated in Chapter 2.

In the literature, there are a few studies involving the use of other control techniques such as Steven et al. \[10\], present a nonlinear driver steering controller using multiple linearized models of the vehicle dynamics and model predictive control theory. A driver steering controller uses central nervous system internal model concepts from neuroscience. The steering controller uses multiple linearized models of the nonlinear vehicle dynamics to generate a steering controller for each linearized model. The switching is done by choosing the best steering controller, available from a database of controllers, which adjusts better to the situation. Thus, the driver steering controller model developed uses those multiple linearized models to estimate/predict the future vehicle trajectory, and then determines an appropriate steering command intended to minimize the path following error. The positive results of this work indicate that in the future there is a substantial improvement to be made in understanding and modeling driver skill based on the internal model concepts.

Frezza et al. \[11\] presents a control strategy for driving a motorcycle along a given path. The objective on their work is more controlling the motorbike than modeling the human pilot. The proposed solution, as control strategy, is based on model predictive control. In order to avoid the problem due to the instability of the roll angle dynamics, the angle is actuated as if it was an input with the steering
1.3 Objectives

The state of the art in Popov et al. [12] addresses steering control in motorcycles and rider models. The review also covers research findings in road preview control. In this article an emphasis is made on recent applications of model predictive control to assess motorcycle performance and driver/rider performance. The works of Frezza et al. [11], Prokop [13], Cole et al. [14] and Rowell et al. [15] [16] addressed the issue of applying model predictive control strategy to motorcycle control. The review concludes with a scope for further study researches.

1.3 Objectives

The thesis aims at developing a control system for an autonomous motorbike moving around a closed circuit using MPC as the main control strategy.

1.4 Thesis Structure

The thesis is structured in the follow order, in Chapter 2, the model predictive control and the theoretical background behind model predictive control is summarily explained. In this Chapter, also, a predictive controller design problem is illustrated. In Chapter 3 are described two methods for the linearization of the motorbike kinematics model. In Chapter 4 is presented three dynamic models for the roll dynamics of a motorbike. In Chapter 5 is explained in what consists the Racing Problem and some implementations details are revealed. Also in this Chapter, the results of the simulations are presented, illustrated, and discussed. Chapter 6 concludes with a summary of the thesis and future work suggestions.

1.5 Contribution of This Work

The kinematics and dynamics equations of a motorbike are described as a system of non linear equations. In this research, was developed a control system for an autonomous motorbike moving around a closed circuit using MPC as the main control strategy. Given the non linear nature of the problem, this work presents a comparison and development of the linear kinematic and dynamic model of the motorbike in order to be applied in a MPC control strategy. The racing problem is also studied.
1. Introduction
2 Model Predictive Control

Contents

2.1 Process Model and Disturbance Model .................................................. 8
2.2 Performance Index .................................................................................... 9
2.3 Prediction .................................................................................................. 10
2.4 Constraints ............................................................................................... 10
2.5 Optimization ............................................................................................. 11
2.6 Receding Horizon Principle ...................................................................... 11
2.7 Tuning ....................................................................................................... 12
2.8 The Standard Predictive Control Problem .............................................. 12
2.9 Application Example: MPC for an elevator ............................................. 16
2. Model Predictive Control

Model Predictive Control “is more of a methodology than a single technique” [17]. The main objective of MPC is to solve a control problem in a limited time interval. To achieve this goal, MPC selects the control inputs to minimize a cost function. The cost function is computed through current values of the output of the process model and through predicted values of the disturbance model.

MPC controllers provide adequate control for complex systems with large time delays and high-order dynamics. In certain conditions MPC can emulate a human rider. This is one of the reasons to use MPC for controlling an autonomous motorbike. For example, for a sudden transition, like right turn to left turn, the dynamics of the motorbike change very quickly and it is important to have a good control strategy to handle with this type of characteristic. Furthermore, another reason to use MPC is its robustness to uncertainty and noise, which ensures stability.

There are several ways to implement a MPC strategy. However, all methods include five important items that make part of the design: Process Model and Disturbance Model, Performance Index, Constraints, Optimization, and Receding Horizon Principle. Other important items are the Tuning of the MPC and Prediction, which play an important role in the outcome. In the following sections these items will be presented.

2.1 Process Model and Disturbance Model

The models used in MPC are usually intended to represent the behavior of complex dynamical systems, and to predict the behavior of future outputs of the process, based on inputs and known disturbances applied to the process in the past. The process model or plant model is used to calculate the input signal to the process that minimizes a specific cost function. The disturbance model is used to model noises and uncertainties that the system is subject.

MPC models predict the change in the dependent variables of the modeled system that will be caused by changes in the independent variables. The dependent variables are the plant outputs which the controller uses to estimate unmeasured quantities as feedback. The independent variables are the plant inputs that affect the process. The controller has to adjust these variables in order to minimize the reference tracking error (Figure 2.1).

![Figure 2.1: Basic MPC design.](image)

The models used to represent the process model and the disturbance model don’t have to be the same. It is usual to use the same model to represent the process model and the disturbance model.
2.2 Performance Index

If they are different it has to be assured that they have a similar behavior in order to ensure good predictions.

The models used describe the input-output behavior of the process and are called Input-Output models (IO). There are two types of IO models that can be applied to MPC: Discrete Input-Output models (DIO) and Increment Input-Output models (IIO). In DIO models the input signal is directly applied to the model. On the other hand, in the IIO models, increments of the input signal are applied to the model instead of applying the input signal directly. In this thesis an IIO model is used to represent the process model and the disturbance model. The main reason of using this type of model is the good steady-state behavior that this type of controller offers, as shown in [17].

Also, in [17] the IIO model is given in a state space representation by

\[
x_{i}(k+1) = A_{i}x_{i}(k) + K_{i}e_{i}(k) + L_{i}d_{i}(k) + B_{i}\Delta u(k)
\]

\[
y(k) = C_{i}x_{i}(k) + H_{i}e_{i}(k) + F_{i}d_{i}(k)
\]

where:
- \(x_{i}(k)\) - is the vector with the state variables;
- \(e_{i}(k)\) - is the Zero Mean White Noise (ZMWN);
- \(d_{i}(k)\) - is the known disturbance signal;
- \(\Delta u(k)\) - is the input increment signal (independent variables);
- \(y(k)\) - is the process output (dependent variables).

2.2 Performance Index

The performance index or cost function is a method to measure the reference tracking error and the control action signal.

As shown in [17], there are three performance index used in MPC: Generalized Predictive Control Performance Index, Linear Quadratic Predictive Control (LQPC) Performance Index, and Zone Performance Index.

In this thesis, the GPC performance index will be used.

According to Boom, [17] the GPC performance index can be described by

\[
J(u,k) = \sum_{j=1}^{Nc} (\Delta u(k+j-1 \mid k))^T \lambda \Delta u(k+j-1 \mid k) + \lambda^2 \sum_{j=1}^{Nm} (\hat{y}_p(k+j \mid k) - y(k+j))^T (\hat{y}_p(k+j \mid k) - y(k+j))
\]

where:
- \(\hat{y}_p(k) = P(q)y(k)\) - is the weighted process output signal;
- \(y(k)\) - is the reference signal;
- \(\Delta u(k)\) - is the process control increment signal;
- \(Nm\) - is the minimum cost horizon;
- \(N\) - is the prediction horizon;
- \(Nc\) - is the control horizon;
- \(\lambda\) - is the weighting on the control signal;
- \(P(q) = 1 + p_1q^{-1} + ... + p_nq^{-n}\) - is a polynomial with desired closed-loops poles.
2. Model Predictive Control

\( \hat{y}_p(k+j | k) \) is the estimation of the prediction of \( y_p(k+j) \), based on the knowledge up to time \( k \). \( \lambda \) determines the trade-off between tracking accuracy and computational effort.

The GPC performance index can be translated into a Standard Predictive Control Problem (SPCP) performance index by choosing

\[
\hat{z}(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = \begin{bmatrix} y_p(k+1) - r(k+1) \\ \lambda \Delta u(k) \end{bmatrix}
\]

Then, the SPCP performance index becomes in 2.5

\[
J(u,k) = \sum_{j=N_m}^{N} \hat{z}_1^T(k) \hat{z}_1(k+j | k) + \sum_{j=1}^{N} \hat{z}_2^T(k+j-1 | k) \hat{z}_2(k+j-1 | k)
\]

where \( \hat{z}_i(k+j | k), i = 1,2 \) is the prediction of \( z_i(k+j) \) at time \( k \).

2.3 Prediction

The purpose of prediction is to find and estimate a prediction signal of the future output of the process. Considering the state-space representation of the standard model

\[
x(k+1) = Ax(k) + Ke(k) + Ld(k) + Bu(k)
\]

\[
y(k) = C_1 x(k) + D_{11} e(k) + D_{12} d(k)
\]

\[
p(k) = C_p x(k) + D_{p1} e(k) + D_{p2} d(k) + D_{p3} u(k)
\]

where \( p(k) \) is the prediction signal.

At each time instant \( k \), all signals are considered over a finite horizon \( N \). This is done by making predictions at time \( k \) of the signal \( p(k+j) \) for \( j = 1, \ldots, N \). These predictions are denoted by \( \hat{p}(k+j|k) \). The prediction \( \hat{p}(k+j|k) \) of the signal \( p(k+j) \) is based on the knowledge at time \( k \) and the future values of the control signal \( u(k|k), u(k+1|k), \ldots, u(k+N-1|k) \). At time instant \( k \) the signal vector \( \hat{p}(k) \) is defined with the predicted estimate signals \( \hat{p}(k+j|k), j = 1, \ldots, N \).

Thus the prediction law is given by

\[
\hat{p}(k) = \hat{p}_0(k) + \hat{D}_{p3} \hat{u}(k)
\]

where \( \hat{p}_0(k) \) is the predicted output signal when the future input signal is set to zero \( (u(k+j|k) = 0 \) for \( j \geq 0 \).

2.4 Constraints

Many systems are subject to constraints resulting from dynamic limitations and safety limitations. So, it is important to specify boundaries for specific signals that cannot be exceeded. Therefore, MPC modifies the optimal unconstrained solution so that the constraints are verified. This can be done using optimization techniques like linear programming or quadratic programming techniques.
the constraints are mapped into bounds on the control signal, state signals or output signals through the following inequalities:

\[
\Delta u_{\text{min}} \leq \Delta u(k) \leq \Delta u_{\text{max}} \quad (2.8)
\]
\[
y_{\text{min}} \leq y(k) \leq y_{\text{max}} \quad (2.9)
\]
\[
x_{\text{min}} \leq x(k) \leq x_{\text{max}} \quad (2.10)
\]

2.5 Optimization

MPC uses an optimization algorithm to compute the future control signal sequence that minimizes the performance index \( J(u, k) \), subject to given constraints. The optimization can be done with multiple programming techniques such as linear and quadratic programming techniques. Due to the nature of the models in this thesis (linear models) and the use of a quadratic (2-norm) performance index, quadratic programming algorithms are used here.

2.6 Receding Horizon Principle

MPC uses receding horizon principle, that is, at time \( k \), only the first input of the optimal control signal is implemented. After that, all the other samples are discarded and the horizon is shifted one sample so that the optimization can be recomputed with new information of the measurements of the process. This iterative process repeats until \( k + p = N \).

Figure 2.2 illustrates the concept of receding horizon.

![Receding Horizon Principle](image)

At time \( k \), the future control sequence \( \{\Delta u(k \mid k), \ldots, \Delta u(k + N_c - 1 \mid k)\} \) is computed and optimized such that the performance index \( J(u, k) \) is minimized subject to specific constraints.
2. Model Predictive Control

2.7 Tuning

In Model Predictive Control (MPC) method it is important to tune properly the predictive controller in order to acquire good signal tracking, and achieve sufficient disturbance rejection and robustness against model mismatch. The rules-of-thumb for a Generalized Predictive Control (GPC) controller were first discussed by Clarke & Mohtadi & Tuffs [18]. For the GPC there are two categories of parameters that can be tuned: the summation parameters and the weighting parameters.

The summation parameters are the minimum cost horizon, $N_m$, the control horizon, $N_c$, and the prediction horizon, $N$. Generally it is chosen $N_m = 1$. The predictive horizon is related to the length of the step response of the process and the control horizon should be less than or equal than the predictive horizon, $N_c \leq N$. If $N_c \ll N$ the control signal tends to be smoother, which is important for stability properties of the system. Another important consequence of decreasing $N_c$ is the reduction of the computational effort, and the faster convergence of the solution.

The weighting parameters are $\lambda$ and $P(q)$. $\lambda$ is the factor that defines the weight of the control action. $P(q)$ is a tracking filter described by a polynomial where the parameters $p_i, i = 1, \ldots, n_p$ are chosen such that the roots of the polynomial are the desired poles of the closed loop. $P(q)$ is responsible for the output of the system tracks the reference signal.

2.8 The Standard Predictive Control Problem

In Section 2.2 the GPC performance index was formulated into a Standard Predictive Control Problem (SPCP) performance index (2.5). Consider a SPCP given by the state-space realization

$$
x(k + 1) = Ax(k) + Ke(k) + Ld(k) + Bu(k)$$

$$
y(k) = C_1 x(k) + D_{11} e(k) + D_{12} d(k)$$

$$
z(k) = C_2 x(k) + D_{21} e(k) + D_{22} d(k) + D_{23} u(k)$$

(2.11)

where $y$ contains the measurements so that at time $k$ all the past inputs $u(0), \ldots, u(k-1)$ and all past and current measurements $y(0), \ldots, y(k-1)$ are available. $z$ contains the variables to be controlled.

The performance index and the constrains are given by

$$
J(u, k) = \sum_{j=0}^{N-1} \bar{z}^T(k+j | k) \bar{\Gamma} \bar{z}(k+j | k) = \bar{z}^T(k) \bar{\Gamma} \bar{z}(k)
$$

(2.12)

$$
\tilde{\phi}(k) = \tilde{C}_3 x(k | k) + \tilde{D}_{31} e(k) + \tilde{D}_{32} d(k) + \tilde{D}_{33} u(k)
$$

(2.13)

$$
\tilde{\psi}(k) = \tilde{C}_4 x(k | k) + \tilde{D}_{41} e(k) + \tilde{D}_{42} d(k) + \tilde{D}_{43} u(k) \leq \tilde{\Psi}(k)
$$

(2.14)

where $\tilde{\phi}(k)$ and $\tilde{\psi}(k)$ are stacked vectors containing predicted states or future inputs on which is imposed either a equality constraint or a inequality constraint and $\bar{\Gamma}$ is the diagonal selection matrix.
2.8 The Standard Predictive Control Problem

\[ \tilde{\psi}(k) = \begin{bmatrix} \psi(k | k) \\ \psi(k + 1 | k) \\ \vdots \\ \psi(k + N_c - 1 | k) \end{bmatrix}, \quad \tilde{\phi}(k) = \begin{bmatrix} \phi(k | k) \\ \phi(k + 1 | k) \\ \vdots \\ \phi(k + N_c - 1 | k) \end{bmatrix} \]

\[ \bar{\Gamma} = \begin{bmatrix} \Gamma(0) & 0 & \ldots & 0 \\ 0 & \Gamma(1) & 0 & \ldots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \Gamma(N - 1) \end{bmatrix}, \quad \Gamma(j) = \begin{bmatrix} \Gamma_i(j) & 0 \\ 0 & I \end{bmatrix} \]

\[ \Gamma_i(j) = \begin{cases} 0 & \text{for } 0 \leq j \leq N_m - 1 \\ I & \text{for } N_m - 1 \leq j \leq N - 1 \end{cases} \]

Summarizing, the standard predictive control problem consists in minimizing (2.12) given the system (2.11) subject to (2.13) and (2.14).

2.8.1 Unconstrained standard predictive control problem

In this problem the aim is to minimize the performance index \( J(u, k) \) for the system as given in (2.11), without any equality or inequality constraints, and for a finite horizon \( N \).

The performance index, given in (2.12), is minimized for each time instant \( k \) through the use of the prediction law (stated in Section 2.3) given by

\[ \tilde{z}(k) = \tilde{C}_2 x(k | k) + \tilde{D}_{21} e(k | k) + \tilde{D}_{22} \tilde{d}(k) + \tilde{D}_{23} \tilde{u}(k) = \tilde{z}_0(k) + \tilde{D}_{23} \tilde{u}(k) \] (2.15)

where \( \tilde{z}_0(k) = \tilde{C}_2 x(k | k) + \tilde{D}_{21} e(k | k) + \tilde{D}_{22} \tilde{d}(k) \) is the free-response signal given in (2.15).

Without constraints, the problem becomes a standard least square problem, and the solution of the predictive control problem can be computed analytically. Assuming that the reference signal to track \( r(k + j) \) is known for the whole prediction horizon, \( j = 1, \ldots, N \), and substituting (2.15) in (2.12) the performance index comes

\[ J(u, k) = \tilde{u}^T(k) \tilde{D}_{23}^T \bar{\Gamma} \tilde{D}_{23} \tilde{u}(k) + 2 \tilde{u}^T(k) \tilde{D}_{23}^T \bar{\Gamma} \tilde{z}_0(k) + \tilde{z}_0^T(k) \bar{\Gamma} \tilde{z}_0(k) \]

\[ J(u, k) = \frac{1}{2} \tilde{u}^T(k) H \tilde{u}(k) + \tilde{u}^T(k) f(k) + c(k) \] (2.16)

where the matrixes \( H \), \( f \) and \( c \) are given as

\[ H = 2 \tilde{D}_{23}^T \bar{\Gamma} \tilde{D}_{23} \] (2.17)

\[ f(k) = 2 \tilde{D}_{23}^T \bar{\Gamma} \tilde{z}_0(k) \] (2.18)

\[ c(k) = \tilde{z}_0^T(k) \bar{\Gamma} \tilde{z}_0(k) \] (2.19)

So, minimizing the performance index amounts to solving a linear algebra problem.
2. Model Predictive Control

2.8.2 Equality constrained standard predictive control problem

In this problem the aim is to minimize the performance index $J(u,k)$ for the system as given in (2.11) subject to equality constraints given by (2.13), and for a finite horizon $N$. A solution to solve this problem is eliminating the equality constraint. In [17], Lemma 12, a solution for $\hat{\phi}(k) = 0$ is given by

$$
\hat{u}(k) = -\hat{D}_{33}^T (\hat{C}_3 x(k | k) + \hat{D}_{31} e(k) + \hat{D}_{32} \hat{d}(k)) + \hat{D}_{33}^T \hat{\mu}(k) 
= \hat{u}_E(k) + \hat{D}_{33}^T \hat{\mu}(k) 
$$

(2.20)

where:

$\hat{u}_E(k) = -\hat{D}_{33}^T (\hat{C}_3 x(k | k) + \hat{D}_{31} e(k) + \hat{D}_{32} \hat{d}(k));$

$\hat{D}_{32}$ - is the right-inverse matrix of $\hat{D}_{33};$

$\hat{D}_{33}^T$ - is the right-complement of $\hat{D}_{33}.$

Substituting (2.20) in (2.15) results

$$
\bar{z}(k) = \hat{C}_2 x(k | k) + \hat{D}_{21} e(k | k) + \hat{D}_{22} \hat{d}(k) + \hat{D}_{23} (\hat{u}_E(k) + \hat{D}_{33}^T \hat{\mu}(k)) = \bar{z}_E(k) + \hat{D}_{23} \hat{D}_{33}^T \hat{\mu}(k) 
$$

(2.21)

Thus the performance index becomes

$$
J(k) = \bar{z}^T(k) \bar{\Gamma} \bar{z}(k) = \frac{1}{2} \hat{\mu}^T H \hat{\mu} + f^T(k) \hat{\mu}(k) + c(k) 
$$

(2.22)

where the matrices $H$, $f$ and $c$ are now given by

$$
H = 2(\hat{D}_{33}^T)^T \hat{D}_{23}^T \bar{\Gamma} \hat{D}_{23} \hat{D}_{33}^T 
$$

(2.23)

$$
f(k) = 2(\hat{D}_{33}^T)^T \hat{D}_{23}^T \bar{\Gamma} \hat{D}_{23} \bar{z}_E(k) 
$$

(2.24)

$$
c(k) = \bar{z}_E^T(k) \bar{\Gamma} \bar{z}_E(k) 
$$

(2.25)

When the equality constraint is eliminated and the inequality constraints are absent, results in an unconstrained optimization problem which has an analytical solution, in the same way as in the previous section.

2.8.3 Full constrained standard predictive control problem

The aim is to minimize the performance index $J(u,k)$ for the system as given in (2.11) subject to an equality constraints and an inequality constraint, given by (2.13) and (2.14), respectively. For a finite horizon $N$.

Let the prediction law be $\tilde{u}(k) = \tilde{u}_E(k) + \tilde{D}_{33}^T \tilde{\mu}(k)$ (stated in Section 2.8.2), $\hat{\phi}(k)$ comes

$$
\hat{\phi}(k) = \hat{C}_3 x(k | k) + \hat{D}_{31} \tilde{e}(k) + \hat{D}_{32} \tilde{d}(k) + \hat{D}_{33} (\tilde{u}_E(k) + \hat{D}_{33}^T \hat{\mu}(k)) = 
= \hat{C}_3 x(k | k) + \hat{D}_{31} \tilde{e}(k) + \hat{D}_{32} \tilde{d}(k) + \hat{D}_{33} \tilde{u}_E(k) + \hat{D}_{33} \hat{D}_{33}^T \hat{\mu}(k) = 0 
$$

(2.26)
The equality constraint is now absent and the optimization vector can be written as

\[ \tilde{\mu}(k) = \tilde{\mu}_E(k) + \tilde{\mu}_I(k) \]  

(2.26)

where \( \tilde{\mu}_E(k) = -H^{-1}f(k) \) is the solution for the equality constraint and \( \tilde{\mu}_I(k) \) is an additional term to take the inequality constraint into account.

Substituting equation (2.26) in (2.22) the performance index comes

\[ \frac{1}{2} \tilde{\mu}^T(k)H\tilde{\mu}(k) + f^T(k)\tilde{\mu}(k) + c(k) = \]

\[ = \frac{1}{2}(-H^{-1}f(k) + \tilde{\mu}_I(k))^T\hat{H}(-H^{-1}f(k) + \tilde{\mu}_I(k)) + f^T(k)(-H^{-1}f(k) + \tilde{\mu}_I(k)) + c(k) = \]

\[ = \frac{1}{2}\tilde{\mu}_I^T(k)H\tilde{\mu}_I(k) - \frac{1}{2}f^T(k)H^{-1}f(k) + c(k) \]

Now, the optimization vector can be written in the following manner

\[ \tilde{\mu}(k) = -H^{-1}f(k) = \]

\[ = -\Xi\bar{e}_E(k) + \tilde{\mu}_I(k) = \]

\[ = -(\Xi\bar{D}_{23}\bar{D}_{33}\bar{C}_3 - \Xi\bar{C}_2)x(k \mid k) + (\Xi\bar{D}_{23}\bar{D}_{33}\bar{D}_{33} - \Xi\bar{D}_{22})\tilde{d}(k) + \tilde{\mu}_I(k) \]  

(2.27)

where \( \Xi = ((\bar{D}_{33}^{1}\bar{D}_{33}^{2}\bar{D}_{23}\bar{D}_{33}^{r})^{-1}(\bar{D}_{33}^{r})^T\bar{D}_{23}^T) \).

The prediction law now can be expressed as

\[ \tilde{u}(k) = \tilde{u}_E(k) + \tilde{D}_{33}^r\tilde{\mu}(k) = \]

\[ = \tilde{u}_E(k) + \tilde{D}_{33}^{r}\tilde{\mu}_E(k) + \tilde{D}_{33}^{r}\tilde{\mu}_I(k) \]  

(2.28)

Thus the inequality constraint is given now by

\[ \tilde{\psi}(k) = \tilde{\psi}_E(k) + \tilde{D}_{43}\tilde{D}_{33}^{r}\tilde{\mu}_I(k) \]  

(2.29)

where

\[ \tilde{\psi}_E = (\bar{C}_4 - \bar{D}_{43}\bar{D}_{33}^r\bar{C}_3)x(k \mid k) + (\bar{D}_{41} - \bar{D}_{43}\bar{D}_{33}^r\bar{D}_{33}c(k \mid k) + (\bar{D}_{42} - \bar{D}_{43}\bar{D}_{33}^r)\tilde{d}(k) + \tilde{D}_{43}\tilde{D}_{33}^{r}\tilde{\mu}_E(k) \]

Now let

\[ A_{\psi} = \tilde{D}_{43}\tilde{D}_{33}^{r} \]  

(2.30)

\[ b_{\psi}(k) = \tilde{\psi}_E(k) - \tilde{\psi}(k) \]  

(2.31)

Then the solution is obtained by solving a quadratic programming problem \( \min \frac{1}{2}\tilde{\mu}_I^T(k)H\tilde{\mu}_I(k) \) subject to \( A_{\psi}\tilde{\mu}_I(k) + b_{\psi}(k) \leq 0 \).
2. Model Predictive Control

2.9 Application Example: MPC for an elevator

The elevator model, assuming a rigid body approximation, uses a triple integrator chain and is given by

\[ y(t) = \ddot{y}(t) = u(t) + \epsilon(t) \] (2.32)

where:
- \( y(t) \) - is the position of the elevator;
- \( \ddot{y}(t) \) - is the jerk input (time derivative of acceleration);
- \( \epsilon(t) = \dot{\epsilon} + 0.5\ddot{\epsilon} + \dot{\epsilon} + 0.1\epsilon(t) \) - is the zero mean white noise model.

This model can be formulated into a state space form with state vector

\[
\begin{bmatrix}
  x_1(t) \\
  x_2(t) \\
  x_3(t)
\end{bmatrix}
= \begin{bmatrix}
  \dot{x}_1(t) \\
  \dot{x}_2(t) \\
  \dot{x}_3(t)
\end{bmatrix}
= \begin{bmatrix}
  \dot{y}(t) \\
  \dot{y}(t) \\
  y(t)
\end{bmatrix}
\]

Then, the continuous-time state space description of the system in (2.32) is given by

\[ \dot{x}_1(t) = u(t) + 0.1\epsilon(t) \] (2.33)

\[ \dot{x}_2(t) = x_1(t) + \epsilon(t) \] (2.34)

\[ \dot{x}_3(t) = x_2(t) + 0.5\epsilon(t) \] (2.35)

\[ y(t) = x_3(t) + \epsilon(t) \] (2.36)

In matrix form it comes

\[ \dot{x}(t) = A_c x(t) + K_c \epsilon(t) + B_c u(t) \] (2.37)

\[ y(t) = C_c x(t) + \epsilon(t) \] (2.38)

where:
- \( A_c = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \), \( B_c = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \), \( K_c = \begin{bmatrix} 0.1 \\ 1 \\ 0.5 \end{bmatrix} \)
- \( C_c = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \)

A zero-order hold transformation with sampling-time, \( T \), gives a discrete-time model given by

\[ x(k+1) = A_o x(k) + K_o \epsilon(k) + B_o u(k) \] (2.39)

\[ y(k) = C_o x(k) + \epsilon(k) \] (2.40)

where:
- \( A_o = \begin{bmatrix} 1 & 0 & 0 \\ T & 1 & 0 \\ T^2/2 & T & 1 \end{bmatrix} \), \( B_o = \begin{bmatrix} T \\ T^2/2 \\ T^3/6 \end{bmatrix} \), \( K_o = \begin{bmatrix} 0.1T \\ T + T^2/20 \\ T/2 + T^2/6 + T^3/60 \end{bmatrix} \)
- \( C_o = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \)

The aim of this example is to direct the elevator from position \( y = 0 \) at time \( t = 0 \) to position \( y = 1 \) as fast as possible. For a soft and safe operation the control is subject to constraints on the jerk, acceleration, speed, and input, given by
2.9 Application Example: MPC for an elevator

\[
\begin{align*}
| \dddot{y}(t) | & \leq 0.4(\text{jerk}) \\
| \dddot{y}(t) | & \leq 0.3(\text{acceleration}) \\
| \ddot{y}(t) | & \leq 0.4(\text{velocity}) \\
| y(t) | & \leq 1.01(\text{position})
\end{align*}
\]

This particular control problem is solved by minimizing the GPC performance index (Section 2.2), subject to the above linear constraints on speed \(\dot{y}\), acceleration \(\ddot{y}\), jerk \(\dddot{y}\) and input \(u\).

A diagram of the implementation can be seen in the Figure 2.3. The simulation has a control and prediction horizon of 30 and a minimum cost-horizon of 1. The reference signal is constant and equal to 1, \(r(k) = 1\), for all \(k\) and the weightings parameters are \(P(q) = 1\), \(\lambda = 0.1\). The sampling-time is \(T = 0.1\) seconds.

![MPC elevator scheme diagram](image)

**Figure 2.3:** MPC elevator scheme.

The optimal control sequence is obtained using a Matlab toolbox [17]. The result is shown in Figure 2.4.

![Simulation result of MPC elevator](image)

**Figure 2.4:** Simulation result of MPC elevator.
2. Model Predictive Control

In Figure 2.4 it can be seen that the jerk (green line), acceleration (dashed red line), speed (back dotted line) and the position (dashed blue line) are all bounded by the constraints. It can be observed that in simulation the elevator as a similar behavior of a real elevator when arriving to a floor.

In this work some difficulties were found in implementing more than one control and reference signal on the MPC toolbox [17]. Furthermore the MPC toolbox requires the use of linear models for the plant model and disturbance model.
3

Kinematics

Contents

3.1 Linearization ......................................................... 20
3.2 Model Validation of the Linearization ............................. 23
3.3 Representation of the Kinematic Model for MPC ............... 27
3. Kinematics

The simplest physical model of a motorbike can be obtained from the kinematic analysis as shown in Limebeer & Sharp [1]. Thus the system of equations for a simple planar motion of a motorbike are described by

\[
\begin{align*}
\dot{x} &= v \cos(\psi) \\
\dot{y} &= v \sin(\psi) \\
\dot{\psi} &= \frac{v \tan(\delta)}{w} \cos(\phi)
\end{align*}
\]

where:
- \( v \) - is the forward velocity;
- \( x \) and \( y \) - are the Cartesian coordinates that indicate the motorbike position in the world frame;
- \( \psi \) - is the orientation angle;
- \( w \) - is the distance between the axle of the rear wheel and the axle of the front wheel;
- \( \delta \) - is the steering angle;
- \( \phi \) - is the leaning angle.

The full system geometry is shown in Figure 3.1:

![Motorbike system geometry](adapted from [1]).

3.1 Linearization

Due to the fact that the MPC toolbox requires a linear model, it is necessary to linearize the motorbike kinematics. To reduce the mathematical complexity of the linearization, it is assumed that the forward velocity is constant and the leaning angle is zero.

As a result of the above assumptions, the kinematics equations are now given by

\[
\begin{align*}
\dot{x} &= V \cos(\psi) \\
\dot{y} &= V \sin(\psi) \\
\dot{\psi} &= \frac{V \tan(\delta)}{w} \cos(\phi)
\end{align*}
\]
This thesis presents two linearizations of the kinematics model of the motorbike to determine which linear model is a better representation of the non-linear kinematics model of the motorbike. The first is based on the first order approximation of the Taylor series, and the second is based on a linear model for the unicycle trajectory tracking [19].

### 3.1.1 Linearization of the kinematics model

For a function \( f(z) \) the first order approximation of the Taylor series around \( z_0 \) is given by,

\[
f(z) \approx f(z_0) + f'(z_0)(z - z_0)
\]

Then the equations of Section 3.1 can be simplified using the first order approximation of the Taylor series around \( \psi_0 \), and so the equations are,

\[
\dot{x} \approx V \cos(\psi_0) - V \sin(\psi_0)(\psi - \psi_0) = V \cos(\psi_0) + V \sin(\psi_0)\psi_0 - V \sin(\psi_0)\psi
\]

\[
\dot{y} \approx V \sin(\psi_0) + V \cos(\psi_0)(\psi - \psi_0) = V \sin(\psi_0) - V \cos(\psi_0)\psi_0 - V \cos(\psi_0)\psi
\]

\[
\dot{\psi} = \frac{V}{w} \tan(\delta)
\]

Setting the continuous kinematics model of the motorbike into a discrete model we have

\[
x(t + \Delta t) \approx x(t) + [V \cos(\psi_0) + V \sin(\psi_0)\psi_0 - V \sin(\psi_0)\psi] \Delta t
\]

\[
y(t + \Delta t) \approx y(t) + [V \sin(\psi_0) - V \cos(\psi_0)\psi_0 - V \cos(\psi_0)\psi] \Delta t
\]

\[
\psi(t + \Delta t) = \psi(t) + \left[ \frac{V}{w} \tan(\delta) \right] \Delta t
\]

Using the small-angle approximation, \( \tan(\delta) \) can be approximated to \( \delta \). So the resulting system of equations is

\[
x(t + \Delta t) \approx x(t) + [V \cos(\psi_0) + V \sin(\psi_0)\psi_0 - V \sin(\psi_0)\psi] \Delta t
\]

\[
y(t + \Delta t) \approx y(t) + [V \sin(\psi_0) - V \cos(\psi_0)\psi_0 - V \cos(\psi_0)\psi] \Delta t
\]

\[
\psi(t + \Delta t) = \psi(t) + \left[ \frac{V}{w} \delta \right] \Delta t
\]

### 3.1.2 Linear model for the motorbike trajectory tracking

Let’s assume that the reference trajectory to track is generated by a virtual motorbike model with constant forward velocity, \( V_{ref} \), and is represented by equations

\[
\dot{x}_{ref} = V_{ref} \cos(\psi_{ref})
\]

\[
\dot{y}_{ref} = V_{ref} \sin(\psi_{ref})
\]

\[
\dot{\psi}_{ref} = \frac{V_{ref} \tan(\delta)}{w \cos(\varphi)}
\]

21
3. Kinematics

![Figure 3.2: Trajectory tracking.](image)

The tracking errors in the world frame are computed as follows

\[
e^W_x = x_{\text{ref}} - x \\
e^W_y = y_{\text{ref}} - y \\
e^W_\psi = \psi_{\text{ref}} - \psi
\]

Assuming that the reference trajectory was generated by a motorbike, under some reference controls as \(V_{\text{ref}}\) and \(\psi_{\text{ref}}\), and \(V_{\text{ref}} = V\),

\[
\dot{e}^W_x = \dot{x}_{\text{ref}} - \dot{x} = V_{\text{ref}}cos(\psi_{\text{ref}}) - Vcos(\psi) \\
\dot{e}^W_y = \dot{y}_{\text{ref}} - \dot{y} = V_{\text{ref}}sin(\psi_{\text{ref}}) - Vsin(\psi) \\
\dot{e}^W_\psi = \dot{\psi}_{\text{ref}} - \dot{\psi} = \frac{V_{\text{ref}}tan(\delta)}{wcos(\varphi)} - \frac{Vtan(\delta)}{wcos(\varphi)}
\]

As a result, the same tracking errors in the motorbike frame are

\[
\begin{pmatrix}
e^B_x \\ e^B_y \\ e^B_\psi
\end{pmatrix} =
\begin{pmatrix}
cos(\psi) & sin(\psi) & 0 \\
-sin(\psi) & cos(\psi) & 0 \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
e^W_x \\ e^W_y \\ e^W_\psi
\end{pmatrix}
\]

and the dynamic of the error is given by the derivative of equation (3.26),

\[
\dot{e}^B =
\begin{pmatrix}
-sin(\psi) & cos(\psi) & 0 \\
-cos(\psi) & -sin(\psi) & 0 \\
0 & 0 & 0
\end{pmatrix}
\dot{\psi}e^W +
\begin{pmatrix}
cos(\psi) & sin(\psi) & 0 \\
-sin(\psi) & cos(\psi) & 0 \\
0 & 0 & 0
\end{pmatrix}
\dot{e}^W
\]

Replacing expressions (3.23), (3.24), and (3.25) on (3.27), \(\dot{e}^B\) becomes,

\[
\dot{e}^B =
\begin{pmatrix}
-sin(\psi) & cos(\psi) & 0 \\
-cos(\psi) & -sin(\psi) & 0 \\
0 & 0 & 0
\end{pmatrix}
\dot{\psi}e^W +
\begin{pmatrix}
cos(\psi) & sin(\psi) & 0 \\
-sin(\psi) & cos(\psi) & 0 \\
0 & 0 & 0
\end{pmatrix}
\dot{e}^W
+ \begin{pmatrix}
0 & 0 & 0 \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
V_{\text{ref}}cos(\psi) - Vcos(\psi) \\
V_{\text{ref}}sin(\psi) - Vsin(\psi) \\
0
\end{pmatrix}
\]

Expressing (3.28) in the motorbike frame, \(\dot{e}^B\) yields
Linearizing the first term of equation (3.29) around $\psi_0$, the linear model for tracking a reference trajectory of a motorbike is

\[
\dot{e}^B = \begin{bmatrix}
  -c_2 & c_1 & 0 \\
  -c_1 & -c_2 & 0 \\
  0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
  \psi_0 \\
  e_\psi \\
  0
\end{bmatrix}
+ \begin{bmatrix}
  0 \\
  e_\psi \\
  0
\end{bmatrix}
V_{\text{ref}}
+ \begin{bmatrix}
  1 & 0 & 0 \\
  0 & e_\psi & 0 \\
  0 & 0 & 0
\end{bmatrix}
\frac{V_{\text{ref}} \cos(e_\psi) - V}{\tan(\delta) \cos(\varphi)} (V_{\text{ref}} - V)
\]

\[(3.30)\]

where:

\[
c_1 = \cos(\psi_0) + \sin(\psi_0) \psi_0 - \sin(\psi_0) \psi
\]

\[(3.31)\]

\[
c_2 = \sin(\psi_0) - \cos(\psi_0) \psi_0 - \cos(\psi_0) \psi
\]

\[(3.32)\]

Note that, $c_1$ and $c_2$ were obtained using the first order approximation of Taylor series, as in Section 3.1.1.

### 3.2 Model Validation of the Linearization

The behavior of the linear models obtained in Sections 3.1.1 and 3.1.2 must be similar to the behavior of the nonlinear model of the motorbike kinematics, described by equations (3.1), (3.2), and (3.3). First, several reference trajectories were generated (straight line, circular trajectory, and an S shaped trajectory). These reference trajectories were generated using Matlab Simulink (see Figure 3.3).

Figure 3.3: Simulink model of the motorbike kinematics.

Next, the behavior of the linear model of the kinematics of the motorbike was tested in a Matlab environment. The analysis was carried out considering a constant velocity, $V$, a zero leaning angle,
3. Kinematics

\[ \varphi = 0, \] and a steering angle, \( \delta \). Note that, in these tests, it is only important to observe if the orientation angle, \( \psi \), goes along with the reference orientation angle, \( \psi_{\text{ref}} \).

The evolution of the orientation angle, \( \psi \), was obtained through a Matlab function that calculates the error between the reference orientation angle of a reference path and the actual orientation, at each time step. When the error is larger than the desired error the linear model is recomputed for a new operation point. The need to recompute the operation point is due to the fact that a model linearized around an operating point is only valid in the neighborhood of that operating point. An operating point of a system is a dynamic configuration that satisfies design and use requirements called operating specifications. So when the error is greater than desired, the operating point must be changed to meet the operation specifications.

3.2.1 Linear kinematics model

The validation of the linear model of the kinematics of the motorbike consists in simulating the motorbike performing a straight line trajectory, a circular trajectory and an S shaped trajectory.

In the first simulation the input variables are a forward speed of 100 km/h, a steering angle of \( \delta = 0^\circ \) and a leaning angle of \( \varphi = 0^\circ \). The result is shown in Figure 3.4. In the second simulation the input variables are a forward speed of 20 km/h, a steering angle of \( \delta = 10^\circ \) and a leaning angle of \( \varphi = 10^\circ \). The result is shown in Figure 3.5. For the third simulation the input variables are a forward speed of 20 km/h, a steering angle of \( \delta = 10^\circ \) for the first curve and \( \delta = -10^\circ \) for the second curve. The result is shown in Figure 3.6.

\[ \begin{align*}
\text{Orientation of the motorbike,} & \quad \psi \\
\text{Reference Orientation,} & \quad \psi_{\text{ref}} \\
\text{Real Orientation,} & \quad \gamma
\end{align*} \]

**Figure 3.4:** Kinematics Linearization - Orientation of the motorbike for a straight line trajectory.
3.2 Model Validation of the Linearization

Figure 3.5: Kinematics Linearization - Orientation of the motorbike for a circular trajectory.

Figure 3.6: Kinematics Linearization - Orientation of the motorbike for an S shaped trajectory.

For the three presented cases related to the linearization of the kinematics model of the motorbike there is an accumulated error associated. The accumulated error along the trajectory is a result of the linearization. For the straight line trajectory, the accumulated error is 0°. For the circular trajectory, the accumulated error is 6.084° and for the S shaped trajectory, the accumulated error is 0.041°.

3.2.2 Motorbike trajectory tracking

The validation of the linear model of motorbike trajectory tracking consists in simulating the motorbike performing a straight line trajectory, a circular trajectory and an S shaped trajectory.

In the first simulation the input variables are a forward speed of 100km/h, a steering angle of \( \delta = 0^\circ \) and a leaning angle of \( \varphi = 0^\circ \). The result is shown in Figure 3.7. In the second simulation the input variables are a forward speed of 20km/h, a steering angle of \( \delta = 10^\circ \) and a leaning angle of \( \varphi = 0^\circ \). The result is shown in Figure 3.8. For the third simulation the input variables are a forward speed of 20km/h, a steering angle of \( \delta = 10^\circ \) for the first curve and \( \delta = -10^\circ \) for the second curve.
3. Kinematics

The result is shown in Figure 3.9.

Figure 3.7: Trajectory Tracking Linearization - Orientation of the motorbike for a straight line trajectory.

Figure 3.8: Trajectory Tracking Linearization - Orientation of the motorbike in a circular trajectory.
3.3 Representation of the Kinematic Model for MPC

For the three cases related to the linearization of the trajectory tracking model of the motorbike there is an accumulated error associated. For the straight line trajectory, the accumulated error is 0°. For the circular trajectory, the accumulated error is 0° and for the S shaped trajectory, the accumulated error is 0°.

3.3 Representation of the Kinematic Model for MPC

As aforementioned in Section 2.1, the state-space representation of an IIO model is given by equations (2.1) and (2.2). The matrices $A$, $B$, $K$, $L$, $C$, $D_H$ and $D_F$ represent the linearized model of the motorbike kinematics around $\psi_0$. These matrices are the plant model.

The two kinematic models obtained in Section 3.1 show similar behavior as regards to the orientation angle. Since the control action is done through the steering of the motorbike and the MPC tracks the reference signal, the model chosen to represented the kinematics of the motorbike is given by the system (3.14), (3.15) and (3.16) (based in the fist order approximation of the Taylor series) instead of using the model based in trajectory tracking method.

Given the linearized kinematic system of the motorbike stated in Section 3.1.1 and the state-space vector $x_i = [x \ y \ \psi]^T$, the system matrices are

$$A = \begin{bmatrix} 0 & 0 & -V \sin(\psi_0) \\ 0 & 0 & V \cos(\psi_0) \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{Vw}{L} \end{bmatrix},$$

$$K = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad L = \begin{bmatrix} V \cos(\psi_0) + V \sin(\psi_0) \psi_0 \\ V \sin(\psi_0) - V \cos(\psi_0) \psi_0 \end{bmatrix},$$

$$C = [0 \ 0 \ 1], \quad D_H = [1], \quad D_F = [0]$$

Figure 3.10 shows a schematic of the MPC implementation of the motorbike kinematics. MPC will act in the steering angle in order to minimize the tracking error between the reference orientation ($\psi_{ref}$) and the actual orientation ($\psi$) of the motorbike.
3. Kinematics

Figure 3.10: Schematic of the MPC implementation of the motorbike kinematics.

The following sections present different types of trajectories as study cases of the feasibility of this method. The analyzes it will be centered in different initial conditions and different MPC gains \((W_y, W_u)\). In the following simulations the summation parameters are \(N_m = 1, N_c = 30\) and \(N = 30\).

3.3.1 Straight line trajectory

Figures 3.11 and 3.12 show the evolution of the orientation angle of the motorbike for a straight line trajectory with different initial orientations and with different MPC tunings, respectively.

Figure 3.11: Different initial conditions for a straight line trajectory.
3.3 Representation of the Kinematic Model for MPC

From Figure 3.11 it can be concluded that regardless of the initial orientation of the motorbike, the orientation angle converges to the reference orientation angle. Figure 3.12 presents three simulations for the weighting parameter \( W_y = 1, 0.7 \) and 0.5 where in each case the weighting parameter \( W_u \) takes the values 1, 0.75, 0.5, 0.25, and 0.01. From simple inspection it can be concluded that the best tuning parameters for the motorbike kinematics are \( W_y = 0.7 \) and \( 0.25 \leq W_u \leq 0.5 \).

3.3.2 Circular trajectory

Figures 3.13 and 3.14 show the evolution of the orientation angle of the motorbike for a circular trajectory with different initial orientations and with different MPC tunings, respectively.

Figure 3.12: Different MPC tunings for a straight line trajectory.

Figure 3.13: Different initial conditions for a circular trajectory.
3. Kinematics

From Figure 3.13 it can be seen that the motorbike follows the reference orientation angle. As in the Section 3.3.1, three simulations for the weighting parameter are presented in Figure 3.14. It can be conclude that for all $W_y = 1, 0.7$ and $0.5$ there is a $W_u$ that minimizes the error between the the final position of the reference orientation and the real orientation of the motorbike. For $W_y = 1$ the parameter $W_u$ is equal to 0.5. For $W_y = 0.75$ the parameter $W_u$ is in the interval $0.5 \leq W_u \leq 0.75$. For $W_y = 0.5$ the parameter $W_u$ is equal to 0.25. For values of $W_y$ below 0.5 the results are the same. In the context of this thesis this is not relevant because in any situation the motorbike has to perform a perfect circular trajectory. So for curves the used parameters are $W_y = 0.5$ and $W_u = 0.25$.

3.3.3 S shaped trajectory

Figures 3.15 and 3.16 show the evolution of the orientation angle of the motorbike for an S shaped trajectory with different initial orientations and with different MPC tunings, respectively.
3.3 Representation of the Kinematic Model for MPC

As in the case of the straight line trajectory, in Figure 3.15 it can be seen that whatever is the initial orientation of the motorbike, the orientation angle converges to the reference orientation angle. Figure 3.16 presents three simulations for the weightings parameters. In this case the best tuning parameters for the motorbike kinematics are $W_y = 1$ and $0.01 \leq W_u \leq 0.25$.

**Figure 3.16**: Different MPC tunings for an S shaped trajectory.
3. Kinematics
## 4 Dynamics

### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Steering Model</td>
<td>34</td>
</tr>
<tr>
<td>4.2</td>
<td>Longitudinal Dynamic Model</td>
<td>34</td>
</tr>
<tr>
<td>4.3</td>
<td>Footpegs Model</td>
<td>35</td>
</tr>
<tr>
<td>4.4</td>
<td>Validation of Models</td>
<td>36</td>
</tr>
<tr>
<td>4.5</td>
<td>Representation of the Dynamic Models for MPC</td>
<td>41</td>
</tr>
</tbody>
</table>
4. Dynamics

A dynamics model is a mathematical representation that includes the physical parameters of the motorbike, such as masses, inertias and constraints. In the literature there are several dynamic models (see for instance [2], [6]) that are complex and hard to work with. The roll dynamics of the motorbike can be treated as an inverted pendulum with an acceleration influence applied at the vehicle’s base [1]. The following equation describes the roll dynamics of a point-mass model

$$ h\ddot{\varphi} = g\sin(\varphi) - [(1 - h\sigma \sin(\varphi))\sigma v^2 + b(\ddot{\psi} + \dot{\psi}(\sigma - \dot{\varphi}v))\cos(\varphi) ] $$  \hspace{1cm} (4.1)

Where:

- $\varphi$ - is the leaning angle;
- $\sigma$ - is the curvature of the path;
- $h$ - is the height from the center of mass to the floor.

There are several approaches to control the roll dynamics of a motorbike. The following sub sections present three of these methods, starting from (4.1).

4.1 Steering Model

The vehicle’s velocity and orientation rate are associated with the curvature of the path through the equation

$$ \sigma v = \dot{\psi} $$  \hspace{1cm} (4.2)

From equation (4.3) the orientation of the motorbike is now given by

$$ \dot{\psi} = \frac{v}{w}\tan(\delta) = v\sigma $$  \hspace{1cm} (4.3)

Substituting (4.2) and (4.3) in (4.1) the roll dynamics yields

$$ h\ddot{\varphi} = g\sin(\varphi) - \tan(\delta)(\frac{v^2}{w} + \frac{b}{w}\dot{\psi} + \tan(\varphi)(\frac{vb}{w}\varphi - \frac{hv^2}{w^2}\tan(\delta))) - \frac{bv}{w\cos^2(\delta)}\dot{\delta} $$  \hspace{1cm} (4.4)

Linearizing (4.4) around a constant velocity ($\dot{\varphi} = 0$) and straight running condition ($\dot{\varphi} = 0$) it is obtained a simple small perturbation linear model given by

$$ \ddot{\varphi} = \frac{g}{h}\varphi - \frac{v^2}{hw}\delta - \frac{bv}{hw}\dot{\delta} $$  \hspace{1cm} (4.5)

4.2 Longitudinal Dynamic Model

The inclusion of the steering angle in the roll dynamics of the motorbike is sufficient to control the motorbike although in some situations it is not the best choice, for example while performing a turn.

It is known that lateral and longitudinal controls of a motorbike are strongly coupled. For example, while performing a curve the driver needs to lean the motorbike into the turn by turning the steer to the opposite direction of the turn, to generate a centrifugal force that pushes the motorbike in the opposite direction of the curve. This behavior is natural and most of the time is not perceived by people.
4.3 Footpegs Model

When exiting from a curve a driver accelerates so that the roll/leaning angle approaches to zero by increasing the centrifugal force that pushes the motorbike in the opposite direction of the curve.

Let equation (4.6) be a simple description of the longitudinal dynamics

\[ \dot{v} = \frac{1}{m} (F_e + F_b - \frac{1}{2} c A_v \rho v^2) \] (4.6)

where \( m \) is the motorbike's mass, and \( F_e \) and \( F_b \) are the longitudinal forces generated by the engine and the brakes of the motorbike, respectively. The remaining term models the aerodynamics of the vehicle, where \( c \) is the drag factor, \( A_v \) is the front projection area of the vehicle and \( \rho \) is the atmospheric air density.

The force generated by the engine is related to the torque of the engine by

\[ F_e = \frac{N_e}{gr R_r} \]

where \( N_e \) is the engine torque, \( gr \) is the transmission ratio and \( R_r \) is the rear wheel radius. To simplify expression (4.6) it is only considered \( F_e \).

Substituting (4.6) in (4.4) the roll dynamics becomes

\[ \ddot{\phi} = g \dot{\phi} - \frac{\delta}{h} \left( \frac{v^2}{w} + \frac{b}{wm} (F_e + F_b - \frac{1}{2} c A_v \rho v^2) + \frac{v b}{w} (\dot{\phi} - \frac{h v^2}{w^2} \delta) \right) - \frac{v b}{h w} \dot{\delta} \] (4.7)

Linearizing equation (4.7) for straight running condition (\( \dot{\phi} = 0 \)) the roll dynamics becomes

\[ \ddot{\phi} = g \dot{\phi} - \frac{\delta}{h} \left( \frac{v^2}{w} + \frac{b}{mw} (F_e + F_b - \frac{1}{2} c A_v \rho v^2) - \frac{h v^2}{w^2} \delta \dot{\phi} \right) - \frac{v b}{h w} \dot{\delta} \] (4.8)

4.3 Footpegs Model

The driver is an important part on the dynamics of a motorbike, so influences the system behavior by applying external torques like the torque generated by the vertical force that the driver’s feet exerts on the footpegs, and the horizontal force applied on the handlebars of the motorbike or even his own weight.

The mass center position of the motorbike can be modified by moving the body horizontally along axis \( y \). In a corner when the mass center moves to the interior of the curve the gravity force increases and so the motorbike tends to fall to the interior of the curve. To counter this effect the driver can increase the forward velocity or steer the handlebar to the opposite direction of the curve. This effect can also be obtained by applying an external force on the footpegs. So, the driver while performing a curve should apply a force on the footpeg of the opposite side of the curve. This makes the motorbike perform a curve with steady leaning inclination.

Figure 4.1 shows a representation of the feet force \( F_p \) applied on the right footpeg of the motorbike.
4. Dynamics

Figure 4.1: Right footpegs force scheme.

The component related to the feet force is

\[ \frac{\varphi d_p}{mh} F_p \]

where:
- \( m \) - is the mass of the motorbike;
- \( h \) - is the height from the center of mass to the contact point between the rear wheel and the floor;
- \( d_p \) - is the length of the footpegs.

Adding the feet force component to (4.5) yields

\[ \ddot{\varphi} = \frac{g}{h} \varphi - \frac{v^2}{hw} \frac{b}{h} \frac{\delta}{\delta} + \frac{d_p}{mh} \varphi F_p \]  \hspace{1cm} (4.9)

4.4 Validation of Models

The dynamical models of the motorbike obtained in the previous sections have to be validated before being used for control purposes. Several tests in a Matlab environment have been made to verify the reliability of the obtained models while the motorbike performs a right turn.

The first two tests analyze the behavior of the motorbike for small perturbations in the steering and in the initial condition. In the first test the motorbike is in a static position (without forward speed). In the second test a forward speed of 100\( \text{km/h} \) is assumed. Finally, the behavior of each model is studied for a specific situation.

The steering model studies the evolution of the roll angle for different velocities and for a fixed steering perturbation of 0.1°. The longitudinal dynamic model studies the evolution of the roll angle for different torques, velocities in order to lift the motorbike from a initial leaning position of 20°. Finally, the footpegs model studies the evolution of the roll angle for different torques applied in the footpegs.

4.4.1 Steering model

Figures 4.2, 4.3, and 4.4 show the results of the tests related to the roll dynamic of the steering model.
4.4 Validation of Models

Figure 4.2: Steering model validation for $v = 0$ km/h.

Figure 4.3: Steering model validation for $v = 100$ km/h.
4. Dynamics

Figure 4.4: Influence of the velocity in the roll dynamics.

Figures 4.2 and 4.3 show that the motorbike behaves as an inverted pendulum. In both cases for an initial condition of $\varphi_0 = 20^\circ$ the motorbike falls to the interior of the curve. In a static position ($v = 0km/h$) and with a small perturbation on the steering angle $\delta = 0.1^\circ$ the motorbike stays in equilibrium. The same does not happen for a forward speed of $100km/h$. The motorbike loses the equilibrium for a small perturbation of $\delta = 0.1^\circ$.

Figure 4.4 shows that the velocity has influence on the behavior of the roll dynamics of the motorbike. The faster the motorbike runs, more quickly it falls when the perturbation occurs. Despite the model in (4.5) is mathematically correct it does not take into account the influence of the longitudinal dynamic.

4.4.2 Longitudinal dynamic model

Figures 4.5, 4.6, and 4.7 show the results of the tests related to the roll dynamic of the longitudinal dynamic model.

Figure 4.5: Longitudinal dynamic model validation for $v = 0km/h$. 
4.4 Validation of Models

Figures 4.5 and 4.6 show that the motorbike have the same behavior as in the previous case. In Figure 4.7 it can be observed that while performing a left turn with initial inclination of $20^\circ$ when the driver accelerates, the velocity of the motorbike increases and tends to reach zero leaning. This behavior arises from the presence of the longitudinal dynamics.

4.4.3 Footpegs model

Figures 4.8, 4.9, and 4.10 show the results of the tests related to the roll dynamic of the footpegs model.
4. Dynamics

Figure 4.8: Footpegs model validation for $v = 0km/h$.

Figure 4.9: Footpegs model validation for $v = 100km/h$. 
The tests results show that the motorbike still behaves as a pendulum, like in section 4.4.1 and 4.4.2. Figure 4.10 shows the behavior of the motorbike when influenced with the feet force applied by the driver. When the feet force decreases it takes more time for the motorbike to fall. Still, due to the mass of the driver and the gravitational force, the motorbike falls.

4.5 Representation of the Dynamic Models for MPC

As in Section 3.3, the models in Sections 4.4.1, 4.4.2, and 4.4.3 have to be converted to a state-space representation of an IIO model.

Figure 4.11 shows a schematic of the MPC implementation of the motorbike dynamics for the three models. The control variable will be changed according to the model in use. The reference signal is the leaning angle \( \phi_{ref} \).

The following sections present the MPC implementations for the steering model, for the longitudinal dynamic model, and for the foot pegs model.
4. Dynamics

4.5.1 Steering model

The roll dynamics in equation (4.5) can be approximated to

\[ \ddot{\phi} = \frac{g}{h} - \frac{v^2}{hw} \delta \]  

(4.10)

considering small variations of \( \delta \), and hence \( \dot{\delta} \) can be approximated to zero.

Given the linearized dynamic system (4.10) and the state-space vector \( x_i = [\phi \ \dot{\phi}]^T \), the system matrices are

\[
A = \begin{bmatrix} 0 & 1 \\ \frac{g}{h} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ -\frac{v^2}{hw} \end{bmatrix}, \\
K = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad L = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \\
C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D_H = [1], D_F = [0]
\]

Figures 4.12 and 4.13 show the evolution of the leaning angle using the MPC to obtain an optimal control signal for two leaning references. The control variable is the steering angle.

![Figure 4.12: MPC simulation for a constant leaning reference of 20°.](image)
4.5 Representation of the Dynamic Models for MPC

Figure 4.13: MPC simulation for a right turn with maximum inclination of 50°.

In the above figures it can be seen that the MPC can conduct the leaning angle to the stipulated value and that MPC doesn’t have any problem to follow the reference inclination of a right turn. It can be concluded that with the steering angle, the control of the roll dynamic of the motorbike is possible although is not the best choice in a racing performance.

4.5.2 Longitudinal dynamic model

As in the previous case, approximating $\dot{\delta}$ to zero the roll dynamics of the longitudinal dynamic model of the motorbike is given by

$$
\ddot{\varphi} = \frac{\delta}{h} \varphi - \frac{\delta}{h} \left( \frac{v^2}{w^2} + b \frac{w}{wm} \left( F_e + F_b - \frac{1}{2} cA_v \rho v^2 \right) \right) - h \frac{v^2}{w^2} \delta \varphi \tag{4.11}
$$

Given the linearized dynamic system (4.11) and the state-space vector $x_i = [\varphi \quad \dot{\varphi}]^T$, the system matrices are

$$
A = \begin{bmatrix}
\frac{g}{h} & 0 & \frac{v^2 c^2 \rho}{w^2} & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
0 \\
-\frac{b \delta}{h \omega m}
\end{bmatrix},
\quad K = \begin{bmatrix}
0 \\
0
\end{bmatrix}, \quad L = \begin{bmatrix}
0 \\
-\frac{\delta \delta}{h \omega m} (F_b - \frac{1}{2} cA_v \rho v^2)
\end{bmatrix},
\quad C = \begin{bmatrix}
1 \\
0
\end{bmatrix}, \quad D_H = [1], \quad D_F = [0]
$$

Figures 4.14 and 4.15 show the evolution of the leaning angle using the MPC to obtain an optimal control signal for two leaning references. The control variable is the force generated by the engine.
4. Dynamics

Figure 4.14: MPC simulation for an exit of a right turn with $\delta = -0.1^\circ$.

Figure 4.15: MPC simulation for an exit of a right turn with $\delta = -9.5^\circ$.

Figure 4.14 shows that a steering angle of $-0.1^\circ$ is not enough to lift the motorbike to its vertical position, while performing a right turn. On the other hand, in Figure 4.15, it is possible to lift the motorbike with a steering angle of $-9.5^\circ$. Although the result of the second simulation is positive, in real driving is not practical due to the fact that the motorbike would fall. Thus this model is not a good choice for control purpose taking into account that the control is made only with one variable.

4.5.3 Footpegs model

The roll dynamics for the footpegs model are given in (4.9). With the approximation of $\dot{\delta}$ to zero the roll dynamics becomes
4.5 Representation of the Dynamic Models for MPC

\[ \ddot{\phi} = \frac{g}{h} \phi - \frac{\delta v^2}{hw} + \frac{d}{mh} \phi F_p \]  \hspace{1cm} (4.12)

From (4.12) it can be seen that the control variable \( F_p \) is linked with the state variable \( \phi \) through a multiplication. In order to put the model in the MPC the roll dynamics is represented by the following state-space model of a continuous time-invariant system

\[ \dot{x}_i = Ax_i + B(t)u \]  \hspace{1cm} (4.13)

where \( B(t) \) is a continuous function of \( \frac{\phi d}{mh} F_p \).

\[ B(t) = \frac{d}{mh} K_c t \]

Thus the roll dynamics becomes

\[ \ddot{\phi} = \frac{g}{h} \phi - \frac{\delta v^2}{hw} + \frac{d}{mh} K_c t F_p \]  \hspace{1cm} (4.14)

where \( K_c \) is the approximation gain of \( \phi \).

Given the linearized dynamic system (4.14) and the state-space vector \( x_i = [\phi \hspace{0.2cm} \dot{\phi}]^T \), the system matrices are

\[ A = \begin{bmatrix} 0 & 1 \\ \frac{g}{h} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{\delta v^2}{hw} K_c t \end{bmatrix} \]

\[ K = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad L = \begin{bmatrix} 0 \\ -\frac{\delta v^2}{hw} \end{bmatrix} \]

\[ C = [1 \hspace{0.2cm} 0], \quad D_H = [1], \quad D_F = [0] \]

Figures 4.16, 4.17 and 4.18 show the leaning angle obtained with the MPC generating controls for three leaning references.

**Figure 4.16:** MPC simulation for a constant leaning reference of 20°.
4. Dynamics

![Graphs showing dynamics of motorbike leaning and force](image)

**Figure 4.17:** MPC simulation for a right turn with maximum inclination of 50°.

**Figure 4.18:** MPC simulation for an exit of a right turn.

Figures 4.16, 4.17, and 4.18 show that the leaning references in all the three presented simulations are followed. The feet force applied by the controller in all three situations corresponds to a real driving situation while exiting from a right turn.
Summarizing, the model chosen to represent the dynamics of the motorbike is the footpegs model. Due to the limitations found in the MPC toolbox in implementing more than one reference and control signals, the steering model and the longitudinal dynamic model can’t be used. The steering model conflicts with the kinematic model because both models use the steering angle as control variable. On the other hand, the longitudinal dynamic model shows results that in a real driving situation are unrealistic. In fact, it is impossible to control the roll dynamics of a motorbike using only the force generated by the engine.
4. Dynamics
5

Racing Problem

Contents

5.1 Racetrack Simulation Background ............................................. 50
5.2 Kinematics Simulation ............................................................. 52
5.3 Dynamics Simulation .............................................................. 53
5. Racing Problem

The racing problem addresses the time a driver takes to do one lap of a closed circuit. In performance driving, such as, Moto Grand Prix, teams try to decrease this time by using light materials and software that helps racing drivers maneuvering the motorbike. On the other hand, drivers are an important factor to decrease this time. Often, in racing there are situations where the driver technique and even the dynamics of the motorbike need to be adjusted to the characteristics of the racetrack, in order to achieve better time laps. Typically, examples are the transitions between dry and wet areas and the adjustments on the suspension that some drivers have to do while racing. This Chapter, presents the racetrack simulation background, a simulation for the kinematics of the motorbike and a simulation for the dynamics of the motorbike.

5.1 Racetrack Simulation Background

The Estoril circuit (see Figure [5.1]), located near Sintra, Portugal, has been chosen to simulate the models. The main reason for choosing this type of racetrack is the challenge that this track offers to the control problem. The presence of fast and slow corners, left and right rotations, straights and other topographic features, makes its analysis interesting from the viewpoint of controlling a motorbike.

![Figure 5.1: Estoril circuit.](image)

The representation of the racetrack in the Matlab simulation is accomplished through image processing. To decrease the complexity and the number of operations of the processing of the track, a scaled version of the racetrack is used and, with basic image processing, the racetrack is built in Matlab environment. The scaled and the processed images are shown in Figures 5.2 and 5.3, respectively.

![Figure 5.2: Image at scale of Estoril circuit.](image)
5.1 Racetrack Simulation Background

5.1.1 Sections of the racetrack

Since the dynamics of the motorbike don’t have the same behavior in the entire track, the race-track is divided in 22 sections. Each section is characterized with different velocities, orientations, inclinations, lengths, and elevations of the racetrack. The partition of the path is shown in Figure 5.4.

5.1.2 Sub sections

The sections A, C, E, G, I, L, N, P, R, T and W are formed, at least, by two sub sections. The first is an acceleration sub section and the second is a breaking sub section. For instance, in section C, the motorbike follows a straight line trajectory while accelerating (C1) and successively breaking (C2). In section A, apart from these subsections, there is a sub section where the velocity remains constant.

The sections B, D, F, H, J, M, O, Q, S and V are corners where the velocity remains constant along the curve.

The reference trajectory was computed for each sub-section.

5.1.3 Velocities

The model of the motorbike that is used for simulation is a Honda CBR 600 RR. For simulation purposes it is important to define the evolution of velocity over time. Knowing that this particular motorbike takes 3.6 seconds from 0 to 100km/h, 11.5 from 0 to 200km/h and it has a maximum speed
5. Racing Problem

of 248.8km/h \[20\], the speed by time curve is shown in Figure 5.5 considering that in this intervals the evolution of velocity over time is linear.

Figure 5.5: Speed by time curve.

In order to determine the distance necessary for the motorbike to go from a velocity to another, or even for braking the Euler-Lagrange motion equations were used.

5.2 Kinematics Simulation

The simulation of the kinematics of the motorbike must analyze if the MPC strategy can be used to make a motorbike follow the circuit path by changing the control variables. For this simulation the control variable is the steering angle of the motorbike. Figures 5.6 and 5.7 show the trajectory of the motorbike, the reference orientation, and the real orientation. More detailed information about the simulation results is presented in Table B.1 (see appendix B).
The motorbike was initially positioned in the racetrack (red dot in Figure 5.6) with initial orientation of $180^\circ$. Figure 5.7 shows that the orientation of the motorbike is almost of the time equal to the reference orientation. Finally, the simulation shows that MPC is able to generate a steering control signal to make the motorbike track the reference orientation.

5.3 Dynamics Simulation

The simulation of the dynamics of the motorbike must analyze if the MPC strategy can be used to make a motorbike perform a run lap by changing the control variables. For this simulation the control variables are the steering angle and the feet force applied on the footpegs of the motorbike. Due to the impossibility to implement more than one control variable and more than one reference signal on the
5. Racing Problem

MPC toolbox, the simulation was achieved by using two control routines. The two control routines are characterized by the racetrack topology. The first routine is used for straight lines with steering angle as the main control variable. The second routine is used for curves with the feet force as the main control variable. Figure 5.8 illustrates the simulation diagram, where the Routine block establishes which MPC is being used.

![Simulation diagram](image)

**Figure 5.8:** Simulation diagram.

In addition to this, there is the need to use a PID controller to compute the steering angle while the second routine is active so that the motorbike stays in the racetrack. The PID controller used is in [2], pages 47-49. The gains were determined by trial and error for each curve starting with the ones determined in [2].

Figure 5.10 shows the diagram for the two routines. The dashed signals are the ones that don’t belong to the routine that is active.
Figure 5.9: Simulation diagram for the first routine.

Figure 5.10: Simulation diagram for the second routine.

The idea of these two routines is to represent a possible behavior by the driver while running in a racetrack. In straight lines the driver tends to focus on the orientation of the motorbike and in curves the driver tends to focus in the inclination of the motorbike to perform the curves.

Figure 5.11 shows in red the estimated trajectory and in blue the real trajectory. In Figure 5.12 it can be seen the reference and real orientation of the motorbike in the racetrack. Lastly, Figure 5.13 shows the reference and real leaning of the motorbike in the racetrack. More detailed information about the simulation results is presented in Table C.1 (see appendix C).
5. Racing Problem

**Figure 5.11:** Estoril racetrack for the dynamics simulation - Trajectories.

**Figure 5.12:** Orientation results for the dynamics simulation - Orientations.
5.3 Dynamics Simulation

Figure 5.13: Inclination results for the dynamics simulation - Inclinations.

As mentioned in the previous Section, the motorbike was initially positioned in the racetrack with the same initial orientation and with an initial inclination of 0°. From the Figure 5.12 it can be seen that the orientation of the motorbike is not always equal to the reference orientation. This result comes from the use of the PID controller that directly influences the position of the motorbike in the racetrack. In fact, this can be seen in Figure 5.11 where the real trajectory is different from the estimated trajectory in sections S and V of the racetrack. Despite this influence the result is positive since the orientation is close to the reference orientation and the motorbike stays in the racetrack. Figure 5.13 shows that the leaning of the motorbike goes along with the leaning reference.

The simulation shows that this strategy with the two MPC routines and a PID controller is able to generate the control signals (steering angle and feet force) to match the orientation and the leaning of the motorbike with the reference orientation and reference leaning, respectively.

The motorbike makes one lap to the racetrack in 2'25" for both simulations starting from a stopped initial position.
5. Racing Problem
Conclusions and Future Work
6. Conclusions and Future Work

6.1 Conclusions

In this work, the Model Predictive Control was used as the main strategy to generate the steering and feet force values in order to control an autonomous motorbike and partially solve the racing problem. Currently, there are many research works on the control of autonomous motorbikes using different control strategies.

Several models of the motorbike have been examined. First the kinematic model of the motorbike was derived from a simple planar motion of a motorcycle. The model was then linearized to be implemented on the MPC and some prior simulations were made to validate the model in the MPC. Then three dynamic models were considered to describe the roll dynamics of the motorbike. The first was based in steering angle control, the second describes the longitudinal dynamic of a motorbike inside the roll dynamics, and the third was based on the feet force that the driver’s feet exerts on the footpegs. The three models were linearized and a few tests were made to determine the best model to be use in the MPC. The model chosen to represent the roll dynamics of the motorbike was the footpegs model (described by [4,11]), due to its mathematical simplicity and its pleasing results on the MPC tests.

The final model was tested on a real driving situation by two simulations. The first one includes only the kinematic model of the motorbike and the second includes the kinematic and roll dynamic model of the motorbike. The behavior of the motorbike model was simulated in the circuit of Estoril. The motorbike reaches its finish mark in a time of 2'25'' starting from a stopped initial position, which can be compared with the time of a real motorbike (2'06'') running on the same racetrack starting from a flying start lap.

The simulations results show that the proposed autonomous motorbike using MPC as the main control strategy is able to generate the control signals to match with the reference signals.

In conclusion it was achieved a control strategy for controlling an autonomous motorbike using simple kinematic and dynamic descriptions models of the motorbike.

6.2 Future Work

The arrival point of this thesis can be considered the starting point of several improvements, developments, and implementations. The positive results of this work show the potential of using MPC to control an autonomous motorbike. In fact, if eliminating the PID controller and using a MPC to handle with multiple reference signals and multiple reference inputs a more compact control system could be obtained.

Other alternative is to design an optimization process to tune the controller gains of the PID controller and the MPC gains. Figure [6.1] shows a schematic of a possible implementation for an optimizer PID gains.
6.2 Future Work

The optimization process can be done using genetic algorithms (see for instance [21] and [22]).

A possible improvement to the footpegs model is to include the force generated by the entire driver’s body while the motorbike is performing turns.

The obtained footpegs model should be tested to disturbances on the steering angle triggered by irregularities on the racetrack and by incorrect tuning of the front and rear suspensions system.

To conclude, the results presented in this thesis incentive the implementation of this control system in a real motorbike.

**Figure 6.1:** Optimizer PID gains.
6. Conclusions and Future Work
Bibliography


Bibliography


Right Inverse and Right Complement
A. Right Inverse and Right Complement

Let $M$ be a full-row rank matrix $\in \mathbb{R}^{m \times n}$ for $m \leq n$. The singular value decomposition of $M$ is given by

$$M = U [\Sigma \ 0] \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$$

where $U$ and $V$ are orthogonal matrixes that contains the eigenvectors of $MM^T$ and the eigenvectors of $M^TM$, respectively. $\sigma$ is a diagonal matrix which entries are the nonnegative square roots of the eigenvalues of $M^TM$ (for $m \leq n$) or $MM^T$ (for $m \geq n$). The matrix $U$ and $V$ are described in (A.1) and (A.2)

$$U = [u_1, \ldots, u_m] \in \mathbb{R}^{m \times m}$$  \hspace{1cm} (A.1)

$$V = [v_1, \ldots, v_n] \in \mathbb{R}^{n \times n}$$  \hspace{1cm} (A.2)

**Right inverse matrix:**

Given a matrix $A$, if $A \in \mathbb{R}^{m \times n}$ then there exist orthogonal matrices $U$ and $V$ such that

$$U^T AV = \sigma$$

The right inverse matrix of $A$ is defined as

$$A^r = V_1 \sigma^{-1} U^T$$

**Right complement matrix:**

Given a matrix $A$, if $A \in \mathbb{R}^{m \times n}$ then there exist orthogonal matrices $U$ and $V$ such that

$$U^T AV = \sigma$$

The right complement matrix of $A$ is defined as

$$A^r \perp = V_2$$

Both matrixes has the following properties

$$AA^r = I$$

$$AA^r \perp = 0$$
### Table B.1: Kinematics simulation data.

<table>
<thead>
<tr>
<th>Section</th>
<th>Sub-Section</th>
<th>Velocity [Km/h]</th>
<th>$\psi$ [°]</th>
<th>$\phi$ [°]</th>
<th>t [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A1</td>
<td>0 - 100</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>A2</td>
<td>100 - 200</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>A3</td>
<td>200 - 203.4</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>A4</td>
<td>203.4 - 80</td>
<td>180</td>
<td>0</td>
<td>0.1965</td>
</tr>
<tr>
<td>B</td>
<td>B1</td>
<td>70</td>
<td>177.9981</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>B2</td>
<td>60</td>
<td>54.1440</td>
<td>10</td>
<td>0.0901</td>
</tr>
<tr>
<td>C</td>
<td>C1</td>
<td>60 - 100</td>
<td>55.7932</td>
<td>10</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>C2</td>
<td>100 - 150</td>
<td>73.5861</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>C3</td>
<td>150 - 80</td>
<td>72.6983</td>
<td>0</td>
<td>0.1084</td>
</tr>
<tr>
<td>D</td>
<td></td>
<td>80</td>
<td>32.908</td>
<td>-19.2664</td>
<td>20</td>
</tr>
<tr>
<td>E</td>
<td>E1</td>
<td>80 - 135</td>
<td>5.8655</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>E2</td>
<td>135 - 80</td>
<td>0.1572</td>
<td>0</td>
<td>0.0867</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>80</td>
<td>0.1285</td>
<td>-160.434</td>
<td>20</td>
</tr>
<tr>
<td>G</td>
<td>G1</td>
<td>80 - 121</td>
<td>-168.6455</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>G2</td>
<td>121 - 80</td>
<td>-169.7509</td>
<td>0</td>
<td>0.1684</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>80</td>
<td>-155.2376</td>
<td>-17.0156</td>
<td>-20</td>
</tr>
<tr>
<td>I</td>
<td>I1</td>
<td>80 - 138</td>
<td>14.5025</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>I2</td>
<td>138 - 110</td>
<td>13.3092</td>
<td>0</td>
<td>0.0101</td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>110</td>
<td>-15.6033</td>
<td>10</td>
<td>0.0101</td>
</tr>
<tr>
<td>L</td>
<td>L1</td>
<td>110 - 204</td>
<td>-14.5472</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>L2</td>
<td>204 - 100</td>
<td>-14.2218</td>
<td>0</td>
<td>0.1030</td>
</tr>
<tr>
<td>M</td>
<td></td>
<td>100</td>
<td>27.8968</td>
<td>-164.0908</td>
<td>-20</td>
</tr>
<tr>
<td>N</td>
<td>N1</td>
<td>100 - 176</td>
<td>165.891</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>N2</td>
<td>176 - 100</td>
<td>165.8761</td>
<td>0</td>
<td>0.0689</td>
</tr>
<tr>
<td>O</td>
<td></td>
<td>100</td>
<td>152.4741</td>
<td>-41.4729</td>
<td>20</td>
</tr>
<tr>
<td>P</td>
<td>P1</td>
<td>100 - 130.5</td>
<td>32.5998</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>P2</td>
<td>130.5 - 100</td>
<td>31.2242</td>
<td>0</td>
<td>0.0325</td>
</tr>
<tr>
<td>Q</td>
<td></td>
<td>100</td>
<td>-27.5995</td>
<td>20</td>
<td>0.0481</td>
</tr>
<tr>
<td>R</td>
<td>R1</td>
<td>100 - 111</td>
<td>-25.6855</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>R2</td>
<td>111 - 60</td>
<td>-25.9107</td>
<td>0</td>
<td>0.0850</td>
</tr>
<tr>
<td>S</td>
<td>S1</td>
<td>60</td>
<td>26.1453</td>
<td>-64.8245</td>
<td>-10</td>
</tr>
<tr>
<td></td>
<td>S2</td>
<td>60</td>
<td>-39.0288</td>
<td>10</td>
<td>0.0998</td>
</tr>
<tr>
<td>T</td>
<td>T1</td>
<td>60 - 136</td>
<td>-50.2136</td>
<td>5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>136 - 90</td>
<td>-80.3544</td>
<td>0</td>
<td>0.0817</td>
</tr>
<tr>
<td>U</td>
<td></td>
<td>90</td>
<td>1.1885</td>
<td>-5</td>
<td>0.0258</td>
</tr>
<tr>
<td>V</td>
<td></td>
<td>100</td>
<td>-30.5759</td>
<td>-179.5462</td>
<td>10</td>
</tr>
<tr>
<td>W</td>
<td>W1</td>
<td>100 - 200</td>
<td>-180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>W2</td>
<td>200 - 248.8</td>
<td>-180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>W3</td>
<td>248.8</td>
<td>-180</td>
<td>0</td>
<td>0.0760</td>
</tr>
</tbody>
</table>

B-2
Dynamics Simulation Data
Table C.1: Dynamics simulation data.

<table>
<thead>
<tr>
<th>Section</th>
<th>Sub-Section</th>
<th>Velocity [Km/h]</th>
<th>$\psi$ [°]</th>
<th>$\phi$ [°]</th>
<th>$t$ [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A1</td>
<td>0 - 100</td>
<td>180</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>A2</td>
<td>100 - 200</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>A</td>
<td>A3</td>
<td>200 - 203.4</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>A</td>
<td>A4</td>
<td>203.4 - 80</td>
<td>180</td>
<td>0</td>
<td>0.1965</td>
</tr>
<tr>
<td>B</td>
<td>B1</td>
<td>70</td>
<td>180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>B</td>
<td>B2</td>
<td>60</td>
<td>53.0181</td>
<td>19.9998</td>
<td>0.0901</td>
</tr>
<tr>
<td>C</td>
<td>C1</td>
<td>60 - 100</td>
<td>54.6186</td>
<td>-20</td>
<td>-</td>
</tr>
<tr>
<td>C</td>
<td>C2</td>
<td>100 - 150</td>
<td>73.5871</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>C</td>
<td>C3</td>
<td>150 - 80</td>
<td>72.6983</td>
<td>0</td>
<td>0.1084</td>
</tr>
<tr>
<td>D</td>
<td>-</td>
<td>80</td>
<td>35.8798 - 18.0027</td>
<td>20</td>
<td>-</td>
</tr>
<tr>
<td>E</td>
<td>E1</td>
<td>80 - 135</td>
<td>5.8666</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>E</td>
<td>E2</td>
<td>135 - 80</td>
<td>0.1572</td>
<td>0</td>
<td>0.0867</td>
</tr>
<tr>
<td>F</td>
<td>-</td>
<td>80</td>
<td>0.1321 - 157.6419</td>
<td>10 - 9.9999</td>
<td>0.1352</td>
</tr>
<tr>
<td>G</td>
<td>G1</td>
<td>80 - 121</td>
<td>-168.6476</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>G</td>
<td>G2</td>
<td>121 - 80</td>
<td>-169.7509</td>
<td>0</td>
<td>0.1684</td>
</tr>
<tr>
<td>I</td>
<td>I1</td>
<td>80 - 138</td>
<td>14.5030</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>I</td>
<td>I2</td>
<td>138 - 110</td>
<td>13.3092</td>
<td>0</td>
<td>0.0667</td>
</tr>
<tr>
<td>J</td>
<td>-</td>
<td>110</td>
<td>-15.576</td>
<td>0</td>
<td>0.0101</td>
</tr>
<tr>
<td>L</td>
<td>L1</td>
<td>110 - 204</td>
<td>-14.5472</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>L</td>
<td>L2</td>
<td>204 - 100</td>
<td>-14.2218</td>
<td>0</td>
<td>0.1030</td>
</tr>
<tr>
<td>M</td>
<td>-</td>
<td>100</td>
<td>22.4215 - 164.158</td>
<td>-10</td>
<td>0.2058</td>
</tr>
<tr>
<td>N</td>
<td>N1</td>
<td>100 - 176</td>
<td>165.891</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>N</td>
<td>N2</td>
<td>176 - 100</td>
<td>165.8761</td>
<td>0</td>
<td>0.0689</td>
</tr>
<tr>
<td>O</td>
<td>-</td>
<td>100</td>
<td>153.1938 - 41.8949</td>
<td>-10 - 9.9988</td>
<td>0.1078</td>
</tr>
<tr>
<td>P</td>
<td>P1</td>
<td>100 - 130.5</td>
<td>32.5094</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>P</td>
<td>P2</td>
<td>130.5 - 100</td>
<td>31.2242</td>
<td>0</td>
<td>0.0325</td>
</tr>
<tr>
<td>Q</td>
<td>-</td>
<td>100</td>
<td>-19.2624</td>
<td>0</td>
<td>0.0481</td>
</tr>
<tr>
<td>R</td>
<td>R1</td>
<td>100 - 111</td>
<td>-25.6889</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>R</td>
<td>R2</td>
<td>111 - 60</td>
<td>-25.9107</td>
<td>0</td>
<td>0.085</td>
</tr>
<tr>
<td>S</td>
<td>S1</td>
<td>60</td>
<td>-14.8016</td>
<td>-24.9998</td>
<td>-</td>
</tr>
<tr>
<td>S</td>
<td>S2</td>
<td>60</td>
<td>-36.0564</td>
<td>24.9998</td>
<td>0.0998</td>
</tr>
<tr>
<td>T</td>
<td>T1</td>
<td>60 - 136</td>
<td>-43.0802 - 80.917</td>
<td>4.9991 - 5</td>
<td>-</td>
</tr>
<tr>
<td>T</td>
<td>T2</td>
<td>136 - 90</td>
<td>-80.61</td>
<td>5</td>
<td>0.0817</td>
</tr>
<tr>
<td>U</td>
<td>-</td>
<td>90</td>
<td>16.8468</td>
<td>-9.9999</td>
<td>0.0258</td>
</tr>
<tr>
<td>V</td>
<td>-</td>
<td>100</td>
<td>-28.4912 - 175.5972</td>
<td>9.999 - 19.9999</td>
<td>0.2562</td>
</tr>
<tr>
<td>W</td>
<td>W1</td>
<td>100 - 200</td>
<td>-180.0018</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>W</td>
<td>W2</td>
<td>200 - 248.8</td>
<td>-180</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>W</td>
<td>W3</td>
<td>248.8</td>
<td>-180</td>
<td>0</td>
<td>0.0760</td>
</tr>
</tbody>
</table>