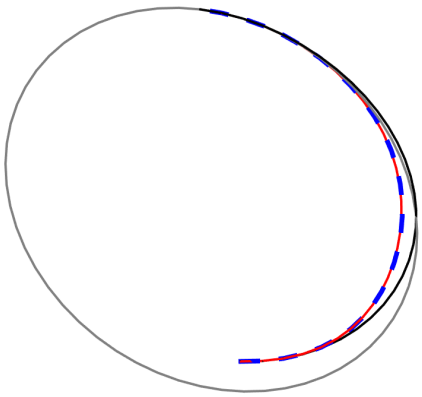
KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Efficient Nonlinear Model Predictive Control Formulations for Economic Objectives with Aerospace and Automotive Applications



Mario Zanon

Dissertation presented in partial fulfilment of the requirements for the degree of Doctor in Engineering Science

November 2015

Efficient Nonlinear Model Predictive Control Formulations for Economic Objectives with Aerospace and Automotive Applications

Mario ZANON

Examination committee:
Prof. dr. ir. Yves Willems, chair
Prof. dr. ir. M. Diehl, supervisor
(KU Leuven and ALU Freiburg)
Prof. Alberto Bemporad, co-supervisor
(IMT Lucca)
Ass. Prof. Sébastien Gros, co-supervisor
(Chalmers University)
Prof. dr. ir. Joseph Vandewalle
Prof. dr. ir. Joris De Schutter
Prof. dr. ir. Jan Swevers
Prof. dr. ir. Johan Meyers
Ass. Prof. Goele Pipeleers
Prof. dr. Lars Grüne
(University of Bayreuth)

Dissertation presented in partial fulfilment of the requirements for the degree of Doctor in Engineering Science

November 2015

© 2015 KU Leuven – Faculty of Engineering Science
Uitgegeven in eigen beheer, Mario Zanon, Kasteelpark Arenberg 10 box 24462, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm, electronic or any other means without written permission from the publisher.

Abstract

This thesis is concerned with optimal control techniques for optimal trajectory planning and real-time control and estimation. The framework of optimal control is a powerful tool which enjoys increasing popularity due to its applicability to a wide class of problems and its ability to deliver solutions to very complicated problems which cannot be intuitively solved.

The downside of optimal control is the computational burden required to compute the optimal solution. Due to recent algorithmic developments and increases in the computational power, this burden has been significantly reduced over the last decades. In order to guarantee effectiveness and reliability of the solver, three main components are necessary: fast and robust algorithms, a good problem formulation, and a mathematical model tailored to optimisation. Indeed, both the model and the optimal control problem can usually be formulated in many different ways, some of which are better suited for optimisation. In this thesis we are concerned with all three components, with a focus on the last two.

Concerning the problem formulation, we propose practical approaches for formulating optimal control, MPC and MHE problems in an optimisation-friendly fashion. Moreover, we analyse the stability properties of various MPC formulations, with a focus on so-called economic MPC, for which the stability theory is still developing.

On the algorithmic level, we review the literature on optimisation and optimal control and we prove that it is possible to tune tracking MPC formulations in order to locally obtain the same behaviour as economic MPC. The main advantages of tuned tracking MPC over economic MPC consist in easier to guarantee closed-loop stability and applicability of efficient real-time algorithms.

On the modelling side, we propose an approach for deriving models of reduced complexity and reduced nonlinearity for multibody mechanical systems. The use of nonminimal coordinates and DAE models enlarges the range of modelling

possibilities and allows the control engineer to derive models which are better suited for optimisation. In order to provide an easy framework for the model derivation, we extend the Euler-Lagrange approach and we demonstrate how to implement the proposed approach in practice.

In order to demonstrate the effectiveness of the proposed techniques, we deploy them for two applications: tethered airplanes and autonomous vehicles. Both examples are characterised by fast nonlinear constrained dynamics for which simple controllers cannot be deployed.

Tethered airplanes are of particular interest because they are an emerging technology for wind energy production. In this thesis, we use optimal control to design trajectories which extract maximum energy from the airmass and compare single and dual-airfoil configurations. We moreover demonstrate the effectiveness of MPC and MHE for controlling the system in real time and apply the new tuning procedure for tracking MPC to show its ability to locally approximate economic MPC.

Beknopte samenvatting

Deze thesis behandelt optimale regeltechnieken voor trajectplanning en real-time regelaars en toestandsschatters. Optimale controle is een krachtig instrument en kent een stijgende populariteit, dankzij de toepasbaarheid op een brede klasse van problemen en het vinden van oplossingen voor gecompliceerde problemen die geen intuïtieve oplossing hebben.

Het grootste nadeel van optimale controle is de computationele kost voor het vinden van oplossingen. Dankzij recente algoritmische ontwikkelingen en verhoogde rekenkracht is dit nadeel sterk afgenomen over de laatste decennia. Om de effectiviteit en betrouwbaarheid van de oplossingsstrategie te garanderen, zijn er drie componenten van belang: snelle en robuuste algoritmes, een gepaste probleemformulering en wiskundige modellen die specifiek zijn ontworpen voor optimalisatie. Zowel het model als het optimalisatieprobleem kunnen op verschillende manieren geformuleerd worden, waarvan sommige meer geschikt zijn voor optimalisatie. In deze thesis worden de drie aspecten behandeld, met een focus op de laatste twee.

Wat betreft de probleemformulering, stellen we een praktische aanpak voor voor de formulering van optimale controle-, MPC- en MHE-problemen. Verder wordt stabiliteit in gesloten lus van verscheidene MPC-formuleringen geanalyseerd, met een nadruk op economische MPC, waarvan de stabiliteitstheorie in volle ontwikkeling is.

Op algoritmisch niveau wordt een literatuurstudie gemaakt van de stand van optimalisatie en optimale controle. Voorts wordt er een bewijs geleverd dat tracking MPC zo getuned kan worden dat het zich lokaal als economische MPC gedraagt. De belangrijkste voordelen van getunede tracking MPC ten opzichte van economische MPC zijn dat stabiliteit eenvoudig aangetoond kan worden, en de toepasbaarheid van efficiënte real-time algoritmes.

Op het vlak van modellering reiken we een aanpak aan voor complexiteitsreductie van multibody mechanische systemen. Het gebruik van niet-minimale

coördinaten en DAE-modellen geeft meer mogelijkheden voor modellering, en maakt het mogelijk modellen af te leiden die beter passen in het kader van optimalisatie. Met het oog op gebruiksvriendelijkheid, breiden we de Euler-Lagrange aanpak uit en tonen we aan hoe de gegeven aanpak toe te passen in de praktijk.

Ter illustratie van de effectiviteit van optimale controle, MPC en MHE, worden twee toepassingen getoond: vlieger-energie systemen en zelfrijdende auto's. Beide voorbeelden bevatten niet-lineaire dynamische systemen met beperkingen waarvoor geen eenvoudige regelaars opgesteld kunnen worden.

Vanwege het feit dat vlieger-energie een opkomende technologie is, wordt er nadruk gelegd op vliegers verankerd aan de grond via een kabel voor het opwekken van energie. In deze thesis worden optimale controletechnieken gebruikt om trajecten te ontwerpen die maximale energie uit de wind halen, en om enkel- en dubbelvliegersystemen te vergelijken. Verder tonen we de real-time toepasbaarheid van MPC en MHE aan. Tevens wordt de behandelde tuning voor tracking MPC gebruikt om de effectiviteit om het benaderen van economische MPC aan te tonen.

Acknowledgements

This is the outcome of four and a half years of research in various universities in Europe and many people have contributed to it by supervising my work, giving me good advice, challenging my opinions and forcing me to justify my thoughts adequately.

My first thanks goes to professor Alberto Bemporad who, after a research period under his supervision, helped me finding a the PhD position at KU Leuven and remained an extremely valuable discussion partner and supervisor.

The greatest thanks goes to professor Moritz Diehl and assistant professor Sébastien Gros. Moritz has been a continuous source of inspiration, ideas and challenges that greatly contributed to my personal development as a researcher. I also thank Moritz for creating a very fruitful environment for cooperation and exchange of ideas within the group, with a particular attention to creating a nice and enjoyable atmosphere at work. Since his first day in our group as a postdoc, Sébastien guided me from the early stages of my PhD by always pushing me to question every idea in a critical way, stimulating my curiosity and helping me to develop the instinct to prove every result thoroughly before relying on it. Also after he became assistant professor in Chalmers we kept a fruitful cooperation and further developed our nice friendship, which was further strengthened by a research period at Chalmers.

A big thanks goes to professor Lars Grüne who welcomed me in his research group in Bayreuth and was a great partner for discussions mainly but not only on economic MPC and a big source of inspiration. Thanks also to junior professor Karl Worthmann, with whom we started a fruitful cooperation and nice friendship when he was a postdoc in Bayreuth.

My PhD has lived several geographic phases. From the period in Leuven I want to thank Adeleh, Andrew, Attila, Boris, Carlo, Greg, Hammad, Janick, Joachim, Joel, Joris, Julia, Kurt, Milan, Quoc, Reinhart, Rien, Sébastien, Slava and Tom for sharing the office with me and making coffee breaks and lunches relaxing

moments for recharging batteries, but also for nice scientific discussions. I also want to thank Attila, Janick, Keith, Milan, Matteo, Reinhart, Rien, Sebastian, Sébastien, Tania and all my other friends outside the office with whom I shared niche goulash sessions and who made me really enjoy my life in Leuven. From the period in Bayreuth I thank Karl, Marleen, Philipp, Roberto and Vryan who have been both colleagues and friends. From the period in Göteborg I thank the whole S2 department for being like a big research family and in particular Emil, Giuseppe, Mitra, Paolo, Robert and Sébastien for sharing the passion for research and extreme grilling. From the period in Freiburg I thank everybody in the new research group of Moritz: Dimitris, Gianni, Greg, Jörg, Jonas, Rien, Robin, Thor for the nice atmosphere and fruitful cooperations. I also want to thank my housemates Capucine, Jan and Miguel who made my stay particularly enjoyable and all the students who cooperated with our group and also became good friends to share good moments with.

I also wat to thank all people involved in the SADCO, HIGHWIND, TEMPO and the starting AWESCO projects for sharing their ideas and friendship with me.

Further thanks go to professor Paolo Falcone for believing in me and giving me the chance to start a postdoc at Chalmers.

A big thank goes to my parents and my sister who always believed in me and pushed me to give my best. Last but not least, I want to thank my girlfriend Linda who, since three years supports me in every moment, understands my moments of stress and encourages me constantly.

Contents

Abstract i

Contents vii

1 Introduction 1

1.1 Contributions 2

1.2 Outline and Structure of the Thesis 4

2 Optimisation, Simulation, Optimal Control and Estimation 7

2.1 Static Optimisation 8

2.2 Dynamic Systems and Initial Value Problems 19

2.3 Optimal Control 24

2.4 Real-Time Optimal Control and Estimation 34

2.5 Conclusions 40

3 Autonomous Driving 43

3.1 Vehicle Model 44

3.2 MPC and MHE for Collision Avoidance 50

3.3 Towards Time Optimal Driving 58

3.4 Conclusions 62

4 Stability Theory for Model Predictive Control 63

4.1 Preliminaries and Definitions 65
4.2 Tracking MPC 67
4.3 Economic MPC 68
4.4 Periodic Economic MPC 71
4.5 Conclusions 92

5 Economic-Based Tuning of Tracking MPC 95

5.1 Problem Setup and Preliminaries 96
5.2 Linear MPC without Active Constraints at Steady State . . . 103
5.3 Linear MPC with Active Constraints at Steady State 114
5.4 Tuning of the Tracking (N)MPC Based on Economic Criteria . 120
5.5 Optimal Periodic Operation 131
5.6 Conclusions 140

6 Modelling of Mechanical Systems aimed at Optimal Control 141

6.1 The Euler-Lagrange Approach 143
6.2 Modelling the Rotational Dynamics 146
6.3 Baumgarte Stabilisation 154
6.4 Conclusions 166

7 Airborne Wind Energy 168

7.1 Introduction 170
7.2 Modelling of Airborne Wind Energy Systems 172
7.3 Offline and Online OCP Formulation 186
7.4 Comparison of Dual and Single Airfoils 193
7.5 Control of a Single Airfoil for Power Generation 206
7.6 Control of Dual Airfoils for Power Generation 215

7.7 Rotational Startup 221
7.8 Towards Tuned Tracking MPC for Power Maximisation 228
7.9 Conclusions 230

8 Conclusions and Outlook **234**

Bibliography **239**

Curriculum Vitae **251**

List of Publications **253**

Chapter 1

Introduction

In the modern world, automation is present in almost every aspect of life. The tasks that a controller is expected to accomplish are becoming more and more complex. Moreover, often one is not only interested in performing a task, but also in doing so in an optimal way. For this reason, in this thesis we focus on optimal control techniques.

Optimal control makes use of a mathematical model of the dynamic system and the constraints within which it must be operated in order to compute a feasible state and control trajectory which minimises a given performance index. In many cases, this task is nontrivial and an approximate solution is hard to be guessed even by an expert human who knows in detail how the considered system works. Moreover, the framework of optimal control is rather generic, and the algorithms are therefore system independent and applicable to a large variety of problems. The possibility to jointly optimise the control strategy as well as some model parameters makes it possible to also design a system which is tailored to the task it needs to perform. All these features make optimal control a very powerful tool for the control engineer.

When applying a control trajectory to a real system, due to model inaccuracy, unmodelled dynamics and external perturbations, the state trajectory will not match the expected one. This open-loop control strategy is in general dangerous, as the system might become unstable, violate the prescribed constraints and fail to accomplish the assigned task. In contrast to open-loop control, closed-loop control techniques make use of state measurements or estimates in order to adapt the control strategy to the perturbed conditions.

Model Predictive Control (MPC) is a feedback control technique based on

optimal control: given the available state measurement or estimate, it solves an optimal control problem in order to compute the feedback control law to apply to the system. MPC therefore inherits the advantages of optimal control: it optimises the prescribed performance index, it keeps the system within the prescribed constraints and it is able to control multiple-input-multiple output systems in a very natural way.

The main limitation of optimal control, and therefore MPC, is the need to solve a rather involved nonconvex optimisation problem. Thanks to a huge increase both in the algorithmic efficiency and in the computational power, we are nowadays able to solve optimisation problems at speeds which were impossible only a few years ago. Therefore, the range of applications of optimal control and MPC is quickly increasing.

The fact that MPC solves an optimisation problem does not directly guarantee that it provides a stabilising feedback law. Only under some assumptions on the system dynamics and the minimised cost function it is possible to prove stability of the closed-loop system. The simplest and most implemented case is that of a quadratic function penalising the distance from a given reference. In this case, the conditions for stability are well understood and a mature literature on the topic is available. In some situations, however, one might be interested in directly optimising a generic performance index. In this second case, often referred to as *economic MPC*, it is much harder to enforce stability and, only in recent years, the first results have been developed, some of which are developed in this thesis.

1.1 Contributions

This thesis presents contributions related to several aspects of optimal control and MPC. We investigate MPC stability theory and extend the stability theory for economic MPC to the periodic case. We prove the possibility of formulating tracking MPC such that the closed-loop behaviour of the system is locally a first-order approximation of that obtained when deploying economic MPC. We propose a modelling framework for mechanical systems aimed at optimal control and MPC. Moreover, we consider two applications: tethered airplanes for energy production and autonomous ground vehicles. We outline in the following the main contributions of this thesis.

Control of Autonomous Ground Vehicles In recent years, driving assistance systems have become widespread on all newly produced cars and the field of autonomous driving has become a popular research field. In this thesis, we

use nonlinear MPC to control an autonomous ground vehicle. The friction coefficient between the tire and the road is particularly critical for guaranteeing an effective control of the vehicle. We use MHE to simultaneously estimate the state and the road-tire friction coefficient. We test the proposed algorithm in simulations in an obstacle avoidance scenario on a low friction road [38, 117, 118]. Moreover, we propose a first step towards time optimal MPC for small-scale race cars [112].

Stability Theory for Periodic Economic MPC As outlined before, stability theory for economic MPC is the object of ongoing research. The steady state case has been studied in [32, 3]. In this thesis, we study the periodic case, both for time-varying and time-invariant systems, and we develop a framework for analysing stability of periodic economic MPC [120, 125].

Tuning of Tracking MPC to Locally Approximate Economic MPC As outlined before, economic MPC has two main drawbacks: it is in general hard to guarantee stability and the efficient algorithms which apply to tracking MPC cannot be applied. For these reasons, in this thesis we develop an approach to tune tracking MPC such that it locally yields the same behaviour as economic MPC while guaranteeing stability of the closed-loop system and making it possible to use the efficient algorithms presented before [122, 123].

Modelling of Multibody Systems Aimed at Optimisation Multibody systems typically have involved nonlinear dynamics, which make them hard to optimise especially in real time. By using nonminimal coordinates it is possible to significantly reduce both the symbolic complexity of the model and the nonlinearities, therefore making it possible to optimise multibody systems reliably and fast. In this thesis we model tethered airplanes using the Euler-Lagrange approach, Cartesian coordinates and nonminimal parametrisations of the rotation group. In order to do so, we use the Euler-Lagrange approach and obtain an index-3 differential-algebraic equation (DAE) for which we treat the problems of reducing the DAE index and stabilising the model invariants [52, 51].

Power Maximisation, Control and Estimation of Tethered Airfoils for Energy Production The potential of tethered airfoils for energy production has been demonstrated in the seminal paper [75]. In particular, as opposed to conventional wind turbines, tethered airplanes have a much lighter structure and are able to fly at higher altitudes, therefore tapping into stronger and more consistent wind. In this context, optimal control is a valuable tool which makes it possible to

design the trajectories which extract the maximum power from the airmass. In this thesis, we develop a strategy to reliably solve trajectory planning problems by using tailored models and efficient optimal control formulations [49]. We compare single and dual-airfoil systems for energy harvesting and conclude that dual airfoils seem to be more advantageous [119, 124]. As tethered airfoils are highly nonlinear constrained systems, automatic flight is a challenging task that we propose to address by means of nonlinear MPC [50, 121, 126]. In order to achieve the required high computational speeds, we make use of the real-time iteration scheme [33]. Finally, we demonstrate the effectiveness of using a tuned tracking MPC formulation to approximate economic MPC on a power-extracting airfoil.

1.2 Outline and Structure of the Thesis

This thesis is structured as follows. First we introduce numerical methods for optimisation, simulation and optimal control, then we already present a challenging application example: we analyse stability of model predictive control (MPC), with a particular focus on so-called economic MPC. In order to use optimal control and MPC for fast mechanical systems, we propose a modelling strategy for multibody systems and apply it to tethered airplanes. Finally, we apply the proposed techniques to tethered airplanes and autonomous ground vehicles. In the following we outline the content chapter by chapter.

In Chapter 2 we treat optimisation, simulation and optimal control. Nonconvex optimisation is introduced first together with two popular classes of optimisation algorithms. Dynamic systems and numerical methods for simulating them are introduced next and optimal control is introduced with a focus on direct methods. Finally, real-time optimal control and estimation, i.e. model predictive control and moving horizon estimation (MHE), are introduced.

In Chapter 3 we consider autonomous driving, where the knowledge of the tire-road friction coefficient is particularly important. We use MHE to simultaneously estimate the state and the friction coefficient and MPC to control the vehicle on an obstacle avoidance scenario on a low-friction road. Finally, we present a preliminary study which aims towards time-optimal race car driving.

In Chapter 4 we analyse stability of MPC. First we consider the simpler case of tracking MPC, where the cost function attains its minimum at the prescribed steady-state reference. We then analyse the stability of economic MPC, which makes use of a generic cost function. We then extend the results obtained for the steady-state case to the periodic economic case.

In Chapter 5, we prove that it is possible to tune tracking MPC with a quadratic cost function such that it locally behaves like economic MPC. We consider both the case of optimal steady-state operation and optimal periodic operation. We propose an approach to compute the quadratic cost in practice and demonstrate the technique using some examples.

In Chapter 6 we propose a modelling approach for multibody systems which allows one to derive models of low complexity specifically tailored to optimisation and optimal control. First we develop a strategy to use the Euler-Lagrange approach with nonminimal coordinates such that the model has a low complexity. Then, we address the problem of stabilising the model invariants so that errors in the numerical integration do not get propagated through the dynamics. Finally, we apply the proposed framework to the modelling of tethered airplanes.

In Chapter 7 we use the previously developed techniques to plan power optimal trajectories for power-harvesting tethered airfoils. We then develop a control scheme based on MHE and MPC in order to control the airfoils in real-time. We compare single and dual-airfoil systems and propose a simple model to model the airfoil-airmass interaction. Finally, we apply the tuning technique developed in Chapter 5 to a single airfoil system to demonstrate the effectiveness of the technique also for practical applications.

Chapter 2

Optimisation, Simulation, Optimal Control and Estimation

In this chapter we present optimal control and estimation problems and the numerical methods to solve them. We will first introduce static optimisation problems and present some popular numerical methods for solving them in Section 2.1. In Section 2.2, we will define dynamic systems and initial value problems, i.e. the simulation of a dynamic system over time. Numerical methods for the simulation of dynamic systems will also be presented.

Optimal control problems will be introduced in Section 2.3. When presenting the numerical approaches for solving optimal control problems, we will focus on direct methods. The connection with the preceding sections will then become clear, as direct methods discretise the continuous-time dynamics in order to obtain a nonlinear programming problem.

Closed-loop control strategies typically rely on an observer which estimates the current state based on a set of measurements and a controller which computes the feedback control action based on the current state estimate. This can be done by solving optimal control and estimation problems online. Efficient tailored algorithms are therefore needed in order to make the solution of these problems real-time feasible. In Section 2.4 we will first define the optimal control and estimation problems and subsequently introduce the numerical methods which make a real-time solution of the problems possible.

Notation In this chapter, we present topics from different domains, where the same function name can have different meanings. Therefore, we use the bar notation when we refer to generic optimisation functions, e.g. $\bar{f}$, $\bar{g}$ and $\bar{h}$, and we use f^{c} to denote generic continuous-time dynamic systems and f to denote generic discrete-time dynamic systems.

Acknowledgement Most of this chapter is a review of the state of the art on the topic. Great sources of inspiration for the writing of this chapter were [80, 18, 14, 56, 57, 30, 31, 103, 95]. The main author of [95] is Rien Quirynen, Mario Zanon helped in the derivation of the obtained results and served as discussion partner.

2.1 Static Optimisation

We define an optimisation problem as follows

$$\underset{w}{\text{minimise}} \quad \bar{f}(w) \tag{2.1a}$$

$$\text{subject to} \quad \bar{g}(w) = 0, \tag{2.1b}$$

$$\bar{h}(w) \geq 0, \tag{2.1c}$$

where $w \in \mathbb{R}^{n_w}$ is the vector of *decision variables*, also called *optimisation variables*, $\bar{f} : \mathbb{R}^{n_w} \to \mathbb{R}$ is the *objective* or *cost function* and functions $\bar{g} : \mathbb{R}^{n_w} \to \mathbb{R}^{n_{\mathrm{eq}}}$ and $\bar{h} : \mathbb{R}^{n_w} \to \mathbb{R}^{n_{\mathrm{ieq}}}$ define the equality and inequality constraints respectively. We define the index set of equality and inequality constraints as $\mathcal{E} := \{1, \ldots, n_{\mathrm{eq}}\}$ and $\mathcal{I} := \{1, \ldots, n_{\mathrm{ieq}}\}$. In the general case, Problem (2.1) is called a Nonlinear Programming (NLP) problem.

Let us provide some basic definitions.

Definition 2.1 (Domain)**.** *We define the* domain *or* feasible set *as follows*

$$\mathcal{F} := \{w \mid \bar{g}(w) = 0,\ \bar{h}(w) \geq 0\}. \tag{2.2}$$

Definition 2.2 (Feasible Point)**.** *A point w is called* feasible *iff $w \in \mathcal{F}$.*

We call Problem (2.1) *feasible* if a feasible point exists.

Definition 2.3 (Global minimum)**.** *A point w^* is called* global minimum *iff*

$$\bar{f}(w^*) \leq \bar{f}(w), \qquad \forall\ w \in \mathcal{F}. \tag{2.3}$$

Definition 2.4 (Local minimum)**.** *A point w^* is called* local minimum *iff there exists an open ball of radius ϵ around w^*, denoted by $\mathcal{B}_\epsilon(w^*)$, such that*

$$\bar{f}(w^*) \leq \bar{f}(w), \qquad \forall\ w \in \mathcal{F} \cap \mathcal{B}_\epsilon(w^*). \tag{2.4}$$

A special category of optimisation problems is that of *convex optimisation problems*, defined as follows.

Definition 2.5 (Convex optimisation problem)**.** *An optimisation problem of the form* (2.1) *is* convex *if it has a convex feasible set $\mathcal{F}$ given by* (2.2) *and the cost function $\bar{f} : \mathcal{F} \to \mathbb{R}$ is convex.*

Theorem 2.6. *For a convex optimisation problem, every local minimum is also a global one.*

Proof. The proof can be found in [80]. □

Definition 2.7 (Lagrangian, Lagrange multipliers)**.** *The Lagrangian is defined as*

$$\mathcal{L}(w, \lambda, \mu) = \bar{f}(w) - \lambda^\top \bar{g}(w) - \mu^\top \bar{h}(w), \tag{2.5}$$

where vectors $\lambda \in \mathbb{R}^{n_{\text{eq}}}$ and $\mu \in \mathbb{R}^{n_{\text{ieq}}}$ are referred to as Lagrange multipliers*, or* dual variables*. The Lagrange multipliers can be interpreted as shadow prices, i.e. they define the cost improvement that one could obtain by relaxing the constraints.*

Definition 2.8 (Active Constraint)**.** *A constraint $\bar{h}_i : \mathbb{R}^{n_{\text{ieq}}} \to \mathbb{R}$, $i \in \mathcal{I}$ is called* active *iff*

$$\bar{h}_i(w) = 0,$$

for a feasible point $w \in \mathcal{F}$.

Definition 2.9 (Active Set)**.** *For a feasible point $w \in \mathcal{F}$, we define the* active set *$\mathcal{A}(w)$ as the set of all indices relative to the active constraints, i.e.*

$$\mathcal{A}(w) := \{i \mid \bar{h}_i(w) = 0\} \subset \mathcal{I}. \tag{2.6}$$

Note that sometimes the equality and inequality constraints are indexed together and the active set includes also all indices related to the equality constraints.

2.1.1 Local Optimality Conditions

In this section, we will define the conditions that define local optimality. Before stating necessary and sufficient conditions, we need to characterise the constraints using a so-called *constraint qualification*. Constraint qualifications are necessary in order to ensure that the constraint linearisation captures the essential geometric features of the feasible set in a neighbourhood of x^* [80]. While several constraint qualifications exist, we will only present one, which is not the weakest, but which serves well for the purpose of this chapter. For more details on constraint qualifications, we refer to [80].

Definition 2.10 (Linear Independence Constraint Qualification)**.** *A feasible point $w \in \mathcal{F}$ satisfies the Linear Independence Constraint Qualification (LICQ) iff the gradients $\nabla \bar{g}_i(w)$, $i \in \mathcal{E}$ and $\nabla \bar{h}_i(w)$, $i \in \mathcal{A}(w)$ are linearly independent.*

First order necessary conditions for optimality are given by the following theorem.

Theorem 2.11 (Karush-Kuhn-Tucker (KKT) Conditions)**.** *Let w^* be a local solution to* (2.1) *that satisfies the LICQ. Then there exist Lagrange multipliers λ^* and μ^* such that the KKT conditions hold:*

$$\nabla_w \mathcal{L}(w^*, \lambda^*, \mu^*) = 0, \tag{2.7a}$$

$$\bar{g}(w^*) = 0, \tag{2.7b}$$

$$\bar{h}(w^*) \geq 0, \tag{2.7c}$$

$$\mu^* \geq 0, \tag{2.7d}$$

$$\mu_i^* \bar{h}_i(w^*) = 0, \qquad i = 0, \ldots, n_{\text{ieq}}. \tag{2.7e}$$

Proof. The proof can be found in [80]. □

Definition 2.12. *A point* $(w^*,\ \lambda^*,\ \mu^*$ *is called* KKT point *or* primal-dual solution *if it satisfies the KKT conditions* (2.7)*.*

Equations (2.7e) are called *complementarity conditions* and imply that either constraint $\bar{h}_i(w^*)$ is active, i.e. $\bar{h}_i(w^*) = 0$, or $\mu_i^* = 0$, or possibly both.

Definition 2.13 (Weakly and Strictly Active Constraints)**.** *Consider a KKT point $(w^*,\ \lambda^*,\ \mu^*)$ and let us define sets*

$$\mathcal{A}_{\text{w}}(w) := \{i \in \mathcal{A} \mid \mu_i^* = 0\} \subset \mathcal{A}(w), \qquad \mathcal{A}_{\text{s}}(w) := \mathcal{A}(w) \setminus \mathcal{A}_{\text{w}}(w).$$

Constraints $\bar{h}_i(w^*),\ i \in \mathcal{A}_{\mathrm{w}}(w)$ *are called* weakly active. *Constraints* $\bar{h}_i(w^*),\ i \in \mathcal{A}_{\mathrm{s}}(w)$ *are called* strictly active.

A special case of complementarity is important for defining some properties of the NLP.

Definition 2.14 (Strict Complementarity)**.** *A KKT point* $(w^*,\ \lambda^*,\ \mu^*)$ *satisfies the* strict complementarity *condition iff there are no weakly active constraints, i.e.* $\mu_i^* > 0$*, for all* $i \in \mathcal{A}(w^*)$*.*

Theorem 2.15 (Uniqueness of the Lagrange Multipliers)**.** *Let a KKT point* $(w^*,\ \lambda^*,\ \mu^*)$ *satisfy LICQ and strict complementarity. Then the Lagrange multipliers* $(\lambda^*,\ \mu^*)$ *are unique.*

Proof. The proof can be found in [18, Theorem 4.15]. □

We define the Jacobian of the strictly active constraints as

$$G_{\mathrm{sa}} = \begin{bmatrix} \nabla \bar{g}(w^*) & \nabla \bar{h}_{\mathcal{A}_{\mathrm{s}}(w^*)}(w^*) \end{bmatrix}^\top,$$

and the set

$$\mathcal{N}_s := \{d \mid G_{\mathrm{sa}} d = 0\}.$$

We are now ready to state sufficient conditions for local optimality.

Definition 2.16 (Strong Second Order Sufficient Conditions)**.** *Let* $(w^*,\ \lambda^*,\ \mu^*)$ *be a KKT point of* (2.1) *satisfying LICQ. If the Hessian of the Lagrangian is strictly positive definite on the nullspace of the strictly active constraints, i.e.*

$$d^\top \nabla_w^2 \mathcal{L}(w^*, \lambda^*, \mu^*) d > 0, \qquad \forall\ d \in \mathcal{N}_s \setminus \{0\}, \tag{2.8}$$

then w^* *is a local minimiser of* (2.1)*.*

Proof. The proof is given in [80]. □

Note that a weaker sufficient condition exists, which requires that

$$d^\top \nabla_w^2 \mathcal{L}(w^*, \lambda^*, \mu^*) d > 0, \quad \forall\ d \in \{d \in \mathcal{N}_s \mid \nabla \bar{h}_{\mathcal{A}_{\mathrm{w}}(w^*)}(w^*)^\top d \geq 0\} \setminus \{0\}. \tag{2.9}$$

We have preferred condition (2.8), as it guarantees that the KKT point is stable against perturbations in the problem functions $\bar{f}$, $\bar{g}$ and $\bar{h}$ [31].

2.1.2 Numerical Methods

In this section, we give a brief overview of some popular numerical methods for solving NLPs of the form (2.1). For a detailed discussion on the topic we refer the reader to e.g. [80, 25] and references therein.

We focus on methods which aim at finding a local minimum by computing a KKT point $v^* = (w^*,\ \lambda^*,\ \mu^*)$ of a given NLP (2.1). If the KKT conditions (2.7) were smooth, one could apply Newton's method in order to compute a KKT point. Unfortunately, Conditions (2.7c)-(2.7e) are nonsmooth and need special care. In the following, we present two popular approaches: Interior Point (IP) methods and Sequential Quadratic Programming (SQP).

Interior Point Methods

The basic idea behind IP methods consists in handling inequality constraints by means of a barrier function, scaled by a barrier parameter. The barrier parameter is initially set to a high value and progressively reduced as the algorithm converges.

IP methods have proven to be highly competitive especially, but not only, for convex problems. One main disadvantage of IP is the need of always going back to the so-called *central path*, even if an initial guess very close to the solution is available (for a proper definition of central path we refer to [80]). This means that even when the solution is very close to the initial guess, the solver needs to start with a large barrier parameter and follow some iterates which take it away from the solution and only afterwards come back to it. This fact can be limiting if one is interested in solving a series of parametric problems for small variations of the parameter: in this context SQP is typically much faster, as it can be warm-started.

In IP methods, a vector of slack variables $s \in \mathbb{R}^{n_{\text{ieq}}}$ is usually introduced and the inequality constraints are reformulated as:

$$\bar{h}(w) - s = 0, \qquad s \geq 0. \tag{2.10}$$

Primal Interior Point Primal interior point methods replace the inequality by adding a barrier function to the cost, e.g. $-\tau \sum_{i=0}^{n_{\text{ieq}}} \log(s_i)$ and solve

$$\underset{w,s}{\text{minimise}} \quad \bar{f}(w) - \tau \sum_{i=0}^{n_{\text{ieq}}} \log(s_i) \tag{2.11a}$$

$$\text{subject to} \quad \bar{g}(w) = 0, \tag{2.11b}$$

$$\bar{h}(w) - s = 0, \tag{2.11c}$$

Note that, for $\tau = 0$ the barrier function becomes the indicator function, i.e.

$$\lim_{\tau \to 0} -\tau \log(s) = i(s) := \begin{cases} 0 & \text{for } s \geq 0 \\ \infty & \text{otherwise} \end{cases}. \tag{2.12}$$

We define $S = \text{diag}(s)$ and $e = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^\top \in \mathbb{R}^{n_{\text{ieq}}}$, then the KKT conditions of this problem read

$$\nabla \bar{f}(w^*) - \nabla \bar{g}(w^*)^\top \lambda^* - \nabla \bar{h}(w^*)^\top \mu^* = 0, \tag{2.13a}$$

$$-\tau (S^*)^{-1} e + \mu^* = 0, \tag{2.13b}$$

$$\bar{g}(w^*) = 0, \tag{2.13c}$$

$$\bar{h}(w^*) - s^* = 0. \tag{2.13d}$$

Note that the conditions $s_i^* \geq 0$ and $\mu_i^* \geq 0$ must be enforced by selecting an appropriate step size [80].

Starting from a rather large value for τ, the approach consists in iteratively solving Problem (2.11) and reducing the value of the barrier parameter τ. Once τ reaches a prescribed tolerance, the problem is solved one last time and the solution is given as output.

Main disadvantages of primal interior point methods consist in the need of first finding a feasible initial guess, and in the fact that Equation (2.13b) becomes quite nonlinear near the solution, as $s \to 0$.

Primal-Dual Interior Point Noting that Equation (2.13b) becomes highly nonlinear when approaching the solution, primal-dual interior point methods

multiply it by matrix S^* and solve the equivalent system

$$\nabla \bar{f}(w^*) - \nabla \bar{g}(w^*)^\top \lambda^* - \nabla \bar{h}(w^*)^\top \mu^* = 0, \tag{2.14a}$$

$$-\tau e + S^* \mu^* = 0, \tag{2.14b}$$

$$\bar{g}(w^*) = 0, \tag{2.14c}$$

$$\bar{h}(w^*) - s^* = 0, \tag{2.14d}$$

with $s^* \geq 0$ and $\mu^* \geq 0$. Note that, if $\tau = 0$ we recover the standard KKT conditions (2.7).

It is not obvious from the primal-dual IP KKT conditions that the method will deliver solutions which lie in the interior of the feasible domain. This only becomes clear when one recalls the equivalence with primal IP methods.

Sequential Quadratic Programming

As opposed to IP methods, SQP does not relax complementary slackness (Equation (2.7e)), but computes the step by using a local quadratic approximation of the NLP and solving a Quadratic Program (QP) at each iteration. Starting from an initial guess $v^{(0)} = (w^{(0)},\ \lambda^{(0)},\ \mu^{(0)})$, SQP computes a sequence of points v_k using

$$v^{(k+1)} = v^{(k)} + \alpha^{(k)} \Delta v^{(k)}, \tag{2.15}$$

with $\alpha^{(k)} \in (0, 1]$ and

$$\Delta v^{(k)} = (\Delta w^{(k)},\ \tilde{\lambda}^{(k)} - \lambda^{(k)},\ \tilde{\mu}^{(k)} - \mu^{(k)}), \tag{2.16}$$

obtained as the solution $(\Delta w^{(k)},\ \tilde{\lambda}^{(k)},\ \tilde{\mu}^{(k)})$ of the QP subproblem

$$\underset{\Delta w \in \Omega^{(k)}}{\text{minimise}} \quad \frac{1}{2} \Delta w^\top H^{(k)} \Delta w + \nabla \bar{f}(w^{(k)})^\top \Delta w \tag{2.17a}$$

$$\text{subject to} \quad \bar{g}(w^{(k)}) + \nabla \bar{g}(w^{(k)})^\top \Delta w = 0, \tag{2.17b}$$

$$\bar{h}(w^{(k)}) + \nabla \bar{h}(w^{(k)})^\top \Delta w \geq 0. \tag{2.17c}$$

Several variants of SQP exist which distinguish themselves in the computations of the step size $\alpha^{(k)}$, the so-called Hessian matrix approximation $H^{(k)}$ and the choice of the set $\Omega^{(k)} \in \mathbb{R}^{n_w}$. Under suitable choice of $H^{(k)}$, $\alpha^{(k)}$, $\Omega^{(k)}$ and the initial guess $v^{(0)}$, SQP is guaranteed to converge to v^* [80].

Note that, if $H^{(k)} = \nabla^2_{ww}\mathcal{L}(w^{(k)}, \lambda^{(k)}, \mu^{(k)})$ is used, then the KKT conditions of each QP subproblem (2.17) coincide with a special form of linearisation of the KKT conditions of NLP (2.1) evaluated at $w^{(k)}$. SQP with exact Hessian can be seen as Newton's method applied to the KKT conditions, where the nonsmooth conditions are handled by the QP solver. The algorithm thus has locally quadratic convergence.

QP Solvers Several approaches are available for solving the QP subproblems. In particular, we mention *active-set* methods, *interior point* methods and *first order* methods. The first two are called second-order methods, because they use the Hessian of the QP problem to compute the optimal solution.

Active-set methods use a strategy similar to the *simplex* algorithm for linear programming in order to identify the active set $\mathbb{A}$. In the following, we present as an illustrative example a primal active-set method. Starting from an initial guess of the active set $\mathbb{A}^{(0)}$, active-set methods iteratively solve the linear system

$$H^{(k)}\Delta\tilde{w}^{(k,i)} + \nabla\bar{f}(w^{(k)}) - \nabla\bar{g}(w^{(k)})\tilde{\lambda}^{(k,i)} - \nabla\bar{h}_{\mathbb{A}^{(i)}}(w^{(k)})\tilde{\mu}^{(k,i)}_{\mathbb{A}^{(i)}} = 0, \quad (2.18)$$

$$\bar{g}(w^{(k)}) + \nabla\bar{g}(w^{(k)})^\top\Delta\tilde{w}^{(k,i)} = 0, \quad (2.19)$$

$$\bar{h}_{\mathbb{A}^{(i)}}(w^{(k)}) + \nabla\bar{h}_{\mathbb{A}^{(i)}}(w^{(k)})^\top\Delta\tilde{w}^{(k,i)} = 0. \quad (2.20)$$

The primal solution of the linear system $\Delta\tilde{w}^{(k,i)}$ is used to refine the guess on the active set using

$$\Delta w^{(k,i)} = \Delta w^{(k,i-1)} + \alpha^{(i)}(\Delta w^{(k,i)} - \tilde{\Delta} w^{(k,i-1)}),$$

where the steplength $\alpha^{(i)} \in [0, 1]$ is chosen to be the largest possible value for which all constraints are satisfied. Note that this value can be computed explicitly in a cheap way. If $\alpha^{(i)} < 1$, then a new constraint is added to the guess for the active set $\mathbb{A}^{(i+1)}$. If $\alpha^{(i)} = 1$, and $\tilde{\mu}^{(k,i)}_{\mathbb{A}^{(i)}} \geq 0$, then the optimum is achieved. If however $\tilde{\mu}^{(k,i)}_j < 0,\ j \in \mathbb{A}^{(i)}$, then constraint j is eliminated from the guess of the active set and the iterations continue until the correct active set has been identified.

Interior point methods use instead a smooth approximation of Conditions (2.7c)-(2.7e) as outlined before for the NLP case. Also for the QP case, the iterative procedure starts with a large value for the barrier parameter τ and progressively reduces it as the iterates converge. The main advantage of IP solvers over active-set solvers is that they do not need to identify the active set, as this happens automatically. The main disadvantage is that they cannot make use of an initial guess to be warm started and, therefore, reduce the number of iterations needed to reach convergence.

First order methods neglect second order information in favour of cheaper computations: typically they need significantly more iterations to converge to the solution, but these iterations are so cheap that, if the method converges in a relatively small number of iterations, first order methods can be competitive for specific applications.

Sequential Convex Programming generalisation of SQP is sequential convex programming (SCP), where the sequence of subproblems does not necessarily involve QPs, but may involve any kind of convex problem. This can be advantageous when constraints involving norms or matrix inequalities are present.

Hessian Approximations and Convergence Rates

When solving an NLP using either an SQP or an IP method, one needs to compute the (exact or approximated) Hessian of the Lagrangian. Because computing the exact Hessian of the Lagrangian can be computationally demanding, in some cases using an approximated Hessian $H^{(k)}$ might yield faster computations. Note that in general the method will need more iterations to converge, but each iteration will be cheaper to compute. We mention in this section two Hessian approximations and, after defining the convergence rate, we also characterise the convergence rate obtained with each Hessian approximation.

Definition 2.17 (q-linear convergence rate)**.** *Let $w^{(k)} \in \mathbb{R}^{n_w}$ with $w^{(k)} \to w^*$ for $k \to \infty$. We say that $w^{(k)}$ converges* q-linearly *if there are constants $r \in (0,1)$ and $\bar{k} \in \mathbb{N}$ such that*

$$\frac{\|w^{(k+1)} - w^*\|}{\|w^{(k)} - w^*\|} \leq r, \qquad \forall\ k \geq \bar{k}.$$

Definition 2.18 (q-superlinear convergence rate)**.** *Let $w^{(k)} \in \mathbb{R}^{n_w}$ with $w^{(k)} \to w^*$ for $k \to \infty$. We say that $w^{(k)}$ converges* q-superlinearly *if*

$$\lim_{k \to 0} \frac{\|w^{(k+1)} - w^*\|}{\|w^{(k)} - w^*\|} = 0.$$

Definition 2.19 (q-quadratic convergence rate)**.** *Let $w^{(k)} \in \mathbb{R}^{n_w}$ with $w^{(k)} \to w^*$ for $k \to \infty$. We say that $w^{(k)}$ converges* q-quadratically *if there are constants $K \in (0, \infty)$ and $\bar{k} \in \mathbb{N}$ such that*

$$\frac{\|w^{(k+1)} - w^*\|}{\|w^{(k)} - w^*\|^2} \leq K, \qquad \forall\ k \geq \bar{k}.$$

Exact Hessian When using the exact Hessian of the Lagrangian, a q-quadratic convergence rate is obtained in a neighbourhood of the KKT point [80].

BFGS Hessian Approximation Starting from an initial guess of the Hessian of the Lagrangian, the BFGS update formula [27, 37, 43, 107] is used at each iteration to refine the Hessian approximation. The convergence rate obtained in this case is q-superlinear.

Gauss-Newton Hessian Approximation The Gauss-Newton Hessian approximation is used in the so-called *Generalised Gauss-Newton* method [20], which can be applied to solve NLPs where the objective function is of least-squares type, i.e.

$$\bar{f}(w) = \frac{1}{2}\|r(w)\|_2^2.$$

where $r : \mathbb{R}^{n_w} \to \mathbb{R}^{n_r}$ is the so-called residual vector. The Hessian of the Lagrangian is given by

$$\begin{aligned}\nabla^2_{ww}\mathcal{L}(w^{(k)}, \lambda^{(k)}, \mu^{(k)}) =& \nabla r(w^{(k)})\nabla r(w^{(k)})^\top + \sum_{i=1}^{n_r} r_i(w^{(k)})\nabla^2 r_i(w^{(k)}) \\ &- \sum_{i=1}^{n_r} \lambda_i^{(k)}\nabla^2 \bar{g}_i(w^{(k)}) - \sum_{i=1}^{n_r} \mu_i^{(k)}\bar{\nabla}^2 h_i(w^{(k)}).\end{aligned}$$

The Gauss-Newton Hessian approximation reads

$$H_{\mathrm{GN}}^{(k)} = \nabla r(w^{(k)})\nabla r(w^{(k)})^\top.$$

One can show that this approximation satisfies

$$\|\nabla^2_{ww}\mathcal{L}(w^{(k)}, \lambda^{(k)}, \mu^{(k)}) - H_{\mathrm{GN}}^{(k)}\| = \mathcal{O}(\|r(w^*)\|),$$

which means that if the optimal residual is small the Gauss-Newton approximation is good. The convergence rate obtained using the Gauss-Newton approximation is only linear [21], however, this approach performs extremely well in practice, provided that the optimal residual vector is small. The Generalised Gauss-Newton method finds interesting applications in e.g. system identification, real-time estimation and control.

Identity Matrix Approximation The gradient method, also known as steepest-descent method can be seen as an approximated Newton method, where the

Hessian of the Lagrangian is approximated by the identity matrix. The gradient method is known to converge at a linear rate. Unlike the Generalised Gauss-Newton method, it typically converges very slowly in practice.

Derivative Computation

In this chapter, we focus on derivative-based optimisation algorithms. In order to be able to compute a descent direction, these algorithms require at least the first-order derivatives of the functions $\bar{f}$, $\bar{g}$ and $\bar{h}$ from NLP (2.1). There exist many approaches for computing the derivatives, including *finite differences*, *algorithmic differentiation* (AD) and the *complex-step derivative*. The last approach can be cast in the category of algorithmic differentiation.

Finite differences are based on computing a directional derivative by perturbing that direction and taking the quotient between the difference of the nominal and perturbed function and the perturbation, i.e.

$$\frac{\partial f(x)}{\partial x} p \approx \frac{f(x + hp) - f(x)}{h},$$

in the case of the forward differences. This method is known to yield poor accuracy due to either truncation errors or cancellation errors [1], so that for forward differences one can expect a precision of no more than half of the number of significant digits.

AD consists essentially in applying the chain rule for differentiation and returns derivatives accurate up to machine precision. It relies on a representation of the function by so-called *elemental functions* [1], for which simple closed-form derivative expressions exist. AD uses these elemental functions to propagate the chain rule through the evaluation tree of the function.

In the so-called *forward mode*, the function and the derivatives are evaluated by following the normal evaluation order. As a result, one obtains the derivative of the function in one direction. The computation of n directional derivatives can be proven to cost no more than $1 + 1.5n$ times the cost of one function evaluation.

In the so-called *reverse mode* (or adjoint mode), the evaluation is done in reverse order. As a result, one obtains the so-called *adjoint derivative*, i.e. the derivative of one component of the function in all directions. The computation of n adjoint derivatives can be proven to cost no more than $1.5 + 2.5n$ times the cost of one function evaluation. This is particularly remarkable since the cost of the evaluation does not depend on the size of the input vector of the function.

If one is interested in computing the full Jacobian matrix, the cost of the evaluation in forward mode scales with the size of the input vector of the function, while in reverse mode it scales with the size of the function output vector. Note, however, that AD in reverse mode typically requires more memory allocation than in forward mode.

2.2 Dynamic Systems and Initial Value Problems

In this section we define the concepts of dynamic system and Initial Value Problem (IVP), i.e. the problem of simulating dynamic systems over time. Then, we briefly introduce numerical methods for solving IVPs.

2.2.1 Dynamic Systems and Initial Value Problems

Many real-world processes can be described mathematically by models in (a) ordinary differential equation (ODE), (b) differential-algebraic equation (DAE), or (c) partial differential equation (PDE) form. In this thesis, we will focus on the first two cases. We remark however, that a PDE can be reduced to an ODE or DAE, e.g. by applying the method of lines.

Definition 2.20 (Continuous time controlled dynamic system)**.** *Let us define time $t \in \mathbb{R}$, differential state $x \in \mathbb{R}^{n_x}$, algebraic state $z \in \mathbb{R}^{n_z}$, control $u \in \mathbb{R}^{n_u}$, parameter $p \in \mathbb{R}^{n_p}$. Then, a continuous time controlled dynamic system is defined by the dynamic equation*

$$f^{\mathrm{c}}(t, \dot{x}, x, z, u, p) = 0, \tag{2.21}$$

where we define $\dot{x} = \frac{\mathrm{d}x}{\mathrm{d}t}$. The distinction between differential and algebraic states x and z is made based on the dependence of the model on the time derivative of the state: all states whose time derivative appears in the model are differential states x and all those whose derivative does not appear in the model are algebraic states z. Note that the explicit dependence on the parameter p does not always need to be specified, however, it can be useful for some optimal control or estimation formulations.

Semi-implicit DAEs are an important subclass of controlled dynamic systems. They are defined as

$$B(t, x, z, u, p)\, \dot{x} = f^{\mathrm{c}}(t, x, z, u, p), \tag{2.22a}$$

$$0 = g^{\mathrm{c}}(t, x, z, u, p), \tag{2.22b}$$

where it is assumed that matrices $B(t,x,z,u,p) \in \mathbb{R}^{n_x \times n_x}$ and $\frac{\partial g^c}{\partial z}$ are regular. This guarantees that DAE (2.22) is of index 1 and can be converted into an explicit ODE (2.24). However, this is not advisable in practice as it is typically numerically more efficient to keep the system dynamics in the form (2.22). Note that there are techniques to reduce high index DAEs to index 1 DAEs or ODEs. For more details on the topic, we refer to [105].

Another important subclass of dynamic systems includes implicit and explicit ODEs, defined as

$$0 = f^c(t, \dot{x}, x, u, p), \tag{2.23}$$

$$\dot{x} = f^c(t, x, u, p). \tag{2.24}$$

Systems which do not explicitly depend on time t are called time invariant. Time-varying systems can be transformed in time-invariant systems by adding a state x_t with dynamics $\dot{x}_t = 1$. The dependence of the model on parameters p can also be suppressed as, in principle, one can always express the parameters as states x_p with dynamics $\dot{x}_p = 0$.

Definition 2.21 (Initial Value Problem)**.** *An Initial Value Problem (IVP) is defined by a dynamic system* (2.21)*, a time interval* $[t_0, t_f]$*, an initial value* $x(t_0)$*, parameter* p *and control* $u(t)$*,* $t \in [t_0, t_f]$*.*

$$\begin{cases} f^c(t, \dot{x}, x, z, u, p) = 0, \quad t \in [t_0, t_f], \\ x(t_0) = x_0 \end{cases} \tag{2.25}$$

The solution to the IVP is the trajectory $x(t)$*,* $z(t)$*,* $t \in [t_0, t_f]$*.*

Note that, in general, IVPs do not have an analytic solution, but can be solved numerically. More details on numerical methods for solving IVPs will be given in the next section.

Theorem 2.22 (Uniqueness of the solution to IVPs)**.** *Consider the IVP*

$$\begin{cases} \dot{x} = f^c(t, x), \quad t \in [t_0, t_f], \\ x(t_0) = x_0, \end{cases} \tag{2.26}$$

where $f^c \in [t_0, t_f] \times \mathbb{R}^{n_x}$ *is continuous in its first variable and Lipschitz in its second variable, i.e.*

$$\|f^c(t, x) - f^c(t, y)\| \leq L_f^c \|x - y\|.$$

Then there exists a unique solution to the IVP (2.26)*.*

Proof. The proof is given in [30]. □

The result of Theorem 2.22 can be extended to the case of the more complicated IVP (2.25). For parameters p, as they are fixed, the extension is straightforward. For the result to also hold for controls u it is necessary to assume that they only have a finite number of discontinuities. Finally, the extension to index-1 DAEs requires that the matrix $\frac{\partial f^{\mathrm{c}}}{\partial z}$ is invertible. Note that in this case the DAE can be transformed into an ODE.

2.2.2 Numerical Methods for IVPs

Solving an IVP is equivalent to computing the state transition map $x_{k+1} = f(x_k, u_k)$ on a prescribed sampling time. As this can rarely be computed exactly, numerical schemes which compute an approximate solution to the IVP have been developed in the literature. In this section we will give a brief introduction on numerical methods for solving IVPs. For more details on the topic, we refer the reader to e.g. [56, 57] and the references therein.

For simplicity, we first consider a simple ODE of the form

$$\dot{x}(t) = f^{\mathrm{c}}(t, x(t)). \tag{2.27}$$

Note, however, that these methods can be extended in order to simulate DAEs of the form (2.21).

The time interval $[t_0, t_{\mathrm{f}}]$ over which the simulation is performed is subdivided into a time grid $t_0 < t_1 < \ldots < t_{n_{\mathrm{sim}}}$. In the following, we will use index j when referring to a discretisation related to the simulation of a dynamic system. We define the so-called *step size* as $h_j = t_{j+1} - t_j$. For simplicity, in the following we will consider a fixed step size, i.e. $h_j = h = \frac{t_{\mathrm{f}} - t_0}{n_{\mathrm{sim}}}$.

The simplest numerical scheme is the Explicit Euler scheme

$$x_{j+1} = h f^{\mathrm{c}}(t_j, x_j) + x_j. \tag{2.28}$$

One can show that, for a sufficiently small stepsize h_j, the sequence x_j converges to the exact solution of ODE (2.27) $x(t_j)$. Unfortunately, this method is usually not the most computationally efficient one. In the following we will briefly introduce a class of simulation methods which can be computationally more efficient.

Runge-Kutta Methods

The integration step of an s-stage Runge-Kutta (RK) method with fixed step size h is defined as

$$k_i = f^{\mathrm{c}}\left(t_j + c_i h,\ x_j + h\sum_{l=1}^{s} a_{il}k_l\right), \qquad i = 1,\dots,s, \tag{2.29a}$$

$$x_{j+1} = x_j + h\sum_{i=1}^{s} b_i k_i. \tag{2.29b}$$

Runge-Kutta methods are hence uniquely defined by the stepsize and the coefficients

$$\begin{array}{c|ccc} c_1 & a_{11} & \dots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array} \tag{2.30}$$

here reported in the so-called *Butcher tableau*. As an example, we report the Butcher tableau of the explicit Euler method, which is of order 1 (ERK1), an explicit Runge-Kutta method of order 4 (ERK4) and an implicit Runge-Kutta method of order 4 (IRK4).

ERK1

$$\begin{array}{c|c} 1 & 0 \\ \hline & 1 \end{array}$$

ERK4

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 0 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

IRK4

$$\begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

In case matrix $A = [a_{il}]$ is lower triangular, each stage k_i is defined only by stages k_m, $m < i$ and each stage can be computed explicitly. Such methods are called *explicit* RK methods. All other methods are called *implicit* RK methods, because each stage k_i is defined implicitly and, in order to compute the integration step, it is necessary to solve a linear system with size $n_x s$.

Clearly, implicit RK methods require heavier computations than their explicit counterpart with the same amount of stages. The advantage of implicit methods over explicit ones is the increased numerical stability [57]. Indeed, implicit methods perform particularly well for so-called *stiff systems*.

Multistep Methods

An alternative approach to RK methods is given by multistep methods, which use a linear combination of the previous integration steps in order to increase the order of accuracy. These approaches can become more competitive than implicit RK methods for systems with a large state space, as the numerical complexity scales better with the system size [56].

Variable Step Size

In general, the step size h_j does not need to be constant and can be chosen adaptively, so as to take longer steps when the integration accuracy is high enough and to reduce the step size when the accuracy is deteriorating too much. We omit the details on variable step size integrators, but remark that such numerical schemes are able to guarantee a prescribed level of accuracy.

Sensitivity Computation

In Section 2.3, it will become clear that, in the context of shooting methods for optimal control, it is necessary to compute the state transition map, possibly the integral of a given function of the trajectory $L(x(t), u(t))$, and their derivatives, i.e. the so-called sensitivities:

$$f(x_k, u_k), \qquad S_x = \frac{\partial f(x_k, u_k)}{\partial x_k}, \qquad S_u = \frac{\partial f(x_k, u_k)}{\partial u_k}, \tag{2.31}$$

$$\ell(x_k, u_k), \qquad \frac{\partial \ell(x_k, u_k)}{\partial x_k}, \qquad \frac{\partial \ell(x_k, u_k)}{\partial u_k}, \tag{2.32}$$

where we lumped all operations needed by the selected numerical method in functions $x(t_{k+1}) = f(x_k, u_k)$ and $\ell(x_k, u_k) = \int_{t_k}^{t_{k+1}} L(x(t), u(t))\,\mathrm{d}t$. Note that a simple way to compute ℓ would be to introduce the additional state

$$\dot{x}_\ell(t) = L(x(t), u(t)). \tag{2.33}$$

However, in some cases this could lead to inefficient computations.

A straightforward way of computing first-order sensitivities consists in augmenting the system dynamics with the so-called Variational Differential Equations (VDE) and simulating the augmented system over the time $t \in [0, T_\mathrm{s}]$.

For an ODE, the VDEs are given by

$$\dot{\bar{S}}_x(t) = \frac{\partial f^{\mathrm{c}}(t, x(t), u(t))}{\partial x} \bar{S}_x(t),$$

$$\dot{\bar{S}}_u(t) = \frac{\partial f^{\mathrm{c}}(t, x(t), u(t))}{\partial x} \bar{S}_u(t) + \frac{\partial f^{\mathrm{c}}(t, x(t), u(t))}{\partial u},$$

$$\bar{S}_x(0) = I, \qquad \bar{S}_u(0) = 0, \qquad S_x = \bar{S}_x(T_{\mathrm{s}}), \qquad S_u = \bar{S}_u(T_{\mathrm{s}}),$$

with $\bar{S}_x(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}x_0}$ and $\bar{S}_u(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}u}$, with x_0 the initial value of the IVP. The extension to DAEs of the form (2.21) is straightforward. In case of an explicit integration method, the VDEs are equivalent to AD applied in forward mode to the integration scheme. For implicit schemes, this differentiation becomes a bit more involved, but can still be done efficiently. For more details on integrators with sensitivities, we refer to [94, 1] and references therein.

2.3 Optimal Control

When formulating control problems for a given system, often the control engineer must consider three main aspects:

1. model the system dynamics appropriately,
2. define the control task,
3. define constraints that the system must satisfy.

As anticipated in the previous section, in many cases Task 1 leads to an ODE or DAE model. Task 2 can often be formulated as the minimisation of a given objective, e.g. execute a given task in minimal time, follow a prescribed path as accurately as possible, execute the prescribed task using minimal control effort, etc. Task 3 is often related to Task 2 and includes constraints such as e.g. prescribed initial and final conditions, bounds on the control variables, bounds on the region of the state space in which the system must remain, periodicity conditions, etc. In some cases, the control engineer might also be interested in jointly controlling the system and choosing some system parameters. Consider e.g. a rocket being sent to the moon: the quantity of fuel necessary in order to complete the mission determines the size of the rocket, which, in turn, affects the fuel consumption.

Optimal control provides an excellent framework to deal with such problems. We define an optimal control problem (OCP) as the following optimisation

problem

$$\underset{x(\cdot),z(\cdot),u(\cdot),p}{\text{minimise}} \quad \int_0^T L(x(t),z(t),u(t),p)\ \mathrm{d}t + E(x(T),z(T),p) \tag{2.34a}$$

$$\text{subject to} \quad f^{\mathrm{c}}(t,\dot{x}(t),x(t),z(t),u(t),p) = 0, \qquad t \in [0,T] \tag{2.34b}$$

$$h(x(t),z(t),u(t),p) \geq 0, \qquad t \in [0,T] \tag{2.34c}$$

$$r(x(0),x(T)) = 0, \tag{2.34d}$$

where one is interested in finding the state and control trajectories $x(t), z(t), u(t)$ and the parameters p which minimise the objective functional (2.34a)

$$J(x(\cdot),u(\cdot),p) = \int_0^T L(x(t),z(t),u(t),p)\ \mathrm{d}t + E(x(T),z(T),p),$$

while satisfying the dynamic constraints (2.34b), the so-called path constraints (2.34c) and given boundary constraints (2.34d). Without loss of generality, we consider here an initial time $t_0 = 0$. Note that the final time T can be an element of vector p, i.e. an optimisation parameter itself. Then the OCP is called a *free final time optimal control problem*.

2.3.1 Numerical Methods for Optimal Control

Numerical methods for solving OCPs are typically classified in three main categories:

1. Hamilton-Jacobi-Bellman (HJB) and Dynamic Programming (DP) approaches
2. Indirect methods, also known as *first optimise, then discretise*
3. Direct methods, also known as *first discretise, then optimise*

In the following, we will give a brief overview on these techniques. For further reading we refer to [14, 12, 13, 28, 87].

We begin by introducing a fundamental theorem of optimal control: Bellman's principle of optimality.

Theorem 2.23 (Principle of optimality [12])**.** *An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.*

Suppose that we follow an optimal trajectory for an arbitrary amount of time and then solve again the optimal control problem on the remaining time horizon starting from the current state. The solution that we obtain must be exactly the remaining part of the optimal trajectory that we were following. If this was not the case, then we could have used the newly computed trajectory to construct a feasible trajectory for the initial OCP which has a lower cost than the solution we obtained in the beginning. However, this would entail that the original solution was not optimal, which contradicts our assumption.

HJB and DP approaches are based on Bellman's principle of optimality: starting from the end of the considered time horizon, they propagate backwards in time the cost-to-go function $V(t, x)$, also called value function, and defined as the cost of applying the optimal control move starting at time t from state x. If the back-propagation of the cost-to-go function is done in continuous time, one obtains the Hamilton-Jacobi-Bellman partial differential equation. If instead the propagation is done in discrete time, one obtains the dynamic programming recursion. In both cases, the global optimum is found. However, the value function needs to be tabulated on a state space discretisation, therefore both approaches suffer from what Bellman called the *curse of dimensionality* [12]. In the special case of control and estimation of linear unconstrained systems with a quadratic cost function, an analytical solution exists.

Indirect methods state the infinite dimensional first order necessary conditions for optimality, typically by formulating *Pontryagin's maximum principle* [87], and solve them numerically by discretising them into a multipoint boundary value problem (BVP). As opposed to HJB/DP, this approach does not suffer from the curse of dimensionality, but it only delivers a local minimum. Indirect methods suffer from at least three major difficulties [14]:

1. The computation of the first order necessary conditions cannot be done automatically and "requires an intelligent user with at least some knowledge of optimal control theory" [14]. Moreover, the derivation can become very difficult and each time a new problem is posed or a problem is slightly modified, a new derivation is necessary.

2. In the presence of path constraints, an a priori estimate on the sequence of the constrained-arc sequence needs to be made. Clearly, such an estimate is quite difficult to obtain, as there is no simple way to guess the number and sequence of the arcs.

3. The method is very sensitive to the provided initial guess and, quoting J. T. Betts: "Even with a reasonable guess for the adjoint variables, the numerical solution of the adjoint equations can be very ill-conditioned!" [14].

Indirect methods can however play an important role for theoretical considerations and can reach very high accuracies.

Direct methods directly discretise the OCP in order to formulate a nonlinear program (NLP). In order to do so, the infinite-dimensional control function is parametrised in time using a finite number of parameters, the dynamic constraints (2.34b) are converted into a finite set of constraints and the path constraints (2.34c) are enforced only on a finite set of time instants. The finite dimensional problem is then solved using an NLP solver. Direct methods are able to handle systems with large state and control spaces subject to path constraints in a very natural way. The derivation of the optimality conditions is automatic and no a-priori knowledge on the structure of the constrained arcs is needed. Because of all these features, direct methods have seen a great success for practical applications and therefore this thesis will focus on direct methods. Similarly to indirect methods, usually only local optimality can be attained.

2.3.2 Direct Methods for Optimal Control

Several numerical schemes can be described as direct methods for Optimal Control. The most known methods have been derived for centralised systems. However, recent interest in distributed systems has resulted in algorithms tailored to this class of systems. We will first present centralised algorithms, which are also the most well known.

The three most known approaches for direct optimal control are

1. (direct) single shooting,
2. (direct) multiple shooting,
3. (direct) collocation.

As we only discuss direct methods, in the following we will drop the term “direct” from the names. It is however important to remember that these algorithms also exist for indirect methods.

For all three approaches, one needs to define a time grid $t_0 < t_1 < \ldots < t_N$ of $N + 1$ time points. Note that the grid does not need to be equidistant, though in practice it is often chosen equidistant, i.e. $t_{k+1} - t_k = T_\mathrm{s}$. The time instants t_k are often referred to as *shooting nodes* in the context of the multiple shooting method. The control function is then parametrised on the time grid, e.g. using basis functions with local support. A popular choice is to take a piecewise constant control parametrisation, though for collocation piecewise polynomial

control parametrisations are also often used. Common to the three approaches is also the handling of path constraints, which are only imposed at a finite number of time instants. A popular choice is to impose the path constraints on the time grid $t_0, t_1, \ldots, t_N$.

The main difference between the three methods is in how the system dynamics are handled. Let us define the state transition map and cost evaluation as

$$x_{k+1} = f(x_k, u_k), \qquad \ell(x_k, u_k) = \int_{t_k}^{t_{k+1}} L(x(t), u(t))\ \mathrm{d}t. \tag{2.35}$$

which results from the solution of an IVP and from computing the integral. For the sake of simplicity, we dropped the dependence on k, $z(t)$ and p and assumed an equidistant time grid $t_{k+1} - t_k = T_\mathrm{s}$. We moreover define $U_k = (u_0, u_1, \ldots, u_{k-1})$, $U_0 = ()$, and introduce the composition of subsequent IVPs and integrations as

$$f_{k+1}(x_0, U_k) = f\left(f_k(x_0, U_k), u_k\right), \qquad f_0(x_0, U_0) = x_0, \tag{2.36}$$

$$L_{k+1}(x_0, U_{k+1}) = \ell(f_k(x_0, U_k), u_{k+1}) + L_k(x_0, U_{k-1}). \tag{2.37}$$

Single Shooting

Single shooting discretises the dynamic constraints (2.34b) using Equation (2.36), i.e. it enforces continuity of the state trajectory by simulation. The optimisation variables are therefore x_0 for the initial state, the control parameter vector for each integration interval u_k and eventually the model parameters p.

This yields the following NLP

$$\underset{x_0, u_0, \ldots, u_{N-1}}{\text{minimise}} \quad L_N(x_0, U_{N-1}) + E(f_N(x_0, U_{N-1})) \tag{2.38a}$$

$$\text{subject to} \quad r(x_0, f_N(x_0, U_{N-1})) = 0, \tag{2.38b}$$

$$h(f_k(x_0, U_{k-1}), u_k) \geq 0, \qquad k = 0, \ldots, N-1. \tag{2.38c}$$

Multiple Shooting

As opposed to single shooting, Multiple shooting discretises the dynamic constraints (2.34b) using Equation (2.35), i.e. it does enforce continuity by simulation only in each time interval $[t_k, t_{k+1})$. Therefore, continuity at each grid point t_k must be enforced by additional constraints in the NLP. The optimisation

variables are therefore the states at each shooting node $x_0, x_1, \ldots, x_N$, the control parameter vector for each integration interval u_k.

This yields the following NLP

$$\underset{\substack{x_0,\ldots,x_N,\\ u_0,\ldots,u_{N-1}}}{\text{minimise}} \quad \sum_{k=0}^{N-1} \ell(x_k, u_k) + E(x_N) \tag{2.39a}$$

$$\text{subject to} \quad r(x_0, x_N) = 0, \tag{2.39b}$$

$$x_{k+1} - f(x_k, u_k) = 0, \tag{2.39c}$$

$$h(x_k, u_k) \geq 0, \qquad k = 0, \ldots, N-1. \tag{2.39d}$$

Collocation

Direct collocation methods for optimal control were introduced by [109], and successively extended by [10, 16, 104] among others. Collocation methods discretise the system dynamics by approximating the state trajectory using a piecewise polynomial function. While this approach can at first sight look completely different from single and multiple shooting, we will show in the next sections that there is a strong connection between the three approaches.

In collocation, each control interval $[t_k, t_{k+1}]$ is subdivided into a time grid $t_{k,0} < t_{k,1} < \ldots < t_{k,n_c}$ with $n_c + 1$ time points. For simplicity, we assume an equidistant time grid, i.e. $t_{k,j+1} - t_{k,j} = h_j = h$. The state trajectory is then approximated by

$$x(t) = \sum_{i=0}^{d} P_i\left(\frac{t - t_{k,j}}{h}\right) x_{k,j,i}, \qquad t \in [t_{k,j}, t_{k,j+1}], \tag{2.40}$$

where $x_{k,j,i}$ are the polynomial coefficients which multiply $P_i(\tau)$, defined as a Lagrangian polynomial basis of order d

$$P_i(\tau) = \prod_{r=0, r\neq i}^{d} \frac{\tau - \tau_r}{\tau_i - \tau_r}, \qquad \tau \in [0, 1]. \tag{2.41}$$

Note that $t_{k,j,i} = t_{k,j} + h\tau_i$ and $x(t_{k,j,i}) = x_{k,j,i}$. The so-called *collocation points* τ_r, with $\tau_0 = 0$, can be chosen in different ways. Two popular choices are *Legendre* schemes, which have a guaranteed accuracy order equal to $2d$, and *Radau* schemes, which have a a higher numerical stability, but a lower guaranteed accuracy order, i.e. $2d - 1$. For more details on direct collocation we refer to [18].

The state derivative is then obtained as

$$\dot{x}(t) = \frac{1}{h}\sum_{i=0}^{d}\frac{\partial P_i(\tau)}{\partial \tau}x_{k,j,i} = \frac{1}{h}\sum_{i=0}^{d}P_i'(\tau)x_{k,j,i}, \tag{2.42}$$

and the dynamic constraints (2.34b) are then enforced at all collocation points $t_{k,j,i}$. For an ODE this reads as

$$f^{\mathrm{c}}(x_{k,j,i}, u_k) - \frac{1}{h}\sum_{i=0}^{d}P_i'(\tau_i)x_{k,j,i} = 0, \qquad \begin{array}{l} k = 0, \ldots, N, \\ j = 0, \ldots, n_j, \\ i = 1, \ldots, d. \end{array} \tag{2.43}$$

For the generic case of an implicit DAE, the collocation equations read

$$f^{\mathrm{c}}\left(t_{k,j,i}, \frac{1}{h}\sum_{i=0}^{d}P_i'(\tau_i)x_{k,j,i}, x_{k,j,i}, u_k, p\right) = 0, \qquad \begin{array}{l} k = 0, \ldots, N, \\ j = 0, \ldots, n_j, \\ i = 1, \ldots, d. \end{array} \tag{2.44}$$

The state $x(t_{k,j+1})$ at the end of each time interval $[t_{k,j}, t_{k,j+1}]$ is given by $x(t_{k,j+1}) = \sum_{i=0}^{d} P_i(1)x_{k,j,i}$, so that the continuity equations read

$$\sum_{i=0}^{d}P_i(1)x_{k,j,i} - x_{k,j+1,0} = 0, \qquad \begin{array}{l} k = 0, \ldots, N, \\ j = 0, \ldots, n_j - 1. \end{array} \tag{2.45}$$

The integral term in the cost can be efficiently evaluated using quadrature. We define $w(X_{k,j}, u_k)$ with $X_{k,j} = (x_{k,j,0}, \ldots, x_{k,j,d})$ the approximation of the integral obtained by using an appropriate quadrature formula. For more details on the topic and on efficient implementations we refer to [18, 5, 58].

The collocation discretisation then yields the following NLP

$$\underset{\substack{x_{0,0,0},\ldots,x_{N-1,n_j,d},x_N,\\ u_0,\ldots,u_{N-1},p}}{\text{minimise}} \quad \sum_{k=0}^{N-1}\sum_{j=0}^{n_j} w(X_{k,j},u_k) + E(x_N) \tag{2.46a}$$

$$\text{subject to} \quad r(x_{0,0,0},x_N) = 0 \tag{2.46b}$$

$$\sum_{i=0}^{d} P_i(1)x_{k,j,i} - x_{k,j+1,0} = 0, \qquad \begin{matrix} k=0,\ldots,N,\\ j=0,\ldots,n_j-1, \end{matrix} \tag{2.46c}$$

$$f^{\mathrm{c}}\left(\frac{1}{h}\sum_{i=0}^{d} P_i'(\tau_i)x_{k,j,i}, x_{k,j,i}, u_k\right) = 0, \quad \begin{matrix} k=0,\ldots,N,\\ j=0,\ldots,n_j,\\ i=1,\ldots,d, \end{matrix} \tag{2.46d}$$

$$h(x_{k,0,0},u_k) \geq 0, \qquad k=0,\ldots,N-1. \tag{2.46e}$$

Comparison of the Three Approaches

The collocation constraints (2.46c) enforce the same continuity condition as the multiple shooting constraints (2.39c). The main difference is that, while in multiple shooting continuity needs to be enforced only at time instants t_k, in collocation, one also needs to enforce continuity at all intermediate time instants $t_{k,j}$. In multiple shooting, continuity at nodes j is enforced by simulation of the system dynamics. In single shooting no continuity constraint is needed, as continuity at both nodes j and k is enforced by simulation.

The so-called *collocation equations* are enforced through Constraints (2.46d) and determine the values of the polynomial coefficients $x_{k,j,i}$. As these coefficients are then used in (2.46c) to determine the states $x(t_k)$, Constraints (2.46d) enforce a way of computing the state transition map $\phi(x_k,u_k)$. One could eliminate all variables $x_{k,j,i}$ with $j,i>0$ by computing analytically the collocation state transition map $\phi(x_{k,0,0},u_k)$ and use this map in a multiple shooting or single shooting framework. Indeed, it can be shown that every collocation scheme is an implicit RK scheme [18]. Therefore, the similarity between shooting methods and collocation becomes evident.

For simplicity, in Problem (2.46) we only enforced the path Constraints (2.46e) at time points t_k. However, one can easily impose the path constraints also at intermediate time instants $t_{k,j,i}$. Note that in general, in all three approaches,

one can impose the path constraints also at a finite number of points which do not necessarily coincide with the time points chosen for discretising the control and state trajectory.

Even though collocation and multiple shooting lead to NLPs with a higher number of decision variables, this is not a disadvantage, on the contrary: while the NLPs have a larger size, they have a sparse structure, which can be exploited by the solvers.

The main difficulties related to single shooting regard the treatment of unstable systems. If the time horizon is long enough, small perturbations in the initial value cause the system to diverge to extremely large values. This has multiple negative effects: (a) the values can get large enough to cause numerical problems, (b) at each NLP iterate, the system is linearised far from the solution and generally a high number of iterates is needed to reach convergence.

Further advantages of multiple shooting and collocation over single shooting consist in the ability to provide an initial guess also for the state trajectory. Multiple shooting is easily parallelisable, as the system dynamics can be simulated independently on every shooting interval.

Both multiple shooting and collocation can be seen as a *lifted* version of single shooting. The *lifted Newton method*, introduced by [2], is a Newton method in which long nonlinear expressions are split up into simpler expressions by introducing auxiliary variables, e.g. define $f_1(f_2(w_1))$ as $f_1(w_2),\ w_2 = f_2(w_1)$. This approach results in a high number of optimisation variables and constraints, but, under mild assumptions, it is known to have faster convergence rate and a computational complexity comparable to a standard Newton approach. For more details on the lifted Newton method we refer to [2].

2.3.3 Distributed Discretisation of Direct Optimal Control Problems

In this section, we present Distributed (direct) Multiple Shooting (DMS), a method for distributed systems that has been proposed in [103, 102] and subsequently extended for real-time NMPC implementations of fast dynamic systems [95]. Though the method has been developed as an extension of multiple shooting, it readily applies also to the framework of collocation. A distributed implementation of single shooting, though possible, is arguably less interesting.

It is important to stress that in this Section we only consider the discretisation part of the optimal control algorithm. Clearly, DMS can be used in combination

with a distributed optimisation algorithm and the development of a fully distributed algorithm for optimal control is the object of ongoing research.

Let us consider a process which can be divided into M coupled subsystems. For the sake of simplicity let us consider an explicit ODE model and drop the explicit dependence on time t and parameters p. Note, however, that this framework directly applies to the more general case presented before. The OCP (2.34) can in this case be formulated as

$$\underset{\substack{x(\cdot),u(\cdot),\\ v(\cdot),y(\cdot)}}{\text{minimise}} \quad \sum_{m=1}^{M} \left(\int_0^T L_m(x_m(t), u_m(t)) \, \mathrm{d}t + E_m(x_m(T)) \right) \tag{2.47a}$$

$$\text{subject to} \quad 0 = f_m^{\mathrm{c}}(\dot{x}_m(t), x_m(t), u_m(t), v_m(t)), \quad t \in [\,0,\, T\,], \quad m = 1, \ldots, M, \tag{2.47b}$$

$$y_m(t) = g_m(\dot{x}_m(t), x_m(t), u_m(t), v_m(t)), \quad t \in [\,0,\, T\,], \quad m = 1, \ldots, M, \tag{2.47c}$$

$$v_m(t) = \sum_{l=1}^{M} K_{ml}\, y_m(t), \quad t \in [\,0,\, T\,], \quad m = 1, \ldots, M, \tag{2.47d}$$

$$0 \leq h_m(x_m(t), u_m(t)), \quad m = 1, \ldots, M, \tag{2.47e}$$

$$0 = r_m(x_m(0), x_m(T)), \quad m = 1, \ldots, M, \tag{2.47f}$$

where $y_m(t)$ and $v_m(t)$ denote the coupling outputs and inputs of subsystem m respectively. For the sake of simplicity, we restrict ourselves to the case where the objective and constraint functions are readily decoupled but in general they can also be made dependent on coupling variables. The linear relation with the coupling matrices K_{ml} in Equation (2.47d) defines the specific interaction among the subsystems.

The main idea of DMS consists in exploiting the distributed structure of the problem by simulating the dynamics of each subsystem j separately. A discontinuity of the system dynamics is therefore not only introduced at a finite number of time instants, but also in the state space. In order to recover continuity of the system dynamics, Constraints (2.47c) and (2.47d) are enforced. In order to solve OCP (2.47), the coupling inputs and outputs of each subsystem must be discretised. A linear combination of basis functions is used to represent the coupling inputs and outputs accurately while keeping the number of variables exchanged between the subsystems sufficiently low. In [102], it is proposed to use Legendre polynomials, while in [95] it is proposed to use Lagrange polynomials, similarly to what is used in collocation schemes.

Once variables $y(t)$ and $z(t)$ have been discretised, the state discretisation can be done in the same way as in centralised methods. Both multiple shooting and collocation discretisations can then easily be derived. Note however, that this distributed method can be of particular interest especially for multiple shooting, as each subsystem can be simulated using a different integration scheme. For more details on DMS, we refer to [103, 102, 95].

2.3.4 Free Final Time OCPs

For some applications the duration of the optimal control problem is not established a priori and must also be chosen optimally. In order to tackle such problems, one can use the following reformulation. We define a fictitious time $\tau \in [0, 1]$ and redefine time as $t = T\tau$, where T is the final time of the OCP, which has to be optimised. Then, by using $\frac{\mathrm{d}t}{\mathrm{d}\tau} = T$, we write

$$x'(t) = \frac{\partial x(t)}{\partial \tau} = \dot{x}(t)\frac{\mathrm{d}t}{\mathrm{d}\tau} = \dot{x}(t)T. \tag{2.48}$$

Then we can include the final time T in the parameter vector p and write OCP (2.34) as

$$\underset{x(\cdot),z(\cdot),u(\cdot),p}{\text{minimise}} \quad T\int_0^1 L(x(\tau), z(\tau), u(\tau), p)\ \mathrm{d}\tau + E(x(1), z(1), p) \tag{2.49a}$$

$$\text{subject to} \quad f^{\mathrm{c}}\left(t, \frac{x'(\tau)}{T}, x(\tau), z(\tau), u(\tau), p\right) = 0, \qquad \tau \in [0, 1] \tag{2.49b}$$

$$h(x(\tau), z(\tau), u(\tau), p) \geq 0, \qquad \tau \in [0, 1] \tag{2.49c}$$

$$r(x(0), x(1)) = 0. \tag{2.49d}$$

2.4 Real-Time Optimal Control and Estimation

Model Predictive Control (MPC) is an advanced control technique for nonlinear constrained systems based on optimal control. It relies on a mathematical model of the system to predict its future evolution and minimise a given cost function over a prescribed prediction horizon. Only the first control move is applied to the system and, at the next sampling time, the new control is computed based on the most current available state estimate. Given its nature of digital controller, MPC is typically analysed in the discrete time framework, even

though it is often applied to continuous time systems. Clearly, a continuous time system can be discretised to obtain a discrete time counterpart. In this context, the connection to direct optimal control becomes evident, even though other techniques can also be used to solve the MPC OCP.

As MPC relies on the most current state estimate for updating the control policy, an observer is needed to provide this estimate. Many techniques have been developed for this purpose. Among them, Moving Horizon Estimation (MHE) shares some similarities to MPC, as it repeatedly solves a dynamic optimisation problem on a fixed horizon in the past, in order to fit the model to the past measurements and thus estimate the current state. A thorough discussion on observers is out of the scope of this thesis, and we will only briefly introduce MHE and show how it can be used in combination with MPC for real world applications.

In the following, we will introduce MPC and MHE in discrete time. Clearly, the same problems can also be formulated in continuous time. However, as these problems are in practice most often solved using direct methods, the continuous time formulation is discretised in time.

2.4.1 Model Predictive Control

Given a sampling time T_s, at every time instant $t_n = n\,T_\mathrm{s}$, MPC computes the feedback control u^MPC by solving the following OCP

$$\begin{aligned}
\underset{x,u}{\text{minimise}} \quad & \sum_{k=0}^{N-1} \ell(x_k, u_k) + V_\mathrm{f}(x_N) && && \text{(2.50a)}\\
\text{subject to} \quad & x_0 = \hat{x}_n, && && \text{(2.50b)}\\
& x_{k+1} = f(x_k, u_k), && k = 0, \dots, N-1, && \text{(2.50c)}\\
& h(x_k, u_k) \geq 0, && k = 0, \dots, N-1, && \text{(2.50d)}\\
& x_N \in \mathbb{X}_\mathrm{f}. && && \text{(2.50e)}
\end{aligned}$$

Function ℓ is called *stage cost*, function V_f is called *terminal cost*, set $\mathbb{X}_\mathrm{f}$ defines the terminal constraint and we define $x = (x_0, \dots, x_N)$ and $(u_0, \dots, u_{N-1})$. Defining the optimal solution as (x^*, u^*), the MPC feedback control is given by $u^\mathrm{MPC} = u_0^*$.

Often, the stage and terminal cost ℓ and V_f are chosen as quadratic functions penalising the deviation from a given reference and the terminal set $\mathbb{X}_\mathrm{f}$ is

often chosen as $\mathbb{R}^{n_x}$. More details on the choice of the cost function, terminal constraint, prediction horizon N and their importance in guaranteeing stability will be given in Chapter 4.

The OCP is subject to the following constraints:

- the *initial value constraint* (2.50b), enforcing the current state as initial state of the predicted trajectory,
- the *dynamic constraints* (2.50c), enforcing that the predicted trajectory satisfies the system dynamics,
- the *path constraints* (2.50d), which define the constraint set $\mathbb{Z} \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$, and constrain the predicted trajectory to lie within a prescribed region of the state-control space,
- the *terminal constraint* (2.50e), the role and importance of which will be detailed in Chapter 4.

Because of unmodelled dynamics, imperfect knowledge of the model parameters and the presence of external (known and unknown, measured and unmeasured) disturbances, closed loop control is necessary. MPC closes the loop by iteratively solving OCP (2.50) at every time instant t_n using the state estimate $\hat{x}_n$.

2.4.2 Moving Horizon Estimation

When formulating the MPC control problem, it is assumed that the current state and model parameters are known. This is in general not the case, as a full state measurement is often impractical, expensive or even impossible. An observer is therefore needed in order to estimate the current state given a set of measurements. Moreover, in many practical applications, some of the model parameters cannot be identified offline and need to be estimated online.

The problem of estimating the state and the unknown parameters online can be formulated as a dynamic optimisation problem. In Moving Horizon Estimation (MHE), a time window of fixed length in the past is considered, on which the deviation of the measurements predicted by the measurement function $y(x_k, u_k)$ from the actual measurements $\tilde{y}_k$ is minimised. While MPC can be seen as an extension of the Linear Quadratic Regulator (LQR), MHE can be seen as the extension of the Kalman filter for constrained nonlinear systems. The MHE

problem can be formalised by the following OCP

$$\underset{x,u}{\text{minimise}} \quad a(x_{-E}) + \sum_{k=-E}^{0} \|y(x_k, u_k) - \tilde{y}_k\|^2_{\Sigma^{-1}} \tag{2.51a}$$

$$\text{subject to} \quad x_{k+1} = f(x_k, u_k), \qquad k = -E, \dots, -1, \tag{2.51b}$$

$$h(x_k, u_k) \leq 0, \qquad k = -E, \dots, -1, \tag{2.51c}$$

$$g(x_0) = 0. \tag{2.51d}$$

We call E the *estimation horizon* and we define $x = (x_{-E}, \dots, x_0)$, $(u_{-E}, \dots, u_{-1})$ and the optimal solution as (x^*, u^*). The MHE estimate is then given by $\hat{x}_n = x_0^*$. Note that, while we formulated the stage cost (2.51a) using the 2 norm of the measurement residual, different choices are possible and sometimes preferable, e.g. using a Huber penalty.

The introduction of path constraints (2.51c) should be done very carefully for several reasons. First, even if one imposes path constraints in MPC, the system can still violate them due to perturbations and, based on the measurements, MHE must be able to also estimate this constraint violation. Secondly, even if small violations will be detected e.g. due to sensor noise, the addition of a path constraints modifies the probability distributions in an artificial way which is undesirable. Path constraints should therefore only be introduced if they eliminate the possibility for unphysical behaviours of the model, e.g. negative concentrations of solutes in a solution. For more details on the topic we refer to [97, 98].

The weighting matrix Σ^{-1} can be used to scale the measurements based on the sensor accuracy. Function $a(\cdot)$ is called *arrival cost* and can be used to summarise all information prior to time $-E$. In case of a linear unconstrained system and Gaussian noise, the equivalence with the Kalman Filter can be obtained by choosing $E = 1$, Σ^{-1} as the inverse of the noise covariance matrix and the arrival cost $a(\cdot)$ as the Kalman estimate covariance update.

Equality constraints (2.51d) are rather non-standard and are added to the formulation when e.g. a model in non-minimal coordinates is used, such that the system dynamics only make physical sense if some invariants are enforced. Provided that the value of the invariants is specified at one time instant, the system dynamics preserve its value throughout the simulation. It is therefore important to constrain the invariants to take a physically meaningful value at one time instant. More details on models in non-minimal coordinates and invariants will be given in Chapter 6.

While a thorough discussion on observers and MHE is out of the scope of this thesis, it is interesting to note that the OCP in MHE solved at each time step has strong similarities to that solved by MPC. Therefore, the same algorithms can be used for both MHE and MPC. For more details on MHE, we refer to [97, 98].

2.4.3 Algorithms for Real-Time Dynamic Optimisation

Due to the tight timing requirements for online applications of fast dynamic systems, generic OCP algorithms may fail to provide a solution within the available time. Therefore, tailored algorithms have been developed specifically for real-time optimal control and estimation. Both MPC and MHE require one to repeatedly solve a sequence of similar problems, for which a very good initial guess is therefore available.

In the context of real-time dynamic optimisation, the time needed to solve the NLP is often not negligible compared to the sampling time. If the computation of the NLP solution lasts longer than the sampling time, the information on the current initial state $\hat{x}_n$ or set of measurements $\tilde{y}$ becomes outdated. Clearly, this problem can be partially attenuated through a translation of the state estimate into the future by an open-loop simulation of the system dynamics. However, due to perturbations and model inaccuracy, open-loop simulations over long time intervals become inaccurate.

A very successful algorithm for fast MPC and MHE is the *Real Time Iteration* (RTI) scheme. Before presenting it, let us first introduce the so-called *real-time dilemma* [31].

Dilemma 2.24 (Real-time dilemma)**.** *Is it better to solve the MPC/MHE NLP problem to convergence using outdated information or to rather compute approximate solutions using the most up-to-date information?*

The RTI scheme is based on the second option, i.e. the computation of approximate solutions using the most up-to-date information. It relies on the fast contraction properties of Newton-type methods in order to track the exact NLP solution. In this way, convergence is obtained while the system evolves in time.

The Real-Time Iteration Scheme

In this section, we will present the RTI scheme for MPC. Note, however, that the adaptation to handle the MHE case is straightforward. In order to allow

for faster feedback, the RTI scheme consists of a modified SQP method which (a) computes only one full SQP step per sampling time and (b) keeps the initial state in MPC as a constraint in the NLP formulation. Feature (a) follows from two observations:

1. SQP methods have a fast convergence rate if initialised close to the solution;
2. subsequent MPC and MHE problems are very similar, and a very good initial guess for the subsequent problem can easily be obtained from the solution to the previous one.

Feature (b), known as *initial value embedding* is important for two reasons:

1. it ensures that the first SQP iterate is a tangential predictor for the new initial state value;
2. it allows one to split computations into a *preparation phase* which is performed before the state estimate becomes available and a *feedback phase*, which is performed after the state estimate becomes available.

For each SQP iterate, the QP subproblem (2.17) must be solved: the main computations consist in (i) computing the cost and constraint sensitivities H, $\nabla_w f$, $\nabla_w g$, $\nabla_w h$ which are needed in order to formulate the QP and (ii) solving the QP. It is easy to note that, by using the initial value embedding, one can perform all computations in (i) *before* the current state estimate is known, i.e. in the preparation phase. Depending on the QP solver used, even some computations related to (ii) can be performed in the preparation phase, e.g. condensing. Once the current state estimate becomes available, the feedback phase consists in solving the QP and the feedback control action can be immediately sent to the plant. It is therefore clear that the latency between the instant at which the state estimate becomes available and the instant at which the feedback control is applied to the plant is reduced to an extremely short time.

Efficient Sensitivity Computation For the preparation phase, embedded integrators with efficient sensitivity propagation are crucial for guaranteeing short computational times. Tailored explicit and implicit structure exploiting Runge Kutta methods have been developed specifically for real-time optimal control in [93, 89, 88, 94, 92, 91, 90].

Solving the QP: Condensing vs Sparse Solution A large variety of tailored QP solvers has been developed for MPC applications, including active set, interior point and first order methods [36, 39, 35, 40, 83]. MPC problems have a specific structure which yields a sparse QP with a block-diagonal structured Hessian and equality constraints which couple "neighbouring" variables. In this context, it can be advantageous to eliminate some variables (often the state variables) and solve a dense QP in a much smaller space. The state elimination procedure is called *condensing* and it has been shown to be beneficial especially for active-set methods, e.g. qpOASES [36]. Other approaches, e.g. interior-point methods like FORCES [35] and HPMPC [40] or active-set methods like qpDUNES [39], exploit the sparsity of the uncondensed problem to efficiently compute the solution.

Real-Time Distributed Multiple Shooting In [95], the DMS framework has been used conjointly with the efficient integrators proposed for the centralised case. Similarly to the condensing strategy, a *compression* technique was used to eliminate the coupling inputs and outputs z and y. In particular, it was shown that the distributed simulation of the system dynamics can be very beneficial also in the centralised setting and a speedup of factor 11 was observed on a nontrivial example. We remark that DMS can be parallelised even more than centralised multiple shooting; moreover, compression can be done in parallel for each shooting interval.

Towards Distributed NMPC In recent years, research interest in distributed systems has increased. In order to deploy a distributed RTI scheme, both the preparation and the feedback phase need to be distributed. DMS provides the framework for distributing the simulation and sensitivity generation (i.e. the preparation phase). The feedback phase, instead, requires tailored distributed QP solvers like those proposed in e.g. [71, 70].

2.5 Conclusions

In this chapter, we have presented a literature review on optimisation, dynamic systems, optimal control, model predictive control and moving horizon estimation. We have introduced algorithms for optimisation and simulation of dynamic systems. For optimal control we have mentioned the main classes of numerical methods and we have described more in detail direct methods also known as "first discretise then optimise". We have then detailed how such algorithms can be adapted for MPC and MHE by exploiting the intrinsic nature

of these problems in order to reduce the computational times and make the implementation of MPC and MHE real-time feasible for a broader class of systems.

This chapter can be considered as a starting point for the rest of the thesis, as its content will be relevant for all chapters but Chapter 4.

Chapter 3

Autonomous Driving

In this chapter we consider the problem of controlling autonomous ground vehicles. First we propose a high-fidelity vehicle model which includes an accurate tire force model and a suspension model. We reparametrise the model from the time domain to a curvilinear coordinate which describes the progress with respect to the centreline of the road. We then use this model to formulate an MHE and MPC scheme to control the vehicle in an obstacle avoidance scenario on a low friction road.

In the second part of the chapter, we propose a first step towards time optimal control of small-scale race cars on a track. We formulate an MPC controller which can make use of the efficient numerical schemes developed for least-squares objectives and test the controller on a real setup.

The chapter is organised as follows. In Section 3.1 an accurate vehicle model is derived. Section 3.2 presents the use of MPC and MHE to control the vehicle. Finally, Section 3.3 presents the first step towards time-optimal MPC for a simplified vehicle model, including experimental results.

Acknowledgement This chapter is based mainly on [38, 117, 118, 111, 112]. Janick Frasch, Andrew Gray and Mario Zanon are the main authors of [38]; Mario Zanon and Janick Frasch are the main authors of [117, 118]. Mario Zanon and Janick Frasch developed the accurate vehicle model. The spatial transformation was developed by Yiqi Gao, Andrew Gray, and Janick Frasch. Mario Zanon contributed most of the numerical simulations presented in this chapter with a substantial help from Janick Frasch. Joachim Ferreau and Milan Vukov provided valuable support with the algorithms for real-time optimisation.

The work of [38, 117, 118] was done under the co-supervision of Sebastian Sager and Moritz Diehl. Francesco Borrelli co-supervised the work presented in [38].

The work on time-optimal MPC is the outcome of Robin Verschueren's Master Thesis [111], which was further extended to be published in [112]. Robin Verschueren is the main author and contributed most of the work. Mario Zanon and Janick Frasch provided supervision on the algorithms and problem formulations. Stijn de Bruyne provided valuable supervision regarding the implementation on the hardware setup at Siemens LMS. Moritz Diehl is the main supervisor and served as valuable discussion partner.

3.1 Vehicle Model

In the following, we present a vehicle model to be used in the formulation of MPC and MHE schemes. In contrast to some of the previous literature on MPC and optimal control [65, 66, 69], which considers a simplified bicycle model, we model each of the four wheels separately. This allows us to introduce load transfer and eventually the suspension in our model, therefore accounting for dynamics which can become important when performing extreme manoeuvres.

3.1.1 System Dynamics in the Time Domain

The chassis position and orientation are defined in the X-Y plane of an absolute reference frame E, while the velocities are given in the local $x-y-z$ frame e. The heave motion of the chassis is neglected and the four wheels are modelled as independent bodies with only spinning inertia. Throughout the chapter, when referring to quantities related to the wheels, subscripts fl, fr, rl, rr denote quantities corresponding respectively to the front left, front right, rear left and rear right wheel. For ease of notation, let $\mathcal{F} := \{\mathrm{f}, \mathrm{r}\}$, $\mathcal{S} := \{\mathrm{l}, \mathrm{r}\}$ and $\mathcal{W} := \mathcal{F} \times \mathcal{S} = \{\mathrm{fl}, \mathrm{fr}, \mathrm{rl}, \mathrm{rr}\}$.

The control inputs are the steering rate $\dot{\delta}$, the accelerating torque T^{a} and the four braking torques of each wheel $T^{\mathrm{b}}_{\diamond\star},\ \forall \diamond \star \in \mathcal{W}$.

Chassis Dynamics

The equations of motion are written with respect to the vehicle's centre of gravity (CoG). Reference frames E and e are chosen orthonormal, right-handed with the z-axis pointing up, and the y-axis pointing left. The chassis equations

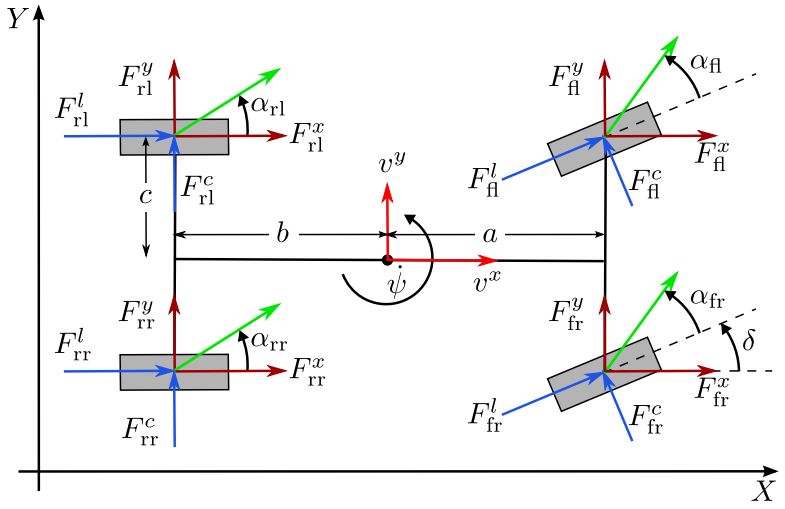


Figure 3.1: Tire forces and slip angles of the 4-wheel vehicle model in inertial coordinates. The tires' directions of movement are indicated by green vectors. Source [38, 117].

of motion thus are

$$m\dot{v}^x = mv^y\dot{\psi} + F^x_{\mathrm{fr}} + F^x_{\mathrm{fl}} + F^x_{\mathrm{rr}} + F^x_{\mathrm{rl}} + F_D \ , \tag{3.1a}$$

$$m\dot{v}^y = -mv^x\dot{\psi} + F^y_{\mathrm{fr}} + F^y_{\mathrm{fl}} + F^y_{\mathrm{rr}} + F^y_{\mathrm{rl}} \ , \tag{3.1b}$$

$$I^z\ddot{\psi} = a(F^y_{\mathrm{fl}} + F^y_{\mathrm{fr}}) - b(F^y_{\mathrm{rl}} + F^y_{\mathrm{rr}}) + c(F^x_{\mathrm{fr}} - F^x_{\mathrm{fl}} + F^x_{\mathrm{rr}} - F^x_{\mathrm{rl}}) \ , \tag{3.1c}$$

$$\dot{X} = v^x\cos\psi - v^y\sin\psi \ , \tag{3.1d}$$

$$\dot{Y} = v^x\sin\psi + v^y\cos\psi \ , \tag{3.1e}$$

where m denotes the mass and I^x, I^y, I^z the moments of inertia of the chassis. The distances of the tires from the vehicle's CoG are characterised by a, b and c, cf. Figure 3.1. The CoG is assumed to be located halfway between the left and right side of the car. The vehicle's yaw angle ψ is obtained by direct integration of $\dot{\psi}$ as is the steering angle δ from input $\dot{\delta}$. The drag force due to air resistance is denoted by F_D, while $F^x_{\cdot\cdot}$, $F^y_{\cdot\cdot}$ denote the components of the tire contact forces along the vehicle's local x and y axis.

The considered vehicle has front steering, thus

$$F^x_{\mathrm{f}\star} = F^{\mathrm{l}}_{\mathrm{f}\star}\cos\delta - F^{\mathrm{c}}_{\mathrm{f}\star}\sin\delta\ , \qquad F^y_{\mathrm{f}\star} = F^{\mathrm{l}}_{\mathrm{f}\star}\sin\delta - F^{\mathrm{c}}_{\mathrm{f}\star}\cos\delta\ , \qquad \forall\ \star \in \mathcal{S}\ ,$$

$$F^x_{\mathrm{r}\star} = F^{\mathrm{l}}_{\mathrm{r}\star}\ , \qquad F^y_{\mathrm{r}\star} = F^{\mathrm{c}}_{\mathrm{r}\star}\ , \qquad \forall\ \star \in \mathcal{S}\ ,$$

where $F^{\mathrm{l}}_{\cdot\cdot}$, $F^{\mathrm{c}}_{\cdot\cdot}$ denote the longitudinal and cornering tire forces respectively.

Tire Contact Forces: Pacejka's Magic Formula

We compute the tire forces using Pacejka's Magic Formula: a very accurate semi-empirical nonlinear tire model that is commonly used for automotive applications. The Magic Formula allows one to compute the longitudinal and cornering forces as a function of the longitudinal slip and slip angle, while taking into account the effect of combined slip. The self-aligning torque $M^z_{\cdot\cdot}$ has a significant contribution only at low speeds [81] and is assumed to be negligible. Longitudinal and cornering forces are thus computed as

$$\left[\ F^{\mathrm{l}}_{\diamond\star},\quad F^{\mathrm{c}}_{\diamond\star}\ \right]^\top = f_{\mathrm{P}}(\alpha_{\diamond\star}, \kappa_{\diamond\star}, \mu, F^z_{\diamond\star})\ , \qquad \forall\ \diamond\star \in \mathcal{W}\ , \tag{3.2}$$

where $f_{\mathrm{P}}(\cdot)$ denotes Pacejka's tire model. The inputs to the Magic formula are: (a) the side slip angle $\alpha_{\cdot\cdot}$, defined as the angle between the wheel's orientation and its velocity (cf. Figure 3.1 and Equation (3.3)), (b) the longitudinal slip $\kappa_{\cdot\cdot}$, (c) the tire-road friction coefficient μ and (d) the vertical load on the wheel $F^z_{\cdot\cdot}$. The longitudinal slip κ and the slip angle α are defined as

$$\kappa_{\diamond\star} = \frac{\omega_{\diamond\star}R_{\mathrm{e}} - v^{\mathrm{l}}_{\diamond\star}}{v^{\mathrm{l}}_{\diamond\star}}\ , \qquad \alpha_{\diamond\star} = \tan^{-1}\left(\frac{v^{\mathrm{c}}_{\diamond\star}}{v^{\mathrm{l}}_{\diamond\star}}\right)\ , \tag{3.3}$$

where $v^{\mathrm{l}}_{\diamond\star}$ and $v^{\mathrm{c}}_{\diamond\star}$ are the wheel longitudinal and cornering (lateral) velocities respectively, $\omega_{\diamond\star}$ is the wheel rotational speed and R_{e} is the effective tire radius.

Note that Equation (3.2) accounts for complex physical phenomena, including combined slip. More details on the formulas used can be found in [44]. We remark here that the force depends linearly on the vertical load $F^z_{\cdot\cdot}$ only for small loads, and saturates for high loads. This means that load transfer due to sharp turns or breaking manoeuvres reduces the overall contact forces that the four wheels can develop, therefore making it important to account for this effect when performing extreme manoeuvres. More details on the computation of slip angles and Pacejka forces can also be found in [68, 81, 66].

Wheel Dynamics

The wheels are modelled as separate bodies with only one rotational degree of freedom. Therefore the dynamic equations only depend on the accelerating and breaking torques $T^{\mathrm{a}}_{\cdot\cdot}$ and $T^{\mathrm{b}}_{\cdot\cdot}$ and on the longitudinal force $F^{\mathrm{l}}_{\cdot\cdot}$. The rotational accelerations are then given by

$$\dot{\omega}_{\diamond\star} = \frac{1}{I^{\mathrm{w}}}(T^{\mathrm{a}}_{\diamond\star} + T^{\mathrm{b}}_{\diamond\star} - R_{\mathrm{e}} F^{\mathrm{l}}_{\diamond\star}) \ , \qquad \forall \ \diamond\star \in \mathcal{W} \ ,$$

where individual wheel braking is considered. For the acceleration torque, a model of the differential is considered, which, for rear-wheel drive, yields

$$T^{\mathrm{a}}_{\mathrm{f}\star} = 0 \ , \qquad T^{\mathrm{a}}_{\mathrm{r}\star} = T^{\mathrm{a}} \left(1 - \frac{\omega_{\mathrm{r}\star}}{\omega_{\mathrm{rl}} + \omega_{\mathrm{rr}}}\right) \ , \qquad \forall \star \in \mathcal{S} \ .$$

Vertical Forces and Load Transfer Model

We obtain the vertical loads $F^{z}_{\cdot\cdot}$ acting on each wheel as the equilibrium of the main body w.r.t. the vertical forces, and roll and pitch torques. For a height h between the vehicle's CoG and its projection onto the wheelbase, the roll and pitch torques are given by

$$2\, T^{y}_{\mathrm{lt}} = (F^{x}_{\mathrm{fl}} + F^{x}_{\mathrm{fr}} + F^{x}_{\mathrm{rl}} + F^{x}_{\mathrm{rr}})\, h, \tag{3.4a}$$

$$2\, T^{x}_{\mathrm{lt}} = (F^{y}_{\mathrm{fl}} + F^{y}_{\mathrm{fr}} + F^{y}_{\mathrm{rl}} + F^{y}_{\mathrm{rr}})\, h. \tag{3.4b}$$

The steady state longitudinal and lateral load transfer are given by

$$\tilde{\Delta}^{z}_{\mathrm{lon}} = \frac{T^{y}_{\mathrm{lt}}}{a+b}, \qquad \tilde{\Delta}^{z}_{\mathrm{lat}} = \frac{T^{x}_{\mathrm{lt}}}{2\,c}. \tag{3.5}$$

By using Equation (3.5) directly in the computation of the vertical loads, the model would have an algebraic loop. While having a DAE model can be an appropriate choice, we relax the algebraic loop by introducing a first order model with time constant τ_{LT}, yielding

$$\dot{\Delta}^{z}_{\bullet} = \frac{1}{\tau_{\mathrm{LT}}}(\tilde{\Delta}^{z}_{\bullet} - \Delta^{z}_{\bullet}) \qquad \forall \bullet \in \{\mathrm{lon}, \mathrm{lat}\}. \tag{3.6}$$

This relaxation yields a more realistic model, which roughly approximates a suspension.

Denoting the rest normal loads by $\bar{F}^{z}_{\cdot\cdot}$, the vertical forces $F^{z}_{\cdot\cdot}$ are given by

$$F^{z}_{\mathrm{fl}} = \bar{F}^{z}_{\mathrm{fl}} - \Delta^{z}_{\mathrm{lon}} - \Delta^{z}_{\mathrm{lat}}, \qquad F^{z}_{\mathrm{fr}} = \bar{F}^{z}_{\mathrm{fr}} - \Delta^{z}_{\mathrm{lon}} + \Delta^{z}_{\mathrm{lat}},$$

$$F^{z}_{\mathrm{rl}} = \bar{F}^{z}_{\mathrm{rl}} + \Delta^{z}_{\mathrm{lon}} - \Delta^{z}_{\mathrm{lat}}, \qquad F^{z}_{\mathrm{rr}} = \bar{F}^{z}_{\mathrm{rr}} + \Delta^{z}_{\mathrm{lon}} + \Delta^{z}_{\mathrm{lat}}.$$

Vertical Forces and Suspension Model

The suspension of the vehicle is assumed to only act on the roll and pitch motions of the chassis. The rotation of the chassis is defined as

$$R = R_y(p)R_x(r) = \begin{bmatrix} \cos p & 0 & \sin p \\ 0 & 1 & 0 \\ -\sin p & 0 & \cos p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos r & \cos r \\ 0 & \sin r & -\sin r \end{bmatrix} ,$$

where r and p denote the roll and pitch angles respectively.

The forces of the suspension due to the spring elasticity and damping are defined respectively as

$$F^{\mathrm{el}}_{\diamond\star} = -k_\diamond \Delta_{\diamond\star} , \qquad F^{\mathrm{d}}_{\diamond\star} = -D_\diamond \dot{\Delta}_{\diamond\star} , \qquad \forall \diamond \star \in \mathcal{W} ,$$

where $k_\cdot$ and $D_\cdot$ denote the elastic and damping constants of the suspension spring. The suspension displacement is defined as $\Delta_{\diamond\star} = R\zeta_{\diamond\star} - \zeta_{\diamond\star}$, $\forall \diamond\star \in \mathcal{W}$, where $\zeta_{\cdot\cdot}$ denotes the wheel position in the body reference frame e.

Denoting the rest vertical forces by $\bar{F}^z_{\cdot\cdot}$, the vertical forces $F^z_{\cdot\cdot}$ are given by

$$F^z_{\diamond\star} = \bar{F}^z_{\diamond\star} + F^{\mathrm{el}}_{\diamond\star} + F^{\mathrm{d}}_{\diamond\star} , \qquad \forall \diamond \star \in \mathcal{W} .$$

The roll and pitch torques acting on the chassis are given by

$$T^x_{\mathrm{s}} = -2\left((k_{\mathrm{f}} + k_{\mathrm{r}})c^2 \sin r + c^2(D_{\mathrm{f}} + D_r)w_x \cos r\right) , \tag{3.7a}$$

$$T^y_{\mathrm{s}} = -2\Big((k_{\mathrm{f}}a^2 + k_{\mathrm{r}}b^2)\cos r \sin p + (D_{\mathrm{f}}a^2 + D_{\mathrm{r}}b^2)w_y \cos p - c^2 w^y (Df + Dr)\frac{\sin^2 r}{\cos r}\Big) . \tag{3.7b}$$

In a small angle approximation the suspension torques read

$$T^x_{\mathrm{sl}} = -2\left((k_{\mathrm{f}} + k_{\mathrm{r}})c^2 r + c^2(D_{\mathrm{f}} + D_r)w^x\right) , \tag{3.8a}$$

$$T^y_{\mathrm{sl}} = -2\left((k_{\mathrm{f}}a^2 + k_{\mathrm{r}}b^2)p + (D_{\mathrm{f}}a^2 + D_{\mathrm{r}}b^2)w^y\right) . \tag{3.8b}$$

The suspension dynamics are given by

$$J\dot{\omega} = T_{\mathrm{lt}} + T_{\mathrm{s}}, \qquad \dot{r} = \omega^x, \qquad \dot{p} = \frac{\omega^y}{\cos r}, \tag{3.9}$$

where the load transfer torques are given by (3.4).

3.1.2 Spatial Transformation

When controlling a vehicle on a road, it might be more advantageous to describe the system dynamics with respect to the position of the vehicle on the road rather than time. More precisely, the absolute coordinates X and Y are projected on a curve σ, which we choose as the centre line of the road and parametrise it by its arc length s. The ODE model is then reformulated as a function of s. The states X, Y and ψ are replaced by

$$e^y = \cos\psi^\sigma (Y - Y^\sigma) - \sin\psi^\sigma (X - X^\sigma), \text{ and}$$

$$e^\psi = \psi - \psi^\sigma,$$

where $[X^\sigma, Y^\sigma]^\top$ and ψ^σ are the position and orientation of the reference point on the path given by s, see Figure 3.2.

The spatial dynamics of the state vector ξ are given by

$$\xi' = \frac{\partial \xi}{\partial s} = \frac{\partial \xi}{\partial t}\frac{\partial t}{\partial s}. \tag{3.10}$$

If $\dot{s} \neq 0$ is assumed at any time, by the inverse function theorem we have $\frac{\partial t}{\partial s} = \frac{1}{\dot{s}}$. It therefore holds

$$\xi' = \frac{1}{\dot{s}}\,\dot{\xi},$$

where $\dot{\xi}$ is defined by the time-dependent system dynamics.

In order to compute the vehicle speed along the centre line, $\dot{s}$, note that from Figure 3.2 we have

$$(\rho^\sigma - e^y)\,\dot{\psi^\sigma} = v^\sigma = v^x\,\cos e^\psi - v^y\,\sin e^\psi,$$

where ρ^σ is the radius of local curvature of σ. The velocity along the path, $\dot{s}$, is then given by

$$\dot{s} = \rho^\sigma\,\dot{\psi}^s = \frac{1}{1 - \frac{e^y}{\rho^\sigma}}\,(v^x \cos(e^\psi) - v^y \sin(e^\psi)).$$

Note that $e^y < \rho^\sigma$ always needs to be fulfilled in order to guarantee uniqueness of the projection onto the centre line. This essentially means that the car needs to drive sufficiently close to the reference when the curvature is high. Also note that ρ^σ only depends on the parametrisation s through $\rho^\sigma(s) = \left(\frac{\partial^2}{\partial s^2}\sigma(s)\right)^{-1}$ but is independent of system inputs.

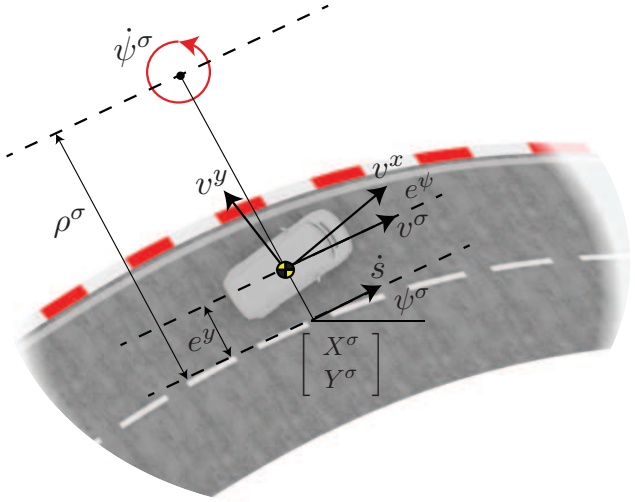


Figure 3.2: Definition of the coordinate system used in the spatial reformulation of the vehicle dynamics. The s coordinate denotes the arc length along the track. Source: [38].

If necessary, time information may be recovered by integrating $\frac{dt}{ds}$ along σ:

$$t(s) = \int_{s_0}^{s} \frac{1}{\dot{s}(\tau)} \, \mathrm{d}\tau \, .$$

Inertial coordinates may be recovered by a transformation from the spatial coordinates to the global coordinates:

$$X = X^\sigma - e^y \, \sin(\psi^\sigma)$$

$$Y = Y^\sigma + e^y \, \cos(\psi^\sigma)$$

$$\psi = \psi^\sigma + e^\psi \, .$$

3.2 MPC and MHE for Collision Avoidance

In this section, the MHE and NMPC schemes used for the simulations are presented. Both schemes are based on a piecewise constant control parametrisation and the system dynamics $f(x, u)$ are discretised in a direct multiple shooting framework using an implicit Runge-Kutta method of order 2 [93].

Sensor	Measurements	Standard deviation σ
IMU	Linear acceleration	10^{-2} m/s^2
IMU	Angular velocity	0.1 rad/s
GPS	Position	10^{-2} m
Encoder	Wheel rotational velocity	10^{-3} rad/s
Encoder	Steering angle	10^{-3} rad

Table 3.1: Available measurements

State or control	e_y, e_ψ	v_x, v_y, $\dot{\psi}$	r, p	ω_x, ω_y	$\omega_{..}$	T^{a}, $T^{\mathrm{b}}_{..}$	$\dot{\delta}$
Associated weight	1	10	1	1	1	10^{-4}	10^2

Table 3.2: Weights for NMPC

MHE Formulation

We formulate the MHE problem in the form (2.51), with a quadratic arrival cost $a(x_{-E}) = \|x_{-E} - \hat{x}_{-E}\|^2_{P_{\mathrm{E}}}$ and no equality constraint (2.51d). The arrival cost prior estimate $\hat{x}_{-E}$ and weighting matrix P_{E} are computed in an analogous way as in [72], where the Kalman update is computed in efficiently and it is ensured that the norm of the arrival cost weighting matrix P_{E} is bounded from above. The estimation horizon has been selected as $S_{\mathrm{E}} = 10$ m, divided into $E = 10$ control intervals of uniform duration $S_{\mathrm{c}} = S_{\mathrm{E}}/E$.

The available measurements come from an inertial measurement unit (IMU), a GPS, and encoders on the wheels and the steering wheel. They are summarised in Table 3.1, together with their standard deviation σ. The weighting matrix Σ^{-1} was chosen diagonal, with all diagonal elements matching the square of the inverse of the standard deviation σ_i they correspond to, i.e. $\Sigma_{i,i} = (\sigma_i)^2$.

In order to accurately estimate the friction coefficient μ, the model needs to account for sudden changes in the road friction. This can be achieved by making the friction coefficient time varying, using a first order model $\dot{\mu} = u_\mu$.

Penalisation of the variable u_μ is needed in order to ensure that the estimate of μ is not strongly affected by sensor noise. Penalising u_μ in a quadratic (L^2) norm has the effect of filtering out noise, but does not allow for fast detection of large jumps in the friction coefficient. Jumps are better detected when penalising the absolute value of u_μ (L^1 norm), as large changes are penalised less than with a L^2 penalty. In this case, though, small variations of μ are filtered out together

with the noise. The Huber penalty combines the benefits of the L^1 and the L^2 penalty. It is defined by

$$H(x) = \begin{cases} \frac{1}{2}x^2 & |x| \leq \rho \\ \rho(|x| - \frac{1}{2}\rho) & |x| \geq \rho \end{cases} . \tag{3.11}$$

We refer to [25] for an algorithmically differentiable implementation using slack variables.

MPC Formulation

We formulate the NMPC problem in the form (2.50). We define the reference trajectory at time k as x_k^{ref}, u_k^{ref} define the stage and terminal cost respectively as

$$\ell(x_k, u_k) = \|x_k - x_k^{\text{ref}}\|_Q^2 + \|u_k - u_k^{\text{ref}}\|_R^2, \qquad V_{\text{f}}(x_N) = \|x_N - x_N^{\text{ref}}\|_{P_N}^2,$$

and impose no terminal constraint. The control horizon for NMPC has been selected as $S_{\text{C}} = 20$ m, divided into $N = 20$ control intervals of uniform duration $S_{\text{c}} = S_{\text{C}}/N$. This longer horizon has been chosen, to guarantee that the obstacles are seen sufficiently in advance to allow for avoidance manoeuvres, including stopping the vehicle in extreme conditions.

The weights Q and R have been chosen as diagonal matrices, with each element selected in accordance with Table 3.2. The units of measure of the weights are selected so as to yield a dimensionless cost. The terminal cost matrix P_{C} has been taken as the solution to the discrete algebraic Riccati equation using the proposed weighting matrices Q and R.

For the inputs, the following constraints have been selected

$$0 \leq T^{\text{a}} \leq \overline{T}^{\text{a}} , \qquad \underline{T}^{\text{b}} \leq T_{\diamond\star}^{\text{b}} \leq 0 , \qquad \forall \diamond \star \in \mathcal{W} .$$

By using the spatial reformulation of the model, the obstacle avoidance constraints become simple bounds, defined as

$$\underline{e}_y \leq e_y \leq \overline{e}_y . \tag{3.12}$$

As this obstacle avoidance constraint drives the trajectory away from the reference, the NMPC scheme avoids the obstacle by steering the vehicle as close to it as possible and bounds (3.12) become active. For a real system, the state estimate will always be noisy due to measurement noise or model inaccuracy. Even the smallest violation of constraint (3.12) might yield an infeasible NLP and make the controller unreliable. Feasibility of the NLP can be guaranteed

by reformulating the obstacle avoidance constraints (3.12) using non-negative slack variables as

$$e_y \leq \overline{e}_y + u_{e_y}^U \ , \qquad u_{e_y}^U \geq 0 \ ,$$

$$e_y \geq \underline{e}_y - u_{e_y}^L \ , \qquad u_{e_y}^L \geq 0 \ .$$

In the proposed formulation, the slack variables $\vec{u}_{e_y} = [\ u_{e_y}^L \ u_{e_y}^U \ ]^T$ are introduced on each interval. They can be seen as a measure of the constraint violation corresponding to each interval. To penalise the constraint violation, the term

$$J_{\text{viol}} = \int_0^{S_c} \vec{w}_1^\top \vec{u}_{e_y}(t) + \|\vec{u}_{e_y}(t)\|_{W_2} \ \mathrm{d}s \ ,$$

is added to the cost. The proposed penalty on the slack variables implements the sum of an L^1 and an L^2 norm, using positive (definite) weights $\vec{w}_1$ and W_2, respectively. This choice allows to add a stronger penalty for large constraint violations (effect of the L^2 norm) while always having a nonzero gradient even when the constraints are not violated (effect of the L^1 norm).

Numerical Results

An obstacle avoidance simulation has been run to demonstrate the performance of the proposed control scheme. The NMPC and MHE schemes have been implemented using the code generation tool of ACADO [61].

As displayed in Figure 3.3, the vehicle is required to travel at a reference speed $v_x = 10$ m/s while avoiding two 6 meter long obstacles positioned at $s = 43$ m and $s = 123$ m on a 200 m long straight road. The first obstacle has to be avoided on the left and is 2 m wide, while the second one needs to be avoided on the right and is 0.8 m wide. The road surface has a very low friction coefficient $\mu = 0.3$, corresponding to a snow-covered or icy road. After 80 m, the friction coefficient increases to $\mu = 0.5$.

The trajectory obtained by applying the proposed control scheme to this scenario is displayed in Figure 3.3, top graph, where it can be seen that the controller is able to avoid the obstacles. In Figure 3.3, bottom graph, the estimated friction coefficient is displayed. It can be noted that the Huber penalty (3.11) successfully rejects the measurement noise, but still allows to detect the jump in the friction coefficient. The jump is detected in an approximate way immediately after it occurs. After this detection, MHE slowly corrects the inaccuracy. The estimation error yielded by MHE is displayed in Figure 3.7. It is interesting to note that, when the jump in the friction coefficient occurs, the quality of the

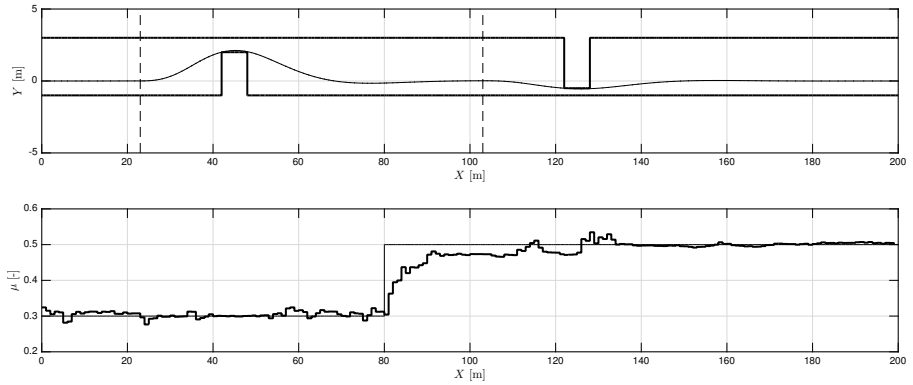


Figure 3.3: Vehicle trajectory for a straight reference (thin line) and obstacles (thick line). The MHE estimates are shown in circles. The friction coefficient is estimated using a Huber norm and is displayed in the bottom figure (thick line), together with the actual friction coefficient (thin line).

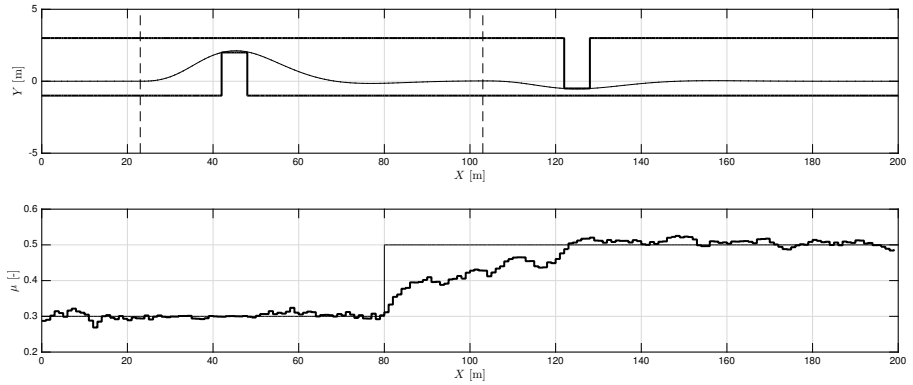


Figure 3.4: Vehicle trajectory for a straight reference (thin line) and obstacles (thick line). The MHE estimates are shown in circles. The friction coefficient is estimated using a L^2 norm and is displayed in the bottom figure (thick line), together with the actual friction coefficient (thin line).

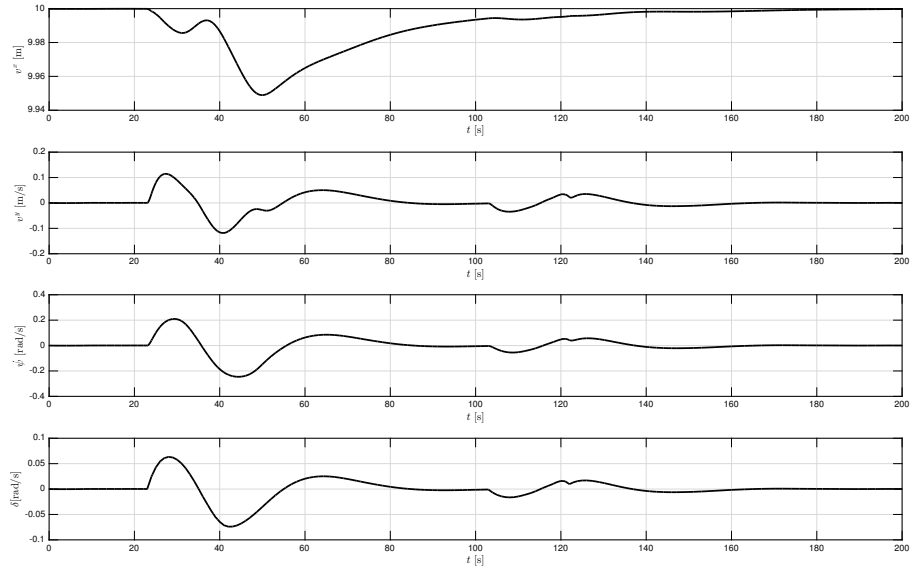


Figure 3.5: Vehicle longitudinal, lateral and yaw velocities and steering angle.

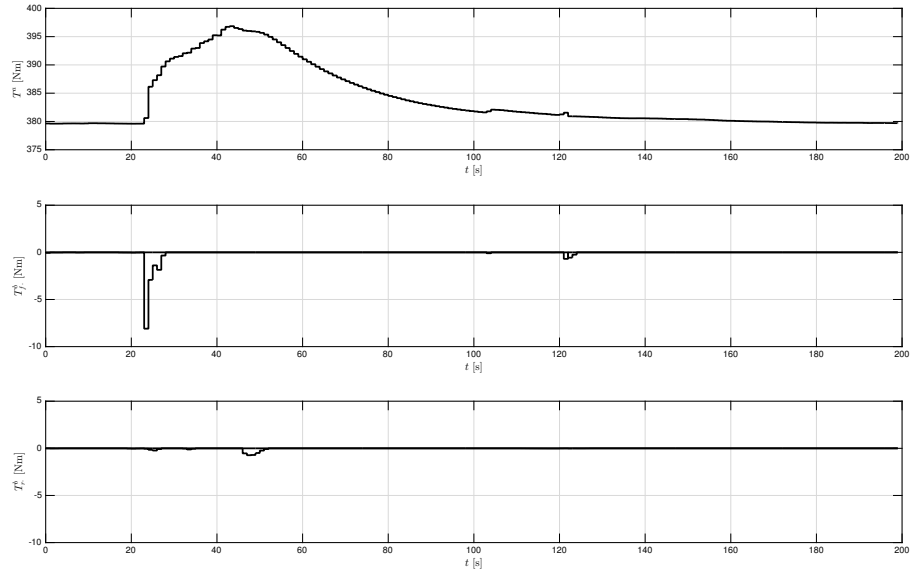


Figure 3.6: Vehicle engine and breaking torques.

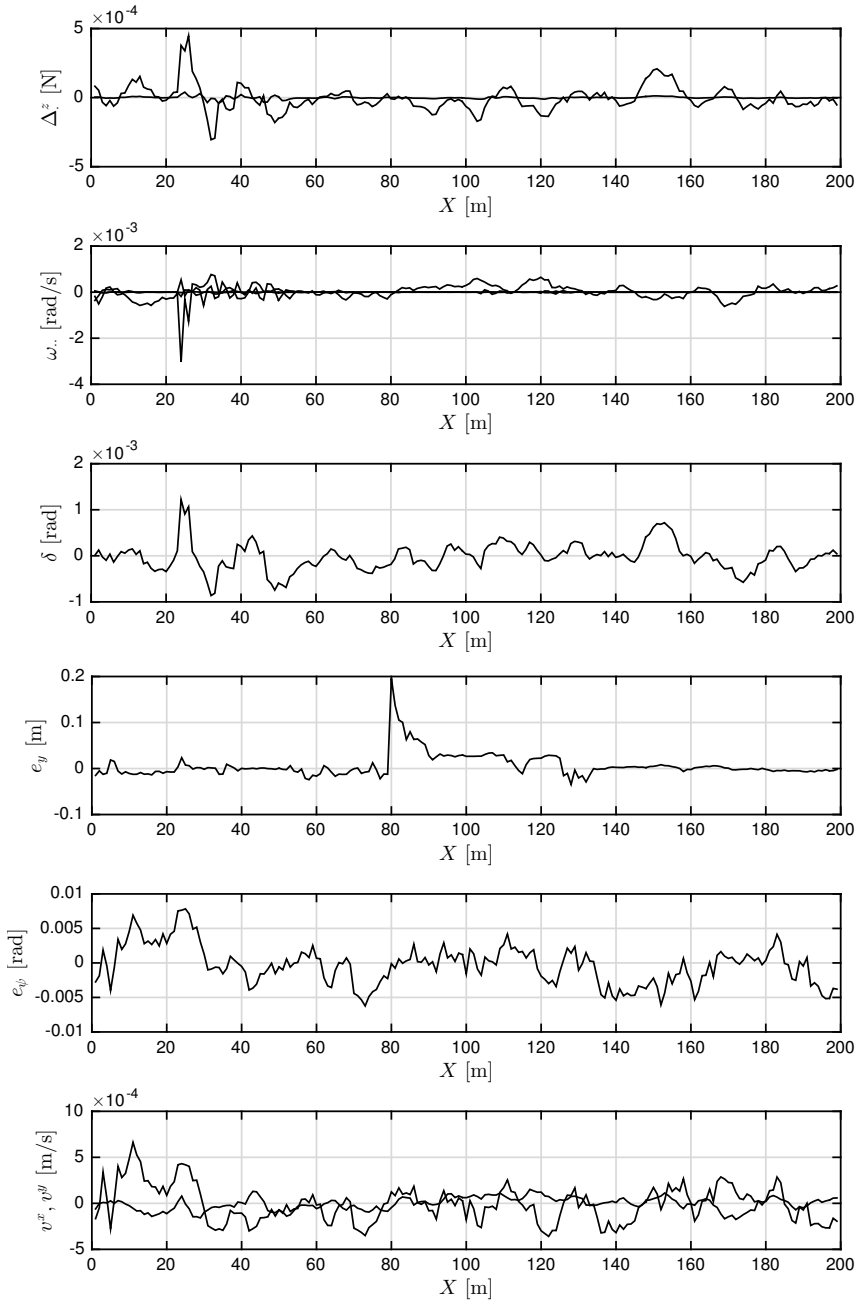


Figure 3.7: Estimation error obtained when using a Huber penalty.

estimate of e_y deteriorates. Indeed, the quality of the estimate of e_y it seems to be strongly connected to the quality of the estimate of the friction coefficient μ. This relation needs to be further studied and will be the object of future research.

The same simulation has been run using a L^2 norm for the friction coefficient estimation and the results are displayed in Figure 3.4. While the car trajectory is the same to eye inspection, the jump in the friction coefficient estimate is detected more slowly than when using a Huber norm. For both simulations, the cost function relative to the fictitious control u_μ has been tuned to try to obtain the best estimate of the friction coefficient μ.

Numerical results obtained by using an L^1 penalty resemble those obtained when using a Huber penalty. The jump in the friction coefficient is quickly detected, i.e. the friction coefficient estimate quickly reaches a neighbourhood of the true value. However, once the estimation error is small, it takes longer to further reduce it until the estimate reaches the true value of the friction coefficient.

The vehicle velocities and steering angle are displayed in Figure 3.5 and the engine and braking torques are displayed in Figure 3.6. It can be seen that the controller maintains the longitudinal speed close to the prescribed one while successfully avoiding the obstacles.

All simulations have been run on an Intel i7 CPU M 640 at 2.8 GHz. The computational times for both MPC and MHE are below 17 ms for the preparation phase and 6 ms for the feedback phase. We remark that, because of the low moment of inertia of each wheel, the model is very stiff and a discretisation approach based on DMS and compression [95] would make it possible to simulate each wheel separately using a tailored integrator. This could probably reduce the computational times even further.

Discussion

The proposed simulation scenario demonstrated the ability of MPC and MHE to safely control the vehicle and estimate the road-tire friction coefficient. An extensive simulation and experimental campaign would be needed in order to test the scheme in several challenging conditions.

Future research directions include the development of an arrival cost update strategy for L^1 and Huber penalties and the implementation on a real vehicle.

3.3 Towards Time Optimal Driving

Due to the natural antagonism of safety and speed in driving, MPC formulations that aim for minimum-time driving while satisfying limitations, such as the boundaries of a race track, seem particularly challenging. Additionally, minimum time objectives do not naturally fit into the class of least-squares objectives for which highly efficient algorithms exist.

Time-optimal MPC approaches have received notable attention in the area of robotics for path following, e.g. [110] and the references therein. There however, the geometric path to be tracked in a minimum-time fashion is predetermined, which renders the problem significantly simpler than the driving problem, where the computation of the path to be taken by the vehicle is part of the online optimisation.

Existing advanced control based attempts to minimum-time driving include [66, 101]. There, however, only offline open-loop solutions were computed. In [26] a cascading controller scheme was used, generating trajectory references from a simple geometric model and tracking these using a higher-detail model. In [112], instead, we propose to use a one-level approach, solving the nonlinear MPC problem with an economic cost function in real-time and we provide experimental results of the proposed nonlinear MPC scheme from a miniature race-car setup.

3.3.1 Current Limitations: Vehicle Model and Available Real-Time Algorithms

Before presenting our approach and the obtained results, we need to underline that the work done so far is a first step towards time optimal driving, whence the title of this section. We discuss in this subsection the main limitations that future research will have to overcome. We also want to underline, however, that these limitations are not fundamental and they are not expected to pose insurmountable obstacles.

Ideally, in order to minimise the time needed to drive on a prescribed road, an accurate vehicle model is needed in order to fully capture the physics of the problem and exploit them. While in principle the model presented in Section 3.1 can be identified, in practice this requires an intensive identification campaign which was not yet possible. Therefore, the vehicle was modelled using a simple slip-free bicycle model. All implementation details can be found in [111, 112]. Future research will aim at identifying a more realistic vehicle model which will also allow to improve the obtained results.

A second limitation comes from the algorithms available for fast real-time feasible NMPC. Currently, the most efficient implementations assume an objective function of least-squares type. While recent extensions towards more generic objective functions have been developed in e.g. [92] where second order derivatives are efficiently computed, some further improvements in the algorithms, e.g. the Hessian regularisation procedure, are still the object of ongoing research. In this Section we restrict ourselves to least-squares cost functions, but future research should also consider directly optimising time, as soon as more advanced real-time algorithms will be available.

Current research is aiming at overcoming these limitations by making use of (a) a more accurate vehicle model and (b) an optimisation algorithm based on exact Hessian.

3.3.2 Problem Formulation

We formulate the time-optimisation problem in the context of system dynamics in the spatial domain. For an approximate solution of the time-optimal driving problem, we aim at minimizing the time required for the race car to reach the end of the fixed-length spatial prediction horizon, i.e.

$$T = \int_{t_0}^{t_f} 1 \, \mathrm{d}t = \int_{s_0}^{s_f} \frac{1}{\dot{s}(\tau)} \, \mathrm{d}\tau \ . \tag{3.13}$$

Note that, in order to solve the problem exactly, the prediction horizon should be infinitely long, or reach the end of the track. However, for sufficiently long prediction horizons, one can expect the solution to approximate well the goal of driving time optimally.

In order to allow for the use of efficient algorithms which rely on least-squares formulations and the generalised Gauss-Newton method, we propose to modify the objective formulation as follows.

Observation 3.1. *Let all track data be fixed. Let T^* be the minimum time required by a vehicle to reach the end of the horizon while satisfying the system dynamics and path constraints. Then, for any $0 < T_{\mathrm{ref}} < T^*$, the global optimum*

(if existent) of the optimisation problem

$$(\hat{\xi}, \hat{u}, \hat{T}) = \underset{\xi(\cdot),u(\cdot),T}{\operatorname{argmin}} \quad \|T - T_{\text{ref}}\|^2_{q_N} \tag{3.14}$$

$$\text{s.t.} \quad \xi'(s) = f(s, \xi(s), u(s)) \tag{3.15}$$

$$e^y(s) \in [e^y_L(s), e^y_U(s)] \tag{3.16}$$

$$u(s) \in [\delta_L, \delta_U] \times [-1, 1] \tag{3.17}$$

$$\xi(0) = \xi_0, \tag{3.18}$$

satisfies $\hat{T} = T^*$.

By providing a sufficiently small (i.e., infeasible) "target time" T_{ref} we can therefore have an approximate time-optimal MPC formulation in least-squares form. Note that, by fixing the sampling grid, the existence of a KKT point is given under rather mild conditions. Note that, in the proposed formulation, a Gauss-Newton algorithm will perform similarly to a steepest descent method. The importance of second order derivatives for guaranteeing fast convergence and the implementation of an exact Hessian algorithm are the object of current research.

3.3.3 Experimental Setup

In our experiments, we use the Kyosho dNaNo model race cars in their 2008 version. Measurements on the car's current position (X, Y), orientation ψ velocity v and yaw rate $\dot{\psi}$ are obtained through a custom-made camera-based infrared sensing system with a sampling speed of 100 Hz. The feedback scheme of the setup is depicted in Fig. 3.8.

State measurements are obtained from a Kalman filter, which also accounts for possibly missing measurements that may arise from sensor defaulting due to external influences. The Kalman filter algorithm assumes a point mass vehicle model to estimate the position, orientation and velocity of the car.

The controller runs on a desktop computer featuring Debian LINUX 7.0 'Wheezy' with its *RT Preempt* real-time kernel patch installed that implements soft real-time guarantees.

The control inputs are sent to the car via a wireless connection. The dNano race cars have been extended with a custom communication module, such that the control signals can be sent over an Asynchronous Connection-Less Bluetooth

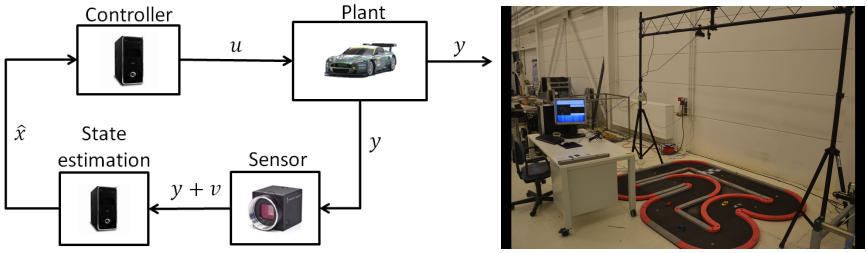


Figure 3.8: Left Figure: schematical view of the experimental setup. Right Figure : the experimental setup.

(ACL) communication link. Further technical details can be found in Appendix A. The considered race track features a chicane, a U-turn and a longer straight section with the car is travelling in clockwise direction. An overview of the setup is shown in Fig. 3.8.

3.3.4 Experimental Results

As previously outlined, the results presented in this section are preliminary and further research is currently aiming at improving them. We compare the time-optimal approach versus tracking the centreline of the track in experiments. We choose a prediction horizon of length 1 m and a target time of $T_{\text{ref}} = 0.24\,\text{s}$ for the time optimal formulation. Note that, as the car maximum velocity is 4 m/s, the chosen target time is infeasible, as required by Observation 3.1. For the trajectory tracking approach, we choose a reference velocity $v_{\text{ref}} = 1.0$ m/s. This was experimentally verified to be the maximum reference speed which could be used for tracking. Higher reference speeds, e.g. $v_{\text{ref}} = 1.2$ m/s resulted in unsafe driving behaviour due to model-plant mismatch [111]. The results are displayed in Figure 3.9. As expected, the time-optimal driving induces the car to drive on the inside of a curve when entering it and on the outside when exiting.

The lap times of the tracking scheme is 9.11 s, while the one of the time-optimal approach is 8.21 s. Both times refer to periodic trajectories, which are obtained after an initial transient of several rounds.

While the proposed approach obtains faster lap times by delivering trajectories which leave the centreline of the track and cut the turns, it is clear that the obtained closed-loop trajectories are still far from being optimal. This fact is to be expected, as we discussed in Section 3.3.1, and several improvements are the object of ongoing research. It is however remarkable that a first improvement

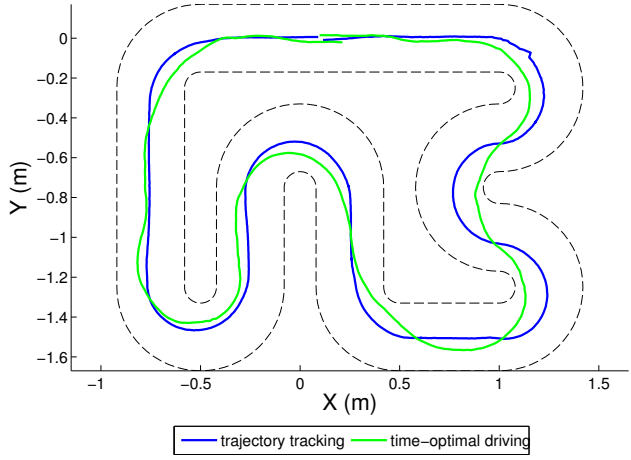


Figure 3.9: Comparison of the performance of trajectory tracking and time-optimal driving on the experimental setup. Note that for tracking $v_{\text{ref}} = 1.0$ m/s and $T_{\text{ref}} = 0.24$ s for time-optimal driving. Source [112].

over a tracking strategy can be obtained even using a very simple vehicle model and problem formulation.

3.4 Conclusions

In this chapter, we have presented a detailed vehicle model which captures the main features of the vehicle dynamics in order to deliver accurate prediction abilities to be used for optimal control, MPC and MHE. We have demonstrated in simulation that it is possible to use such a detailed model in real-time to control a vehicle in an obstacle avoidance scenario while estimating the road-tire friction coefficient.

Having the aim of developing time-optimal MPC, we have developed a first step towards this aim by running experiments on a small-scale race car setup. Even though we used a rather simplified approach, a significant improvement in lap times has been observed in experiments versus a tracking MPC strategy. Current research is aiming at further improving these results by making use of a more accurate model and more advanced optimisation algorithms.

Chapter 4

Stability Theory for Model Predictive Control

In this chapter we will consider discrete time systems of the form

$$x_{k+1} = f(x_k, u_k). \tag{4.1}$$

Note that, given a chosen sampling time T_s, any continuous time system can be transformed in its discrete time form, cf. Section 2.2 and 2.3.

Model Predictive Control (MPC) is a digital control strategy based on predicting the future state and control trajectory in order to minimise a given cost function. This is achieved by repeatedly solving the OCP

$$V_N(\hat{x}_0) = \min_{w} \quad \sum_{k=0}^{N-1} \ell(x_k, u_k) + V_\mathrm{f}(x_N) \tag{4.2a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{4.2b}$$

$$x_{k+1} = f(x_k, u_k), \qquad k = 0, \ldots, N-1, \tag{4.2c}$$

$$(x_k, u_k) \in \mathbb{Z}, \qquad k = 0, \ldots, N-1, \tag{4.2d}$$

$$x_N \in \mathbb{X}_\mathrm{f}. \tag{4.2e}$$

Here $\hat{x}_0$ is the current state, and we use the shorthands $w = (\, w_0, \ldots, w_{N-1}, x_N \,)$, and $w_k = (\, x_k, u_k \,)$. Sets $\mathbb{Z}$ and $\mathbb{X}_\mathrm{f}$ define the path and terminal constraints

respectively. For completeness, we underline that

$$\mathbb{Z} := \{\ (x,u) \mid h(x,u) \geq 0\ \}.$$

We define the MPC feasible set as

$$\mathbb{Z}_N = \{\ (x_0,\, u_0\,) \mid \exists\ (u_1,\ldots,u_{N-1}) \text{ such that}$$
$$x_{k+1} = f(x_k,u_k),\ h(x_k,u_k) \geq 0,\ \text{for all } k=0,\ldots,N-1,\ x_N \in \mathbb{X}_{\mathrm{f}}\ \}.$$

We moreover define

$$\mathbb{X}_N = \{\ x_0 \mid \exists\ (u_0,\ldots,u_{N-1}) \text{ such that}$$
$$x_{k+1} = f(x_k,u_k),\ h(x_k,u_k) \geq 0,\ \forall\, k=0,\ldots,N-1,\ x_N \in \mathbb{X}_{\mathrm{f}}\ \},$$

$$\mathbb{U}_N = \{\ u_0 \mid \exists\ x_0 \in \mathbb{X}_N \text{ and } (u_1,\ldots,u_{N-1}) \text{ such that}$$
$$x_{k+1} = f(x_k,u_k),\ h(x_k,u_k) \geq 0,\ \forall\, k=0,\ldots,N-1,\ x_N \in \mathbb{X}_{\mathrm{f}}\ \},$$

so that $\mathbb{Z}_N \subseteq \mathbb{X}_N \times \mathbb{U}_N$.

Most MPC implementations make use of a quadratic cost function of the form

$$\ell(x_k,u_k) = \begin{bmatrix} x_k - x_{\mathrm{s}} \\ u_k - u_{\mathrm{s}} \end{bmatrix}^{\top} H \begin{bmatrix} x_k - x_{\mathrm{s}} \\ u_k - u_{\mathrm{s}} \end{bmatrix}, \quad V_{\mathrm{f}}(x_N) = (x_N - x_{\mathrm{s}})^{\top} P (x_N - x_{\mathrm{s}}),$$

with $H, P \succ 0$. This choice is justified for applications in which one is interested in stabilising the system at a prescribed equilibrium x_{s}. In general, every MPC scheme having a possibly non quadratic stage cost which penalises the deviation from a given feasible reference is called *tracking MPC*. Stability theory for tracking MPC is rather mature [99, 54].

In many applications, however, one is not only interested in stabilising the system, but a given objective needs to be minimised. This is the case of e.g.

- chemical plants, where one is interested in maximising the revenue by producing the most product at the least cost,
- power plants, where one is interested in producing energy at least cost by taking into account variable prices,
- time optimal plant operation.

In all these cases the cost function can be a general nonlinear function which does not explicitly consider a given reference. All schemes which do not explicitly

penalise the deviation from a given feasible reference are called *economic MPC*. Stability theory for this case is much less mature than for the tracking case.

We remark here that the optimal steady-state or periodic operation could be pre-computed and a tracking MPC scheme could be used to track it. Economic MPC, however, directly optimises the economic objective and therefore usually outperforms tracking MPC in transients. Because systems are typically operated in perturbed environments, the system will mostly be controlled in such transients, so that the advantage of economic MPC is clear. The approach recently developed in this thesis and published in [122, 123] consists in tuning the stage cost for tracking MPC so that it locally approximates the behaviour of economic MPC.

The computational methods for solving Problem (4.2) have been presented in Chapter 2. As optimality does not necessarily imply stability, in this chapter we analyse the stability properties of the MPC closed-loop system. In Section 4.1 we introduce some notation and definitions which we will use throughout the chapter. We prove stability for tracking MPC in Section 4.2 and for economic MPC in Section 4.3. Finally, we present the main contribution of this chapter in Section 4.4, i.e. the extension of the stability theory to periodic economic MPC for both time-invariant and time-varying systems.

Acknowledgement The first part of this Chapter reviews the state of the art and is inspired by [99, 3, 8]. The part on stability of periodic economic MPC (Section 4.4) is based on the two papers [120, 125]. Mario Zanon is the main author of both papers and developed the theoretical results under the supervision of Lars Grüne, Sébastien Gros and Moritz Diehl.

4.1 Preliminaries and Definitions

In this section, we define the concepts needed to establish asymptotic stability. We begin by giving some useful definitions.

Definition 4.1 (Equilibrium point)**.** *A point* $(x_\mathrm{s}, u_\mathrm{s})$ *is an* equilibrium point *of* $x_{k+1} = f(x_k, u_k)$ *if*

$$x_\mathrm{s} = f(x_\mathrm{s}, u_\mathrm{s}). \tag{4.3}$$

For autonomous systems, a point x_s *is an* equilibrium point *of* $x_{k+1} = f(x_k)$ *if*

$$x_\mathrm{s} = f(x_\mathrm{s}). \tag{4.4}$$

Definition 4.2 (Positive invariant set)**.** *A set $\mathcal{A}$ is positive invariant for the system $x_{k+1} = f(x_k)$ if $x \in \mathcal{A}$ implies $f(x) \in \mathcal{A}$.*

A special case is obtained for the set $\mathcal{A} = \{x_s\}$ consisting of a single equilibrium x_s. We define the distance from a set as follows.

Definition 4.3 (Set distance)**.** *The distance of a point x from a set $\mathcal{A}$ is given by*

$$\|x\|_{\mathcal{A}} := \inf_{z \in \mathcal{A}} \|x - z\|. \tag{4.5}$$

Note that, in the special case $\mathcal{A} = \{x_s\}$, then $\|x\|_{\mathcal{A}} = \|x - x_s\|$. If $x_s = 0$, then $\|x_k\|_{\mathcal{A}} = \|x_k\|$.

Definition 4.4 (Stability, attractivity and asymptotic stability)**.** *The closed positive invariant set $\mathcal{A}$ is*

(i) stable *for system $x_{k+1} = f(x_k)$ if, for all $\epsilon > 0$, there exists a $\delta > 0$ such that $\|x_0\|_{\mathcal{A}} < \delta$ implies $\|x_k\|_{\mathcal{A}} < \epsilon$, for all $k \geq 0$;*

(ii) locally stable *in $\mathcal{X}$ for system $x_{k+1} = f(x_k)$ if, for each $\epsilon > 0$, there exists a $\delta > 0$ such that $x_0 \in \mathcal{X}$ and $\|x_0\|_{\mathcal{A}} \leq \delta$ imply $\|x_k\|_{\mathcal{A}} < \epsilon$, for all $k \geq 0$;*

(iii) globally attractive (attractive in $\mathcal{X}$) *for the system $x_{k+1} = f(x_k)$ if $\|x_k\|_{\mathcal{A}} \to 0$ as $k \to \infty$ for all $x_0 \in \mathbb{R}^{n_x}$ ($x_0 \in \mathcal{X}$);*

(iv) globally asymptotically stable *if it is stable and globally attractive;*

(v) asymptotically stable with region of attraction $\mathcal{X}$ *if it is locally stable in $\mathcal{X}$ and attractive in $\mathcal{X}$.*

We now define some classes of functions that we will use to establish stability properties.

Definition 4.5 ($\mathcal{K}$ function)**.** *A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{K}$ (is a $\mathcal{K}$ function) if it is continuous, zero at zero and strictly increasing*

$$\alpha(0) = 0, \qquad \alpha(z) > \alpha(y), \ \forall \ z > y. \tag{4.6}$$

Definition 4.6 ($\mathcal{K}_\infty$ function)**.** *A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{K}_\infty$ (is a $\mathcal{K}_\infty$ function), if $\alpha \in \mathcal{K}$ and it is unbounded, i.e.*

$$\alpha(0) = 0, \qquad \alpha(z) > \alpha(y), \ \forall \ z > y \qquad \lim_{z \to \infty} \alpha(z) = \infty. \tag{4.7}$$

Definition 4.7 ($\mathcal{KL}$ function)**.** *A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{I} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{KL}$ (is a $\mathcal{KL}$ function), if $\beta(\cdot, k) \in \mathcal{K}$ and for each $s \geq 0$, $\beta(s, \cdot)$ is non increasing and $\lim_{k\to\infty} \beta(s, k) = 0$.*

Definition 4.8 (Positive definite function)**.** *A function $\alpha : \mathbb{R} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{PD}$ (is* positive definite*) if it is continuous, and positive everywhere except at the origin, i.e.*

$$\alpha(0) = 0, \qquad \alpha(x) > 0, \ \forall\ x \neq 0. \tag{4.8}$$

Definition 4.9 (Lyapunov function)**.** *A function $V : \mathbb{R}^{n_x} \to \mathbb{R}$ is a* Lyapunov function *for the system $x_{k+1} = f(x_k)$, and the set $\mathcal{A}$ if there exist functions $\alpha_i \in \mathcal{K}_\infty$, $i = 1, 2$ and $\alpha_3 \in \mathcal{PD}$ such that for any $x \in \mathbb{R}^{n_x}$*

$$\alpha_1(\|x\|_{\mathcal{A}}) \leq V(x) \leq \alpha_2(\|x\|_{\mathcal{A}}), \tag{4.9a}$$

$$V(f(x)) - V(x) \leq -\alpha_3(\|x\|_{\mathcal{A}}). \tag{4.9b}$$

If $V(\cdot)$ satisfies Equations (4.9) for all $x \in \mathcal{X}$, where $\mathcal{X} \supset \mathcal{A}$ is a positive invariant set for $x_{k+1} = f(x_k)$, then V is a Lyapunov function in $\mathcal{X}$ for the system $x_{k+1} = f(x_k)$ and set $\mathcal{A}$.

Theorem 4.10 (Lyapunov stability)**.** *Suppose $\mathcal{X} \subset \mathbb{R}^{n_x}$ is positive invariant for $x_{k+1} = f(x_k)$, that $\mathcal{A}$ is closed and positive invariant for $x_{k+1} = f(x_k)$, and that $\mathcal{A}$ lies in the interior of $\mathcal{X}$. If there exists a Lyapunov function $V : \mathcal{X} \to \mathbb{R}$ for the system $x_{k+1} = f(x_k)$ and set $\mathcal{A}$ with $\alpha_3(\cdot)$ a $\mathcal{K}_\infty$ function, then $\mathcal{A}$ is asymptotically stable for $x_{k+1} = f(x_k)$ with a region of attraction $\mathcal{X}$.*

Proof. The proof can be found in [99, 67]. □

Remark 4.11. *Note that in Theorem 4.10, if the set $\mathcal{X}$ is compact, then the same stability result holds if functions α_i, $i = 1, 2$ are $\mathcal{K}$ functions and $\alpha_3 \in \mathcal{PD}$ on $\mathcal{X}$.*

4.2 Tracking MPC

In order to show stability for MPC, we will show that, under some mild assumptions, the value function $V_N(\cdot)$ of the MPC Problem (4.2) is a Lyapunov function for the closed loop system on $\mathbb{X}_N$.

Without loss of generality we consider the case of stabilising the system to a steady state in the origin, i.e. $(x_\mathrm{s}, u_\mathrm{s}) = (0, 0)$.

Assumption 4.12 (Continuity of the cost and system dynamics)**.** *The system dynamics $f : \mathbb{Z} \to \mathbb{R}^{n_x}$, stage cost $\ell : \mathbb{Z} \to \mathbb{R}$, and terminal cost $V_\mathrm{f} : \mathbb{X}_\mathrm{f} \to \mathbb{R}$, are continuous. Moreover, without loss of generality, $f(0,0) = 0$, $\ell(0,0) = 0$, $V_\mathrm{f}(0) = 0$.*

Assumption 4.13. *The set $\mathbb{X}_N$ is closed, the sets $\mathbb{X}_\mathrm{f} \subseteq \mathbb{X}_N$ and $\mathbb{U}_N$ are compact, $x_\mathrm{s} \in \mathbb{X}_\mathrm{f}$ and $(x_\mathrm{s}, u_\mathrm{s}) \in \mathbb{Z}$.*

Assumption 4.14. *There exists a control law $\kappa : \mathbb{X}_\mathrm{f} \to \mathbb{U}$ such that*

(i) $(x_N, \kappa(x_N)) \in \mathbb{Z}$

(ii) $f(x_N, \kappa(x_N)) \in \mathbb{X}_\mathrm{f}$

(iii) $V_\mathrm{f}(f(x_N, \kappa(x_N))) - V_\mathrm{f}(x_N) \leq -\ell(x_N, \kappa(x_N))$

Assumption 4.15 (Bounds on stage and terminal costs)**.** *The stage cost and terminal cost satisfy*

$$\ell(x,u) \geq \alpha_1(\|x\|), \ \forall \ (x,u) \in \mathbb{Z}_N, \tag{4.10}$$

$$V_\mathrm{f}(x) \leq \alpha_2(\|x\|), \ \forall \ x \in \mathbb{X}_\mathrm{f}, \tag{4.11}$$

with $\alpha_i(\cdot) \in \mathcal{K}_\infty$, $i = 1, 2$.

Theorem 4.16 (Tracking MPC stability)**.** *Suppose that Assumptions 4.12, 4.13, 4.14, and 4.15 are satisfied, then x_s is an asymptotically stable equilibrium for the closed loop system with region of attraction $\mathbb{X}_N$.*

Proof. The proof can be found in [99, 54]. □

A stability proof for the case of no terminal cost nor constraints, i.e. $V_\mathrm{f}(x) = 0$ and $\mathbb{X}_\mathrm{f} = \mathbb{X}_N$, has been provided in [54] under some additional assumptions. In particular, it is shown that, provided that the horizon N is long enough and some technical assumptions are met, then MPC with neither terminal cost nor terminal constraints satisfies a relaxed Lyapunov inequality and is able to stabilise the system. We remark, however, that in the absence of terminal cost and constraints, depending on the specific system dynamics and stage cost, closed-loop stability might not be obtained for any prediction horizon N.

4.3 Economic MPC

In the stability proof for tracking MPC, Assumption 4.15 plays a very important role. Clearly, a large class of stage costs violate Assumption 4.15: when aiming

e.g. at maximising the revenue, the energy production or minimising time, no explicit information about a steady state is used in the stage cost and, therefore, the minimum of the stage cost is in general not attained at steady state. All MPC formulations for which the stage cost violates Assumption 4.15 are called *economic MPC*. Therefore, a new theoretical framework needs to be developed for ensuring stability of this class of MPC formulations. For this aim, we first need to provide some definitions.

Definition 4.17 (Optimal steady state)**.** *We define the optimal steady state as*

$$(x_\mathrm{s}, u_\mathrm{s}) = \underset{x,u}{\operatorname{argmin}}\ \ell(x,u) \quad \text{s.t. } x = f(x,u), \quad (x,u) \in \mathbb{Z}. \tag{4.12}$$

Definition 4.18 (Optimal operation at steady state)**.** *The system $x_{k+1} = f(x_k, u_k)$ is* optimally operated at steady state *with respect to the stage cost ℓ if, for each solution satisfying $(x_k, u_k) \in \mathbb{Z}$, $k = 0, 1, \ldots$, the following holds*

$$\liminf_{T\to\infty} \frac{\sum_{k=0}^{T} \ell(x_k, u_k)}{T+1} \geq \ell(x_\mathrm{s}, u_\mathrm{s}). \tag{4.13}$$

If, in addition, one of the following two conditions hold

$$\textit{either:} \quad \liminf_{T\to\infty} \frac{\sum_{k=0}^{T} \ell(x_k, u_k)}{T+1} > \ell(x_\mathrm{s}, u_\mathrm{s}), \tag{4.14a}$$

$$\textit{or:} \quad \liminf_{T\to\infty} \|x_k - x_\mathrm{s}\| = 0, \tag{4.14b}$$

the system is suboptimally operated off steady state*.*

Definition 4.19 (Dissipativity)**.** *The system $x_{k+1} = f(x_k, u_k)$ is dissipative on a set $\mathbb{W} \subseteq \mathbb{W}_x \times \mathbb{W}_u$ such that $\mathbb{W} \subseteq \mathbb{Z}$ with respect to the supply rate $s : \mathbb{W} \to \mathbb{R}$ if there exists a function $\lambda : \mathbb{W}_x \to \mathbb{R}$, which is bounded from below on $\mathbb{W}_x$ and such that the following inequality is satisfied for all $(x_k, u_k) \in \mathbb{W}$:*

$$\lambda(f(x_k, u_k)) - \lambda(x_k) \leq s(x_k, u_k). \tag{4.15}$$

If, in addition, there exists a function $\rho \in \mathcal{PD}$ such that for all $(x_k, u_k) \in \mathbb{W}$:

$$\lambda(f(x_k, u_k)) - \lambda(x_k) \leq -\rho(\|x_k - x_\mathrm{s}\|) + s(x_k, u_k), \tag{4.16}$$

then the system is strictly dissipative on $\mathbb{W}$.

Assumption 4.20 (Compact constraint set)**.** *The set $\mathbb{Z}$ is compact.*

Assumption 4.21 (Strict dissipativity)**.** *The system $x_{k+1} = f(x_k, u_k)$ is strictly dissipative on $\mathbb{Z}_N$ with respect to the supply rate $s(x_k, u_k) = \ell(x_k, u_k) - \ell(x_\mathrm{s}, u_\mathrm{s})$ and the corresponding storage function λ is continuous at x_s.*

We are now ready to state two important theorems.

Theorem 4.22 (Optimal steady state operation)**.** *Suppose that the system $x_{k+1} = f(x_k, u_k)$ is dissipative (strictly dissipative) on $\mathbb{Z}_N$ with respect to the supply rate $s(x_k, u_k) = \ell(x_k, u_k) - \ell(x_\mathrm{s}, u_\mathrm{s})$. Then the system is optimally operated at steady state (suboptimally operated off steady state).*

Proof. The proof is given in [8]. □

Theorem 4.23 (Economic MPC stability)**.** *Suppose that Assumptions 4.12, 4.20, 4.14, and 4.21 are satisfied, then x_s is an asymptotically stable equilibrium for the closed loop system with region of attraction $\mathbb{X}_N$.*

Proof. The proof is given in [3]. □

The stability proof relies on an auxiliary *rotated problem* which is shown to be equivalent to the original problem, but has a *rotated cost* which satisfies Assumption 4.15. By using the so-called storage function $\lambda(\cdot)$, we define the so-called *rotated stage cost* and *rotated terminal cost* respectively as

$$L(x_k, u_k) := \ell(x_k, u_k) - \ell(x_\mathrm{s}, u_\mathrm{s}) + \lambda(x_k) - \lambda(f(x_k, u_k)), \tag{4.17}$$

$$\bar{V}_\mathrm{f}(x_N) := V_\mathrm{f}(x_N) + \lambda(x_N). \tag{4.18}$$

The strict dissipativity condition directly implies that $L(x, u) \geq \rho(\|x - x_\mathrm{s}\|)$, which in turn means that the rotated stage cost is of tracking type, i.e. it satisfies Assumption 4.15. Continuity of the stage and terminal cost and of the storage function and some mild assumptions on V_f are used to prove positive definiteness of the rotated terminal cost, i.e. $\bar{V}_\mathrm{f}(x_N) \geq \rho(\|x_N - x_\mathrm{s}\|)$ so that it also satisfies Assumption 4.15.

We call *rotated MPC problem* the following MPC problem

$$\bar{V}_N(\hat{x}_0) = \min_{w} \quad \sum_{k=0}^{N-1} L(x_k, u_k) + \bar{V}_\mathrm{f}(x_N) \tag{4.19a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{4.19b}$$

$$x_{k+1} = f(x_k, u_k), \qquad k = 0, \ldots, N-1, \tag{4.19c}$$

$$(x_k, u_k) \in \mathbb{Z}, \qquad k = 0, \ldots, N-1, \tag{4.19d}$$

$$x_N \in \mathbb{X}_\mathrm{f}. \tag{4.19e}$$

The rotated MPC problem (4.19) provides the same solution as the original MPC problem (4.2), as the rotated cost only differs from the original one only by a constant term. It can indeed be verified that

$$\sum_{k=0}^{N-1} L(x_k, u_k) + \bar{V}(x_N) = \sum_{k=0}^{N} \ell(x_k, u_k) + V(x_N) + \underbrace{(N-1)\ell(x_\mathrm{s}, u_\mathrm{s}) + \lambda(\hat{x}_0)}_{\text{constant}}.$$

Under a local controllability assumption, dissipativity has been proven to be not only sufficient but also necessary for optimal steady state operation [**?**]. However, there is no result on the necessity of strict dissipativity for stability of economic MPC. In [53], under some additional assumptions, the stability theory has been extended to the case of no terminal cost or constraints.

4.4 Periodic Economic MPC

In some cases, the steady-state performance is outperformed by periodic operation of the system. It is therefore important to extend the stability theory to cover the periodic case. The results have been extended in two different directions in [120, 125]. In [125] the case of time invariant stage cost and system dynamics has been considered using a periodic nonlinear storage function. In [120] time-varying system dynamics and stage cost have been considered using a periodic linear storage function.

4.4.1 Periodic Economic MPC of The Time-Invariant Systems

In the following, we extend the concepts of (optimal) steady state, optimal operation at steady state and strict dissipativity to the periodic case. To this end, we define (optimal) periodic orbit and trajectory, optimal P-periodic operation and P-periodic strict dissipativity.

Definition 4.24 (Periodic orbit)**.** *An ordered p-tuple of points $\Pi = (\bar{x}_0^\mathrm{p}, \dots, \bar{x}_p^\mathrm{p})$, $p \geq 1$, is called a* feasible p-periodic orbit *with control sequence $(\bar{u}_0^\mathrm{p}, \dots, \bar{u}_{p-1}^\mathrm{p})$ if $(\bar{x}_k^\mathrm{p}, \bar{u}_k^\mathrm{p}) \in \mathbb{Z}$, $k = 0, \dots, p-1$, $\bar{x}_0^\mathrm{p} = \bar{x}_p^\mathrm{p}$ and*

$$\bar{x}_{k+1}^\mathrm{p} = f(\bar{x}_k^\mathrm{p}, \bar{u}_k^\mathrm{p}) \quad \text{for } k = 0, \dots, p-1.$$

The number p is called the period *of the orbit Π and if there is no $q \geq 1$ with $q < p$ such that $(\bar{x}^{\mathrm{p}}_k, \bar{u}^{\mathrm{p}}_k) = (\bar{x}^{\mathrm{p}}_{k+q}, \bar{u}^{\mathrm{p}}_{k+q})$ for all $k = 0, \dots, p-q$, then p is called the* minimal period *of Π. Given the corresponding control sequence $\bar{u}^{\mathrm{p}}_0, \dots, \bar{u}^{\mathrm{p}}_{p-1}$ we define the tuple of state-control pairs $\Pi_U := \big((\bar{x}^{\mathrm{p}}_0, \bar{u}^{\mathrm{p}}_0), \dots, (\bar{x}^{\mathrm{p}}_{p-1}, \bar{u}^{\mathrm{p}}_{p-1})\big)$.*

Note that, in this terminology, an equilibrium is a periodic orbit with period $p = 1$. Moreover, for $p > 1$, the periodic orbit is not unique, as phase shifts produce an orbit which is defined by the same states and controls, but in a shifted order. For this reason, we define the periodic trajectory as a periodic orbit with a fixed phase, extended infinitely long into the future.

Definition 4.25 (Periodic trajectory)**.** *A periodic trajectory is defined as:*

(i) A sequence $X^p = (x_0, x_1, x_2, \dots)$, is called a feasible p-periodic trajectory *with control sequence $U^p = (u_0, u_1, u_2, \dots)$ if $(x_k, u_k) \in \mathbb{Z}$, $x_k = x_{k+p}$, $u_k = u_{k+p}$ for all $k = 0, 1, \dots$, and*

$$x_{k+1} = f(x_k, u_k) \quad \text{for } k = 0, 1, \dots.$$

(ii) Given a p-periodic orbit $\Pi = (\bar{x}^{\mathrm{p}}_0, \dots, \bar{x}^{\mathrm{p}}_p)$ and a phase $\phi \in \{0, \dots, p-1\}$*. we define the infinite sequence*

$$X^p_\phi(\Pi) := (\bar{x}^{\mathrm{p}}_\phi, \dots, \bar{x}^{\mathrm{p}}_{p-1}, \bar{x}^{\mathrm{p}}_0, \dots, \bar{x}^{\mathrm{p}}_{p-1}, \dots).$$

The points on $X^p_\phi(\Pi)$ will be denoted by x^ϕ_k, i.e, $x^\phi_k = \bar{x}^{\mathrm{p}}_{(k+\phi) \bmod p}$, and the corresponding control values by u^ϕ_k.

For any p-periodic trajectory, the ordered tuple $(\bar{x}^{\mathrm{p}}_0, \dots, \bar{x}^{\mathrm{p}}_p) = (x^{\mathrm{p}}_0, \dots, x^{\mathrm{p}}_p)$ is a p-periodic orbit Π. Conversely, for every p-periodic orbit Π and any $\phi \in \{0, \dots, p-1\}$ the sequence $X^p_\phi(\Pi)$ from (ii) is a p-periodic trajectory in the sense of (i).

Definition 4.26 (Optimal p-periodic orbit)**.** *A p-periodic orbit Π^* with corresponding control sequence U^{p*} is called* optimal *if it has minimal period p^* and corresponds to the state-control pairs Π^*_U defined as*

$$(p^*, \Pi^*_U) \in \operatorname*{argmin}_{p, \Pi_U} \frac{1}{p} \sum_{k=0}^{p-1} \ell(x_k, u_k), \quad \text{s.t.} \quad (x_k, u_k) \in \mathbb{Z}, \quad k = 0, \dots, p-1. \tag{4.20}$$

where minimization is carried out over all periods $p \geq 1$ and all periodic feasible state-control sequences Π_U of minimal period p. We emphasise that, in general, the argmin is not unique. Also note that the minimum might not exist.

Given a phase ϕ, we denote the elements of the p-periodic trajectory $X_\phi^{p*}(\Pi^*)$ by $(x_0^{\phi*}, x_1^{\phi*}, \ldots)$ and the control values by $(u_0^{\phi*}, u_1^{\phi*}, \ldots)$. The average optimal p-periodic cost (which is independent of ϕ) is given by

$$\ell_p^* := \frac{1}{p} \sum_{k=0}^{p-1} \ell(x_k^{\phi*}, u_k^{\phi*}).$$

For a point x and the periodic orbit Π^*, let us define the distance from Π^* as

$$\|x\|_{\Pi^*} := \min_{\bar{x}_k^{\mathrm{p}*} \in \Pi^*} \|x - \bar{x}_k^{\mathrm{p}*}\|, \tag{4.21}$$

and the distance from Π_U^* as

$$\|(x,u)\|_{\Pi_U^*} := \min_{(\bar{x}_k^{\mathrm{p}*}, \bar{u}_k^{\mathrm{p}*}) \in \Pi_U^*} \|x - \bar{x}_k^{\mathrm{p}*}\| + \|u - \bar{u}_k^{\mathrm{p}*}\|. \tag{4.22}$$

Let us define functions $\sigma^\bullet(x,u)$, as

$$\sigma^{\mathrm{A}}(x,u) := \rho(\|(x,u)\|_{\Pi_U^*}) \tag{4.23}$$

and

$$\sigma^{\mathrm{B}}(x,u) := \rho(\|x\|_{\Pi^*}), \tag{4.24}$$

with ρ being a positive definite function. We remark that in case of (4.24) function σ^{B} does not depend on u, but in order to obtain a uniform notation in what follows we always write $\sigma^\bullet(x,u)$.

Definition 4.27 (Optimal P-periodic operation)**.** *The system $x_{k+1} = f(x_k, u_k)$ is* optimally p-periodically operated *at a periodic orbit Π^* with respect to the stage cost ℓ if, for each solution satisfying $(x_k, u_k) \in \mathbb{Z}$, $k = 0, 1, \ldots$, the following holds*

$$\liminf_{T\to\infty} \frac{\sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \ell(x_{pk+j}, u_{pk+j})}{Tp} \geq \ell_p^*. \tag{4.25}$$

If, in addition, there exist functions $\sigma^\bullet$ of the form (4.23) *or* (4.24) *such that one of the following two conditions hold*

$$\textit{either} \qquad \liminf_{T\to\infty} \frac{\sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \ell(x_{pk+j}, u_{pk+j})}{Tp} > \ell_p^*, \tag{4.26a}$$

$$\textit{or} \qquad \textit{there is } \phi \in \{0, \ldots, p-1\} \textit{ with } \liminf_{k\to\infty} \sigma^\bullet(x_k, u_k) = 0, \tag{4.26b}$$

the system is suboptimally non p-periodically operated *at a periodic orbit Π^* with respect to the stage cost ℓ and the function $\sigma^\bullet$ from* (4.23) *or* (4.24)*.*

If, in addition, for each $\delta > 0$ there exists an integer $\bar{t} \geq 1$ such that one of the following two conditions holds:

$$\sum_{k=0}^{t} \sum_{j=0}^{p} \frac{\ell(x_{pk+j}, u_{pk+j})}{pt} \geq \ell_p^*, \text{for all } t \geq \bar{t}, \tag{4.27a}$$

$$\begin{aligned} &\text{there is } \phi \in \{0, \dots, p-1\} \text{ with} \\ &\sigma^{\bullet}(x_k, u_k) \leq \delta, \text{ for } p \text{ consecutive } k \in [1, \bar{t}], \end{aligned} \tag{4.27b}$$

the system is uniformly suboptimally non p-periodically operated *at a periodic orbit Π^* with respect to the stage cost ℓ and the function $\sigma^{\bullet}$ from* (4.23) *or* (4.24).

Definition 4.28 (P-periodic dissipativity)**.** *The system $x_{k+1} = f(x_k, u_k)$ is P-periodic dissipative on a set $\mathbb{Z}$ with respect to the supply rate $\ell(x_k, u_k) - \ell(x_k^{\phi*}, u_k^{\phi*})$ if there exists a feasible p-periodic orbit Π^* a phase ϕ and bounded storage functions $\lambda_0, \dots, \lambda_{p-1}, \lambda_p, \dots : X \subset \mathbb{R}^{n_x} \to \mathbb{R}$, with $\lambda_{k+p} = \lambda_k$ such that the inequalities*

$$\begin{aligned} L_k(x_k, u_k) := \ell(x_k, u_k) - \ell(x_k^{\phi*}, u_k^{\phi*}) \\ + \lambda_k(x_k) - \lambda_{k+1}(f(x_k, u_k)) \geq 0 \end{aligned} \tag{4.28}$$

hold for all $(x_k, u_k) \in \mathbb{Z}$ and all $k = 0, 1, \dots$, where $x_k^{\phi*}$ are the elements of the sequence $X_{\phi}^{p*}(\Pi^*)$. If, in addition, there exist functions of the form (4.23) *or* (4.24) *such that*

$$L_k(x_k, u_k) \geq \sigma^{\bullet}(x_k, u_k), \quad \bullet \in \{\mathrm{A}, \mathrm{B}\} \tag{4.29}$$

holds, then the system $x_{k+1} = f(x_k, u_k)$ satisfies p-periodic strict dissipativity of type A *or* B*, respectively, on $\mathbb{Z}$.*

It is easily seen that for σ^{B}, i.e. from (4.24), this definition is equivalent to Definition 4.19 in case $p = 1$. Moreover, for $p > 1$, (strict) dissipativity might hold for more than one phase ϕ. While this can be restrictive if one is interested in the actual computation of $L_k(x_k, u_k)$, this does not constitute any problem for the theoretical results that we aim at establishing next, i.e. sufficiency of strict p-periodic dissipativity for optimal p-periodic operation (uniform suboptimal non p-periodic operation), and p-periodic stability of EMPC.

As it holds that $\|(x_k, u_k)\|_{\Pi_U^*} \geq \|x_k\|_{\Pi^*}$, Definition (4.29) in the sense A implies Definition (4.29) in the sense B.

Remark 4.29. *The time-varying and phase-dependent definition*

$$\sigma_k^{\mathrm{C}}(x_k, u_k) := \rho(\|x_k - x_k^{\phi}\|),$$

would at first look like the natural extension of the steady state case. However, in contrast to the time varying case in [120], this definition does not work in the time invariant setting of this paper. More precisely, if $L_k(x,u) \geq \sigma_k^{\mathrm{C}}(x,u)$ for phase ϕ_1 and the rotated cost of the p-periodic optimal trajectory is evaluated for phase $\phi_2 \neq \phi_1$, then we obtain the inequality $\sum_{k=0}^{p} L_k(x_k^{\phi_2}, u_k^{\phi_2}) \geq \rho(\|x_k^{\phi_2} - x_k^{\phi_1}\|)$, which can never be satisfied since $\sum_{k=0}^{p} L_k(x_k^{\phi_2}, u_k^{\phi_2}) = 0$ and $\rho(\|x_k^{\phi_2} - x_k^{\phi_1}\|) > 0$.

Remark 4.30. *We note that the actual behaviour of the trajectories satisfying* (4.27b) *differs depending on $\sigma^{\bullet}$.*

In case of σ^{A}, i.e. from (4.23)*, if Property* (4.27b) *holds for sufficiently small δ, then by continuity of f from $\rho(\|x_k - \bar{x}_k^{\mathrm{P}*}\| + \|u_k - \bar{u}_k^{\mathrm{P}*}\|) \leq \delta$ we obtain $f(x_k, u_k) \approx \bar{x}_{k^+}^{\mathrm{P}*}$ with $k^+ = k+1 \,(mod\ P^*)$. Since the periodic orbit consists of finitely many distinct points, for sufficiently small $\delta > 0$ this implies $\sigma^{\mathrm{A}}(f(x_k, u_k), u_{k+1}) \geq \rho(\|f(x_k, u_k) - \bar{x}_j^{\mathrm{P}*}\| + \|u_{k+1} - \bar{u}_j^{\mathrm{P}*}\|) > \delta$ for all $j \neq k^+$ which together with $\sigma^{\mathrm{A}}(f(x_k, u_k), u_{k+1}) \leq \delta$ yields $\rho(\|f(x_k, u_k) - \bar{x}_{k^+}^{\mathrm{P}*}\| + \|u_{k+1} - \bar{u}_{k^+}^{\mathrm{P}*}\|) \leq \delta$. As a consequence, any state-control sequence sufficiently close to Π_U^* and satisfying strict dissipativity with respect to the supply rate $\ell(x_k, u_k) - \ell(x_k^{\mathrm{P}*}, u_k^{\mathrm{P}*})$ and σ^{A}, approximately follows the periodic motion.*

In contrast to this, in case of σ^{B}, i.e. from(4.24)*, we can only conclude that the solution stays near the set Π_U^* but it need not approximately follow the periodic motion. While it is possible to re-establish approximate periodicity in case Π_U^* is the unique minimiser of $\sum_{p=0}^{p-1}(x_k, u_k)$ over all (not necessarily periodic) orbits of length p, this will require additional arguments in the subsequent proofs and does not directly follow from* (4.24)*, see also Remark 4.38.*

We can now state the following theorems relating dissipativity, optimal operation of the system and economic MPC stability.

Theorem 4.31. *Assume that system $x_{k+1} = f(x_k, u_k)$ is (strictly) p-periodically dissipative on $\mathbb{Z}$ with respect to the supply rate $\ell(x_k, u_k) - \ell(x_k^{\phi*}, u_k^{\phi*})$ and $\sigma^{\bullet}$, $\bullet \in \{\mathrm{A}, \mathrm{B}\}$ from* (4.23) *or* (4.24)*. Then the system is optimally p-periodically operated (uniformly suboptimally non p-periodically operated) at the optimal p-periodic trajectory $X_{\phi}^{p*}(\Pi^*)$.*

Proof. The proof is given in [125] and we recall it in the following. It follows with appropriate adaptations from the one given in [8, Proposition 6.4] and [79,

Theorem 1] for the case $p = 1$. We have

$$\begin{aligned} 0 &= \lim_{T\to\infty} \frac{\lambda_{pT}(x_{pT}) - \lambda_0(x_0)}{pT} \\ &= \lim_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\lambda_{pk+j+1}(x_{pk+j+1}) - \lambda_{pk+j}(x_{pk+j})}{pT} \\ &\leq \liminf_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\ell(x_{pk+j}, u_{pk+j})}{pT} - \ell_p^*. \end{aligned}$$

This establishes the first claim. If strict p-periodic dissipativity holds, then there is a phase ϕ with

$$\begin{aligned} 0 \leq \liminf_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\sigma^\bullet(x_{pk+j}, u_{pk+j})}{pT} \\ \leq \liminf_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\ell(x_{pk+j}, u_{pk+j})}{pT} - \ell_p^*, \end{aligned}$$

and two cases are possible:

1. $\liminf_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\ell(x_{pk+j}, u_{pk+j})}{pT} > \ell_p^*$, or
2. $\liminf_{T\to\infty} \sum_{k=0}^{T-1} \sum_{j=0}^{p-1} \frac{\ell(x_{pk+j}, u_{pk+j})}{pT} = \ell_p^*$, hence $\liminf_{k\to\infty} \sigma^\bullet(x_k, u_k) = 0$.

This proves that strict p-periodic dissipativity entails suboptimal non p-periodic operation. It remains to prove uniform suboptimal non p-periodic operation.

For each feasible solution and $t \geq 0$, (4.29) and boundedness of the λ_k entails that

$$\begin{aligned} -c := -2 \sup_{\substack{0\leq k\leq p-1 \\ x\in\mathbb{X}_\mathbb{Z}}} |\lambda_k(x)| \leq \lambda_{pt}(x_{pt}) - \lambda_0(x_0) \\ \leq \sum_{k=0}^{t-1} \sum_{j=0}^{p-1} [\ell(x_{pk+j}, u_{pk+j}) - \sigma^\bullet(x_{pk+j}, u_{pk+j})] - pt\ell_p^*. \end{aligned}$$

Let $\delta > 0$ be fixed and choose $\bar{t} := \left\lceil \frac{c}{\rho(\delta)} \right\rceil + 1$. Then two cases are possible:

1. $\sum_{k=0}^{t-1}\sum_{j=0}^{p-1}\ell(x_{pk+j},u_{pk+j}) > pt\ell_p^*$ for all $t \geq \bar{t}$, or
2. $\sum_{j=0}^{p-1}\sigma(x_{pk+j},u_{pk+j}) \leq c/\bar{t}$ for at least one $k \in [1,\bar{t}]$, implying $\sigma^\bullet(x_j,u_j) \leq c/\bar{t}$ and thus $\sigma^\bullet(x_j,u_j) \leq \delta$ for $j = pk,\dots,(p+1)k-1$

which concludes the proof. □

Let us consider the following MPC problem at time instant i

$$V^i(\hat{x}_0) = \min_{x,u} \; J_N^i(x,u) = \sum_{k=0}^{N-1}\ell(x_k,u_k) + V_{\mathrm{f}}^{N+i}(x_N) \tag{4.30a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{4.30b}$$

$$x_{k+1} = f(x_k,u_k), \tag{4.30c}$$

$$(x_k,u_k) \in \mathbb{Z}, \tag{4.30d}$$

$$x_N \in \mathbb{X}_{\mathrm{f}}^{N+i}, \tag{4.30e}$$

where the (periodic) terminal set and cost depend on the current time instant i. We note that this time-dependence can be used in order to induce a fixed phase for the EMPC closed loop trajectory. Note however that even if a terminal constraint with phase ϕ_1 is introduced, the closed-loop solution can in general have a phase $\phi_2 \neq \phi_1$. We may also use terminal costs and constraints which are independent of i, in which case the phase is not fixed. We also remark that for non constant $\mathbb{X}_{\mathrm{f}}^{N+i}$ the feasible sets $\mathbb{X}_N^i$, i.e., the sets of all x for which the constraints in (4.30) can be satisfied, depend periodically on i.

The following assumptions are the periodic version of the assumptions used for proving stability of economic MPC in the steady-state case.

Assumption 4.32. *The sets $\mathbb{X}_{\mathrm{f}}^{N+i}$ are compact.*

Assumption 4.33. *The stage cost ℓ and system dynamics f are continuous on $\mathbb{Z}$. The terminal cost function V_{f}^{N+i} is continuous on the terminal region $\mathbb{X}_{\mathrm{f}}^{N+i}$.*

Assumption 4.34 (p-Periodic Strict Dissipativity)**.** *System $x_{k+1} = f(x_k,u_k)$ is p-periodically strictly dissipative at a periodic orbit Π^* with respect to the supply rate $\ell(x_k,u_k) - \ell(x_k^{\phi*},u_k^{\phi*})$ and $\sigma^\bullet$, $\bullet \in \{\mathrm{A},\mathrm{B}\}$ from (4.23) or (4.24). Moreover, the storage functions λ_k are bounded and continuous in every point $x_k^{\phi*} \in \Pi^*$.*

Assumption 4.35. *The value function V^i is bounded on $\mathbb{X}_N^i$ and continuous in every point $x_k^{\phi*} \in \Pi^*$.*

Assumption 4.36. *The terminal regions are periodic, i.e., $\mathbb{X}_\mathrm{f}^{k+p} = \mathbb{X}_\mathrm{f}^k$ for all $k \geq 0$ and the periodic terminal regions $\mathbb{X}_\mathrm{f}^k$ contain the states $x_k^{\phi*}$ of the periodic trajectory $X_\phi^p(\Pi^*)$ from Assumption 4.34. Moreover, the terminal costs are periodic, i.e., $V_\mathrm{f}^{k+p} = V_\mathrm{f}^k$ for all $k \geq 0$ and there exist a terminal control law $\kappa_\mathrm{f}^k : \mathbb{X}_\mathrm{f}^k \to \mathbb{U}$ with $\kappa_\mathrm{f}^{k+p} = \kappa_\mathrm{f}^k$ for all $k \geq 0$ such that, at a given time instant i, for all $x_N \in \mathbb{X}_\mathrm{f}^{N+i}$ the inclusion $f(x, \kappa_\mathrm{f}^{N+i}(x_N)) \in \mathbb{X}_\mathrm{f}^{N+i+1}$ and the inequality*

$$V_\mathrm{f}^{N+i+1}(f(x_N, \kappa_\mathrm{f}^{N+i}(x_N))) \leq V_\mathrm{f}^{N+i}(x_N) - \ell(x, \kappa_\mathrm{f}^{N+i}(x_N)) + \ell(x_{N+i}^{\phi*}, u_{N+i}^{\phi*}),$$

hold.

We remark that in case $\mathbb{X}_\mathrm{f}^k = \{x_k^{\phi*}\}$, Assumption 4.36 is satisfied with $\kappa_\mathrm{f}^k \equiv u_k^{\phi*}$ and $V_\mathrm{f}^k \equiv 0$. The simplest example for time invariant terminal conditions is $\mathbb{X}_\mathrm{f} = \{x_k^{\phi*} \in \Pi^*\}$ with $\kappa_\mathrm{f}(x_k^{\phi*}) = u_k^{\phi*}$ and again $V_\mathrm{f} \equiv 0$.

Note that Assumptions 4.33, 4.34 and 4.36, are the equivalents of Assumptions 4.12, 4.21 and 4.14 for the periodic case. Moreover, Assumption 4.35 follows from Assumptions 4.32, 4.20, 4.34 and 4.36.

Let us define the rotated MPC problem and the corresponding rotated value function as

$$\bar{V}^i(\hat{x}_0) = \min_{x,u} \; \bar{J}_N^i(x,u) = \sum_{k=0}^{N-1} L_{k+i}(x_k, u_k) + \bar{V}_\mathrm{f}^{N+i}(x_N) \tag{4.31a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{4.31b}$$

$$x_{k+1} = f(x_k, u_k), \tag{4.31c}$$

$$(x_k, u_k) \in \mathbb{Z}, \tag{4.31d}$$

$$x_N \in \mathbb{X}_\mathrm{f}^{N+i}, \tag{4.31e}$$

where the rotated terminal and stage cost are phase-dependent and defined respectively as $\bar{V}_\mathrm{f}^{N+i}(x_N) := V_\mathrm{f}^{N+i}(x_N) + \lambda_{N+i}(x_N)$ and $L_k = L_{k(\mathrm{mod}P)}$ from (4.28). These definitions imply

$$\bar{J}_N^i(x,u) = J_N^i(x,u) + \lambda_i(\hat{x}_0) - \sum_{k=0}^{N-1} \ell(x_k^{\phi*}, u_k^{\phi*}) \tag{4.32}$$

and thus the rotated MPC problem delivers the same optimal trajectories and control sequences as the original problem, see also [120].

Theorem 4.37. *Let Assumptions 4.32, 4.33, 4.34, 4.35 and 4.36 hold. Then the rotated optimal value function $\bar{V}(\hat{x}_0)$ is a Lyapunov function and*

(i) for σ^{A}, i.e. from (4.23), there exists a phase ϕ such that the trajectory $x_k^{\phi*}$ corresponding to the optimal periodic orbit Π^* is asymptotically stable for the closed loop system;

(ii) for σ^{B}, i.e. from (4.24), the optimal periodic orbit Π^* is an asymptotically stable set for the closed loop system.

Proof. The proof is given in [125] and we recall it in the following. It uses ideas similar to the steady state case [3] with appropriate adaptations. We define $\sigma^*(x) := \inf_{u\in U} \sigma^{\mathrm{A}}(x,u) = \inf_{u\in U} \sigma^{\mathrm{B}}(x,u)$. Formula (4.32) and the boundedness and continuity in every $x_k^{\phi*}$ from Π^* of V^i and λ_i ensured by Assumptions (4.34) and (4.35) imply that $\bar{V}^i$ is also bounded on $\mathbb{X}_N^i$ and continuous in every $x_k^{\phi*}$ from Π^*. Moreover, strict dissipativity implies $\bar{V}^i(x) \geq L_k(x,u) \geq \sigma^*(x)$ and $L_k(x_k^{\phi*}, u_k^{\phi*}) = 0$ implies $\bar{V}^i(x_i^{\phi*}) = 0$. Together, these properties ensure the existence of $\mathcal{K}$ functions $\hat{\alpha}$ and α such that

$$\hat{\alpha}(\sigma^*(x)) \leq \bar{V}^i(x) \leq \alpha(\sigma^*(x)).$$

Note that local loss of controllability near the periodic optimal trajectory can entail a discontinuity of $V^i(\cdot)$ at Π^* and hence of $\bar{V}^i(\cdot)$. If the cost $\bar{V}^i(\cdot)$ is not continuous at the periodic optimal trajectory, we cannot establish the upper bound $\bar{V}^i(x) \leq \alpha(\sigma^*(x))$.

In order to prove descent of the rotated value function $\bar{V}^i(x)$, let us define the optimal state and control trajectory as

$$X_i^{\mathrm{MPC}} = (x_{0,i}^{\mathrm{MPC}}, x_{1,i}^{\mathrm{MPC}}, \ldots, x_{N,i}^{\mathrm{MPC}}),$$

$$U_i^{\mathrm{MPC}} = (u_{0,i}^{\mathrm{MPC}}, u_{1,i}^{\mathrm{MPC}}, \ldots, u_{N-1,i}^{\mathrm{MPC}}).$$

Let us moreover define a feasible candidate trajectory for the MPC problem at the next time step as

$$\bar{X}_{i+1} = (x_{1,i}^{\mathrm{MPC}}, \ldots, x_{N,i}^{\mathrm{MPC}}, f(x_{N,i}^{\mathrm{MPC}}, \kappa_{\mathrm{f},i}^k(x_{N,i}^{\mathrm{MPC}}))),$$

$$\bar{U}_{i+1} = (u_{1,i}^{\mathrm{MPC}}, \ldots, u_{N-1,i}^{\mathrm{MPC}}, \kappa_{\mathrm{f},i}^k(x_{N,i}^{\mathrm{MPC}})).$$

The rotated objective value associated with this trajectory is given by

$$\begin{aligned}\bar{J}^{i+1}(x_{1,i}^{\mathrm{MPC}}) = \bar{V}^i(x_{0,i}^{\mathrm{MPC}}) &- L_i(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}}) \\ &- V_{\mathrm{f}}^{N+i}(x_{N,i}^{\mathrm{MPC}}) + V_{\mathrm{f}}^{N+i}(f(x_{N,i}^{\mathrm{MPC}}, \kappa_{\mathrm{f}}^{N+i}(x_{N,i}^{\mathrm{MPC}}))) \\ &+ \ell(x_N, \kappa_{\mathrm{f}}^{N+i}(x_N)) - \ell(x_{N+1+i}^{\phi*}, u_{N+1+i}^{\phi*}) \\ &\leq \bar{V}^i(x_{0,i}^{\mathrm{MPC}}) - L_i(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}}).\end{aligned}$$

Optimality implies $\bar{V}^{i+1}(x_{1,i}^{\mathrm{MPC}}) \leq \bar{J}^{i+1}(x_{1,i}^{\mathrm{MPC}})$ and hence

$$\begin{aligned}\bar{V}^{i+1}(x_{1,i}^{\mathrm{MPC}}) - \bar{V}^i(x_{0,i}^{\mathrm{MPC}}) &\leq -L_i(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}}) \\ &\leq -\sigma^{\bullet}(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}}).\end{aligned}$$

The periodic family of rotated value functions is hence a family of Lyapunov functions for the nonlinear system; particularly, $\bar{V}^i$ converges to 0 along the closed loop trajectory. From this, for σ^{B} from (4.24), the claimed stability properties immediately follow.

For σ^{A} from (4.23), the lower bound $\hat{\alpha}(\sigma^*(x))$ of the Lyapunov functions only implies the convergence of the states of the closed loop to Π but not necessarily of the controls. Hence, the proof so far only shows asymptotic stability of the set Π but not of the periodic trajectory $x_k^{\phi*}$ corresponding to Π^*. However, from the last inequality, above, we obtain

$$\sigma^{\mathrm{A}}(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}}) \leq \bar{V}^i(x_{0,i}^{\mathrm{MPC}})$$

implying that since V^i tends to 0 the value $\sigma^{\mathrm{A}}(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}})$ also tends to 0. By (4.23) this yields that $\|(x_{0,i}^{\mathrm{MPC}}, u_{0,i}^{\mathrm{MPC}})\|_{\Pi_u}$ tends to 0 as V_i tends to 0 and thus asymptotic stability of the trajectory corresponding to the periodic orbit Π follows by similar arguments as in Remark 4.30. □

Remark 4.38. *In case of strict dissipativity of type B, i.e. with σ^{B} from* (4.24)*, asymptotic stability of the periodic trajectory $x_k^{\phi*}$ follows if the optimal periodic orbit Π^* is the unique minimiser of $J_p(x,u)$ over all (not necessarily periodic) orbits of length p. Indeed, in this case for $x_{0,i}^{\mathrm{MPC}}$ sufficiently close to Π^*, due to continuity X_i^{MPC} must approximately follow $x_k^{\phi*}$ because otherwise we would obtain a contradiction to the optimality of X_i^{MPC}.*

The following examples illustrate the proposed concepts.

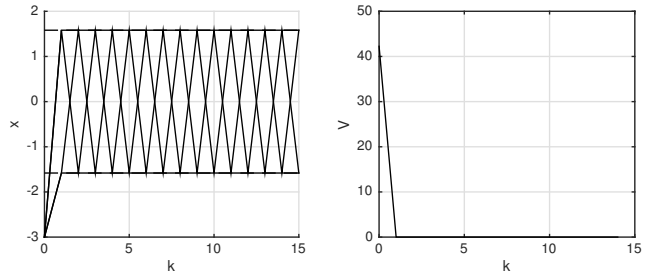


Figure 4.1: Example 4.39. Left graph: closed-loop trajectory (continuous line) obtained starting from $\hat{x}_0 = -3$. The periodic optimal states are displayed in dashed line. Right graph: Value function of the rotated MPC problem.

Example 4.39 (Strict Dissipativity of type B)**.** *Consider the 1d nonlinear dynamics* $f(x,u) = -x + u$ *and stage cost*

$$\ell(x,u) = (x-2)(x-1)(x+1)(x+2).$$

The optimal trajectory can either be of period $p = 1$*, i.e. one of the two steady states* $x_{\mathrm{s}}^{1,2} = \pm\frac{\sqrt{10}}{2}$*, or of period* $p = 2$*, with* $\Pi^* = \left(\frac{\sqrt{10}}{2}, -\frac{\sqrt{10}}{2}\right)$ *and* $\bar{u}_1^{\mathrm{p}*} = \bar{u}_0^{\mathrm{p}*} = 0$*. Using* $\lambda_0(x) = \lambda_1(x) = 0$*, it can be verified that* $L_0(x,u) = L_1(x,u)$ *satisfy the strict dissipation of type B, i.e. with* $\sigma^{\mathrm{B}}(\cdot,\cdot)$ *from* (4.24)*.*

As the control is not bounded and it does not enter the cost, any MPC scheme will stabilise the system in one step. The solution is however not unique and we can conclude from Definition (4.24) *that the system will be stabilised to the set of states which includes the periodic optimal trajectory. However, both staying at one of the steady states and moving from one to the other are optimal moves. Using the initial condition* $\hat{x}_0 = -3$ *and the terminal constraint* $x_N = \left(-\sqrt{10}/2\right)^{i+N+1}$*, all possible closed-loop trajectories and the value of the rotated problem are displayed in Figure 4.1. Note that, as the control effort is not penalised by the stage cost, the system is stabilised to the optimal operation in one step.*

Example 4.40 (Strict Dissipativity of type A)**.** *Consider the 1d system with dynamics* $f(x,u) = u$ *and stage cost*

$$\ell(x,u) = x^4 - \frac{x^3}{3} - 2x^2 + x + (x+u)^2.$$

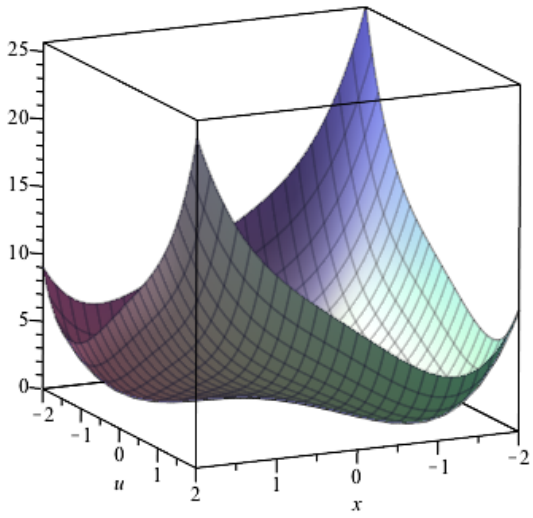


Figure 4.2: Example 4.40: graph of the rotated stage costs $L_1 = L_2$.

Using

$$\lambda_1(x) = -\frac{x^4}{2} + \frac{x^3}{6} + x^2 - \frac{x}{2} + \frac{2}{3}, \quad \lambda_2(x) = -\frac{x^4}{2} + \frac{x^3}{6} + x^2 - \frac{x}{2},$$

$\Pi^* = (1,-1)$ *and* $\Pi_U^* = ((1,-1),(-1,1))$*, one obtains*

$$L_1(x,u) = L_2(x,u) = \frac{x^4}{2} - \frac{x^3}{6} + \frac{x}{2} + 1 + \frac{u^4}{2} - \frac{u^3}{6} + \frac{u}{2} + 2xu.$$

One checks that this polynomial has exactly two local minima at $(1,-1)$ *and* $(-1,1)$ *at which its value is* 0*, cf. Figure 4.2. Hence, it is positive elsewhere and since it grows unboundedly for* $|x|, |u| \to \infty$*, we can find* $\rho \in \mathcal{K}_\infty$ *such that* (4.29) *holds with* $\sigma^\bullet(\cdot,\cdot)$*,* $\bullet = \{\mathrm{A}, \mathrm{B}\}$*, i.e. from* (4.23) *or* (4.24)*.*

For an MPC scheme with horizon $N = 5$*, initial condition* $\hat{x}_0 = -3$ *and terminal constraint* $x_N = (-1)^{i+N+1}$*, the closed-loop trajectory and the value of the rotated problem are displayed in Figure 4.3.*

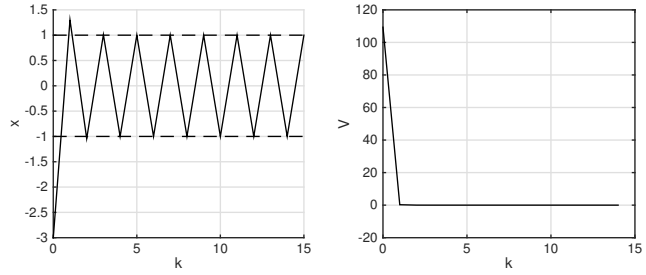


Figure 4.3: Example 4.40. Left graph: closed-loop trajectory (continuous line) obtained starting from $\hat{x}_0 = -3$. The periodic optimal states are displayed in dashed line. Right graph: Value function of the rotated MPC problem.

Example 4.41 (Strict Dissipativity of type B for a 2d system)**.** *Define* $x = [z,\, y]^\top$ *and consider the nonlinear dynamics and stage cost*

$$f(x,u) = \begin{bmatrix} -0.9\,z + y\,u \\ -y \end{bmatrix},$$

$$\ell(x,u) = (z-1.9)(z-0.9)(z+1.1)(z+2.1) + (u-20)^2,$$

with constraint $y \in \{-1, 1\}$*.*

Let us consider the case $\{\bar{y}_0^{\mathrm{p}*}, \bar{y}_1^{\mathrm{p}*}\} = \{1, -1\}$*. The optimal orbit is periodic with period* $P = 2$ *and can be computed numerically:* $\Pi^* = \{(\bar{z}_0^{\mathrm{p}}, \bar{y}_0^{\mathrm{p}*})^\top, (\bar{z}_1^{\mathrm{p}*}, \bar{y}_1^{\mathrm{p}*})^\top\}$ *with* $\bar{z}_0^{\mathrm{p}*} \approx -1.8294$*,* $\bar{z}_1^{\mathrm{p}*} \approx 1.6719$*,* $\{\bar{u}_0^{\mathrm{p}*}, \bar{u}_1^{\mathrm{p}*}\} \approx \{0.0254, 0.3247\}$ *and Lagrange multipliers associated to the* z*-variable of the dynamic constraints* $\{\lambda_0^{\mathrm{p}}, \lambda_1^{\mathrm{p}}\} \approx \{39.9492, -39.3506\}$*.*

Using

$$\lambda_0(x) = (dy - c)(z - z_0) + (1-y)e \qquad \lambda_1(x) = (dy - c)(z - z_1),$$

with $d = \frac{\lambda_0^{\mathrm{p}} - \lambda_1^{\mathrm{p}}}{2}$ *and* $c = \frac{\lambda_0^{\mathrm{p}} + \lambda_1^{\mathrm{p}}}{2}$*, we obtain*

$$L_0(x,u) = \ell(x,u) - \ell(x_0,u_0) + \lambda_0(x) - \lambda_1(f(x,u)) + (1-y)e,$$

$$L_1(x,u) = \ell(x,u) - \ell(x_1,u_1) + \lambda_1(x) - \lambda_0(f(x,u)) - (1+y)e.$$

Computing the minima of these functions reveals that for $e \approx 6.89763344$ *the functions* L_k *satisfy the strict dissipation inequalities* (4.29) *for* $\sigma^{\mathrm{A}}(\cdot,\cdot)$*, cf. Figure 4.4.*

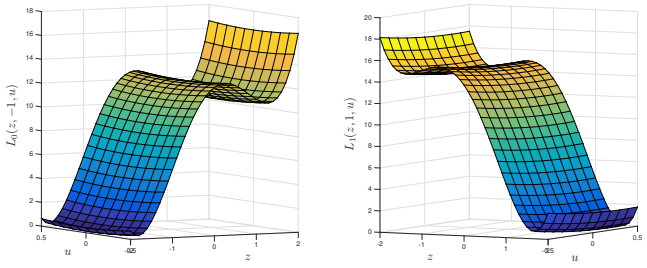


Figure 4.4: Example 4.41: Rotated stage cost $L_0(z, y, u)$ with $y = -1$ fixed (left) and $L_1(z, y, u)$ with $y = 1$ fixed (right).

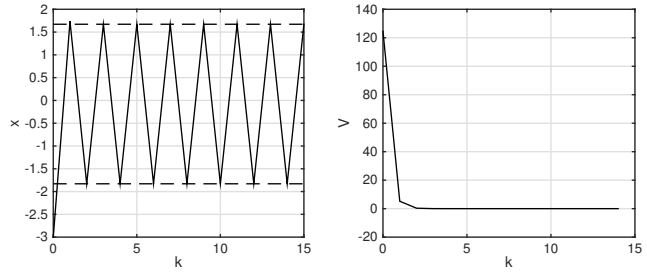


Figure 4.5: Example 4.41. Left graph: closed-loop trajectory (continuous line) obtained starting from $\hat{x}_0 = -3$. The periodic optimal states are displayed in dashed line. Right graph: Value function of the rotated MPC problem.

For an MPC scheme with horizon $N = 5$ and an initial condition $\hat{x}_0 = (-3, 1)$ with terminal constraint $x_N = (-1)^{i+N+1}(\bar{z}_0^{\mathrm{P}*}, -1)$, the closed-loop trajectory and the value of the rotated problem are displayed in Figure 4.5.

4.4.2 Periodic Economic MPC of Time-Varying Systems

Periodic Economic MPC of time-varying systems has been analysed in [120] under more restrictive assumptions than those used for time-invariant systems in the previous section. The extension of these results to a framework using less restrictive assumptions is the object of ongoing research.

In the following we will refer to all time-varying variables and functions in their trajectory form, so that, for a given period p, quantity a_k will satisfy $a_k = a_{k+p}$, for all $k \in \{\ldots, -1, 0, 1, \ldots\}$. We define the state and control

trajectories with index starting from i over a horizon N as $x_i^N = (x_i, \dots, x_{N+i})$ and $u_i^{N-1} = (u_i, \dots, u_{N+i-1})$.

We define the periodic optimal trajectory as $x^{\mathrm{p}} = x_0^{p*}$,$u^{\mathrm{p}} = u_0^{p*}$, i.e. as the solution of the periodic optimal control problem

$$\min_{x_0^p, u_0^p} \sum_{j=0}^{p-1} \ell_j(x_j, u_j) \tag{4.33a}$$

$$\text{s.t. } h_j(x_j, u_j) \geq 0, \qquad \forall\ j = 0, \dots, p-1. \tag{4.33b}$$

Let us consider the following MPC problem at time instant i

$$V^i(\hat{x}_i) = \min_{x_i^N, u_i^{N-1}} \sum_{k=i}^{N+i-1} \ell_k(x_k, u_k) \tag{4.34a}$$

$$\text{s.t. } x_i = \hat{x}_i, \tag{4.34b}$$

$$x_{k+1} = f_k(x_k, u_k), \qquad k = i, \dots, N+i-1, \tag{4.34c}$$

$$(x_k, u_k) \in \mathbb{Z}_k, \qquad k = i, \dots, N+i-1, \tag{4.34d}$$

$$x_N = x_{N+i}^{\mathrm{p}}. \tag{4.34e}$$

We define the MPC feasible set as

$$\mathbb{Z}_N^i = \{\ (x_0,\ u_0\ ) \mid \exists\ (u_{i+1}, \dots, u_{N+i-1}) \text{ such that } x_{k+1} = f_k(x_k, u_k),$$
$$h_k(x_k, u_k) \geq 0, \text{ for all } k = i, \dots, N+i-1,\ x_{N+i} = x_{N+i}^{\mathrm{p}}\ \}.$$

We moreover define

$$\mathbb{X}_N^i = \{\ x_0 \mid \exists\ (u_i, \dots, u_{N+i-1}) \text{ such that } x_{k+1} = f_k(x_k, u_k),$$
$$h_k(x_k, u_k) \geq 0,\ \forall\, k = i, \dots, N-1,\ x_{N+i} = x_{N+i}^{\mathrm{p}}\ \},$$

$$\mathbb{U}_N^i = \{\ u_0 \mid \exists\ x_i \in \mathbb{X}_N^i \text{ and } (u_{i+1}, \dots, u_{N+i-1}) \text{ such that } x_{k+1} = f_k(x_k, u_k),$$
$$h_k(x_k, u_k) \geq 0,\ \forall\, k = i, \dots, N+i-1,\ x_{N+i} = x_{N+i}^{\mathrm{p}}\ \},$$

so that $\mathbb{Z}_N^i \subseteq \mathbb{X}_N^i \times \mathbb{U}_N^i$.

In order to state the stability result, we require the following assumptions to hold.

Assumption 4.42 (Lipschitz Continuity of f_k and ℓ_k)**.** *We assume Lipschitz continuity of $f_k(\cdot)$ and $l_k(\cdot)$ on the admissible set $\mathbb{Z}_N^i$, i.e. there exist constants $L_f, L_\ell > 0$ such that*

$$\begin{matrix} \|f_k(x_k,u_k) - f_k(\hat{x}_k,\hat{u}_k)\| & \leq L_f\|(x_k,u_k) - (\hat{x}_k,\hat{u}_k)\| \\ \|\ell_k(x_k,u_k) - \ell_k(\hat{x}_k,\hat{u}_k)\| & \leq L_\ell\|(x_k,u_k) - (\hat{x}_k,\hat{u}_k)\| \end{matrix}, \ k = 0,\dots,P-1,$$

where $\|.\|$ is any appropriate norm.

Assumption 4.43 (Weak Controllability)**.** *There exists a $\mathcal{K}_\infty$ function $\gamma(\cdot)$ such that for every $\hat{x}_i \in \mathbb{X}_N^i$*

$$\sum_{k=i}^{N+i-1} \|u_k - u_k^{\mathrm{p}}\| \leq \gamma(\|\hat{x}_i - x_i^{\mathrm{p}}\|). \tag{4.35}$$

We define the rotated stage cost using a linear time varying storage function as

$$L_k(x_k,u_k) := \ell_k(x_k,u_k) - \ell_k(x_k^{\mathrm{p}},u_k^{\mathrm{p}}) + (x_k - x_k^{\mathrm{p}})^\top \lambda_k^{\mathrm{p}} - (f_k(x_k,u_k) - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}}. \tag{4.36}$$

Assumption 4.44 (Strong Duality of the Periodic Problem)**.** *There exists a series of multipliers λ_j^{p} so that the trajectory x^{p}, u^{p} uniquely solves*

$$\min_{x_0^p, u_0^p} \ \sum_{j=0}^{p-1} L_j(x_j,u_j) \tag{4.37a}$$

$$\text{s.t. } \ h_j(x_j,u_j) \geq 0, \qquad j = 0,\dots,p-1. \tag{4.37b}$$

Moreover, there exists a $\mathcal{K}_\infty$ function $\beta(\cdot)$ such that the rotated stage cost functions satisfy

$$L_k(x_k,u_k) \geq \beta(\|x_k - x_k^{\mathrm{p}}\|), \tag{4.38}$$

for all $(x_k,u_k) \in \mathbb{Z}_k$, i.e. satisfying $h_k(x_k,u_k) \geq 0$.

Assumption 4.43 bounds the cost of steering the system from an admissible initial state $\hat{x}_0$ to x_N^{p} in N steps while satisfying the state and control constraints.

Assumption 4.44 holds for strictly convex problems, which include periodic optimal control of linear time-varying (LTV) systems subject to convex constraints $h_k(x_k,u_k) \geq 0$. This assumption could be replaced by a

generalization of the dissipativity assumption proposed in [3, 125]. Yet, for the sake of simplicity, the more restrictive Assumption 4.44 is used here.

Note that Assumptions 4.42 and 4.43 directly enforce Assumptions 4.33 and 4.35. Assumptions 4.32 and 4.36 are not necessary since we only consider the simpler but more restrictive case of a point terminal constraint. Finally, Assumption 4.34 is replaced by the more restrictive Assumption 4.44.

Theorem 4.45. *Let Assumptions 4.42, 4.43 and 4.44 hold. Then the trajectory x_k^{p} is asymptotically stable for the closed-loop system with $\mathbb{X}_N^i$ as the region of attraction.*

Proof. The proof was given in [121] and we report it here for completeness. The proof is based on the following rotated MPC scheme:

$$\bar{V}^i(\hat{x}_i) := \min_{x_i^N, u_i^{N-1}} \sum_{k=i}^{N+i-1} L_k(x_k, u_k) \tag{4.39a}$$

$$\text{s.t.} \quad x_i = \hat{x}_i, \tag{4.39b}$$

$$x_{k+1} = f_k(x_k, u_k), \qquad k = i, \dots, N+i-1, \tag{4.39c}$$

$$(x_k, u_k) \in \mathbb{Z}_k, \qquad k = i, \dots, N+i-1, \tag{4.39d}$$

$$x_N = x_{N+i}^{\mathrm{p}}, \tag{4.39e}$$

First, we will prove that Problem (4.39) delivers the same primal solution as Problem (4.34). Second, we will prove that the rotated value function $\bar{V}^i(\hat{x}_i)$ is a Lyapunov function for the closed-loop system, which yields the desired stability property [63].

Problem (4.39) is subject to the same constraints as Problem (4.34). Moreover, the rotated cost only differs from the original cost by a constant term, as it can

be seen by using (4.36) to obtain the telescopic sum:

$$\begin{aligned}\sum_{k=i}^{N+i-1} L_k(x_k,u_k) &= \sum_{k=i}^{N+i-1} \Big(\ell_k(x_k,u_k) - \ell_k(x_k^{\mathrm{p}},u_k^{\mathrm{p}}) + (x_k - x_k^{\mathrm{p}})^\top \lambda_k^{\mathrm{p}}\Big) \\ &\quad - \sum_{k=i}^{N+i-1} (f_k(x_k,u_k) - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}} \\ &= \sum_{k=i}^{N+i-1} \Big(\ell_k(x_k,u_k) - \ell_k(x_k^{\mathrm{p}},u_k^{\mathrm{p}})\Big) \\ &\quad + (x_i - x_i^{\mathrm{p}})^\top \lambda_i^{\mathrm{p}} + \sum_{k=i}^{N+i-2} (x_{k+1} - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}} \\ &\quad - \sum_{k=i}^{N+i-1} (f_k(x_k,u_k) - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}}\end{aligned}$$

For all feasible trajectories $f_k(x_k,u_k) = x_{k+1}$ so that

$$\begin{aligned}&\sum_{k=i}^{N+i-1} (f_k(x_k,u_k) - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}} = \\ &\qquad \sum_{k=i}^{N+i-2} (x_{k+1} - x_{k+1}^{\mathrm{p}})^\top \lambda_{k+1}^{\mathrm{p}} - (x_{N+i} - x_{N+i}^{\mathrm{p}})^\top \lambda_{N+i}^{\mathrm{p}}.\end{aligned}$$

It therefore holds that that

$$\sum_{k=i}^{N+i-1} L_k(x_k,u_k) = \sum_{k=i}^{N+i-1} \ell_k(x_k,u_k) \underbrace{- \sum_{k=i}^{N+i-1} \ell_k(x_k^{\mathrm{p}},u_k^{\mathrm{p}}) + (\hat{x}_i - x_i^{\mathrm{p}})^\top \lambda_i^{\mathrm{p}}}_{\text{constant}},$$

where feasibility of the initial and terminal constraints implies $x_i = \hat{x}_i$ and $x_{N+i} - x_{N+i}^{\mathrm{p}} = 0$ respectively. This proves that Problem (4.39) delivers the same primal solution as Problem (4.34).

We now turn to proving that the rotated value function $\bar{V}^i(\hat{x}_i)$ is a Lyapunov function for the closed-loop system: we will first prove the bound from below, then the descent condition and finally the bound from above.

By Assumption 4.44, we have that $L_k(x_k,u_k) \geq \beta(\|x_k - x_k^{\mathrm{p}}\|)$, therefore $\bar{V}^i(\hat{x}_i) \geq \beta(\|\hat{x}_i - x_i^{\mathrm{p}}\|)$, i.e. the rotated value function is bounded from below by the $\mathcal{K}_\infty$ function β.

In order to prove descent of the rotated value function, let us define the optimal state and control trajectory as

$$X_i^{\mathrm{MPC}} = (\hat{x}_i, x_{i+1,i}^{\mathrm{MPC}}, \dots, x_{N+i,i}^{\mathrm{p}}), \quad U_i^{\mathrm{MPC}} = (u_{i,i}^{\mathrm{MPC}}, u_{i+1,i}^{\mathrm{MPC}}, \dots, u_{N+i-1,i}^{\mathrm{MPC}}).$$

Let us moreover define a feasible candidate trajectory for the MPC problem at the next time step as

$$\bar{X}_{i+1} = (x_{i+1,i}^{\mathrm{MPC}}, \dots, x_{N+i,i}^{\mathrm{p}}, x_{N+i+1,i}^{\mathrm{p}}), \quad \bar{U}_{i+1} = (u_{i+1,i}^{\mathrm{MPC}}, \dots, u_{N+i-1,i}^{\mathrm{MPC}}, u_{N+i,i}^{\mathrm{p}}).$$

The rotated objective value associated with this trajectory is given by

$$\bar{J}^{i+1}(x_{i+1,i}^{\mathrm{MPC}}, \bar{U}_{i+1}) = \bar{V}^i(\hat{x}_i) - L_i(\hat{x}_i, u_{i,i}^{\mathrm{MPC}})$$

Optimality implies $\bar{V}^{i+1}(x_{i+1,i}^{\mathrm{MPC}}) \leq \bar{J}^{i+1}(x_{i+1,i}^{\mathrm{MPC}}, \bar{U}_{i+1})$ and hence

$$\bar{V}^{i+1}(x_{i+1,i}^{\mathrm{MPC}}) - \bar{V}^i(\hat{x}_i) \leq -L_i(\hat{x}_i, u_{i,i}^{\mathrm{MPC}}) \leq -\beta(\|\hat{x}_i - x_i^{\mathrm{p}}\|).$$

We prove now the upper bound $\bar{V}^i(\hat{x}_i) \leq \alpha(\|\hat{x}_i - x_i^{\mathrm{p}}\|)$. Consider an initial state $\hat{x}_i$ and a control sequence $U = (u_i, \dots, u_{N+i-1})$ which yield a feasible trajectory $x_i, \dots, x_{N+i}$. The rotated cost of the obtained trajectory is given by

$$\begin{aligned}\bar{J}^i(\hat{x}_i, U) &= (\hat{x}_i - x_i^{\mathrm{p}})^\top \lambda_i^{\mathrm{p}} + \sum_{k=i}^{N+i-1} \Big(\ell_k(x_k, u_k) - \ell_k(x_{k+i}^{\mathrm{p}}, u_{k+i}^{\mathrm{p}}) \Big) \\ &\leq \|\hat{x}_i - x_i^{\mathrm{p}}\| \|\lambda_i^{\mathrm{p}}\| + \sum_{k=i}^{N+i-1} L_\ell \Big(\|x_k - x_{k+i}^{\mathrm{p}}\| + \|u_k - u_{k+i}^{\mathrm{p}}\| \Big),\end{aligned}$$

where we used Lipschitz continuity of the stage cost ℓ_k and the triangle inequality. Using the system dynamics and Lipschitz continuity of f_k we obtain

$$\|x_k - x_{k+i}^{\mathrm{p}}\| \leq L_f^k \|\hat{x}_i - x_i^{\mathrm{p}}\| + \sum_{j=i}^{k-1} L_f^{k-i-j} \|u_j - u_{j+i}^{\mathrm{p}}\|,$$

so that

$$\sum_{k=i}^{N+i-1} L_\ell \|x_k - x_k^{\mathrm{p}}\| \leq L_{\mathrm{F}} \|\hat{x}_i - x_i^{\mathrm{p}}\| + L_{\mathrm{F}} \sum_{k=i}^{N+i-1} \|u_k - u_k^{\mathrm{p}}\|, \tag{4.40}$$

with $L_{\mathrm{F}} = \sum_{j=0}^{N-1} L_f^j$. By using Equation (4.40) and the weak controllability Bound (4.35) from Assumption 4.43 we obtain

$$\begin{aligned}\sum_{k=i}^{N+i-1} L_\ell \Big(\|x_k - x_k^{\mathrm{p}}\| + \|u_k - u_k^{\mathrm{p}}\| \Big) \leq & L_\ell L_{\mathrm{F}} \|x_k - x_k^{\mathrm{p}}\| \\ & + L_\ell (1 + L_{\mathrm{F}}) \gamma(\|x_k - x_k^{\mathrm{p}}\|).\end{aligned}$$

By defining $\alpha(\cdot) := (\|\lambda_i^{\mathrm{p}}\| + L_\ell L_{\mathrm{F}})(\cdot) + L_\ell(1 + L_{\mathrm{F}})\gamma(\cdot)$ we obtain

$$\bar{V}^i(\hat{x}_i) \leq \bar{J}^i(\hat{x}_i, U) \leq \alpha(\|\hat{x}_i - x_i^{\mathrm{p}}\|),$$

where the first inequality follows from optimality of the rotated value function. We have therefore proven that the rotated value function is a periodic Lyapunov function, i.e.

$$\beta(\|\hat{x}_i - x_i^{\mathrm{p}}\|) \leq \bar{V}^i(\hat{x}_i) \leq \alpha(\|\hat{x}_i - x_i^{\mathrm{p}}\|),$$

$$\bar{V}^{i+1}(x_{1,i}^{\mathrm{MPC}}) - \bar{V}^i(\hat{x}_i) \leq -L_i(\hat{x}_i, u_{0,i}^{\mathrm{MPC}}) \leq -\beta(\|\hat{x}_i - x_i^{\mathrm{p}}\|).$$

□

Numerical Examples

In the following, we illustrate the obtained stability result using two numerical examples. We also show that the stability result proposed in [62] does not hold for the first example. For the second example, the framework proposed in this section gives the same result as the one proposed in [62].

Example 4.46 (Unicycle)**.** *A simple car model is considered. The system dynamics are given by*

$$\begin{bmatrix} \dot{z} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ u \end{bmatrix},$$

where the position is given by (z, y)*, the angle* θ *indicates the orientation of the vehicle, the control* u *is a rotational velocity and the vehicle has constant linear velocity* $v = 1$ m/s*.*

All optimization problems have been solved using the direct multiple shooting [22] method. The system dynamics have been translated into discrete time using 50 *steps of a Runge-Kutta integrator of order 4, with a sampling time* $T_{\mathrm{s}} = 0.1$ s *and a piecewise constant control parametrization. The period has been chosen as* $p = 50$*. The problems have been formulated using the open source software CasADi [6] and the NLP solver Ipopt [113].*

We use the time-invariant stage cost $\ell(x_j, u_j) = u_j^2 + z_j^2 + 5y_j^2$*. Since for the obtained periodic optimization problem strong duality does not hold, we consider the Linear Time Varying (LTV) model for the MPC formulation. The LTV model is obtained by linearising the integrator on the optimal periodic trajectory.*

Starting from an initial state $(z_{\mathrm{init}}, y_{\mathrm{init}}, \theta_{\mathrm{init}})$*, at each time instant* i*, the Economic MPC Problem* (4.34) *is solved using a horizon length* $N = 20$*. The*

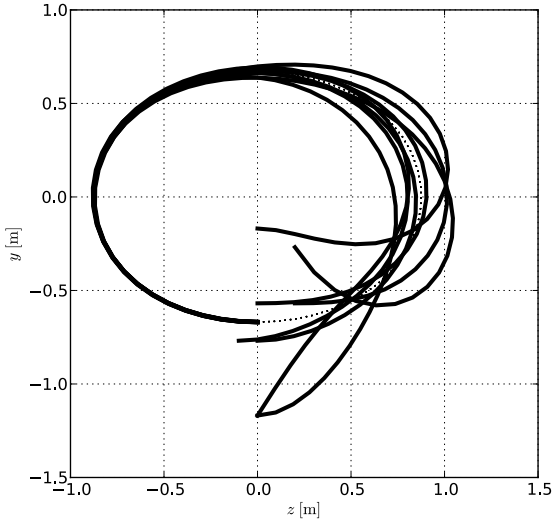


Figure 4.6: The trajectories obtained by applying the economic MPC scheme to the system starting from different initial conditions are displayed in thick line. The system is steered to the periodic optimal trajectory (thin line).

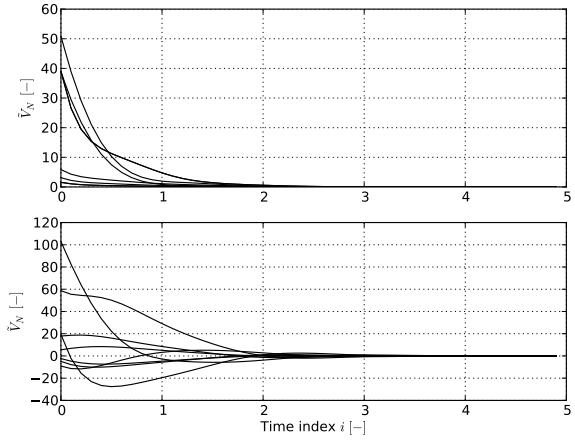


Figure 4.7: Rotated cost $\bar{V}^i$ over time. In the top graph, the rotated cost obtained with the proposed framework monotonically decreases and is a Lyapunov function. In the bottom graph, the cost rotated as proposed in [62] is not a Lyapunov function for the given example.

obtained closed-loop trajectory of the LTV system is shown in Figure 4.6, where it can be noted that the system is stabilized to the optimal periodic trajectory. Figure 4.7 displays the value functions $\bar{V}^i(\hat{x}_i)$ at each time instant in the top graph. It satisfies the requirements of a Lyapunov function as it is bounded from below by a K_∞-function and decreases monotonically. As a term of comparison, the approach proposed in [62] has also been implemented and the obtained value function is displayed in Figure 4.7, bottom graph. It can be seen that this function does not monotonically decrease and even becomes negative.

Example 4.47 (Double Tank System [62])**.** *We consider now an interconnected double tank system. The control u is the liquid inlet flow into the first tank. The sates are the liquid heights in the two tanks, x_1, x_2. The system dynamics are given by*

$$\dot{x}_1 = 0.16\,u - 0.4\sqrt{x_1}, \qquad \dot{x}_2 = 0.4\sqrt{x_1} - 0.4\sqrt{x_2}.$$

The system is discretised using a sampling time $T_\mathrm{s} = 1$ s and 100 steps of an explicit Runge-Kutta integrator of order 4.

The time-varying stage cost is given by

$$\ell_k(x_k, u_k) = u_k p_k + \|x_k - \bar{x}_k\|_Q^2,$$

where p_k is a time-varying price displayed in Figure 4.8, $Q = \mathrm{diag}(\begin{bmatrix} 10^{-3} & 10^{-3} \end{bmatrix})$ and

$$\begin{aligned}\bar{x}_{0,\dots,10} = [&(1.118, 0.4028), (0.7351, 0.5126), (0.4321, 0.5254), (0.2092, 0.4654),\\ &(0.0662, 0.3517), (0.0033, 0.2037), (0.1761, 0.1599), (0.1513, 0.1606),\\ &(0.1701, 0.1612), (0.7056, 0.2495)],\end{aligned}$$

which is the periodic optimal trajectory obtained using $\ell_k(x_k, u_k) = u_k p_k$ and a period $p = 10$.

The numerical results are displayed in Figure 4.8. It can be seen that the value function of the rotated problem is positive and monotonically decreasing. For this example, the rotated cost (4.36) *coincides with the one proposed in [62].*

4.5 Conclusions

In this chapter, we have presented the state of the art on stability theory for MPC. We have illustrated the difference between tracking and economic MPC. In the first case the stage cost has its unconstrained minimum at a feasible

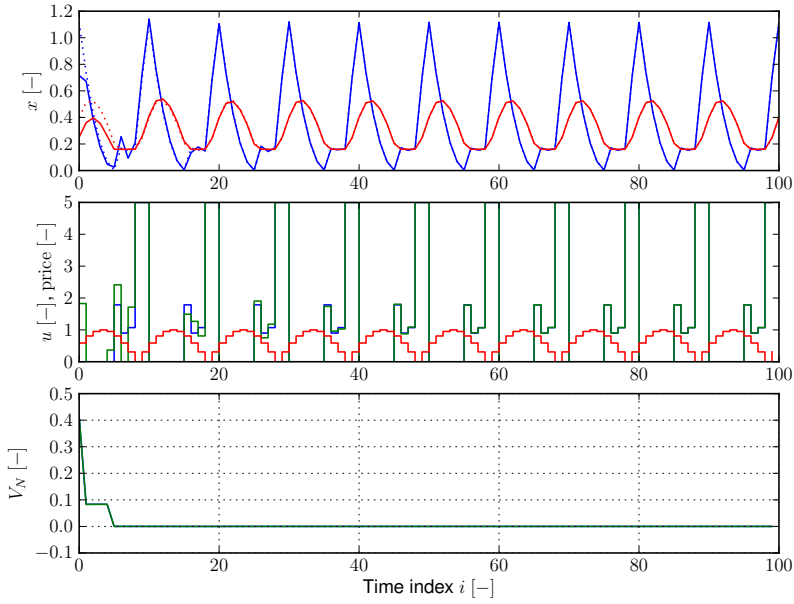


Figure 4.8: In the top graph: plant state in thick line, optimal periodic state in dotted line. Middle graph: applied controls (green), optimal periodic controls (blue) and price (red). Bottom graph: rotated cost $\tilde{V}_N^{[0]}$ over time. In this example the rotated cost $\tilde{V}_N^{[0]}$ (blue line) coincides with the cost rotated as proposed in [62] (green line).

steady-state and it is rather easy to prove stability of the closed-loop system. In the second case, the stage cost does not have its unconstrained minimum at the optimal steady state. The stability of the closed-loop system can be established if the strict dissipativity condition is satisfied.

Steady-state operation is not always optimal and, in some cases, periodic operation can outperform it. In order to analyse this case, we have extended the stability theory to cover the periodic case. We have developed two different extensions: the first one for time-invariant formulations and the second one for time-varying formulations. For each case, we have illustrated the theoretical developments with numerical examples.

Chapter 5

Economic-Based Tuning of Tracking MPC

In this chapter, we propose a strategy to tune tracking NMPC with quadratic stage cost so that it locally yields the behaviour of economic NMPC. The main motivation for investigating tuned tracking MPC schemes is twofold:

1. As discussed in Chapter 4, guaranteeing stability for tracking MPC hinges on designing a proper terminal cost and constraint. For economic MPC instead, it is also necessary that the system and stage cost satisfy the strict dissipativity inequality, which is in general hard to prove.

2. Efficient real-time algorithms exist for least squares objective functions, i.e. quadratic tracking stage costs. Such algorithms are based on the generalised Gauss-Newton method, which does not require the computation of second order derivatives and guarantee to deliver a positive (semi)definite Hessian approximation [20, 31]. For the more generic cost functions typically encountered in economic MPC, real-time algorithms are still under development and it is currently impossible to achieve computational times which are comparable to those observed in tracking MPC.

In this chapter, we focus on proving under which conditions such a construction is feasible and leave to future research a thorough comparison on the computational times and stability properties of tuned tracking MPC.

The problem is described in Section 5.1, and some useful results are established or recalled. The case of optimal operation at steady-state is analyse first for linear MPC in Section 5.2. Starting from the case of no active constraints at the optimum, the result is extended to the case of active constraints at the optimum in Section 5.3. Subsequently, in Section 5.4, the obtained results are further extended to the nonlinear case. A numerical example is given in Section 5.4.1. Finally, in Section 5.5, we also analyse the case of optimal periodic operation, which is a direct extension of the results obtained for the steady-state case.

Acknowledgement This chapter is mostly based on new work, due to be submitted for publication. Section 5.2 is largely based on the article [122], of which Mario Zanon is the first author. For all results in this chapter Mario Zanon is the main contributor and developed the theoretical results under the supervision of Moritz Diehl and Sébastien Gros.

5.1 Problem Setup and Preliminaries

This chapter is concerned with time-invariant nonlinear discrete-time systems $x_{k+1} = f(x_k, u_k)$ that shall be operated such that constraints $h(x_k, u_k) \geq 0$ are satisfied and the cost $\sum_{k=0}^{\infty} \ell(x_k, u_k)$ is minimised. Because the infinite horizon problem is computationally intractable, MPC approximates the infinite horizon problem by optimizing over a finite horizon N. We lump all states and controls in a vector $w = (w_0, w_1, \ldots, w_{N-1}, x_N)$, with $w_k = (x_k, u_k)$. In the following, we will refer to the optimal solution by adding to the variable as a superscript the equation number of the problem it refers to, i.e. $w^{(5.1)}$ is the optimal primal solution of Problem (5.1). We will analyse in the following two approaches towards this aim.

The first approach is *economic MPC*, where at each time step, given the current state $\hat{x}_0$, one solves the following optimal control problem

$$\min_{w} \quad \sum_{k=0}^{N-1} \ell(x_k, u_k) + V_\mathrm{f}(x_N) \tag{5.1a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{5.1b}$$

$$x_{k+1} - f(x_k, u_k) = 0, \qquad k = 0, \ldots, N-1, \tag{5.1c}$$

$$h(x_k, u_k) \geq 0, \qquad k = 0, \ldots, N-1, \tag{5.1d}$$

$$x_N \in \mathbb{X}_\mathrm{f}, \tag{5.1e}$$

where we define the optimal primal solution as $w^{(5.1)}(\hat{x}_0) = (x_0^{(5.1)}, u_0^{(5.1)}, \ldots, x_N^{(5.1)})$ and the economic MPC feedback as $u^{\mathrm{e}}(\hat{x}_0) = u_0^{(5.1)}$.

We remark that the terminal cost and constraint in the economic MPC Problem (5.1) ought to be chosen together with the prediction horizon in order to guarantee stability [99, 54]. For a given stage cost, this choice can be made so as to provide a good approximation of the infinite-horizon problem. For $\mathbb{X}_{\mathrm{f}} = \{x_{\mathrm{s}}\}$ we also define the Lagrangian of the economic MPC problem as

$$\mathcal{L}(w, \lambda, \mu) = \sum_{k=0}^{N-1} \ell(x_k, u_k) - \lambda_{k+1}^{\top}(x_{k+1} - f(x_k, u_k)) - \mu_k^{\top} h(x_k, u_k) \\ - \lambda_0^{\top}(x_0 - \hat{x}_0) - \lambda_{N+1}^{\top}(x_N - x_{\mathrm{s}}).$$

The optimal Lagrange multipliers are then $\lambda_k^{(5.1)}$ and $\mu_k^{(5.1)}$.

For notational simplicity and without loss of generality we assume that $\ell(0,0) = 0$ and that $x_{\mathrm{s}} = 0$, $u_{\mathrm{s}} = 0$ is an optimal steady state, i.e. it solves the steady state problem

$$\min_{x,u} \quad \ell(x,u) \qquad \text{s.t.} \quad x - f(x,u) = 0, \qquad h(x,u) \geq 0. \tag{5.2}$$

We define $w_{\mathrm{s}} = (x_{\mathrm{s}}, u_{\mathrm{s}})$ and denote by λ_{s} and μ_{s} the optimal Lagrange multipliers associated with the equality and inequality constraints of Problem (5.2) respectively.

An alternative approach to economic MPC is *tracking MPC*, where, at each time step, given the current state $\hat{x}_0$, one solves the following optimal control problem

$$\min_{w} \quad \sum_{k=0}^{N-1} \ell^{\mathrm{t}}(x_k, u_k) + V_{\mathrm{f}}^{\mathrm{t}}(x_N) \tag{5.3a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{5.3b}$$

$$x_{k+1} - f(x_k, u_k) = 0, \qquad k = 0, \ldots, N-1, \tag{5.3c}$$

$$h(x_k, u_k) \geq 0, \qquad k = 0, \ldots, N-1, \tag{5.3d}$$

$$x_N \in \mathbb{X}_{\mathrm{f}}, \tag{5.3e}$$

where the stage and terminal cost satisfy $\ell^{\mathrm{t}}(x,u) \geq \alpha(\|x\|)$ for all feasible u and $V_{\mathrm{f}}^{\mathrm{t}}(x) \geq \alpha(\|x\|)$, where α is a class $\mathcal{K}$ function [99]. Typically, the tracking stage and terminal cost are chosen as quadratic functions. In the

remainder of the paper, whenever we refer to tracking MPC, we will implicitly assume such a choice. We define the optimal primal solution as $w^{(5.3)}(\hat{x}_0) = (x_0^{(5.3)}, u_0^{(5.3)}, \ldots, x_N^{(5.3)})$ and the tracking MPC feedback as $u^{\mathrm{t}}(\hat{x}_0) = u_0^{(5.3)}$. We define the optimal Lagrange multipliers analogously to Problem (5.1).

The main difference between economic and tracking MPC is that the former typically outperforms the latter during transients. Proving stability for economic MPC, however, is much more involved than for tracking MPC. Moreover, efficient numerical schemes for fast real-time NMPC are based on the generalised Gauss-Newton method can only be applied to tracking MPC. For these reasons, in this chapter, we aim at computing a positive definite stage cost for tracking NMPC such that its feedback control law $u^{\mathrm{t}}(\hat{x}_0)$ is equal to the one delivered by the economic MPC scheme up to first order $u^{\mathrm{e}}(\hat{x}_0)$, i.e. $\|u^{\mathrm{t}}(\hat{x}_0) - u^{\mathrm{e}}(\hat{x}_0)\| = O(\|\hat{x}_0\|^2)$. When aiming at designing a tracking MPC scheme which provides a local approximation of the behaviour of economic MPC, the choice of the appropriate stage cost only depends on the economic stage cost and not on the terminal cost or constraint.

Before starting our analysis, we give some useful definitions and state some important properties of the tracking and economic MPC problems.

For tracking MPC, Lyapunov stability is guaranteed by the condition $\ell^{\mathrm{t}}(x, u) \geq \alpha(\|x\|)$ for all feasible u, in combination with continuity of the cost and system dynamics, an appropriate choice of the terminal cost, and the satisfaction of a technical controllability assumption. For economic MPC, the condition $\ell(x, u) \geq \alpha(\|x\|)$ is in general violated and the current stability theory relies on a strict dissipativity assumption to replace it (see Section 4.3, Definition 4.19 and Assumption 4.21). In [32] and [3], it has been proven that the MPC problem (5.1) formulated using the rotated cost delivers the same primal solution as the original one. Clearly, if strict dissipativity holds, $L(x_k, u_k) \geq \alpha(\|x_{\mathrm{s}}\|)$ holds on a compact set so that the rotated economic MPC problem satisfies all the assumptions used to prove Lyapunov stability of tracking MPC. Because there exists no systematic method to find or dismiss the existence of a storage function $\lambda(x)$ such that the system satisfies strict dissipativity in the general case, it is hard to guarantee Lyapunov stability for economic MPC schemes. Note that one equivalent definition of strict dissipativity would be

Definition 5.1 (Strict dissipativity)**.** *The system $x_{k=1} = f(x_k, u_k)$ is strictly dissipative on a set $\mathbb{W} = \mathbb{W}_{\mathbb{X}} \times \mathbb{W}_{\mathbb{U}}$ with respect to the supply rate $\ell : \mathbb{W} \to \mathbb{R}$ if there exists a function $\lambda : \mathbb{W}_{\mathbb{X}} \to \mathbb{R}$, which is bounded from below on $\mathbb{W}_{\mathbb{X}}$ and such that the rotated stage cost $L(x_k, u_k)$ given by Equation* (4.17) *is a positive definite tracking cost, i.e. $L(x_k, u_k) \geq \rho(\|x_k\|)$.*

The use of a nonlinear storage function $\lambda(\cdot)$ is a generalisation of the rotation first

introduced in [32], which was based on using the optimal Lagrange multipliers of the steady-state problem to construct the linear storage function $\lambda(x) = \lambda_s^\top x$. In order to distinguish it from the general case, we call *linear rotation* the rotation which uses a linear storage function. We establish next an important property of the rotated cost.

Lemma 5.2. *Suppose that the steady-state Problem* (5.2) *has no active constraints. Then the rotated cost* L *using* $\lambda(x) = \lambda_s^\top x$ *has zero gradient at steady state. If there are strictly active constraints at steady state, then the following holds:*

$$\left.\frac{\partial L(x_k, u_k)}{\partial w_k}\right|_{w=0} = \mu_s^\top \left.\frac{\partial h(x_k, u_k)}{\partial w_k}\right|_{w=0}.$$

Proof. In the case of no active constraints at steady-state, the rotated cost coincides with the Lagrangian of the optimisation Problem (5.2) which, by definition of optimality, has zero gradient at the optimum. If instead some constraints are active at steady state, the desired equality is directly obtained by writing the KKT conditions of Problem (5.2). □

In this chapter, we generalise the concept of linear rotation to generic NLP formulations and investigate the rotation of inequality constraints. In particular, one might be interested in formulating a rotated NLP problem which is equivalent to the original one but makes use of a rotated stage cost such that the rotated steady-state problem

$$\min_{x,u} \quad L(x,u) \qquad \text{s.t.} \quad x - f(x,u) = 0, \qquad h(x,u) \geq 0.$$

delivers a zero Lagrange multiplier associated with the inequality constraints. We prove in the following lemma that this cannot be done unless the original stage cost already delivers $\mu_s = 0$. In the case of economic MPC, this entails that there are no strictly active constraints at steady state. We will prove in the following sections that this is an important fact which needs to be taken into account when tuning a tracking stage cost in order to locally approximate the economic MPC behaviour.

Lemma 5.3. *Consider vectors* $\bar{\lambda}$ *and* $\bar{\mu}$*, a vector of fixed parameters* θ*, and the two parametric NLPs*

$$\min_w \quad \bar{f}(w,\theta) \qquad \text{s.t.} \quad \bar{g}(w,\theta) = 0, \quad \bar{h}(w,\theta) \geq 0. \tag{5.4}$$

$$\min_w \quad \bar{f}(w,\theta) - \bar{\lambda}^\top \bar{g}(w,\theta) - \bar{\mu}^\top \bar{h}(w,\theta) \quad \text{s.t.} \quad \bar{g}(w,\theta) = 0, \quad \bar{h}(w,\theta) \geq 0. \tag{5.5}$$

We will call NLP (5.4) *the original NLP and NLP* (5.5) *the rotated NLP. For NLP* (5.4)*, let us introduce the optimal Lagrange multipliers* $\lambda^{(5.4)}(\theta)$, $\mu^{(5.4)}(\theta)$

associated with the equality and inequality constraints, respectively. Equivalently, for NLP (5.5)*, we introduce* $\lambda^{(5.5)}(\theta)$ *and* $\mu^{(5.5)}(\theta)$*. Then, the original and rotated NLPs* (5.4) *and* (5.5) *deliver the same primal solution for all* θ *iff* $\bar{\mu} = 0$*. Moreover, it holds that* $\lambda^{(5.5)}(\theta) = \lambda^{(5.4)}(\theta) - \bar{\lambda}$ *and* $\mu^{(5.5)}(\theta) = \mu^{(5.4)}(\theta)$*.*

Proof. Let us denote the optimal solution of Problem (5.4) by $w^*(\theta) = w^{(5.4)}(\theta)$. The KKT conditions of NLP (5.4) read

$$\nabla_w \bar{f}(w^*(\theta),\theta) - \nabla_w \bar{g}(w^*(\theta),\theta)\lambda^{(5.4)}(\theta) - \nabla_w \bar{h}(w^*(\theta),\theta)\mu^{(5.4)}(\theta) = 0, \quad (5.6a)$$

$$\bar{g}(w^*(\theta),\theta) = 0, \quad (5.6b)$$

$$\bar{h}(w^*(\theta),\theta) \geq 0, \quad (5.6c)$$

$$\mu^{(5.4)}(\theta) \geq 0, \quad (5.6d)$$

$$\mu_i^{(5.4)}(\theta)\bar{h}_i(w^*(\theta),\theta) = 0, \quad i = 1,\ldots,n_{\text{ieq}}. \quad (5.6e)$$

If the original and rotated NLPs deliver the same primal solution, then $w^*(\theta)$ must also be a KKT point for the rotated problem. The KKT conditions of NLP (5.5) then read

$$\nabla_w \bar{f}(w^*(\theta),\theta) - \nabla_w \bar{g}(w^*(\theta),\theta)(\lambda^{(5.5)}(\theta) + \bar{\lambda}) - \nabla_w \bar{h}(w^*(\theta),\theta)(\mu^{(5.5)}(\theta) + \bar{\mu}) = 0, \quad (5.7a)$$

$$\bar{g}(w^*(\theta),\theta) = 0, \quad (5.7b)$$

$$\bar{h}(w^*(\theta),\theta) \geq 0, \quad (5.7c)$$

$$\bar{\mu}^{(5.5)}(\theta) \geq 0, \quad (5.7d)$$

$$\bar{\mu}_i^{(5.5)}(\theta)\bar{h}_i(w^*(\theta),\theta) = 0, \quad i = 1,\ldots,n_{\text{ieq}}. \quad (5.7e)$$

Conditions (5.6a) and (5.7a) are equivalent for $\lambda^{(5.5)}(\theta) = \lambda^{(5.4)}(\theta) - \bar{\lambda}$ and for $\mu^{(5.5)}(\theta) = \mu^{(5.4)}(\theta) - \bar{\mu}$. However, we will now show that one can in general only satisfy Conditions (5.7d) and (5.7e) for all θ iff $\bar{\mu} = 0$. Let us consider the original problem and a variation $\delta\theta$ which yields $0 = \mu_i^{(5.4)}(\theta + \delta\theta) < \mu_i^{(5.4)}(\theta)$. In this case the only feasible choice which still satisfies (5.7d) is $\bar{\mu}_i \leq 0$. However, in case $\bar{h}_i(w^*(\theta + \delta\theta), \theta + \delta\theta) > 0$, then the only admissible choice is $\bar{\mu}_i = 0$, otherwise Condition (5.7e) cannot be satisfied.

Therefore, Problems (5.4) and (5.5) are always delivering the same primal solution if and only if $\bar{\mu} = 0$, i.e. $\mu^{(5.5)}(\theta) = \mu^{(5.4)}(\theta)$. □

We remark that, in the context of MPC, the NLP parameter is the initial condition, i.e. $\theta = \hat{x}_0$ and it is well known that, depending on the initial state, active constraints can become inactive and vice versa.

Finally, we establish an important property relating the economic MPC Problem (5.1) to the steady-state Problem (5.2).

Lemma 5.4. *Consider a stabilising economic MPC formulation of the form* (5.1) *such that (a) the horizon is infinite, or (b) the gradient of the terminal cost $V_\mathrm{f}(\cdot)$ satisfies $\nabla V_\mathrm{f}|_{x=x_\mathrm{s}} = \lambda_\mathrm{s}$ and $\nabla^2 V_\mathrm{f} \geq \nabla^2 V$ with $V(x)$ the cost-to-go of Problem* (5.1) *with an infinite horizon, or (c) the terminal constraint is $x_N = x_\mathrm{s}$. Then, the Lagrange multipliers $\lambda_k,\ \mu_k$ of the MPC problem solved for the initial state $\hat{x}_0 = x_\mathrm{s}$ coincide with those of the steady state problem $\lambda_\mathrm{s},\ \mu_\mathrm{s}$.*

Proof. The proof is directly obtained by writing the KKT conditions of the MPC problem and the steady state problem. If the economic MPC problem is stabilising, then the primal solution is $x_k = x_\mathrm{s}$ and $u_k = u_\mathrm{s}$. Then, by replacing $\lambda_k = \lambda_\mathrm{s}$ and $\mu_k = \mu_\mathrm{s}$, one obtains that the KKT conditions are satisfied. □

Note that we implicitly assume that the economic MPC problem and the steady-state problem are formulated using the same stage cost, so that, if the rotated stage cost is used, $\lambda_\mathrm{s} = 0$, while otherwise, in the general case, $\lambda_\mathrm{s} \neq 0$ holds.

We conclude this section by introducing a result from parametric optimisation which will be used both in Section 5.3 and in Section 5.4.

Theorem 5.5. *(Continuous differentiability [55, 31]) Let us consider a parametric optimisation problem which depends on parameter t*

$$\min_{w} \quad \bar{f}(w,t) \qquad \text{s.t.} \quad \bar{g}(w,t) = 0, \qquad \bar{h}(w,t) \geq 0. \tag{5.8}$$

We define the Lagrangian as $\bar{\mathcal{L}}(w,t,\lambda,\mu) = \bar{f}(w,t) - \lambda^\top \bar{g}(w,t) - \mu^\top \bar{h}(w,t)$, the solution points depending on t as $(w^*(t), \lambda^*(t), \mu^*(t))$, and the set of indices of the active constraints for $t = 0$ as $\mathbb{A}$.

Let us assume that the KKT point $(w^*(0), \lambda^*(0), \mu^*(0))$ satisfies LICQ and strong second order sufficient conditions and strict complementarity. Let us moreover assume that the solution $(\delta w^*, \delta\lambda^*, \delta\mu^*_\mathbb{A})$ of the following quadratic

program (QP)

$$\min_{w} \quad \frac{1}{2}\delta w^\top \nabla_w^2 \bar{\mathcal{L}} \delta w + \left(\frac{\partial}{\partial t}\nabla_w \bar{\mathcal{L}}\right)^\top \delta w \tag{5.9a}$$

$$\text{s.t.} \quad \nabla_t \bar{g} + \nabla_w \bar{g}^\top \delta w = 0, \tag{5.9b}$$

$$\nabla_t \bar{h}_{\mathbb{A}} + \nabla_w \bar{h}_{\mathbb{A}}^\top \delta w = 0, \tag{5.9c}$$

where all derivatives are evaluated at $(w^*(0), \lambda^*(0), \mu^*(0))$*, satisfies the strict complementarity condition. Then*

(i) there exists an ϵ *and a differentiable curve* $v(t) = (w^*(t), \lambda^*(t), \mu^*(t))$ *of KKT points that satisfy the optimality conditions for Problem* (5.8)*, for* $t \in (-\epsilon, \epsilon)$*;*

(ii) at $t = 0$ *the one sided derivative of this curve is given by*

$$\frac{\partial v}{\partial t} = \lim_{t \to 0^+} \frac{1}{t} \begin{bmatrix} w^*(t) - w^*(0) \\ \lambda^*(t) - \lambda^*(0) \\ \mu^*(t) - \mu^*(0) \end{bmatrix} = \begin{bmatrix} \delta w^* \\ \delta \lambda^* \\ \begin{bmatrix} \delta \mu^*_{\mathbb{A}} \\ 0 \end{bmatrix} \end{bmatrix},$$

where we ordered μ^* *such that the multipliers corresponding to the active constraints are followed by those corresponding to the inactive ones.*

Proof. The proof is given in [31] and [55, Theorem 3.3.4 and Corollary 3.3.1]. □

This theorem is particularly important because (a) it ensures that there is an interval $(-\epsilon, \epsilon)$ inside which the set of strongly active constraints remains unchanged and (b) it states that $w^*(t) = w^*(0) + t\delta w^* + O(t^2)$, for $t \in (-\epsilon, \epsilon)$.

The original version of the theorem considered milder assumptions that allowed the existence of weakly active constraints for $t = 0$. In that case one can only prove the existence of the one-sided derivative of the solution curve, as $t = 0$ can be a point of non-differentiability of the solution curve. For simplicity, in this chapter we only consider economic MPC formulations which do not have weakly active constraints at steady state. An extension to the case of weakly active constraints at steady state seems possible but requires a more detailed technical discussion which is left for future research.

An implication of Theorem 5.5 is given in the following lemma.

Lemma 5.6. *Consider any NLP of the form* (5.8) *which satisfies the assumption of Theorem 5.5 and the following QP*

$$\min_{w} \quad \frac{1}{2}\delta w^\top \nabla_w^2 \bar{\mathcal{L}} \delta w + \left(\frac{\partial}{\partial t}\nabla_w \bar{\mathcal{L}}\right)^\top \delta w \tag{5.10a}$$

$$\text{s.t.} \quad \bar{g}(w^{(5.8)}, t) + \nabla_w \bar{g}^\top \delta w = 0, \tag{5.10b}$$

$$\bar{h}(w^{(5.8)}, t) + \nabla_w \bar{h}^\top \delta w \geq 0, \tag{5.10c}$$

where all derivatives are evaluated at the optimal point $w^*(t), t, \lambda^*(t), \mu^*(t)$. *Then the following holds at* $t = 0$*:*

$$\frac{\partial v^{(5.10)}}{\partial t} = \frac{\partial v^{(5.8)}}{\partial t},$$

i.e. the two problems locally deliver the same solution up to first-order.

Proof. The proof is obtained by noting that both Problem (5.8) and (5.10) deliver the same QP (5.9). □

In the following we will first analyse the case of linear economic MPC, by recalling the main result of [122], and further analysing its implications. We will then extend the result to the case of strictly active constraints at steady state. We will afterwards propose a tuning strategy for tracking nonlinear MPC. In order to clearly distinguish between linear and nonlinear MPC, in the remainder of the chapter we will use LMPC and NMPC respectively.

5.2 Linear MPC without Active Constraints at Steady State

The case of economic LMPC has been first analysed in [122] for the case of no active constraints at steady state. In the following, we first summarise the main result of [122], which states that every stabilising LMPC scheme can be formulated as a positive definite one. Afterwards, we extend that result, by proposing a new method for computing the positive definite stage cost which also makes it possible to prove the necessity of strict dissipativity for LMPC in the case of no active constraints at steady state.

5.2.1 The LQR case

We consider first the case of discrete time infinite horizon OCPs for linear systems and quadratic cost. The linear quadratic regulator (LQR) is defined by the following OCP

$$V(\bar{x}_0) = \inf_{x,u} \sum_{k=0}^{\infty} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top H \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.11a}$$

$$\text{s.t. } x_0 = \hat{x}_0, \tag{5.11b}$$

$$x_{k+1} = Ax_k + Bu_k, \quad k \geq 0, \tag{5.11c}$$

$$\lim_{k\to\infty} x_k = 0, \tag{5.11d}$$

with state $x = [x_0, x_1, \ldots]$ and control $u = [u_0, u_1, \ldots]$ and matrix $H = \begin{bmatrix} Q & S^\top \\ S & R \end{bmatrix}$ symmetric and possibly indefinite. This optimisation problem has been widely studied in the literature and excellent analyses can be found in [78, 96, 122] for discrete time systems and in [115] for continuous time systems. In particular, it is well known that the LQR can be stabilising even if the stage cost matrix H is indefinite.

The solution to the LQR problem (5.11) can be obtained by computing the stabilizing solution of the discrete algebraic Riccati equation (DARE), defined as

$$\begin{aligned} \mathcal{D}(A,B,Q,R,S) := \{(P,K) |\ & Q + A^\top PA - P - (S^\top + A^\top PB)K = 0, \\ & K = (R + B^\top PB)^{-1}(S + B^\top PA), \\ & \rho(A - BK) < 1\}, \end{aligned} \tag{5.12}$$

where ρ denotes the spectral radius, i.e. $\rho(A) = \max\{|\lambda| \mid \lambda \text{ is eigenvalue of } A\}$. In general, the DARE has many solutions but only one, if it exists, is stabilizing the system (A, B), yields the solution to the LQR problem. Then, the LQR cost-to-go is given by $V(\hat{x}_0) = \hat{x}_0^\top P \hat{x}_0$ and the LQR feedback law is given by $u_{\text{LQR}}(\hat{x}_0) = -K\hat{x}_0$. Note that, in the case of an indefinite cost, i.e. $H \not\succeq 0$, also the cost-to-go matrix P can be indefinite.

We prove in the following that any stabilizing LQR can be reformulated as a positive definite one, i.e. such that $P \succ 0$ and $H \succ 0$, which yields the same feedback law. We introduce the following notation. The DARE $\mathcal{D}(A, B, Q, R, S)$ delivers a solution (P, K), where $P = P^\top$ is a real symmetric matrix and K is a

feedback gain. We define the LQR feedback law as $u_{\text{LQR}}(\hat{x}_0) := \mathcal{P}(A, B, Q, R, S)$. Moreover, we define two LQR problems equivalent if they deliver the same feedback law, i.e. $\mathcal{P}(A_1, B_1, Q_1, R_1, S_1) = \mathcal{P}(A_2, B_2, Q_2, R_2, S_2)$.

Definition 5.7. *Problem $\mathcal{P}(A_2, B_2, Q_2, R_2, S_2)$ is a positive definite reformulation of the possibly indefinite LQR problem $\mathcal{P}(A_1, B_1, Q_1, R_1, S_1)$ if $\begin{bmatrix} Q_2 & S_2^\top \\ S_2 & R_2 \end{bmatrix} \succ 0$ and $\mathcal{P}(A_2, B_2, Q_2, R_2, S_2) = \mathcal{P}(A_1, B_1, Q_1, R_1, S_1)$.*

Theorem 5.8. *Any stabilizing LQR problem can be reformulated as a positive definite LQR problem.*

Before proving the theorem, two lemmas are introduced that will help in establishing the desired result. The following lemma considers the case of a system (A, B) which is pre-stabilised using a linear feedback $\bar{K}$. It proves that, by suitably modifying the cost, an LQR can be defined for the pre-stabilised system $(A_{\bar{K}}, B)$, which is equivalent to the original LQR defined for (A, B).

Lemma 5.9. *Let us assume that a stabilizing LQR exists for system (A, B), with weighting matrices Q, R, S. The equivalence $\mathcal{P}(A, B, Q, R, S) = \mathcal{P}(A_{\bar{K}}, B, Q_{\bar{K}}, R, S_{\bar{K}})$ holds for any arbitrary feedback matrix $\bar{K}$, when choosing $A_{\bar{K}} = A - B\bar{K}$, $Q_{\bar{K}} = Q - S^\top \bar{K} - \bar{K}^\top S + \bar{K}^\top R\bar{K}$, $S_{\bar{K}} = S - R\bar{K}$. Moreover, the real symmetric matrix $P_{\bar{K}}$, computed as $(P_{\bar{K}}, \bar{K}) = \mathcal{D}(A_{\bar{K}}, B, Q_{\bar{K}}, R, S_{\bar{K}})$, satisfies $P_{\bar{K}} = P$ with $(P, K) = \mathcal{D}(A, B, Q, R, S)$. The feedback gain $K_{\bar{K}}$ is given by $K_{\bar{K}} = K - \bar{K}$.*

Proof. The proof follows by substitution of $A_{\bar{K}}$, $Q_{\bar{K}}$ and $S_{\bar{K}}$ into (5.12) and simplification [78]. Before providing the details, we remark that the proposed reformulation is equivalent to choosing for the control input $u_k = -\bar{K}x_k + v_k$, i.e.

$$x_{k+1} = (A - B\bar{K})x_k + Bv_k = A_{\bar{K}}x_k + Bv_k. \tag{5.13}$$

From $u_k = -\bar{K}x_k + v_k$ and $v_k = -K_{\bar{K}}x_k$, one obtains

$$-Kx_k = u_k = -\bar{K}x_k + v_k = -\bar{K}x_k - K_{\bar{K}}x_k,$$

which also entails $K_{\bar{K}} = K - \bar{K}$.

By replacing the definitions of $A_{\bar{K}}$, $Q_{\bar{K}}$ and $S_{\bar{K}}$ in $\mathcal{D}(A_{\bar{K}}, B, Q_{\bar{K}}, R, S_{\bar{K}})$, one obtains

$$\begin{aligned}
0 =& Q_{\bar{K}} + A_{\bar{K}}^\top P_{\bar{K}} A_{\bar{K}} - P_{\bar{K}} - (S_{\bar{K}}^\top + A_{\bar{K}}^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S_{\bar{K}} + B^\top P_{\bar{K}} A_{\bar{K}}) \\
=& Q - S^\top \bar{K} - \bar{K}^\top S + \bar{K}^\top R\bar{K} + (A - B\bar{K})^\top P_{\bar{K}}(A - B\bar{K}) - P_{\bar{K}} \\
& - (S^\top - \bar{K}^\top R + (A - B\bar{K})^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S - R\bar{K} + B^\top P_{\bar{K}}(A - B\bar{K}))
\end{aligned}$$

The last term of the sum can be rewritten as

$$\begin{aligned}
&(S^\top - \bar{K}^\top R + (A - B\bar{K})^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S - R\bar{K} + B^\top P_{\bar{K}}(A - B\bar{K})) \\
&= -\bar{K}^\top \underbrace{(R + B^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}}_{I}(S - R\bar{K} + B^\top P_{\bar{K}}(A - B\bar{K})) \\
&\quad + (S^\top + A^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S - R\bar{K} + B^\top P_{\bar{K}}(A - B\bar{K})) \\
&= -\bar{K}^\top S + \bar{K}^\top R\bar{K} - \bar{K}^\top B^\top P_{\bar{K}} A + \bar{K}^\top B^\top P_{\bar{K}} B\bar{K} \\
&\quad + (S^\top + A^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S + B^\top P_{\bar{K}} A) \\
&\quad - (S^\top + A^\top P_{\bar{K}} B)\underbrace{(R + B^\top P_{\bar{K}} B)^{-1}(R + B^\top P_{\bar{K}} B)}_{I}\bar{K} \\
&= -\bar{K}^\top S + \bar{K}^\top R\bar{K} - \bar{K}^\top B^\top P_{\bar{K}} A + \bar{K}^\top B^\top P_{\bar{K}} B\bar{K} - S^\top \bar{K} - A^\top P_{\bar{K}} B\bar{K} \\
&\quad + (S^\top + A^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S + B^\top P_{\bar{K}} A).
\end{aligned}$$

Replacing this expression in $\mathcal{D}(A_{\bar{K}}, B, Q_{\bar{K}}, R, S_{\bar{K}})$ and simplifying one obtains

$$Q + A^\top P_{\bar{K}} A - P_{\bar{K}} - (S^\top + A^\top P_{\bar{K}} B)(R + B^\top P_{\bar{K}} B)^{-1}(S + B^\top P_{\bar{K}} A) = 0,$$

which directly entails $P_{\bar{K}} = P$.

The result $K_{\bar{K}} = K - \bar{K}$ is obtained as follows

$$\begin{aligned}
K_{\bar{K}} &= (R + B^\top PB)^{-1}(S_{\bar{K}} + B^\top PA_{\bar{K}}) \\
&= (R + B^\top PB)^{-1}(S - R\bar{K} + B^\top PA - B^\top PB\bar{K}) \\
&= (R + B^\top PB)^{-1}(S + B^\top PA) - (R + B^\top PB)^{-1}(R + B^\top PB)\bar{K} \\
&= K - \bar{K}.
\end{aligned}$$

□

Corollary 5.10. *By choosing $\bar{K} = K$, the DARE reduces to $Q_K + A_K^\top PA_K - P = 0$. This entails that, for any arbitrarily chosen matrix R_K such that $R_K + B^\top PB \succ 0$, the solution remains unchanged, i.e. $\mathcal{P}(A_K, B, Q_K, R, S_K) = \mathcal{P}(A_K, B, Q_K, R_K, S_K)$. One obtains thus a problem whose solution is $K_K = 0$ and $v_k = 0$.*

Proof. The DARE corresponding to $\mathcal{P}(A_K, B, Q_K, R_K, S_K)$ is given by

$$Q_K + A_K^\top PA_K - P - (S_K^\top + A_K PB)(R_K + B^\top PB)^{-1}(S_K + B^\top PA_K) = 0.$$

The last term is always zero, as can be seen by replacing the expression for K:

$$\begin{aligned} S_K + B^\top P A_K =& S - RK + B^\top PA - B^\top PBK \\ =& S + B^\top PA - (R + B^\top PB)K \\ =& S + B^\top PA - (R + B^\top PB)(R + B^\top PB)^{-1}(S + B^\top PA) \\ =& 0. \end{aligned}$$

For any matrix $R_K \succ B^\top PB$, the DARE simplifies and P is defined by

$$Q_K + A_K^\top P A_K - P = 0.$$

□

The following lemma states that the LQR cost can be modified such that the feedback matrix K remains unchanged while the cost-to-go matrix P can be any arbitrary symmetric matrix.

Lemma 5.11. *Let us assume that a stabilizing LQR exists for system (A, B), with weighting matrices Q, R, S. Given any real symmetric matrix $\bar{P}$, the equivalence $\mathcal{P}(A, B, Q, R, S) = \mathcal{P}(A, B, Q_{\bar{P}}, R_{\bar{P}}, S_{\bar{P}})$ holds when choosing $Q_{\bar{P}} = Q + A^\top \bar{P} A - \bar{P}$, $S_{\bar{P}} = S + A^\top \bar{P} B$ and $R_{\bar{P}} = R + B^\top \bar{P} B$. The solution $P_{\bar{P}}$ to $\mathcal{D}(A, B, Q_{\bar{P}}, R_{\bar{P}}, S_{\bar{P}})$ is given by $P_{\bar{P}} = P - \bar{P}$ and the resulting feedback gain is $K_{\bar{P}} = K$.*

Proof. By writing the DARE $\mathcal{D}(A, B, Q_{\bar{P}}, R_{\bar{P}}, S_{\bar{P}})$, and replacing the expressions for $Q_{\bar{P}}, R_{\bar{P}}, S_{\bar{P}}$, one immediately obtains $\mathcal{D}(A, B, Q, R, S)$, which is solved by matrix P and feedback gain K.

By definition $P = \bar{P} + P_{\bar{P}}$ and one can write

$$\begin{aligned} 0 =& Q + A^\top PA - P - (S^\top + A^\top PB)(R + B^\top PB)^{-1}(S + B^\top PA) \\ =& Q + A^\top \bar{P} A - \bar{P} + A^\top P_{\bar{P}} A - P_{\bar{P}} \\ & - (S^\top + A^\top \bar{P} B + A^\top P_{\bar{P}} B)(R + B^\top \bar{P} B + B^\top P_{\bar{P}} B)^{-1}(S + B^\top \bar{P} A + B^\top P_{\bar{P}} A) \\ =& Q_{\bar{P}} + A^\top P_{\bar{P}} A - P_{\bar{P}} - (S_{\bar{P}}^\top + A^\top P_{\bar{P}} B)(R_{\bar{P}} + B^\top P_{\bar{P}} B)^{-1}(S_{\bar{P}} + B^\top P_{\bar{P}} A) \end{aligned}$$

□

Note that choosing $\bar{P} = P$, one obtains $P_{\bar{P}} = 0$.

Corollary 5.12. *By applying Lemma 5.9 using $\bar{K} = K$ and Lemma 5.11, one obtains that $\mathcal{P}(A_K, B, Q_K, R_K, S_K) = \mathcal{P}(A_K, B, Q_{K,\bar{P}}, R_{K,\bar{P}}, S_{K,\bar{P}})$, where $Q_{K,\bar{P}} = Q_K + A_K^\top \bar{P} A_K - \bar{P}$, $S_{K,\bar{P}} = S_K + A_K^\top \bar{P} B$ and $R_{K,\bar{P}} = R_K + B^\top \bar{P} B$. Note that, as R_K can be chosen arbitrarily large, also $R_{K,\bar{P}}$ can be chosen arbitrarily large.*

Using these lemmas and corollaries, we can now prove Theorem 5.8.

Proof (Theorem 5.8). Using the equivalence given in Corollary 5.12, it follows that $\mathcal{P}(A, B, Q, R, S) = \mathcal{P}(A_K, B, Q_{K,\bar{P}}, R_{K,\bar{P}}, S_{K,\bar{P}})$ and $K_{K,\bar{P}} = 0$. As we are interested in possibly indefinite but stabilizing LQR formulations, by hypothesis, the feedback gain K is stabilizing, i.e. $\rho(A_K) = \rho(A - BK) < 1$, where ρ denotes the spectral radius, i.e. $\rho(A) = \max\{|\lambda| \mid \lambda \text{ is eigenvalue of } A\}$. We now use the Lyapunov inequality, which states that for any symmetric matrix $Q_\mathrm{L} \succ 0$ there exists a matrix P_L such that $A_K^\top P_\mathrm{L} A_K - P_\mathrm{L} + Q_\mathrm{L} = 0$. Choosing $\bar{P} = P - P_\mathrm{L}$, and recalling the result of Corollary 5.10, one obtains $A_K^\top \bar{P} A_K - \bar{P} = Q_\mathrm{L} - Q_K$. Then $Q_{K,\bar{P}} = Q_K + A_K^\top \bar{P} A_K - \bar{P} = Q_\mathrm{L} \succ 0$. One last step is needed to make matrix $H_{K,\bar{P}} = \begin{bmatrix} Q_{K,\bar{P}} & S_{K,\bar{P}}^\top \\ S_{K,\bar{P}} & R_{K,\bar{P}} \end{bmatrix}$ positive definite. By Corollary 5.12, one can arbitrarily choose matrix $R_{K,\bar{P}}$ arbitrarily large without changing the feedback matrix $K_{K,\bar{P}}$, nor the cost-to-go $\bar{P}$. The result $H_{K,\bar{P}} \succ 0$ is then obtained by selecting $R_{K,\bar{P}} \succ S_{K,\bar{P}} Q_{K,\bar{P}}^{-1} S_{K,\bar{P}}^\top$.

While this reformulated problem already yields an LQR with positive definite cost, it applies to the system (A_K, B). By applying Lemma 5.9 using $\bar{K} = -K$ one obtains an LQR with positive definite cost, which applies to the system (A, B), i.e. $\mathcal{P}(A, B, \tilde{Q}, \tilde{R}, \tilde{S}) = \mathcal{P}(A, B, Q, R, S)$, where $\tilde{Q} = Q_{K,\bar{P}} + S_{K,\bar{P}}^\top K + K^\top S_{K,\bar{P}} + K^\top R_{K,\bar{P}} K$, $\tilde{S} = S_{K,\bar{P}} + R_{K,\bar{P}} K$ and $\tilde{R} = R_{K,\bar{P}}$. This can be shown by applying the Schur complement to check positive definiteness. Matrix $\tilde{R}$ is positive definite, so it remains to prove that $\tilde{Q} \succ \tilde{S}^\top \tilde{R}^{-1} \tilde{S}$. This follows immediately from the definition:

$$\tilde{Q} = Q_{K,\bar{P}} + S_{K,\bar{P}}^\top K + K^\top S_{K,\bar{P}} + K^\top R_{K,\bar{P}} K,$$

$$\tilde{S}^\top \tilde{R}^{-1} \tilde{S} = S_{K,\bar{P}}^\top R_{K,\bar{P}}^{-1} S_{K,\bar{P}} + S_{K,\bar{P}}^\top K + K^\top S_{K,\bar{P}} + K^\top R_{K,\bar{P}} K.$$

The result $\tilde{Q} \succ \tilde{S}^\top \tilde{R}^{-1} \tilde{S}$ directly follows from $Q_{K,\bar{P}} \succ S_{K,\bar{P}}^\top R_{K,\bar{P}}^{-1} S_{K,\bar{P}}$. □

5.2.2 The Linear MPC case

We consider now the following LMPC problem where, without loss of generality, we assume to use the linearly rotated stage cost

$$\min_{w} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \underbrace{\begin{bmatrix} Q & S^\top \\ S & R \end{bmatrix}}_{H \not\succ 0} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + \frac{1}{2} x_N^\top P_N x_N \tag{5.14a}$$

$$\text{s.t. } x_0 = x, \tag{5.14b}$$

$$x_{k+1} = Ax_k + Bu_k, \quad k = 0, \dots, N-1, \tag{5.14c}$$

$$C \begin{bmatrix} x_k \\ u_k \end{bmatrix} + e \geq 0, \quad k = 0, \dots, N-1, \tag{5.14d}$$

where we assume that $e > 0$, so that no constraint is active at steady state and the gradient of the stage cost in the origin is 0. The prediction horizon N and terminal cost matrix P_N are tuning parameters to be selected by the control engineer and do not need to be discussed in this paper. For more details on the topic we refer to [100, 54]. Note that matrix P_N can in principle be indefinite.

When no inequality constraints are imposed, the solution to this problem can be obtained by propagating backwards the discrete Riccati equation (DRE)

$$\begin{aligned} \mathcal{R}_N(A, B, Q, R, S, P_N) := \{(P_0, P_1, \dots, P_N, K_0, K_1, \dots, K_{N-1}) \mid \\ P_{k-1} = Q + A^\top P_k A - (S^\top + A^\top P_k B) K_{k-1}, \\ K_{k-1} = (R + B^\top P_k B)^{-1} (S + B^\top P_k A), \\ k = N, \dots, 1.\}. \end{aligned} \tag{5.15a}$$

The closed loop MPC trajectory can then be computed by simulating forward in time the system $x_{k+1} = A_{K_0} x_k = (A - BK_0)x_k$, while the MPC predicted trajectory can be computed by simulating forward in time $x_{k+1} = A_{K_k} x_k = (A - BK_k)x_k$. If the terminal cost matrix is chosen as $P_N = P$, where P is obtained by solving the LQR, then the previously developed LQR theory applies. This section focuses on the case $P_N \neq P$.

Definition 5.13. *The DRE $\mathcal{R}_N(\tilde{A}, \tilde{B}, \tilde{Q}, \tilde{R}, \tilde{S}, \tilde{P}_N)$ is a positive definite reformulation of the indefinite DRE $\mathcal{R}_N(A, B, Q, R, S, P_N)$ if $\begin{bmatrix} \tilde{Q} & \tilde{S}^\top \\ \tilde{S} & \tilde{R} \end{bmatrix} \succ 0$, $\tilde{P}_N \succ 0$ and it yields the same feedback sequence $(K_0, K_1, \dots, K_{N-1})$ as the indefinite DRE.*

Theorem 5.8 considered the case of an infinite horizon linear quadratic OCP (LQR problem). The following theorem considers the case of a finite horizon OCP with a terminal cost which is different from the LQR cost to go (MPC problem).

Theorem 5.14. *Suppose that both (a) the LQR formulated using matrices A, B, H and (b) the economic LMPC scheme* (5.14) *are stabilizing. Then, there exists a positive definite LMPC scheme which yields the same primal solution as the economic LMPC scheme* (5.14)*.*

Before proving the theorem, let us first introduce the following lemma.

Lemma 5.15. *Two DREs $\mathcal{R}_N(A, B, Q, R, S, P_N)$ and $\mathcal{R}_N(A, B, \tilde{Q}, \tilde{R}, \tilde{S}, \tilde{P}_N)$ yield the same feedback sequence $K_0, \ldots, K_{N-1}$ if*

$$\tilde{P}_N - \tilde{P} = P_N - P, \quad \tilde{R} + B^T \tilde{P} B = R + B^T P B, \quad \tilde{S} + B^T \tilde{P} A = S + B^T P A, \tag{5.16}$$

with P and $\tilde{P}$ computed by solving the associated DAREs $\mathcal{D}(A, B, Q, R, S)$ and $\mathcal{D}(A, B, \tilde{Q}, \tilde{R}, \tilde{S})$.

Proof. Let us first consider the two equivalent LQR problems $\mathcal{P}(A, B, Q, R, S) = \mathcal{P}(A, B, \tilde{Q}, \tilde{R}, \tilde{S})$ with associated DAREs $(P, K) = \mathcal{D}(A, B, Q, R, S)$ and $(\tilde{P}, K) = \mathcal{D}(A, B, \tilde{Q}, \tilde{R}, \tilde{S})$. Defining $\Delta_k = P_k - P$ and $\tilde{\Delta}_k = \tilde{P}_k - \tilde{P}$, the two DREs can be rewritten as $\mathcal{R}_N(A, B, Q, R, S, P + \Delta_N)$ and $\mathcal{R}_N(A, B, \tilde{Q}, \tilde{R}, \tilde{S}, \tilde{P} + \tilde{\Delta}_N)$, i.e.

$$P + \Delta_{k-1} = Q + A^\top P A + A^\top \Delta_k A - (S^\top + A^\top P B + A^\top \Delta_k B) K_{k-1}, \tag{5.17a}$$

$$K_{k-1} = (R + B^\top P B + B^\top \Delta_k B)^{-1} (S + B^\top P A + B^\top \Delta_k A), \tag{5.17b}$$

and

$$\tilde{P} + \tilde{\Delta}_{k-1} = \tilde{Q} + A^\top \tilde{P} A + A^\top \tilde{\Delta}_k A - (\tilde{S}^\top + A^\top \tilde{P} B + A^\top \tilde{\Delta}_k B) K_{k-1}, \tag{5.18a}$$

$$K_{k-1} = (\tilde{R} + B^\top \tilde{P} B + B^\top \tilde{\Delta}_k B)^{-1} (\tilde{S} + B^\top \tilde{P} A + B^\top \tilde{\Delta}_k A). \tag{5.18b}$$

The identity $\Delta_k = \tilde{\Delta}_k$ is obtained if condition (5.16) is satisfied. This can be seen by replacing the values of P and $\tilde{P}$ from the DARE into equations (5.17) and (5.18), which gives

$$\Delta_{k-1} = A^\top \Delta_k A - (S^\top + A^\top P B) K - (S^\top + A^\top P B + A^\top \Delta_k B) K_{k-1}, \tag{5.19a}$$

$$K_{k-1} = (R + B^\top P B + B^\top \Delta_k B)^{-1} (S + B^\top P A + B^\top \Delta_k A), \tag{5.19b}$$

and

$$\tilde{\Delta}_{k-1} = A^\top \tilde{\Delta}_k A - (\tilde{S}^\top + A^\top \tilde{P} B)K - (\tilde{S}^\top + A^\top \tilde{P} B + A^\top \tilde{\Delta}_k B)K_{k-1}, \tag{5.20a}$$

$$K_{k-1} = (\tilde{R} + B^\top \tilde{P} B + B^\top \tilde{\Delta}_k B)^{-1}(\tilde{S} + B^\top \tilde{P} A + B^\top \tilde{\Delta}_k A). \tag{5.20b}$$

By replacing condition (5.16) into (5.20) one immediately obtains (5.19). □

Remark 5.16. *Lemma 5.15 provides sufficient conditions for the equivalence of two DREs but does not provide any guarantee on the existence of two DREs which satisfy those conditions. It will be clear in the proof of Theorem 5.14 that this is indeed possible.*

We can now prove Theorem 5.14.

Proof (Theorem 5.14). The proof follows from combining the results of Theorem 5.8 and Lemma 5.15. In particular, conditions (5.16) can be easily enforced in the framework of Theorem 5.8. From Corollary 5.10 we have that $S_K^\top + B^\top P A_K = S_{K,\bar{P}}^\top + B^\top P_{\bar{P}} A_K = 0$, for any choice of matrix $\bar{P}$. As matrix $R_{K,\bar{P}}$ can be arbitrarily chosen, one can choose $R_{K,\bar{P}} = R + B^\top P B - B^\top P_{\bar{P}} B$.

As in Theorem 5.8, we can choose $\bar{P} = P - P_\mathrm{L}$, to obtain $P_{\bar{P}} = P_\mathrm{L}$. In order to guarantee positive definiteness, let us choose Q_L arbitrarily but positive definite and scale it by a factor $\gamma > 0$. The Lyapunov equation becomes $\gamma A_K^\top P_\mathrm{L} A_K - \gamma P_\mathrm{L} + \gamma Q_\mathrm{L} = 0$ and, subsequently,

$$S_{K,\bar{P}}^\top = -\gamma B^\top P_\mathrm{L} A_K,$$

$$R_{K,\bar{P}} = R + B^\top P B - \gamma B^\top P_\mathrm{L} B.$$

The condition for ensuring positive definiteness $R_{K,\bar{P}} \succ S_{K,\bar{P}}^\top Q_{K,\bar{P}}^{-1} S_{K,\bar{P}}$ thus becomes

$$R + B^\top P B \succ \gamma B^\top P_\mathrm{L} B + \gamma (B^\top A_K^\top P_\mathrm{L} B) Q_\mathrm{L}^{-1} (B^\top P_\mathrm{L} A_K B), \tag{5.21}$$

which holds for γ sufficiently small. The transformation back from system (A_K, B) to system (A, B) can then be done as in Theorem 5.8. □

Theorem 5.14 implies that for all stabilising schemes the stage and terminal cost matrices $H, P_N \not\succ 0$ can always be replaced by appropriately selected matrices $\tilde{H}, \tilde{P}_N \succ 0$ without changing the (primal) solution of the LMPC scheme. In this chapter, we will denote by the term *convexification* the procedure of computing the positive definite stage cost.

We define the feedback matrix K and the cost-to go matrix P as the ones obtained from the LQR formulated using the original cost. Note that the LQR does not necessarily exist for all stabilising LMPC formulations (5.14): this is the case when the infinite horizon economic problem is not stabilising, and the system is not optimally operated at steady state. The case of non optimal steady state operation is out of the scope of this section.

Because the choice of $\tilde{P}$, $\tilde{H}$ is not unique, it was proposed in [122] to compute them by solving the following SDP

$$\min_{\tilde{P},\tilde{Q},\tilde{R},\tilde{S},\tilde{H}} \quad \|\tilde{H} - I\|^2 \tag{5.22a}$$

$$\text{s.t.} \quad \tilde{H} = \begin{bmatrix} \tilde{Q} & \tilde{S}^\top \\ \tilde{S} & \tilde{R} \end{bmatrix} \tag{5.22b}$$

$$\tilde{H} \succeq 0 \tag{5.22c}$$

$$\tilde{P} \succeq 0 \tag{5.22d}$$

$$\tilde{Q} + A^\top \tilde{P} A - \tilde{P} - (\tilde{S}^\top + A^\top \tilde{P} B)K = 0, \tag{5.22e}$$

$$(\tilde{R} + B^\top \tilde{P} B)K - (\tilde{S} + B^\top \tilde{P} A) = 0, \tag{5.22f}$$

$$\tilde{R} + B^T \tilde{P} B = R + B^T P B, \tag{5.22g}$$

$$\tilde{S} + B^T \tilde{P} A = S + B^T P A. \tag{5.22h}$$

The convexification of the terminal cost matrix is given by $\tilde{P}_N = P_N - P + \tilde{P}$.

In order to analyse the modification of the stage cost matrix introduced by $\tilde{P}$, we define $\delta P = P - \tilde{P}$ and introduce the linear operator

$$\mathcal{H}(\delta P) = \begin{bmatrix} A^\top \delta P A - \delta P & A^\top \delta P B \\ B^\top \delta P A & B^\top \delta P B \end{bmatrix}. \tag{5.23}$$

We then define $\tilde{H} = H + \mathcal{H}(\delta P)$. It can be easily checked that this construction directly enforces Constraints (5.22e)-(5.22h). Then, one can convexify the cost by solving the following SDP which is equivalent to (5.22)

$$\min_{\delta P} \quad \|H + \mathcal{H}(\delta P) - I\|^2 \tag{5.24a}$$

$$\text{s.t.} \quad H + \mathcal{H}(\delta P) \succeq 0. \tag{5.24b}$$

Note that $\tilde{H} \succeq 0$ implies that the cost-to-go matrix $\tilde{P}$ of the LQR computed using $\tilde{H}$ satisfies $\tilde{P} \succeq 0$. Moreover, no knowledge about the cost-to-go P

or the feedback gain matrix K is needed in order to compute the stage cost convexification given by $\tilde{H} \succeq 0$. Note that the cost function of the SDP is somewhat arbitrary and has been chosen to favour well-conditioned stage cost matrices. However, other choices are possible and even preferable and will be discussed in Section 5.3.2.

This formulation of the cost convexification also highlights the connection with the concepts of strict dissipativity and rotated cost. In the following, we prove an important implication of Theorem 5.14 on the necessity of strict dissipativity.

Corollary 5.17. *Consider an economic LMPC formulation of the form* (5.14) *with no active constraints at steady state. Strict dissipativity is not only sufficient but also necessary for the stability of the closed-loop system.*

Proof. By Theorem 5.14, if the economic LMPC (5.14) with no active constraints at steady state is stabilising, then there exists a matrix δP such that $\tilde{H} := H + \mathcal{H}(\delta P) \succ 0$. It is easy to see that the cost convexification (5.23) is a nonlinear cost rotation of the form (4.17) with storage function $\lambda(x_k) = -x_k^\top \delta P x_k$.

Therefore, the convexified stage cost satisfies

$$\begin{aligned} L(x,u) &:= \ell(x,u) + \lambda(x) - \lambda(Ax+Bu) \\ &= \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \tilde{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \geq \epsilon \|x\|^2, \end{aligned}$$

with $\epsilon > 0$. As a result, strict dissipativity is therefore not only sufficient, but also necessary for closed-loop stability of economic LMPC with no active constraints at steady state. □

Note that the original stability proof based on strict dissipativity only requires the rotated stage cost to be positive definite on a compact set $\mathbb{Z}$. In the framework of LMPC with no active constraints at steady state, however, we obtain the stronger result of Theorem 5.17. Moreover, Equation (5.23) proves that the convexification procedure is a rotation of the cost in the sense of [3]. We remark that in the economic LMPC Formulation (5.14) we assumed without loss of generality that the cost was linearly rotated. Then, in the general case, for economic LMPC without active constraints at steady state, the rotated cost is given by the composition of a linear rotation and a convexification, i.e. $\lambda(x) = \lambda_s^\top x - x^\top \delta P x$. In the following section, we analyse the case of active constraints at steady state.

5.3 Linear MPC with Active Constraints at Steady State

In this section, we study the properties of economic LMPC formulations in case some constraints are active at steady state. First, we establish a theoretical framework for proving the equivalence of the original indefinite formulation and a positive definite one. Subsequently, we provide a practical approach for computing a positive definite LMPC formulation which locally gives the same feedback law as the original indefinite one.

5.3.1 Equivalent MPC Formulations with Active Constraints at Steady State

In the case of a linear system, a quadratic cost and strictly active constraints at steady state, the cost can in general have a non-zero gradient at the optimum even after a linear rotation of the stage cost (cf. Section 5.1). Also in this section, we consider an LMPC problem formulated using the linearly rotated stage cost, given by

$$\min_{w} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top H \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.25a}$$

$$\text{s.t. } x_0 = \hat{x}_0, \tag{5.25b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \tag{5.25c}$$

$$C \begin{bmatrix} x_k \\ u_k \end{bmatrix} + e \geq 0, \quad k = 0, \dots, N-1, \tag{5.25d}$$

$$x_N = x_\mathrm{s}. \tag{5.25e}$$

where the non-zero gradient at the origin is due to the presence of strictly active constraints. We consider a terminal point constraint for simplicity, however, any formulation satisfying the conditions of Lemma 5.4 can be used in our derivation.

We analyse now some local properties of Problem (5.25), which hold for a given fixed active set. Note that the problem formulations we will consider can be too restrictive and inadequate for practical applications. However, they are helpful for establishing some properties which can then be exploited for problem formulations which can be applied in practice. We discuss these issues in Remark 5.20.

Let us denote the Jacobian of the constraints which are strictly active at steady state as $C_{\mathbb{A}_\mathrm{s}}$, which we assume to be full row rank. Note that, the steady state being $x_\mathrm{s} = 0$, $u_\mathrm{s} = 0$, the affine term in the active inequality constraints must disappear, i.e. $e_{\mathbb{A}_\mathrm{s}} = 0$.

Theorem 5.18. *Let us consider the region $\mathbb{X}_0$ of initial states $\hat{x}_0$ around the origin for which the optimal active set of Problem* (5.25) *includes all constraints which are strictly active for $\hat{x}_0 = 0$. Assume that the LMPC problem*

$$\min_{w} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top H \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.26a}$$

$$s.t.\ \ x_0 = \hat{x}_0, \tag{5.26b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \tag{5.26c}$$

$$C_{\mathbb{A}_\mathrm{s}} \begin{bmatrix} x_k \\ u_k \end{bmatrix} = 0, \quad k = 0, \dots, N-1, \tag{5.26d}$$

$$x_N = x_\mathrm{s}. \tag{5.26e}$$

is stabilising for all $\hat{x}_0 \in \mathbb{X}_0$. Then there exist matrices δP and F such that

$$H + \mathcal{H}(\delta P) + C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}} \succ 0.$$

Additionally,

$$Z^\top (H + \mathcal{H}(\delta P)) Z \succ 0,$$

holds for any Z with $C_{\mathbb{A}_\mathrm{s}} Z = 0$, $Z^\top Z = I$ and $\begin{bmatrix} C_{\mathbb{A}_\mathrm{s}}^\top & Z \end{bmatrix}$ invertible.

Proof. For any matrix F, Problem (5.26) can be reformulated as

$$\min_{w} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \Big(\underbrace{H + C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}}}_{\tilde{H}} \Big) \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix}$$

$$\text{s.t.}\ \ x_0 = \hat{x}_0, \tag{5.27a}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \tag{5.27b}$$

$$C_{\mathbb{A}_\mathrm{s}} \begin{bmatrix} x_k \\ u_k \end{bmatrix} = 0, \quad k = 0, \dots, N-1, \tag{5.27c}$$

$$x_N = x_\mathrm{s}. \tag{5.27d}$$

By construction, for any $\hat{x}_0 \in \mathbb{X}_0$, Problem (5.27) delivers the same solution as Problem (5.26). Moreover, there exists a matrix $F \succ 0$, such that $C_{\mathbb{A}_\mathrm{s}} \bar{H} C_{\mathbb{A}_\mathrm{s}}^\top \succ 0$.

We now focus on the special case $\hat{x}_0 = 0$. Note that Lyapunov stability of Problem (5.26) entails that the primal solution of Problem (5.27) is zero, i.e. $w^{(5.27)}(0) = w^{(5.26)}(0) = 0$. In the following, we dualise the constraints, i.e. we replace the original cost with the Lagrangian of the problem. By Lemma 5.4, the Lagrange multipliers of Problem (5.27) with $\hat{x}_0 = 0$ coincide with those of the steady-state problem. Moreover, the primal solution is not changed by the following linear rotation:

$$\begin{aligned} J = &\sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \bar{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} + \frac{1}{2} x_N^\top P_N x_N + p^\top x_N \\ &- \lambda_\mathrm{s}^\top x_0 + \sum_{k=0}^{N-1} -\lambda_\mathrm{s}^\top (x_{k+1} - Ax_k - Bu_k) - \mu_\mathrm{s}^\top C_{\mathbb{A}_\mathrm{s}} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \\ = &\sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \bar{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + \frac{1}{2} x_N^\top P_N x_N, \end{aligned}$$

where, by optimality, all linear terms cancel out. Note that our new cost is given by the Lagrangian function of Problem (5.27) evaluated at the optimal Lagrange multipliers.

We obtain then the following problem

$$\min_{w} \quad \sum_{k=0}^{N} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \bar{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.28a}$$

$$\text{s.t.} \quad x_0 = \hat{x}_0, \tag{5.28b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1. \tag{5.28c}$$

$$C_{\mathbb{A}_\mathrm{s}} \begin{bmatrix} x_k \\ u_k \end{bmatrix} = 0, \quad k = 0, \dots, N-1, \tag{5.28d}$$

$$x_N = x_\mathrm{s}. \tag{5.28e}$$

For the initial condition $\hat{x}_0 = 0$, the optimal Lagrange multipliers of Constraints (5.28d) are $\mu^{(5.28)}(0) = 0$ and the primal solution is $x_k = 0$ and $u_k = 0$. We recall that $C_{\mathbb{A}_\mathrm{s}} \bar{H} C_{\mathbb{A}_\mathrm{s}}^\top \succ 0$ and, because the primal solution is zero, the Lagrange multipliers associated with constraints (5.28d) are also zero. Then,

also the following problem

$$\min_{w} \sum_{k=0}^{N} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \bar{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.29a}$$

$$\text{s.t. } x_0 = \hat{x}_0, \tag{5.29b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \tag{5.29c}$$

$$x_N = x_\mathrm{s}, \tag{5.29d}$$

delivers the primal solution $x_k = 0$ and $u_k = 0$ for $\hat{x}_0 = 0$. This entails that Problem (5.29) must be stabilising for any initial state $x_0 = \hat{x}_0 \in \mathbb{R}^{n_x}$. Hence by Theorem 5.14

$$\exists\ \delta P \text{ s.t.} \qquad H + C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}} + \mathcal{H}(\delta P) \succ 0, \tag{5.30}$$

and, equivalently,

$$\exists\ \delta P \text{ s.t.} \qquad Z^\top \left(H + \mathcal{H}(\delta P)\right) Z \succ 0. \tag{5.31}$$

□

Remark 5.19. *Note that Problem* (5.29) *is not equivalent to Problem* (5.28)*. Indeed, the feedback control law is expected to be different for the two problems. Nevertheless, the fact that the unconstrained Problem* (5.29) *must be stabilising allows us to establish the desired result, i.e. the existence of a positive definite stage cost matrix.*

Remark 5.20. *Note that in the proposed framework we did not make any direct assumption on the satisfaction of any constraint qualification. Clearly, the LMPC Problem* (5.25) *needs to be formulated in a way which can be implemented in practice. The result we obtained only concerns the properties of the stage cost. Consider e.g. the case of pure state constraints. For practical implementations, one would impose the state constraints starting from $k = 1$ (we call this problem MPC1), as opposed to $k = 0$ (we call this problem MPC0). Indeed, the initial state cannot be controlled and, therefore, satisfaction of the state constraints cannot be guaranteed for MPC0. Moreover, if the system starts at steady state, most constraint qualifications are violated for MPC0. Indeed both the linear independence constraint qualification (LICQ) and the Mangasarian-Fromovitz constraint qualification (MFCQ) are violated, while the Abadie constraint qualification holds. Note however that for all initial states satisfying the given state bound, the solution of MPC1 and MPC0 must coincide. In practice, in order to avoid problems related to infeasibility, it is advisable to introduce so-called soft constraints using slack variables with an adequately*

chosen L1 penalty. These slack variables can be treated in the proposed framework as extra controls which do not affect the system dynamics but which enter the constraints.

We formulate the LMPC problem with positive definite stage cost as

$$\min_{w} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \tilde{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} + \frac{1}{2} x_N^\top \tilde{P}_N x_N + p^\top x_N \quad (5.32a)$$

$$\text{s.t. } x_0 = \hat{x}_0, \quad (5.32b)$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \quad (5.32c)$$

$$C \begin{bmatrix} x_k \\ u_k \end{bmatrix} + e \geq 0, \quad k = 0, \dots, N-1. \quad (5.32d)$$

with $\tilde{H} = H + \mathcal{H}(\delta P) + C_{\mathbb{A}_s}^\top F C_{\mathbb{A}_s}$ and $\tilde{P}_N = P_N - \delta P$.

Theorem 5.21. *Consider any linear MPC scheme* (5.25) *which is stabilising in the sense of Lyapunov with region of attraction $\mathcal{X} \subseteq \mathbb{X}_0$ and which satisfies the assumptions of Theorem 5.5. Then, the positive definite LMPC scheme* (5.32) *delivers the same primal solution as the original problem for initial conditions $\hat{x}_0$ in a neighbourhood of the origin, i.e. $w^{(5.32)}(\hat{x}_0) = w^{(5.25)}(\hat{x}_0),\ \forall\ \hat{x}_0 \in \bar{\mathbb{X}}_0$, where we define the set $\bar{\mathbb{X}}_0$ as the set of initial conditions $\hat{x}_0$ for which the set of active constraints for both Problem* (5.25) *and* (5.32) *coincides and includes all constraints that are active for $\hat{x}_0 = 0$.*

Moreover, if the positive definite stage cost matrix can be computed using $F = 0$, then the equivalence holds for all initial conditions $\hat{x}_0$, i.e. $w^{(5.32)}(\hat{x}_0) = w^{(5.25)}(\hat{x}_0),\ \forall\ \hat{x}_0$.

Proof. Given the stage cost matrix H, Theorem 5.18 guarantees the existence of a positive definite stage cost matrix $\tilde{H} = H + \mathcal{H}(\delta P) + C_{\mathbb{A}_s}^\top F C_{\mathbb{A}_s}$. We now distinguish two cases.

Let us first consider the case $F = 0$. Because the convexification (5.23) is a rotation in the sense of [3], the addition of the term $\mathcal{H}(\delta P)$ to the stage cost does not change the primal solution of the problem, regardless of the active set. Then, for $F = 0$, the equivalence of Problems (5.25) and (5.32) is directly obtained, regardless of the active set.

For $F \neq 0$ instead, the equivalence of Problems (5.25) and (5.32) will in general depend on the active set. Theorem 5.5 guarantees that there exists a neighbourhood of initial conditions $\hat{x}_0$ around the origin in which the constraints

that are strictly active for $\hat{x}_0 = 0$ remain active. As path constraints are local in time, we can analyse the active set of each stage separately. Let us define the nullspace of the strictly active constraints at steady state as Z, so that $C_{\mathbb{A}_s} Z = 0$. If some inactive constraint becomes strictly active and all previously active constraints remain active, then the nullspace of the new set of strictly active constraints $\bar{Z}$ will by definition satisfy $C_{\mathbb{A}_s} \bar{Z} = 0$. Then, the contribution of the term $C_{\mathbb{A}_s}^\top F C_{\mathbb{A}_s}$ to the reduced Hessian will be 0, i.e. $\bar{Z}^\top C_{\mathbb{A}_s}^\top F C_{\mathbb{A}_s} \bar{Z} = 0$. Therefore, if in addition to the constraints which are active at steady state more constraints become active, the reduced Hessian of Problems (5.25) and (5.32) coincide and the two problems deliver the same primal solution.

Then the primal solution of the positive definite Problem (5.32) will be the same as that of the original Problem (5.25), i.e. $w^{(5.32)}(\hat{x}_0) = w^{(5.25)}(\hat{x}_0)$, for all $\hat{x}_0 \in \bar{\mathbb{X}}_0$. Therefore the control laws of the two MPC schemes are identical, i.e. $u^{(5.32)}(\hat{x}_0) = u^{(5.25)}(\hat{x}_0)$. □

5.3.2 A Practical Approach for Convexifying the Stage Cost

The convexification procedure can be formulated as the following Linear Matrix Inequality (LMI)

$$\tilde{H} = H + \mathcal{H}(\delta P) + C_{\mathbb{A}_s}^\top F C_{\mathbb{A}_s} \succ 0. \tag{5.33}$$

Note that, similarly to the unconstrained case, by solving the LMI (5.33) in variables δP and F the stage cost matrix $\tilde{H}$ can be badly conditioned. Moreover, as stated in Theorem 5.21, if the LMI admits a solution for $F = 0$, then the positive definite LMPC scheme (5.32) delivers the same primal solution as the original LMPC scheme (5.25) for all feasible initial conditions.

In order to mitigate the problem of bad conditioning, one can formulate an SDP problem as follows. Add the constraint $\tilde{H} \succeq \alpha I$, and maximise the value of α. This ensures that the stage cost matrix is made positive definite. Introduce the constraints $\beta I \succeq \tilde{H}$, and minimise β. This allows one to keep the condition number of the stage cost matrix low. Note that minimising the condition number directly would lead to a nonconvex optimisation problem. Instead, we propose to minimise $\gamma\beta - \alpha$, with $\gamma \geq 0$ a parameter of choice.

In order to also address the second observation, we propose a two-step procedure in which it is first attempted to solve the convexification problem using $F = 0$ and, only when necessary, $F \neq 0$ is used. We construct the convexification

procedure using the following SDP:

$$\min_{\delta P, F, \alpha, \beta} \quad \gamma\beta - \alpha + \rho\|F\| \tag{5.34a}$$

$$\text{s.t.} \quad Z^\top (H + \mathcal{H}(\delta P))Z \succeq 0, \tag{5.34b}$$

$$H + \mathcal{H}(\delta P) + \eta C_{\mathbb{A}_\text{s}}^\top F C_{\mathbb{A}_\text{s}} \succeq \alpha I, \tag{5.34c}$$

$$\beta I \succeq H + \mathcal{H}(\delta P), \tag{5.34d}$$

with $\gamma \geq 0$, $\rho \geq 0$ two tuning parameters and $\|\cdot\|$ and appropriately chosen matrix norm. Parameter $\eta \in \{0, 1\}$ is used to construct the aforementioned 2-step procedure. First, SDP (5.34) is solved using $\eta = 0$, $\rho > 0$, which results in $F = 0$. If the solution satisfies $\alpha^* > 0$, we have found the stage cost matrix for a positive definite reformulation of the original problem which is globally equivalent to it. If even with $\gamma = 0$ we obtain $\alpha^* < 0$, we set $\eta = 1$ and solve the SDP again. In this second case the equivalence of the positive definite and original problems will only be local.

For the second step of the convexification procedure, one can decide to set $\rho = 0$ and constrain δP to match the one found by the previous solution. This way, the second solution only acts on matrix F in order to make the stage cost matrix positive definite and reduce its condition number.

Note that, if one chooses $\gamma = 0$, it might be necessary to introduce the additional constraint $\alpha \leq \bar{\alpha}$ with $\bar{\alpha}$ a parameter of choice. This ensures that the SDP is bounded.

We also remark that the condition number is given by $\kappa = \frac{\beta}{\alpha}$ and its first order Taylor expansion around $(\alpha, \beta) = (a, b)$ yields $\kappa \approx \frac{b}{a} + \frac{1}{a}\beta - \frac{b}{a^2}\alpha$. One can therefore interpret SDP (5.34) as the subproblem of a sequential convex programming (SCP) method for minimising the condition number κ of matrix $\tilde{H}$.

5.4 Tuning of the Tracking (N)MPC Based on Economic Criteria

Using the results obtained previously, we aim at formulating a tracking NMPC scheme which is locally equivalent to economic NMPC (ENMPC) up to first order. We begin by defining the equivalence between MPC problems.

Definition 5.22 (Equivalent problems)**.** *We consider two MPC problems* A *and* B *delivering feedback laws* $u^{\mathrm{A}}(\hat{x}_0)$ *and* $u^{\mathrm{B}}(\hat{x}_0)$ *respectively and we define the following:*

(i) MPC problems A *and* B *are* equivalent *iff they deliver the same feedback law, i.e. for all* $\hat{x}_0$ *it holds that* $u^{\mathrm{A}}(\hat{x}_0) = u^{\mathrm{B}}(\hat{x}_0)$*;*

(ii) MPC problems A *and* B *are* locally equivalent *iff there exists a neighbourhood* $\mathcal{N}$ *of the origin so that for all* $\hat{x}_0 \in \mathcal{N}$ *it holds that* $u^{\mathrm{A}}(\hat{x}_0) = u^{\mathrm{B}}(\hat{x}_0)$*;*

(iii) MPC problems A *and* B *are* locally first-order equivalent *iff there exists a neighbourhood* $\mathcal{N}$ *of the origin so that for all* $\hat{x}_0 \in \mathcal{N}$ *it holds that* $\|u^{\mathrm{A}}(\hat{x}_0) - u^{\mathrm{B}}(\hat{x}_0)\| = O(\|\hat{x}_0\|^2)$.

We can now state the main result of this section in the following theorem.

Theorem 5.23. *Given any stabilising economic NMPC of the form* (5.1) *satisfying the assumptions of Theorem 5.5, it is possible to formulate both a tracking linear MPC and a tracking nonlinear MPC with positive definite stage cost which are locally first-order equivalent to the economic NMPC problem* (5.1)*.*

The result of Theorem 5.23 is strongly related to the results obtained in the previous Sections. However, these results are only valid for linear MPC and do not directly apply to the nonlinear case. Before proving the theorem, we establish some intermediate results.

We will (i) formulate a linear MPC scheme which is locally first-order equivalent to economic NMPC, (ii) convexify its stage cost to obtain a locally equivalent linear MPC problem with positive definite cost and (iii) formulate a tracking NMPC problem whose quadratic expansion at the origin coincides with that of the economic NMPC scheme. Finally, we will prove that the tracking NMPC problem formulated following this procedure is locally first-order equivalent to the original economic NMPC problem.

We aim at constructing a series of MPC problems such that

$$\text{ENMPC (5.1)} \leftrightarrow \text{ELMPC (5.35)} \leftrightarrow \text{PD LMPC (5.36)} \leftrightarrow \text{PD NMPC (5.3)},$$

where we use the symbol $\leftrightarrow$ to denote local first-order equivalence and we label an MPC problem with positive definite stage cost as PD. For the sake of simplicity, we will consider here a terminal point constraint, i.e. $x_N = x_{\mathrm{s}}$. Note however that any other choice of terminal cost and constraint can be used in the proposed framework.

The Lagrangian Hessian of Problem (5.1) is block diagonal, with each block given by

$$\nabla^2_{w_k}\mathcal{L}_k(w,\lambda,\mu) = \nabla^2\ell(x_k,u_k) + \sum_{i=0}^{n_x}\lambda_{k+1,i}\nabla^2 f_i(x_k,u_k) - \sum_{j=0}^{n_\mu}\mu_{k,j}\nabla^2 h_j(x_k,u_k).$$

We define

$$H = \nabla^2_{w_k}\mathcal{L}_k(w,\lambda,\mu), \qquad q = \nabla_{w_k}\ell(x_k,u_k),$$
$$A = \nabla_{x_k} f(x_k,u_k)^\top, \qquad B = \nabla_{u_k} f(x_k,u_k)^\top,$$
$$C = \nabla_{w_k} h(x_k,u_k)^\top,$$

evaluated at the primal dual solution of the steady-state Problem (5.2), i.e. w_s, λ_s, μ_s. Then, the following problem is the local quadratic expansion of the economic NMPC problem around the steady state located at the origin

$$\min_{\substack{x_0,\dots,x_N\\ u_0,\dots,u_{N-1}}} \sum_{k=0}^{N-1}\frac{1}{2}\begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top H \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.35a}$$

$$\text{s.t. } x_0 = \hat{x}_0, \tag{5.35b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0,\dots,N-1, \tag{5.35c}$$

$$C\begin{bmatrix} x_k \\ u_k \end{bmatrix} + e \geq 0, \quad k = 0,\dots,N-1, \tag{5.35d}$$

$$x_N = x_\mathrm{s}. \tag{5.35e}$$

Lemma 5.6 ensures that $\frac{\partial w^{(5.35)}(\hat{x}_0)}{\partial \hat{x}_0} = \frac{\partial w^{(5.1)}(\hat{x}_0)}{\partial \hat{x}_0}$, so that the Problem (5.35) is locally first-order equivalent to the economic NMPC problem. Note that, as the stage cost matrix H is the Hessian of the Lagrangian of a constrained optimisation problem, it is in general indefinite. Theorem 5.21 states that, if the original economic NMPC problem is stabilising in the sense of Lyapunov on an infinite horizon, there exists a positive definite stage cost matrix $\tilde{H} \succ 0$

which can be used to formulate the following problem

$$\min_{\substack{x_0,\dots,x_N \\ u_0,\dots,u_{N-1}}} \sum_{k=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \tilde{H} \begin{bmatrix} x_k \\ u_k \end{bmatrix} + q^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \tag{5.36a}$$

$$\text{s.t. } x_0 = \hat{x}_0, \tag{5.36b}$$

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0, \dots, N-1, \tag{5.36c}$$

$$C \begin{bmatrix} x_k \\ u_k \end{bmatrix} + e \geq 0, \quad k = 0, \dots, N-1, \tag{5.36d}$$

$$x_N = x_\mathrm{s}, \tag{5.36e}$$

which is locally equivalent to (5.35), and therefore also locally first-order equivalent to (5.1).

Until now, we have proven all equivalences but the last one. By relying on the equivalence result of Lemma 5.6 it is possible to formulate a tracking NMPC problem whose QP approximation at the origin is given by Problem (5.36), so that the tracking NMPC problem is a local approximation of the economic NMPC problem. However, some difficulties can be encountered if this last equivalence needs to be satisfied by using a positive definite stage cost also in the tracking NMPC formulation. We explain in the following why this is not straightforward and we propose a way to tackle this problem.

Matrix $\tilde{H}$ is given by (5.30), and reads as:

$$\tilde{H} = \nabla^2_{w_k} \mathcal{L}_k(w, \lambda, \mu) + \mathcal{H}(\delta P) + C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}} \succ 0,$$

where the Hessian of the Lagrangian is evaluated at $w_k = w_\mathrm{s}$, $\lambda_k = \lambda_\mathrm{s}$, $\mu_k = \mu_\mathrm{s}$. In order to define the proposed tracking NMPC scheme, we need to use a stage cost that yields a problem whose quadratic expansion is (5.36). It is easy to see that the choice of a quadratic cost

$$\sum_{k=0}^{N-1} w_k^\top \hat{H}_\mathrm{t} w_k + q_\mathrm{t}^\top w_k,$$

$$\hat{H}_\mathrm{t} = \nabla^2 \ell(x_k, u_k) + \mathcal{H}(\delta P) + C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}}, \qquad q_\mathrm{t} = \nabla \ell(x_k, u_k),$$

does deliver the desired property.

However, we are interested in a stage cost matrix $\hat{H}_\mathrm{t} \succ 0$. Because

$$\hat{H}_\mathrm{t} = \underbrace{\tilde{H}}_{\succ 0} + \underbrace{(\sum_{i=0}^{n_x} \lambda_\mathrm{s} \nabla^2 f_i(x_k, u_k)}_{\not\succeq 0} \underbrace{- \sum_{j=0}^{n_\mu} \mu_\mathrm{s} \nabla^2 h_j(x_k, u_k))}_{\not\succeq 0},$$

in the general case, we do not have the guarantee of obtaining the desired property.

In nonlinear MPC, the Hessian of equality constraint i, i.e. $\nabla^2_{w_k} f_i(x_k, u_k)$ is in general nonzero. Moreover, also path constraints are possibly nonlinear, in which case they yield nonzero contributions to the Hessian of the Lagrange function. In order to solve the problem relative to the Hessian of the equality constraints, one can consider the rotated version of the economic NMPC problem, so that $\lambda_\mathrm{s} = 0$ and, therefore $\sum_{i=0}^{n_x} \lambda_\mathrm{s} \nabla^2 f_i(x_k, u_k) = 0$. However, we have proven in Lemma 5.3 that it is not possible to rotate the cost with respect to the inequality multipliers without changing the primal solution, so that one still needs to address the problem of the contribution of the path constraints to the Hessian.

If $\tilde{H} \not\succ -\sum_{j=0}^{n_\mu} \mu_{k,j} \nabla^2_w h_j(x_k, u_k)$, we propose to tackle the problem by introducing slack variables s and replacing the inequality constraints $\bar{h}(w, \theta) \geq 0$ by the equality constraints $\bar{h}(w, \theta) - s = 0$ and inequality constraints $s \geq 0$. Then, the inequality constraints do not contribute to the Lagrangian Hessian because they are linear. At the same time, the contribution to the Hessian which was previously due to inequality constraints is now due to equality constraints which can be rotated.

Lemma 5.24. *Consider vectors $\bar{\lambda}$ and $\bar{\mu}$, a vector of fixed parameters θ, and the two parametric NLPs*

$$\min_{w,s} \quad \bar{f}(w, \theta) \tag{5.37a}$$

$$\text{s.t.} \quad \bar{g}(w, \theta) = 0, \qquad \bar{h}(w, \theta) - s = 0, \qquad s \geq 0. \tag{5.37b}$$

and

$$\min_{w,s} \quad \bar{f}(w, \theta) - \bar{\lambda}^\top \bar{g}(w, \theta) - \bar{\mu}^\top (\bar{h}(w, \theta) - s) \tag{5.38a}$$

$$\text{s.t.} \quad \bar{g}(w, \theta) = 0, \qquad \bar{h}(w, \theta) - s = 0, \qquad s \geq 0. \tag{5.38b}$$

For NLP (5.37)*, let us introduce the Lagrange multipliers associated with $\bar{g}(w, \theta) = 0$ as $\lambda^{(5.37)}(\theta)$, those associated with $\bar{h}(w, \theta) - s = 0$ as $\mu^{(5.37)}(\theta)$ and those associated with $s \geq 0$ as $\nu^{(5.37)}$. Equivalently, for NLP* (5.38)*,*

we introduce $\lambda^{(5.38)}(\theta)$, $\mu^{(5.38)}(\theta)$ *and* $\nu^{(5.38)}(\theta)$. *Then, both NLP* (5.37) *and* (5.38) *deliver the same primal solution for all* $\bar{\lambda}$, $\bar{\mu}$. *Moreover, it holds that* $\lambda^{(5.38)}(\theta) = \lambda^{(5.37)}(\theta) - \bar{\lambda}$, $\mu^{(5.38)}(\theta) = \mu^{(5.37)}(\theta) - \bar{\mu}$ *and* $\nu^{(5.38)}(\theta) = \nu^{(5.37)}(\theta)$.

Proof. As we only rotate equality constraints, Lemma 5.3 directly applies. □

The proposed slack reformulation of the economic NMPC Problem (5.1) has the nice property that, after rotation of the cost, none of the constraints contributes to the Hessian of the Lagrangian for an initial state $\hat{x}_0 = 0$. Note that the proposed rotation is different from the one proposed in both [32] and [3]. We are now ready to prove Theorem 5.23.

Proof (Theorem 5.23). One can always formulate Problem (5.1) using the proposed slack variable Formulation (5.38). Then, none of the inequality constraints contributes to the Lagrangian Hessian at steady state. The existence of the positive definite tracking formulation is then a direct consequence of Lemma 5.24 and Theorem 5.21. Indeed, one can rotate the economic NMPC NLP in the form (5.38) using $\tilde{\lambda} = \lambda_\mathrm{s}$ and $\tilde{\mu} = \mu_\mathrm{s}$. The corresponding locally equivalent linear MPC problem is obtained as Problem (5.35), which has a possibly indefinite stage cost. Note that the slack variables introduced by the formulation (5.38) can be treated as dummy controls which do not affect the system dynamics. Theorem 5.21 then states that there also exists a locally equivalent positive definite linear MPC Formulation (5.36). Moreover, as we use Formulation (5.38), the inequality constraints do not contribute to the Lagrangian Hessian, and thus one can directly formulate a tracking NMPC scheme (5.3) which has the same local behaviour as the economic NMPC problem.

Using Theorem 5.5 and Lemma 5.6 we obtain that

$$\underbrace{\frac{\partial w^{(5.35)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{Indefinite MPC}} = \underbrace{\frac{\partial w^{(5.1)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{EMPC}} \quad \text{and} \quad \underbrace{\frac{\partial w^{(5.3)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{PD tracking NMPC}} = \underbrace{\frac{\partial w^{(5.36)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{PD MPC}}.$$

Moreover, Theorem 5.18 proves that $w^{(5.36)} = w^{(5.35)}$ locally holds. Therefore

$$\underbrace{\frac{\partial w^{(5.36)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{PD MPC}} = \underbrace{\frac{\partial w^{(5.35)}}{\partial \hat{x}_0}\bigg|_{\hat{x}_0=0}}_{\text{Indefinite MPC}},$$

and we obtain the desired equivalence

$$\|u^\mathrm{t}(\hat{x}_0) - u^\mathrm{e}(\hat{x}_0)\| = O(\|\hat{x}_0\|^2).$$

□

Remark 5.25. *It is important to remark that, in order to formulate the tracking NMPC problem which locally approximates economic NMPC, we first rotate the stage cost of economic NMPC. This implies that, as stated in Lemma 5.2, the gradient of the rotated cost in the origin is given by:*

$$\left.\frac{\partial L(x_k,u_k)}{\partial w_k}\right|_{w=0}^{\top} = \left.\frac{\partial h(x_k,u_k)}{\partial w_k}\right|_{w=0}^{\top} \mu_{\mathrm{s}}.$$

Note that, because we formulate the economic NMPC Problem (5.1) *in its slack form, i.e. as Problem* (5.38)*, we have* $h(x_k,u_k) = s_k$ *where* s_k *are the slack variables added as fictitious controls in vector* u_k*.*

Remark 5.26. *Note that the proposed tuned tracking cost is a second-order approximation of the rotated cost of the economic MPC problem. Indeed, the tuned tracking cost is obtained by using a second-order approximation of the economic MPC cost rotated with a quadratic storage function.*

5.4.1 Numerical Example: Evaporation Process

In this section, we present a numerical example which illustrates the theoretical concepts presented in the previous sections. We remark that the scope of this section is not to assess how well a tuned tracking NMPC scheme can perform, but rather to show that the tuning procedure allows one to formulate an NMPC scheme whose closed loop behaviour is closer than that of a generic NMPC to the one obtained by the economic NMPC scheme, which also results in an improved performance. Clearly, the economic performance of the tuned tracking NMPC scheme can be extremely good for some applications, while for some other applications it can still be rather poor, as all derived results concern local properties. As opposed to the economic NMPC formulation, however, the tuned tracking NMPC formulation does always provide stability guarantees. Moreover, more efficient and faster numerical schemes are available for tracking formulations.

In order to compare the performance of the different MPC schemes, we use the following performance measure

$$G = \frac{P_{\mathrm{eco}} - P_{\mathrm{track}}}{\sum_{k=0}^{T-1} P_{\mathrm{s}}}, \tag{5.39}$$

where P_{eco} and P_{track} are the cumulative costs of economic and tracking NMPC respectively over T time steps and P_{s} is the steady-state optimal cost.

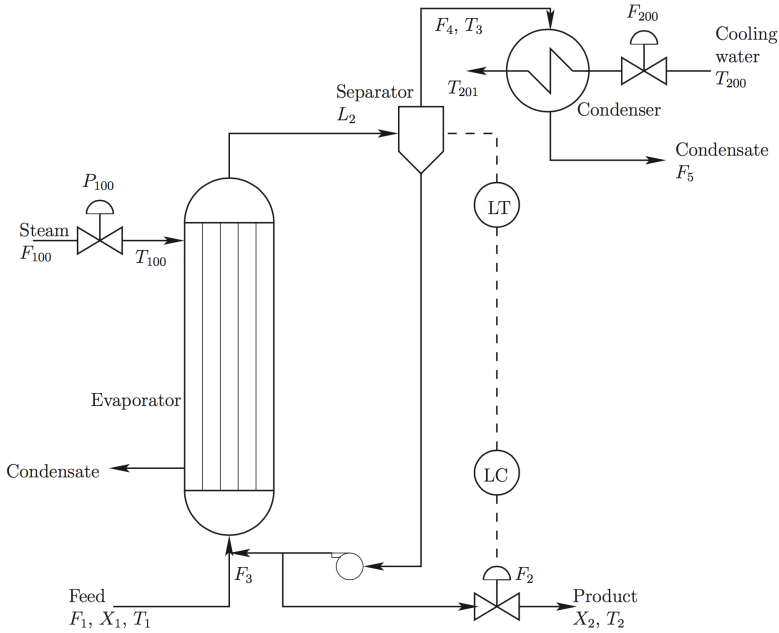


Figure 5.1: Evaporation process scheme (source [4]).

We consider an evaporation process that removes a volatile species from a solvent. Let us remark that the interest of this example is to demonstrate the tuning technique on a real system rather than an academic one. Therefore, we will not focus on the modelling and all model details can be found in [4] and references therein. The states are $x = (X_2, P_2)$, i.e. the product composition and operating pressure. The controls are $u = (P_{100}, F_{200})$, i.e. the steam pressure and cooling water flow rate. The model is depicted in Figure 5.1 and the equations are given by [4]

$$M\dot{X}_2 = F_1X_1 - F_2X_2, \qquad C\dot{P}_2 = F_4 - F_5, \tag{5.40}$$

where the liquid energy balance yields

$$T_2 = aP_2 + bX_2 + c, \quad T_3 = dP_2 + e, \quad \lambda F_4 = Q_{100} - F_1C_{\mathrm{p}}(T_2 - T_1),$$

a	b	c	d	e	f	g	h
0.5616	0.3126	48.43	0.507	55	0.1538	55	0.16
M	C	UA_2	C_p	λ	λ_s	F_1	X_1
20	4	6.84	0.07	38.5	36.6	10	5 %
F_3	T_1	T_{200}					
50	40	25					

Table 5.1: Model Parameters. The units are omitted and are consistent with the physical quantities they correspond to.

the heat stream in the jacket is modelled as

$$T_{100} = fP_{100} + g, \qquad Q_{100} = UA_1(T_{100} - T_2),$$

$$UA_1 = h(F_1 + F_3), \qquad F_{100} = \frac{Q_{100}}{\lambda_\mathrm{s}},$$

the condenser equations are

$$Q_{200} = \frac{UA_2(T_3 - T_{200})}{1 + UA_2/(2C_\mathrm{p}F_{200})}, \qquad \lambda F_5 = Q_{200},$$

and the level controller gives

$$F_2 = F_1 - F_4.$$

The model parameters are given in Table 5.1. The economic objective is given by

$$\ell(x, u) = 10.09(F_2 + F_3) + 600F_{100} + 0.6F_{200}.$$

The system is subject to the following constraints

$$X_2 \geq 25\,\%, \qquad 40\,\mathrm{kPa} \leq P_2 \leq 80\,\mathrm{kPa},$$

$$P_{100} \leq 400\,\mathrm{kPa}, \qquad F_{200} \leq 400\,\mathrm{kg/min}.$$

The optimal steady state is given by

$$x_\mathrm{s} = (25,\, 49.743), \qquad u_\mathrm{s} = (191.713,\, 215.888). \tag{5.41}$$

We consider a sampling time $t_\mathrm{s} = 1\,\mathrm{s}$ and formulate the NMPC scheme using a piecewise constant control parametrisation, a prediction horizon $N = 200$ and

the terminal point constraint $x_N = x_\mathrm{s}$. In order to avoid feasibility problems we reformulate the concentration constraint using a slack variable s (introduced as a fictitious control) so that the constraint reads $X_2 + s \geq 25\,\%$, $s \geq 0$, and the cost is given by

$$\ell(x,u) = 10.09(F_2 + F_3) + 600F_{100} + 0.6F_{200} + 10^3 s.$$

A purely economic NMPC scheme has been compared in a simulation scenario to a conventional tracking NMPC implementation, the proposed economically tuned NMPC scheme and a simplified version of the tuned NMPC scheme which uses a diagonal weighting matrix. On a 300 s long simulation, a pressure disturbance $\Delta P_2 = 1\,\mathrm{kPa}$ is applied to the system at time instants $t_0 = 0\,\mathrm{s}$, $t_1 = 20\,\mathrm{s}$, and $t_2 = 40\,\mathrm{s}$. For the tracking schemes we choose the following "normal" stage cost matrix (denoted by subscript n), we compute the following tuned stage cost matrix (denoted by subscript t) and its diagonal version (denoted by subscript td)

$$H_\mathrm{n} = \mathrm{diag}(\begin{bmatrix} 10 & 10 & 0.1 & 0.1 & 0.1 \end{bmatrix}),$$

$$H_\mathrm{t} = \begin{bmatrix} 6.96 \cdot 10^{0} & -7.42 \cdot 10^{-1} & 1.54 \cdot 10^{-1} & -9.55 \cdot 10^{-4} & 0 \\ -7.42 \cdot 10^{-1} & 1.23 \cdot 10^{-1} & -1.62 \cdot 10^{-2} & 6.86 \cdot 10^{-5} & 0 \\ 1.54 \cdot 10^{-1} & -1.62 \cdot 10^{-2} & 7.93 \cdot 10^{-3} & -2.10 \cdot 10^{-5} & 0 \\ -9.55 \cdot 10^{-4} & 6.86 \cdot 10^{-5} & -2.10 \cdot 10^{-5} & 4.53 \cdot 10^{-3} & 0 \\ 0 & 0 & 0 & 0 & 1.49 \cdot 10^{1} \end{bmatrix},$$

$$H_\mathrm{td} = \mathrm{Diag}(H_\mathrm{t}),$$

where the units of the weights are chosen consistently with the physical quantities they correspond to, so as to yield a dimensionless cost. We remark that, in order to be able to compute a convex tuned tracking stage cost, we needed to make use of the term $C_{\mathbb{A}_\mathrm{s}}^\top F C_{\mathbb{A}_\mathrm{s}}$. The operator diag constructs a diagonal matrix with the diagonal elements taken from the given vector and the operator Diag returns a matrix of the same size as the original with the diagonal elements are unchanged and all off-diagonal elements are set to 0.

The tracking cost has then been defined as

$$\ell_\bullet^\mathrm{t}(x,u) = \begin{bmatrix} x \\ u \end{bmatrix}^\top H_\bullet \begin{bmatrix} x \\ u \end{bmatrix} + \nabla_w \ell(x,u)\Big|_{w_\mathrm{s}}^\top \begin{bmatrix} x \\ u \end{bmatrix}, \qquad \bullet \in \{\mathrm{n}, \mathrm{t}, \mathrm{td}\},$$

where $w = \begin{bmatrix} x & u \end{bmatrix}^\top$. We remark that, as there are active constraints at steady state, the gradient of the cost at steady state can not be made equal to zero and needs to be included in the tracking cost formulation. For comparison, we also implemented a tracking NMPC formulation which uses the stage cost

matrix H_{n} and zero gradient, i.e.

$$\ell^{\mathrm{t}}_{\mathrm{n,ng}}(x,u) = \begin{bmatrix} x \\ u \end{bmatrix}^{\top} H_{\mathrm{n}} \begin{bmatrix} x \\ u \end{bmatrix}.$$

The state and control trajectories resulting from the closed loop simulations are displayed in Figure 5.2. It can be seen that the tuned tracking NMPC scheme is so close to the economic NMPC one that the trajectories are indistinguishable by eye inspection. The NMPC scheme formulated using the diagonal of the tuned stage cost matrix performs better than the normal though there is no a priori guarantee that this should be the case. The performance index for the three schemes reflects this situation:

$$G_{\mathrm{n}} = -3.2 \cdot 10^{-4}, \qquad G_{\mathrm{n,ng}} = -1.5 \cdot 10^{-3},$$

$$G_{\mathrm{td}} = -1.1 \cdot 10^{-4}, \qquad G_{\mathrm{t}} = -1.2 \cdot 10^{-7}.$$

The tuned NMPC scheme performs 3 orders of magnitude better than the two other tracking NMPC schemes. It is therefore clear that the off-diagonal elements can be very important in order to have an accurate approximation of the economic NMPC behaviour.

In order to confirm the theoretical result that $\|u^{\mathrm{t}}(\hat{x}_0) - u^{\mathrm{e}}(\hat{x}_0)\| = O(\|\hat{x}_0\|^2)$, we have perturbed the initial condition in a neighbourhood of the optimal steady state. The resulting control laws are displayed in Figure 5.3, where it can be seen that the tuned tracking nonlinear MPC scheme is the only tracking scheme for which it holds that $\frac{\partial u^{\mathrm{t}}(\hat{x}_0)}{\partial \hat{x}_0}\Big|_{\hat{x}_0 = x_{\mathrm{s}}} = \frac{\partial u^{\mathrm{e}}(\hat{x}_0)}{\partial \hat{x}_0}\Big|_{\hat{x}_0 = x_{\mathrm{s}}}$.

From Figure 5.3 we also deduce that the tracking NMPC scheme with zero gradient performs very differently from the others if the pressure P_2 drops. By running the same scenario with negative perturbations of the pressure, we obtain the following performance indices:

$$G_{\mathrm{n}} = -5.4 \cdot 10^{-6}, \qquad G_{\mathrm{n,ng}} = -1.6 \cdot 10^{-1},$$

$$G_{\mathrm{td}} = -1.1 \cdot 10^{-4}, \qquad G_{\mathrm{t}} = -1.2 \cdot 10^{-7}.$$

All the tracking NMPC schemes with the correct gradient perform very well, while the tracking NMPC scheme with zero gradient performs much worse than all others.

Finally, on a 300 s long simulation, a concentration disturbance $\Delta X_2 = 1\,\%$ is applied to the system at time instants $t_0 = 0\,\mathrm{s}$, $t_1 = 20\,\mathrm{s}$, and $t_2 = 40\,\mathrm{s}$. We

obtain the following performance indices:

$$G_{\mathrm{n}} = -2.6 \cdot 10^{-5}, \qquad G_{\mathrm{n,ng}} = -4.2 \cdot 10^{-2},$$

$$G_{\mathrm{td}} = -1.6 \cdot 10^{-5}, \qquad G_{\mathrm{t}} = -7.4 \cdot 10^{-9}.$$

Again, all tracking NMPC schemes which make use of the non-zero gradient at the origin perform well compared to the economic NMPC, while the one which has zero gradient at the origin is by far the worst. We remark that this last tracking NMPC scheme is the only one which does not bring the concentration X_2 back to its steady state value in one step.

5.5 Optimal Periodic Operation

The previous analysis focused on the steady-state case, however, the developed theory can be extended to the periodic case as well. In this section, we sketch how the theory can be extended.

5.5.1 Periodic Linear MPC

Let us consider p-periodic linear time-varying systems $x_{k+1} = A_k x_k + B_k u_k$, $A_{k+p} = A_k,\ B_{k+p} = B_k$ with time varying quadratic stage cost

$$\ell_k(x_k, u_k) = \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top \underbrace{\begin{bmatrix} Q_k & S_k^\top \\ S_k & R_k \end{bmatrix}}_{H_k} \begin{bmatrix} x_k \\ u_k \end{bmatrix}, \quad \ell_{k+p}(x_k, u_k) = \ell_k(x_k, u_k),$$

where w.l.o.g. the optimal trajectory is staying at the origin. Note that any system $\bar{x}_{k+1} = A_k \bar{x}_k + B_k \bar{u}_k + c_k$ having a periodic optimal trajectory $\bar{x}_k^{\mathrm{p}},\ \bar{u}_k^{\mathrm{p}}$ can be converted in this form by shifting the system dynamics using $x_k = \bar{x}_k - \bar{x}_k^{\mathrm{p}}$, $u_k = \bar{u}_k - \bar{u}_k^{\mathrm{p}}$ and noting that

$$\begin{aligned} x_{k+1} &= \bar{x}_{k+1} - \bar{x}_{k+1}^{\mathrm{p}} = A_k \bar{x}_k + B_k \bar{u}_k + c_k - A_k \bar{x}_k^{\mathrm{p}} - B_k \bar{u}_k^{\mathrm{p}} - c_k \\ &= A_k x_k + B_k u_k. \end{aligned}$$

The steady state case can be seen as a specific case of periodic system with $p = 1$.

We first consider the case of no active constraints at the optimal periodic trajectory. For each variable M, we define the p-periodic sequence $[M]^p =$

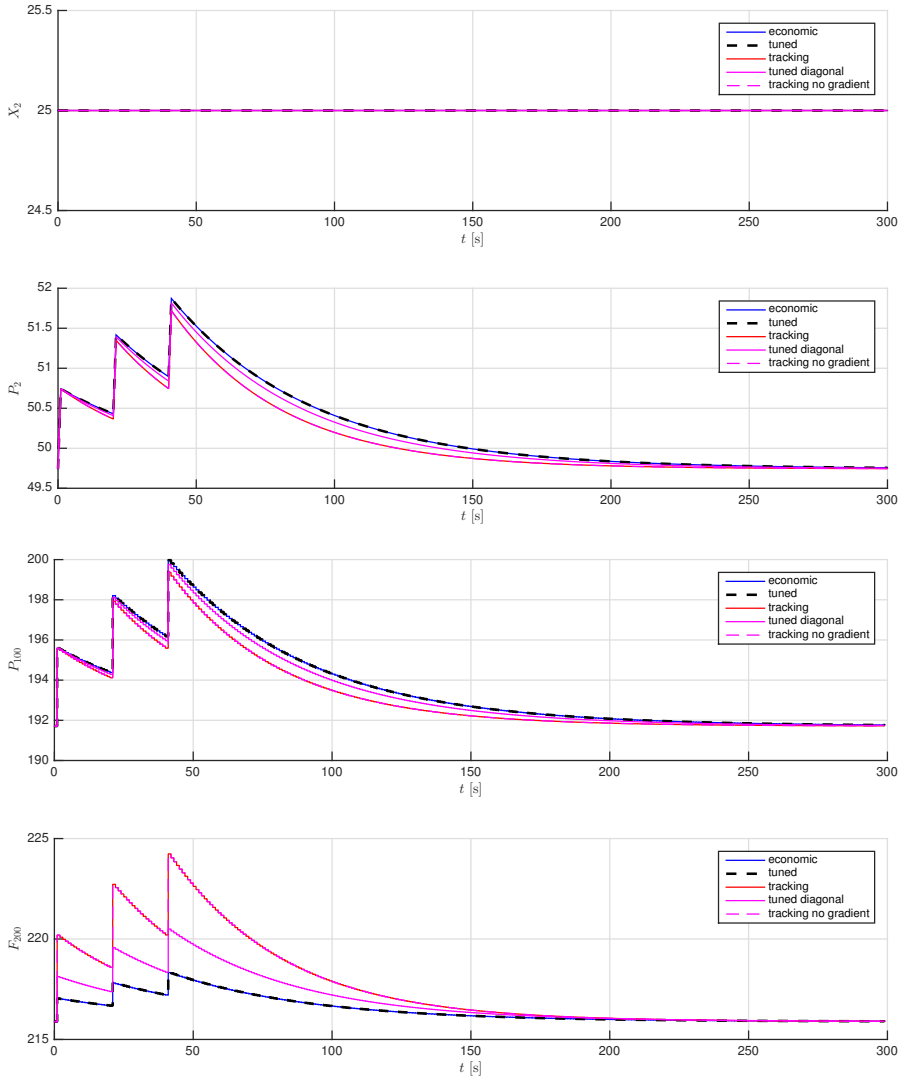


Figure 5.2: States and controls for the evaporation process. Blue = EMPC, dashed black = tuned tracking, red = normal tracking, magenta = tuned diagonal, dashed magenta = normal tracking with zero gradient.

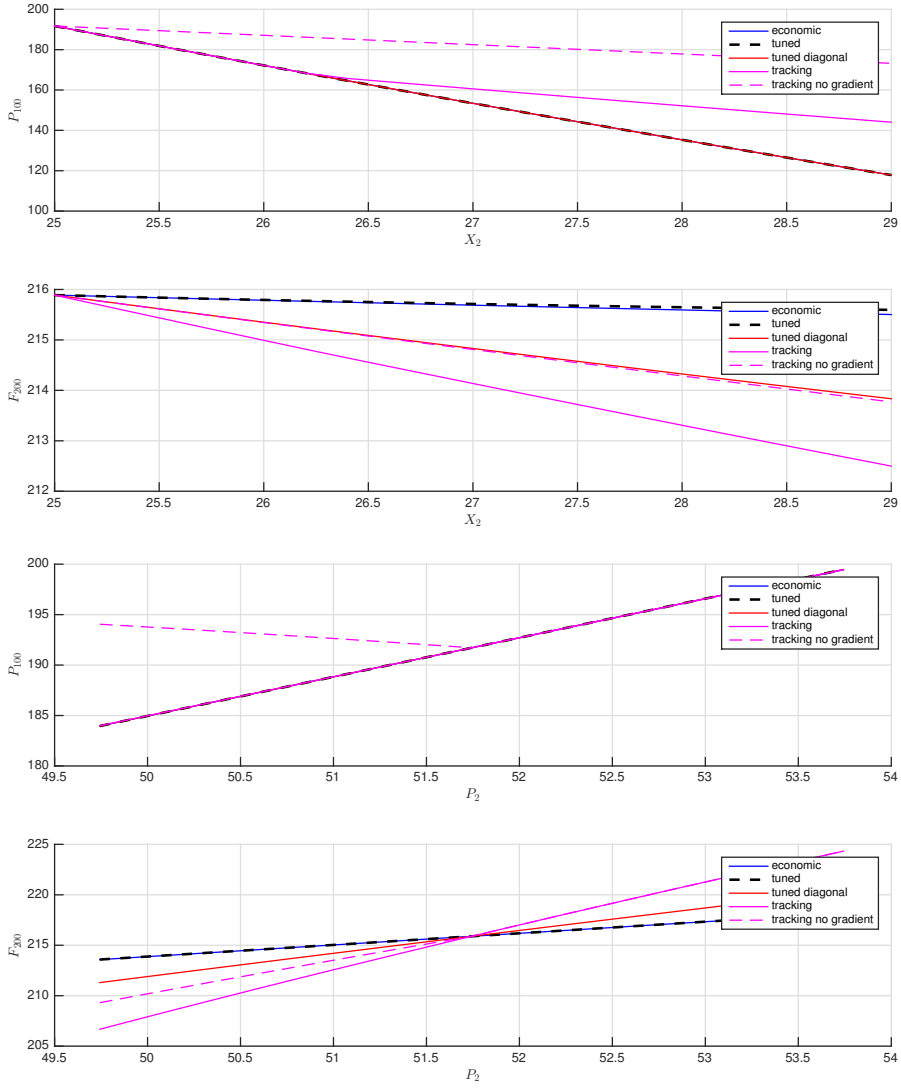


Figure 5.3: MPC control law depending on initial conditions close to the optimal steady state. In the top graph we have perturbed X_2, while in the bottom one we have perturbed P_2. In all graphs, the optimal steady state is displayed in dashed black line and the bounds in thick continuous black line. Blue = EMPC, dashed black = tuned tracking, red = tuned diagonal, magenta = normal tracking, dashed magenta = normal tracking with zero gradient.

$(M_0, \dots, M_{p-1})$ and we assume $M_{k+p} = M_k$ for all $k = \dots, -1, 0, 1, \dots$. Moreover, we define the periodic Riccati operator as

$$\begin{aligned}\mathcal{PR}([A]^p, [B]^p, [Q]^p, [R]^p, [S]^p) := \\ \{\ ([P]^p, [K]^p) \mid P_{k-1} = Q_{k-1} + A_{k-1}^\top P_k A_{k-1} - (S_{k-1}^\top + A_{k-1}^\top P_k B_{k-1}) K_{k-1}, \\ K_{k-1} = (R_{k-1} + B_{k-1}^\top P_k B_{k-1})^{-1} (S_{k-1} + B_{k-1}^\top P_k A_{k-1}), \\ P_k = P_{k+p},\ K_k = K_{k+p}.\ \}\end{aligned} \tag{5.42}$$

This defines the cost-to-go matrix P_k and the feedback matrix K_k at each stage k.

The results developed for the steady-state case can then be extended to the periodic setting as follows.

Lemma 5.27. *Let us assume that a stabilizing periodic LQR with feedback and cost-to-go matrices $[K]^p$ and $[P]^p$ exists for the periodic system (A_k, B_k) with periodic weighting matrices $[Q]^p$, $[R]^p$, $[S]^p$. Given any arbitrary p-periodic feedback matrix sequence $[\bar{K}]^p$, we define matrices $P_k^{\bar{K}^p}$ and $K_k^{\bar{K}^p}$ by the periodic recursion:*

$$\left(\left[P^{\bar{K}^p}\right]^p, \left[K^{\bar{K}^p}\right]^p\right) = \mathcal{PR}\left(\left[A^{\bar{K}^p}\right]^p, [B]^p, \left[Q^{\bar{K}^p}\right]^p, [R]^p, \left[S^{\bar{K}^p}\right]^p\right), \tag{5.43}$$

with $A_k^{\bar{K}^p} = A_k - B_k \bar{K}_k$, $Q_k^{\bar{K}^p} = Q_k - S_k^\top \bar{K}_k - \bar{K}_k^\top S_k + \bar{K}_k^\top R_k \bar{K}_k$, and $S_k^{\bar{K}^p} = S_k - R_k \bar{K}_k$. Then it holds that

$$P_k^{\bar{K}^p} = P_k, \qquad K_k^{\bar{K}^p} = K_k - \bar{K}_k.$$

Proof. The proof is analogous to the steady state case with the proper indices added and is omitted for the sake of brevity. We remark that the introduction of the feedback matrix sequence $\bar{K}^p$ can be seen as the change of variable $u_k = -\bar{K}_k x_k + v_k$, where v_k becomes the new control, so that the optimal feedback control must be given by $u_k^* = -\bar{K}_k x_k - K_k^{\bar{K}^p} x_k = -K_k x_k$, while the cost-to-go must remain unchanged. □

Lemma 5.28. *Let us assume that a stabilizing periodic LQR with feedback and cost-to-go matrices $[K]^p$ and $[P]^p$ exists for the periodic system (A_k, B_k) with periodic weighting matrices $[Q]^p$, $[R]^p$, $[S]^p$. Given any p-periodic sequence of real symmetric matrices $[\bar{P}]^p$, we define matrices $P_{k-1}^{\bar{P}^p}$ and $K_{k-1}^{\bar{P}^p}$ by the periodic recursion:*

$$\left(\left[P^{\bar{P}^p}\right]^p, \left[K^{\bar{P}^p}\right]^p\right) = \mathcal{PR}\left([A]^p, [B]^p, \left[Q^{\bar{P}^p}\right]^p, \left[R^{\bar{P}^p}\right]^p, \left[S^{\bar{P}^p}\right]^p\right), \tag{5.44}$$

with $Q_{k-1}^{\bar{P}^p} = Q_{k-1} + A_{k-1}^\top \bar{P}_k A_{k-1} - \bar{P}_k$, $S_{k-1}^{\bar{P}^p} = S_{k-1} + A_{k-1}^\top \bar{P}_k B_{k-1}$ *and* $R_{k-1}^{\bar{P}^p} = R_{k-1} + B_{k-1}^\top \bar{P}_k B_{k-1}$. *Then it holds that*

$$P_k^{\bar{P}^p} = P_k - \bar{P}_k, \qquad K_k^{\bar{P}^p} = K_k.$$

Proof. The proof is analogous to the steady state case with the proper indices added and is omitted for the sake of brevity. □

Theorem 5.29. *Let us assume that a stabilizing periodic LQR with feedback and cost-to-go matrices* $[K]^p$ *and* $[P]^p$ *exists for the periodic system* (A_k, B_k) *with periodic weighting matrices* $[Q]^p$, $[R]^p$, $[S]^p$. *Then, there exists a positive definite LQR formulation which yields the same feedback matrix sequence* $[K]^p$ *as the original problem. Consider, moreover, any stabilising linear MPC formulation which makes use of the same quadratic stage cost as the stabilising LQR, i.e. defined by matrices* $[Q]^p$, $[R]^p$, $[S]^p$, *and a quadratic terminal cost defined by matrix* $[P_N]^p$. *There exists a linear MPC formulation which makes use of positive definite stage and terminal cost matrices and which delivers the same feedback matrix sequence as the original indefinite linear MPC formulation.*

Proof. The proof is a straightforward adaptation of the steady state case. The first claim is an adaptation of Theorem 5.8 In particular, Lemma 5.9 and 5.11 are replaced by their time varying equivalents Lemma 5.27 and 5.28 respectively. Moreover, for periodic linear systems, the Lyapunov inequality becomes periodic and reads $A_k P_{k+1}^{\mathrm{L}} A_k - P_k^{\mathrm{L}} + Q_k^{\mathrm{L}} = 0$, with $P_0^{\mathrm{L}} = P_p^{\mathrm{L}}$ [19]. The rest of the proof then directly applies, with the proper time indices added. The second claim is a direct adaptation of Theorem 5.14 with the proper time indices added. □

In the case of active constraints at steady state, the following theorem holds.

Theorem 5.30. *Consider the following linear p-periodic (time-varying) MPC formulation at time i*

$$\min_{\substack{x_i,\ldots,x_{N+i} \\ u_i,\ldots,u_{N+i-1}}} \sum_{k=i}^{N+i} \frac{1}{2} \begin{bmatrix} x_k \\ u_k \end{bmatrix}^\top H_k \begin{bmatrix} x_k \\ u_k \end{bmatrix} + x_{N+i}^\top P_{N+i} x_{N+i} \tag{5.45a}$$

$$s.t.\ \ x_0 = \hat{x}_0, \tag{5.45b}$$

$$x_{k+1} = A_k x_k + B_k u_k, \quad k = i, \ldots, N+i-1, \tag{5.45c}$$

$$C_k \begin{bmatrix} x_k \\ u_k \end{bmatrix} \geq e_k, \quad k = i, \ldots, N+i-1. \tag{5.45d}$$

with $H_k \not\succ 0$. If Problem (5.45) is stabilising, then there exists an equivalent positive definite problem formulation which locally yields the same feedback matrix sequence $[K]^p$ as the original problem.

Proof. The proof is based on the same arguments used for the steady-state case Theorem 5.21 with the addition of the proper time indices.

Note that in the periodic case, also the Jacobian of the strictly active constraints at steady state $[C_{\mathbb{A}_\mathrm{s}}]^p$ and, therefore, the relative nullspace basis $[Z]^p$ is periodic. Let us define the periodic matrix sequence $[\delta P]^p$, and

$$\tilde{H}_k := H_k + \mathcal{H}_k([\delta P]^p), \tag{5.46a}$$

$$\mathcal{H}_k([\delta P]^p) := \begin{bmatrix} A_k^\top \delta P_{k+1} A_k - \delta P_k & A_k^\top \delta P_{k+1} B_k \\ B_k^\top \delta P_{k+1} A_k & B_k^\top \delta P_{k+1} B_k \end{bmatrix}, \tag{5.46b}$$

for $k = 0, \dots, p-1$. Then, analogously to the steady state case, it holds that

$$\exists\ \delta P^p \text{ s.t. } Z_k^\top \left(H_k + \mathcal{H}_k(\delta P^p)\right) Z_k \succ 0, \qquad k = 0, \dots, p-1. \tag{5.47}$$

Moreover, there exist matrices $F_k \succ 0$, such that $C_{k,\mathbb{A}_\mathrm{s}}(H_k + F_k)C_{k,\mathbb{A}_\mathrm{s}}^\top \succ 0$, where $C_{k,\mathbb{A}_\mathrm{s}}$ denotes the Jacobian of the strictly active constraints at time k. Then, it also holds that

$$\exists\ \delta P^p \text{ s.t. } H_k + \mathcal{H}_k(\delta P^p) + C_{k,\mathbb{A}_\mathrm{s}}^\top F_k C_{k,\mathbb{A}_\mathrm{s}} \succ 0, \quad k = 0, \dots, p-1. \tag{5.48}$$

Analogously to the steady state case, the open-loop prediction yielded by the MPC problem formulated using the new cost is equivalent to the open-loop prediction of the original MPC problem only as long as the constraints which are active on the periodic optimal trajectory remain active. □

The convexification procedure can be formulated analogously to the steady-state case with the addition of the proper time indices, i.e. as the following SDP

$$\min_{\substack{\alpha,\beta,\\ [\delta P]^p,[F]^p}} \quad \gamma\beta - \alpha + \rho \sum_{k=0}^{p-1} \|F_k\| \tag{5.49a}$$

$$\text{s.t.} \quad Z_k^\top (H_k + \mathcal{H}_k([\delta P]^p)) Z_k \succeq 0, \qquad k = 0, \dots, p-1, \tag{5.49b}$$

$$H_k + \mathcal{H}_k([\delta P]^p) + \eta C_{k,\mathbb{A}_\mathrm{s}}^\top F_k C_{k,\mathbb{A}_\mathrm{s}} \succeq \alpha I, \quad k = 0, \dots, p-1, \tag{5.49c}$$

$$\beta I \succeq H_k + \mathcal{H}_k([\delta P]^p), \qquad k = 0, \dots, p-1, \tag{5.49d}$$

with $\gamma \geq 0$, $\rho \geq 0$ two tuning parameters and $\|\cdot\|$ and appropriately chosen matrix norm. As for the steady-state case, parameter $\eta \in \{0, 1\}$ is used to construct a 2-step procedure. First, SDP (5.49) is solved using $\eta = 0$, $\rho > 0$, which results in $F_k = 0$. If the solution satisfies $\alpha^* > 0$, we have found the stage cost matrix for a positive definite reformulation of the original problem which is globally equivalent to it. If even with $\gamma = 0$ we obtain $\alpha^* < 0$, we set $\eta = 1$ and solve the SDP again. In this second case the equivalence of the positive definite and original problems will only be local.

Periodic Nonlinear MPC

The nonlinear case is again a direct adaptation of the theory developed for the steady-state case and is covered by the following theorem.

Theorem 5.31. *Given any stabilising periodic economic NMPC satisfying the assumptions of Theorem 5.5, it is possible to formulate both a periodic tracking linear MPC and a periodic tracking nonlinear MPC with positive definite stage cost which are locally first-order equivalent to the economic NMPC problem* (5.1).

Proof. The proof follows the same arguments used in the proof of the steady-state case, i.e. Theorem 5.23, with the proper time indices added. □

5.5.2 Numerical Example: Periodic MPC for a Unicycle Model

We consider the following unicycle model

$$\begin{bmatrix} \dot{z} \\ \dot{y} \\ \dot{e}_z \\ \dot{e}_y \end{bmatrix} = \begin{bmatrix} ve_z \\ ve_y \\ -ue_y - \rho e_z(e_z^2 + e_y^2 - 1) \\ ue_z - \rho e_y(e_z^2 + e_y^2 - 1) \end{bmatrix}, \tag{5.50}$$

with $v = 1$ the velocity of the vehicle and $\rho > 0$ a parameter. Note that this model was previously presented by representing the orientation using an angle. In this section instead, we define the orientation of the vehicle by the unitary vector e, and its components along the z and y axes e_z and e_y respectively as states. Quantities z and y indicate the position of the vehicle on the z and y axes. We chose to represent the orientation using $e_z,\ e_y$ in order to allow for truly periodic orbits even when the vehicle keeps turning.

This modelling choice is explained in Chapter 6. As it involves the use of nonminimal coordinates, the system dynamics preserve the invariant $e_z^2 + e_y^2 =$

1. In order to both compensate for numerical errors and make the system stabilisable, we stabilise the invariant by adding to the dynamics the terms containing ρ. Note that, without these terms, the uncontrollable dynamics of the invariant would only be marginally stable and the existence of a periodic LQR at the optimal solution could not be guaranteed. Also for this reason, we stabilise the invariant by choosing $\rho = 0.001$. For more details on invariant stabilisation, we refer to Section 6.3 and [51, 52, 11].

We use the time-invariant stage cost $\ell(x, u) = u^2 + z^2 + 5y^2$, consider a horizon $T = 5$ s and discretise the system dynamics using 50 steps of a Runge-Kutta integrator of order 4 over a sampling time $T_s = T/30$ with a piecewise constant control parametrisation. We convexify the stage cost by linearising the system dynamics at the optimal periodic orbit and solving SDP (5.49).

In order to compare the performance of the different MPC schemes, we use the following performance measure

$$G = \frac{P_{\text{eco}} - P_{\text{track}}}{\sum_{k=0}^{T-1} P_k^{\text{p}}}, \tag{5.51}$$

where P_{eco} and P_{track} are the cumulative costs of economic and tracking MPC respectively over T time steps and P_k^{p} is the periodic optimal cost.

We consider a simulation scenario over a time horizon $T = 41.66$ s, in which we perturb state z with perturbations $\Delta z_0 = \Delta z_1 = 0.1$ m, $\Delta z_2 = \Delta z_3 = 0.5$ m, at times $t_0 = 0$ s, $t_1 = 5.83$ s, $t_2 = 11.67$ s, $t_3 = 17.5$ s respectively. We control the system using economic NMPC, standard tracking NMPC and the tuned tracking NMPC and compare the obtained closed-loop trajectories, which are displayed in Figure 5.4. It can be seen that the behaviour of tuned tracking MPC scheme is closer to that of the economic MPC than that of the normal MPC scheme. However, the behaviour of both tracking schemes is rather close to that of the economic MPC. This is also reflected in the performance coefficients $G_{\text{tuned}} = -6.2 \cdot 10^{-4}$ and $G_{\text{tracking}} = -3.6 \cdot 10^{-3}$.

The stage cost deviation from the periodic optimal cost, i.e. $\ell(x_k, u_k) - \ell(x_k^{\text{p}}, u_k^{\text{p}})$, is displayed in Figure 5.5. It can be seen that, for the small perturbations occurring at time instants $t_0 = 0\,\text{s}$ and $t_1 = 4\,\text{s}$, the closed-loop behaviour of the tuned tracking MPC formulation has a cost which is extremely close to that of the economic MPC formulation. Instead, for the larger perturbations occurring at time instants $t_2 = 8\,\text{s}$ and $t_3 = 12\,\text{s}$, the tuned tracking MPC behaviour deviates more from the economic MPC one, but it is still closer to it than that of the normal MPC.

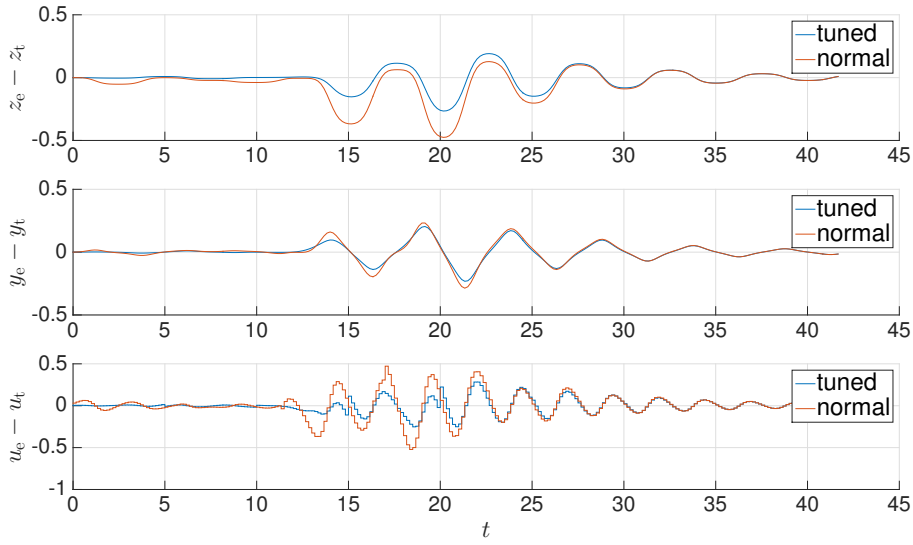


Figure 5.4: Difference between the tracking and the economic NMPC schemes.

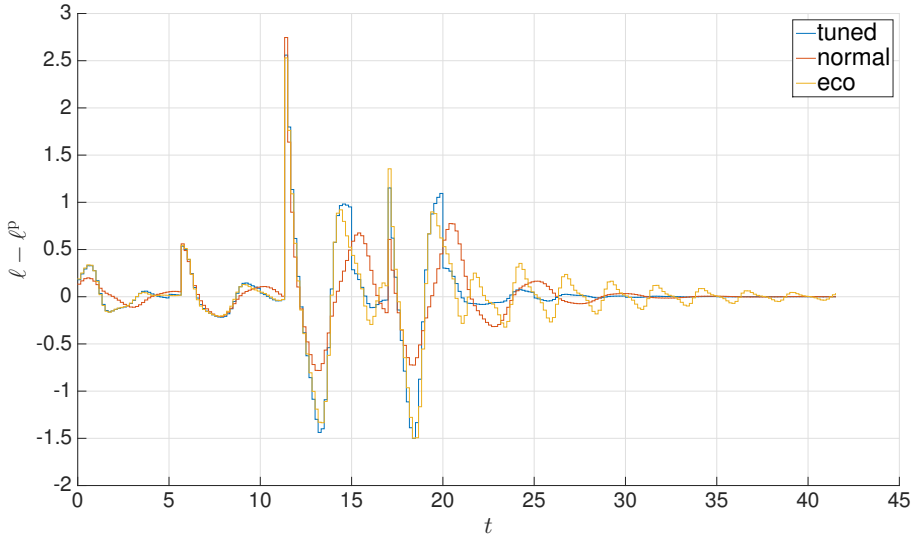


Figure 5.5: Stage cost deviation from the periodic optimal stage cost.

5.6 Conclusions

In this chapter we have developed a tuning strategy for tracking MPC based on economic criteria. We have proven that, whenever economic MPC is stabilising for the set of active constraints at steady state, then it is possible to tune tracking MPC with a quadratic stage cost such that it yields a control law which approximates that yielded by economic MPC for all initial conditions for which all constraints active at steady state remain active.

We have proposed a strategy to compute the stage cost matrices such that their condition number is approximately minimised in order to avoid ill conditioned stage costs.

We have extended the framework developed for the steady state case in order to include the case of optimal periodic operation. The obtained tracking MPC formulation has in this case a periodically time-varying stage cost.

We have illustrated the theoretical developments using two numerical examples. The comparison with standard tracking MPC formulations illustrates the advantage of using our tuning procedure in terms of closed-loop cost. Moreover, we also illustrate that the control law obtained by using our tuned tracking MPC is a first-order approximation of the one obtained by economic MPC.

In the examples presented in this thesis, the tuning strategy performs well in terms of closed-loop cost when compared to economic MPC. Clearly, this tuning approach can be expected to deliver a worse performance especially when the system or the stage cost are strongly nonlinear or when perturbations become large. In particular for the periodic case, tracking MPC formulations rely on a fixed phase of the reference trajectory: for large perturbations a different phase might become optimal, however, while economic MPC will automatically adapt, tracking MPC will force the system to track the suboptimal phase. Especially this last aspect deserves further studies and future research will aim at further studying the potential and the limitations of the tuning approach.

In Chapter 7, we will apply the theory developed in this chapter to tethered airplanes for energy harvesting.

Chapter 6

Modelling of Mechanical Systems aimed at Optimal Control

When formulating an optimal control problem, particular attention needs to be dedicated to the mathematical model of the dynamic system. Often, the same system can be modelled in many different ways, each of which has different advantages. As nonlinearities in the model translate into nonconvexity in the NLP, a strong effort should be put into avoiding or at least reducing nonlinearities whenever possible.

In order to be able to treat complicated mechanical systems, characterised by fast highly nonlinear constrained dynamics, efficient models and reliable modelling procedures are fundamental.

Often, modelling a system using a minimal set of coordinates appears appealing because it seems more intuitive or easier to derive. However, minimal coordinates often have severe disadvantages, e.g. singularities might be present in the system dynamics, unnecessarily strong nonlinearities might be introduced and large expression trees might be created, which make the simulation of the system dynamics slow. It has been observed that often the model efficiency and simplicity can be significantly improved by using a multi-body approach [106], where each body is modelled separately and linked to the other bodies through algebraic equations. This often leads to non-minimal coordinates models, which, in the case of mechanical systems, are often index-3 DAEs, where the algebraic equations constrain the model dynamics to evolve according to the

links between the bodies. It has been shown in [47] that for tethered airplanes minimal coordinates yield a model which has an evaluation tree with 2.9 times more nodes and can be evaluated in a time 1.74 times longer than the one obtained by using nonminimal coordinates. This difference increases when considering the model Jacobian and Hessian, in which the ratios become 6.6 times for the nodes and 3.4 times for the evaluation of the Jacobian and 6.7 times for the nodes and 6.1 times for the evaluation of the Hessian.

In the context of optimal control with high-index DAEs, index reduction techniques are often used in order to reduce the DAE to index 1 by differentiating the constraints in time [105]. One of the main reasons for performing index reduction is due to the existence of efficient numerical methods for optimal control of index 1 DAEs. While some numerical methods have been developed for the simulation of high index DAEs, to the author's knowledge, performing index reduction and handling index 1 DAEs is today still the most efficient approach for optimal control applications. Index-reduced DAEs typically retain a low numerical complexity. One disadvantage resulting from index reduction is the drift of the invariants in long horizon simulations / optimal control problems. In order to tackle this issue, computationally inexpensive remedies have been developed [105].

The $SO(3)$ special orthogonal Lie group is used to represent the orientation of a body in a 3-dimensional space. It can be parametrised in many different ways, the most popular of which are (a) Euler angles, (b) quaternions and (c) the Direct Cosine Matrix (DCM) representation [64]. These three parametrisations distinguish themselves for the number of parameters needed, the degree of nonlinearity of the rotational dynamics and the fact that representation (a) can incur singularities, while (b) and (c) are always non-singular.

In this chapter, we propose a modelling approach for mechanical systems, with a particular attention to multi-body systems, e.g. tethered airplanes. We introduce the Euler-Lagrange modelling approach in Section 6.1. The modelling of the rotational dynamics are discussed in Section 6.2. Section 6.3 explains how to apply a Baumgarte stabilisation of the invariants to the proposed modelling framework.

Acknowledgement The first part of this chapter introduces state-of-the art modelling techniques. Sections 6.2 and 6.3 are largely based on [52, 51]. Sébastien Gros is the main author of [52] and developed the theoretical results on the direct cosine matrix therein under the supervision of Moritz Diehl. Mario Zanon contributed some numerical results, most theoretical developments on quaternions and served as a discussion partner. Sébastien Gros and Mario Zanon are the main authors of [51] and worked under the supervision of Moritz

Diehl. Sébastien Gros contributed most of the theoretical results and Mario Zanon contributed most of the simulations.

6.1 The Euler-Lagrange Approach

A popular modelling approach derives the equations of motion using the Euler-Lagrange equation. The first step consists in defining the set of parameters which is needed to describe the system. These parameters must uniquely define any configuration of the system and are known as the *generalised coordinates*, usually denoted by q. It is important to remark that the choice of generalised coordinates to be used for modelling a system is not unique and the mathematical properties of the model strongly depend on this choice. The number of degrees of freedom of a system is the minimum number of generalised coordinates which uniquely defines its configuration.

Having defined the generalised coordinates, the *Lagrangian* [1] is defined as

$$\mathcal{L}(\dot{q}, q) = \mathcal{T}(\dot{q}, q) - \mathcal{V}(q) - \lambda^\top \mathcal{C}(q), \tag{6.1}$$

where $\mathcal{T}(\dot{q}, q)$ and $\mathcal{V}(q)$ denote respectively the kinetic and the potential energy of the system. The system dynamics might be constrained by $\mathcal{C}(q)$, which enters the Lagrangian $\mathcal{L}$ in a scalar product with the algebraic variable λ. Note that in general the constraint can also depend on the generalised velocities, i.e. $\mathcal{C}(\dot{q}, q)$, however, for the applications we are interested in, the velocities never appear in the constraint. In the following, for the sake of simplicity, we will omit the arguments and write (6.1) as

$$\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda^\top \mathcal{C}, \tag{6.2}$$

The system dynamics can then be obtained by writing the *Euler-Lagrange* equations, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}}\right) - \frac{\partial \mathcal{L}}{\partial q} = F^q, \tag{6.3}$$

where F^q denotes the generalised forces. The generalised forces are defined by the *virtual work* principle: any infinitesimal displacement δq of the system configuration yields the infinitesimal work

$$\delta W = \langle \delta q, F^q \rangle, \tag{6.4}$$

[1] In the context of the Euler-Lagrange approach the Lagrangian is at first sight not the Lagrangian of an optimisation problem. Though there exists a clear connection between Lagrange mechanics and optimisation, one can think of function $\mathcal{L}$ without necessarily having in mind an optimisation problem.

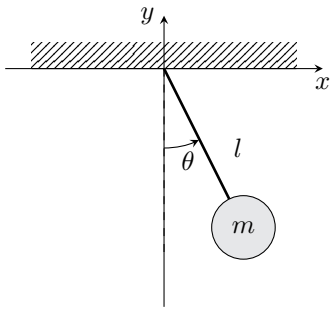


Figure 6.1: Schematics of a pendulum.

where $\langle \cdot, \cdot \rangle$ is the scalar product.

It is interesting to note that the Euler-Lagrange equations are Newton's equations projected on the space of the generalised coordinates [85].

6.1.1 A Simple Guiding Example: the Pendulum

The pendulum has been widely used as an example in many different contexts due to its simplicity. We will also use it in order to illustrate how to derive the equations of motion using the Euler-Lagrange approach. As pointed out before, several choices are possible for the generalised coordinates. We will investigate two choices in particular: (a) choosing $q_{\mathrm{a}} = \theta$ and (b) choosing $q_{\mathrm{b}} = \begin{bmatrix} x & y \end{bmatrix}^\top$.

Polar Coordinates: Let us consider case (a) first. The kinetic and potential energy of the system are given by

$$\mathcal{T}(\dot{q}_{\mathrm{a}}) = \frac{1}{2} m l^2 \dot{\theta}^2, \qquad \mathcal{V}(q_{\mathrm{a}}) = -mgl\cos(\theta).$$

Using

$$\mathcal{L} = \mathcal{T} - \mathcal{V} = \frac{1}{2} m l^2 \dot{\theta}^2 + mgl\cos(\theta)$$

in the Euler-Lagrange equations, we obtain the equations of motion $ml^2\ddot{\theta} + mgl\sin(\theta) = 0$, which reduce to the well-known form

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta).$$

Cartesian Coordinates: Let us now consider case (b). The kinetic, potential energy and the constraint of the system are given by

$$\mathcal{T}(\dot{q}_\mathrm{b}) = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2), \qquad \mathcal{V}(q_\mathrm{b}) = -mgy, \qquad \mathcal{C}(q_\mathrm{b}) = \frac{1}{2}\left(x^2 + y^2 - l^2\right).$$

Using

$$\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda^\top \mathcal{C} = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + mgy - \frac{1}{2}\lambda\left(x^2 + y^2 - l^2\right)$$

in the Euler-Lagrange equations, we obtain the equations of motion

$$\begin{bmatrix} m\ddot{x} \\ m\ddot{y} \\ 0 \end{bmatrix} - \begin{bmatrix} -\lambda x \\ -mg - \lambda y \\ \frac{1}{2}\left(x^2 + y^2 - l^2\right) \end{bmatrix} = 0.$$

As opposed to case (a), in which one obtains an ODE, in case (b) one obtains an index-3 DAE. While numerical methods for solving index-3 DAEs are available, it can be preferable to use index reduction techniques to obtain an index-1 DAE or an ODE. By differentiating the constraint equation one gets

$$\dot{\mathcal{C}} = x\dot{x} + y\dot{y}, \qquad\qquad \ddot{\mathcal{C}} = x\ddot{x} + y\ddot{y} + \dot{x}^2 + \dot{y}^2,$$

and the index-1 DAE in semi-implicit form reads

$$\underbrace{\begin{bmatrix} m & 0 & x \\ 0 & m & y \\ x & y & 0 \end{bmatrix}}_{M} \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ mg \\ -\dot{x}^2 - \dot{y}^2 \end{bmatrix}.$$

The ODE form can be obtained by inverting the so-called mass matrix M and using $x^2 + y^2 = l^2$:

$$\ddot{x} = -\frac{x(\dot{x}^2 + \dot{y}^2 + gy)}{l^2},$$

$$\ddot{y} = -\frac{y(\dot{x}^2 + \dot{y}^2) - gx^2}{l^2},$$

$$\lambda = \frac{m(\dot{x}^2 + \dot{y}^2 + gy)}{l^2}.$$

Note that the Lagrange Multiplier does have a physical meaning, as the tension in the pendulum rod is given by $F_\mathrm{T} = \lambda l$.

Constrained Polar Coordinates In this paragraph we aim at further clarifying why in case (a) we directly obtain an ODE while in case (b) we obtain an index-3 DAE which we then reduce to an index-1 DAE or to an ODE. Indeed, in case (a) we chose to only use one generalised coordinate, even though the system evolves in a 2-dimensional space. An alternative modelling choice (c) could consist in using $q_\mathrm{c} = \begin{bmatrix} \theta & r \end{bmatrix}^\top$. Then the kinetic, potential energy and the constraint of the system read

$$\mathcal{T}(\dot{q}_\mathrm{c}) = \frac{1}{2} m r^2 \dot{\theta}^2 + \frac{1}{2} m \dot{r}^2, \qquad \mathcal{V}(q_\mathrm{c}) = -mgr\cos(\theta), \qquad \mathcal{C}(q_\mathrm{c}) = r - l.$$

Using

$$\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda^\top \mathcal{C} = \frac{1}{2} m r^2 \dot{\theta}^2 + \frac{1}{2} m \dot{r}^2 + mgr\cos(\theta) - \lambda(r - l)$$

in the Euler-Lagrange equations, we obtain the equations of motion

$$\begin{bmatrix} mr^2\ddot{\theta} \\ m\ddot{r} \\ 0 \end{bmatrix} - \begin{bmatrix} -2mr\dot{r}\dot{\theta} - mgr\sin(\theta) \\ mr\dot{\theta}^2 + mg\cos(\theta) - \lambda \\ r - l \end{bmatrix} = 0, \tag{6.5}$$

which is again an index-3 DAE. However, by differentiating the constraint in order to perform index reduction one obtains

$$\dot{\mathcal{C}} = \dot{r}, \qquad\qquad \ddot{\mathcal{C}} = \ddot{r}, \tag{6.6}$$

which can be used in order to simplify the equations and obtain

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta), \qquad\qquad \lambda = mg\cos(\theta) + ml\dot{\theta}^2,$$

where we used (6.6) and we used the third equation of (6.5) to replace $r = l$ in the first two equations. As expected, the same equations of motion as in case (a) are obtained. Note that, in this case, the tension in the pendulum rod is given by $F_\mathrm{T} = \lambda$.

6.2 Modelling the Rotational Dynamics

The rotational dynamics of a free body with rotational velocity ω and moment of inertia J are given by

$$J\dot{\omega} = T - \omega \times J\omega = T - \omega_\times J\omega, \tag{6.7}$$

with $\omega_\times := \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$ and T the torques applied to the body given in the reference frame attached to the body.

Equations (6.7) only describe the relation between torques and rotational accelerations. In order to obtain the dynamics of the body orientation, we need to integrate the rotational velocities consistently with the chosen representation of the body orientation.

Given two reference frames E and e in $\mathbb{R}^3$, we define the rotation matrix as $R = \begin{bmatrix} e_x & e_y & e_z \end{bmatrix}$, where e_a, $a \in \{x, y, z\}$ is a unit vector describing the orientation of axis a in frame E. Note that the rotation matrix can be interpreted in several ways. In our definition, Rv transforms the coordinates of vector v from frame e to frame E. The rotation of a given vector v_1 into vector v_2 can be described as

$$v_2 = R^\top v_1.$$

A composition of two subsequent rotations R_1 and R_2 is given by $R = R_2 R_1$, as it can easily be verified. In the following, we present three popular parametrisations of the SO(3) rotation group.

Euler Angles A very popular representation of the rotation is given by the Euler angles: three subsequent rotations around given axes are used in order to define the rotation matrix R:

$$R = R_a(\theta) R_b(\phi) R_c(\psi),$$

where $a, b, c \in \{x, y, z\}$ define the axis of rotation and the rotations around the axes x, y, z are defined as

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix},$$

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix},$$

$$R_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

As opposed to other formalisms, Euler angles have an intuitive definition, as they represent 3 subsequent rotations. This makes it easy to interpret the effect

of small rotations, as it can be seen by writing the infinitesimal rotation around $\theta, \phi, \psi = 0$, i.e.

$$\delta R = \delta R_x(\delta\theta)\delta R_y(\delta\phi)\delta R_z(\delta\psi) = \begin{bmatrix} 0 & -\delta\psi & \delta\phi \\ \delta\psi & 0 & -\delta\theta \\ -\delta\phi & \delta\theta & 0 \end{bmatrix}.$$

Note that, for infinitesimal rotations, the subsequent rotations commute, therefore the order in which they happen is irrelevant.

Because they are simple to manipulate, especially in the context of linear control theory, Euler angles are used for many applications. However, Euler angles suffer from a major defect: for any choice of subsequent rotations there always exists a configuration in which a singularity arises. This difficulty is often referred to as the gimbal lock. It can be proven that no minimal parametrisation of SO(3) exists which does not incur in singularities. In order to avoid that problem, higher-dimensional representations need to be used. The two most commonly used higher-dimensional representations of the SO(3) rotation group are the unit quaternion and the Direct-Cosine Matrix (DCM).

The Euler angle dynamics are given by

$$\begin{bmatrix} \dot{\theta} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = V(\theta, \phi, \psi)\ \omega, \tag{6.8}$$

where $V(\theta, \phi, \psi)$ is a nonlinear matrix function of the three angles and depends on the chosen sequence of rotations. It is important to underline that, for any choice of rotations, function $V(\theta, \phi, \psi) \neq I$ depends on the angles through trigonometric functions and can become singular for some configurations.

Quaternion Quaternions are unit vectors in a four-dimensional vector space which allow one to describe the SO(3) rotation group. The SO(3) rotation group is then covered by $\mathfrak{q} \in \left\{\mathfrak{q} \in \mathbb{R}^4 \,|\, \mathfrak{q}^\top \mathfrak{q} = 1\right\}$ with [108]:

$$R(\mathfrak{q}) = E(\mathfrak{q})G(\mathfrak{q})^\top \in \mathrm{SO}(3),$$

$$G(\mathfrak{q}) = \begin{bmatrix} -\mathfrak{q}_1 & \mathfrak{q}_0 & \mathfrak{q}_3 & -\mathfrak{q}_2 \\ -\mathfrak{q}_2 & -\mathfrak{q}_3 & \mathfrak{q}_0 & \mathfrak{q}_1 \\ -\mathfrak{q}_3 & \mathfrak{q}_2 & -\mathfrak{q}_1 & \mathfrak{q}_0 \end{bmatrix},$$

$$E(\mathfrak{q}) = \begin{bmatrix} -\mathfrak{q}_1 & \mathfrak{q}_0 & -\mathfrak{q}_3 & \mathfrak{q}_2 \\ -\mathfrak{q}_2 & \mathfrak{q}_3 & \mathfrak{q}_0 & -\mathfrak{q}_1 \\ -\mathfrak{q}_3 & -\mathfrak{q}_2 & \mathfrak{q}_1 & \mathfrak{q}_0 \end{bmatrix}.$$

The quaternion dynamics are given by

$$\dot{\mathfrak{q}} = \frac{1}{2} G(\mathfrak{q})^\top \omega, \tag{6.9}$$

where $\omega \in \mathbb{R}^3$ is the angular velocity in the reference frame attached to the rotating body. We remark that the quaternion dynamics (6.9) have an invariant given by

$$\mathfrak{q}^\top \mathfrak{q} - 1 = 0. \tag{6.10}$$

Note that quaternions can be interpreted geometrically as a rotation of an angle θ around a prescribed axis of rotation defined by the unitary vector e_r [108], i.e.

$$\mathfrak{q} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ e_\mathrm{r} \sin\left(\frac{\theta}{2}\right) \end{bmatrix}.$$

Note also that $\mathfrak{q}$ and $-\mathfrak{q}$ yield the same rotation, i.e. $R(\mathfrak{q}) = R(-\mathfrak{q})$.

Direction Cosine Matrix The Direction Cosine Matrix (DCM) representation considers each entry of the rotation matrix R as a parameter, i.e. it parametrises SO(3) using 9 parameters. The rotation dynamics are given by

$$\dot{R} = R\omega_\times. \tag{6.11}$$

We remark that the DCM dynamics (6.11) have an invariant given by

$$R^\top R - I = 0. \tag{6.12}$$

6.2.1 Lagrangian Formalism and Rotations

When the rotational and translational dynamics of the system are decoupled, one can model them separately and the rotational dynamics are given by

$$J\dot{\omega} + \omega \times J\omega = T$$

The dynamics of the Euler angles, quaternions or DCM are then obtained by adequately integrating in time the rotational velocity.

However, in case the translational and rotational dynamics are coupled, the system dynamics need to be derived together using the generalised coordinates $q = \begin{bmatrix} q_\mathrm{t} & q_\mathrm{r} \end{bmatrix}^\top$, where q_t and q_r denote the generalised coordinates relative to the translation and the rotation respectively. In general, the coordinates relative

to the translation define the body position in $\mathbb{R}^3$, e.g. $q_{\mathrm{t}} = p = \begin{bmatrix} x & y & z \end{bmatrix}^\top$. The coordinates relative to the rotation can be e.g. a quaternion $q_{\mathrm{r}} = \mathfrak{q}$ or a DCM $q_{\mathrm{r}} = R$.

The Lagrangian of a rigid body subject to constraints is then given by

$$\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda^\top \mathcal{C} + \langle Z, \mathcal{C}_{\mathrm{r}} \rangle,$$

where $\langle Z, \mathcal{C}_{\mathrm{r}} \rangle$ is given by $Z(\mathfrak{q}^\top \mathfrak{q} - 1)$ and $\mathrm{tr}(Z(R^\top R - I))$ for the quaternion and DCM respectively. The Euler-Lagrange equations are given by

$$\frac{\mathrm{d}}{\mathrm{d}t} \nabla_{\dot{q}_{\mathrm{t}}} \mathcal{L} - \nabla_{q_{\mathrm{t}}} \mathcal{L} = F^{q_{\mathrm{t}}}, \quad \frac{\mathrm{d}}{\mathrm{d}t} \nabla_{\dot{q}_{\mathrm{r}}} \mathcal{L} - \nabla_{q_{\mathrm{r}}} \mathcal{L} = F^{q_{\mathrm{r}}}, \quad C = 0, \quad C_{\mathrm{r}} = 0. \tag{6.13}$$

These equations define the dynamics of $\ddot{q}$ by a possibly implicit DAE $f(\ddot{q}, \dot{q}, q, \lambda, Z, F^q) = 0$ whose index is typically reduced and for which the differential part is reduced to its first-order representation, i.e.

$$\dot{q} = v_q, \qquad f(\dot{v}_q, v_q, q, \lambda, Z, F^q) = 0, \qquad \ddot{C} = 0, \qquad C_{\mathrm{r}} = 0.$$

Note that, using the quaternion or the DCM parametrisation, the equations for the rotational part are 1×4 and 3×3 differential equations defining $\ddot{\mathfrak{q}}$ and $\ddot{R}$ respectively. Clearly, the principle of virtual work can be applied in order to compute the generalised forces $F^{q_{\mathrm{r}}}$. However, constraint $C_{\mathrm{r}} = 0$ constrain the accelerations $\ddot{\mathfrak{q}}$ or $\ddot{R}$ to be tangential to the quaternion and DCM manifolds respectively. At the same time, the generalised forces $F^{q_{\mathrm{r}}}$ relative to the rotational dynamics arise from the torques applied to the body and therefore by definition only act in the directions which are tangential to the quaternion or DCM manifolds. The system dynamics can then be projected on the appropriate manifold, so as to reduce the number of variables, differential and algebraic equations.

For a free body with decoupled translational and rotational dynamics, the projection must yield the well-known equations

$$J\dot{\omega} + \omega \times J\omega = T, \qquad \dot{\mathfrak{q}} = \frac{1}{2} G(\mathfrak{q})^\top \omega, \qquad \dot{R} = R^\top \omega_\times. \tag{6.14}$$

In the following, we propose a projection technique in order to define a systematic procedure to obtain Equations (6.14) directly from the Euler-Lagrange equations.

Quaternion Representation

We consider the case $q_{\mathrm{r}} = \mathfrak{q}$ and define the quaternion projection operator as

$$\mathcal{P}_{\mathfrak{q}}(v) = \frac{1}{2} G(\mathfrak{q}) v. \tag{6.15}$$

Theorem 6.1. *Consider the Lagrangian* $\mathcal{L} = \mathcal{T}_{\mathrm{t}} + \mathcal{T}_{\mathrm{r}} - \mathcal{V} - \lambda^\top \mathcal{C} + \langle Z, \mathcal{C}_{\mathfrak{q}} \rangle$, *with* $\mathcal{T}_{\mathrm{r}} = \frac{1}{2}\omega^\top J\omega$ *and* $C_{\mathfrak{q}} = (\mathfrak{q}^\top \mathfrak{q} - 1)$. *By projecting the rotational dynamics using* (6.15), *one obtains the following dynamic equations*

$$\frac{\mathrm{d}}{\mathrm{d}t}\nabla_{\dot{q}_{\mathrm{t}}}\mathcal{L} - \nabla_{q_{\mathrm{t}}}\mathcal{L} = F^{q_{\mathrm{t}}}, \tag{6.16a}$$

$$J\dot{\omega} + \omega \times J\omega - \mathcal{P}_{\mathfrak{q}}(\nabla_{\mathfrak{q}}(\mathcal{V} + \lambda^\top \mathcal{C})) = T, \tag{6.16b}$$

$$\dot{\mathfrak{q}} = \frac{1}{2}G(\mathfrak{q})^\top \omega, \qquad C = 0. \tag{6.16c}$$

Before proving the theorem, let us introduce some useful equivalences and lemmas

$$G(\mathfrak{q})G(\mathfrak{q})^\top = I, \qquad G(\mathfrak{q})G(\dot{\mathfrak{q}})^\top = (G(\mathfrak{q})\dot{\mathfrak{q}})_\times, \qquad G(\mathfrak{q})\dot{\mathfrak{q}} = -G(\dot{\mathfrak{q}})\mathfrak{q},$$
$$G(\mathfrak{q})\mathfrak{q} = 0, \qquad G(\dot{\mathfrak{q}})\dot{\mathfrak{q}} = 0. \tag{6.17}$$

From the quaternion dynamics (6.9), using (6.17), it holds that

$$\omega = 2G(\mathfrak{q})\dot{\mathfrak{q}}, \qquad \dot{\omega} = 2G(\mathfrak{q})\ddot{\mathfrak{q}} + \underbrace{2G(\dot{\mathfrak{q}})\dot{\mathfrak{q}}}_{=0}.$$

Then the rotational kinetic energy is given by

$$\mathcal{T}_{\mathrm{r}} = \frac{1}{2}\omega^\top J\omega = 2\dot{\mathfrak{q}}^\top G(\mathfrak{q})^\top J G(\mathfrak{q})\dot{\mathfrak{q}}, \tag{6.18}$$

Lemma 6.2. *The following equalities hold*

$$\mathcal{P}_{\mathfrak{q}}\left(\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{T}}{\partial \dot{q}} - \frac{\partial \mathcal{T}}{\partial q}\right) = J\dot{\omega} + \omega \times J\omega.$$

and

$$\mathcal{P}_{\mathfrak{q}}\left(F^{\mathfrak{q}}\right) = T.$$

Proof. By using (6.18) and (6.17), one obtains

$$\begin{aligned}\mathcal{P}_{\mathfrak{q}}\left(\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{T}}{\partial \dot{q}} - \frac{\partial \mathcal{T}}{\partial q}\right) &= \mathcal{P}_{\mathfrak{q}}\left(4G(\mathfrak{q})^\top J G(\mathfrak{q})\ddot{\mathfrak{q}} - 8G(\dot{\mathfrak{q}})^\top J G(\dot{\mathfrak{q}})\mathfrak{q}\right) \\ &= 2JG(\mathfrak{q})\ddot{\mathfrak{q}} - 2\,\omega \times JG(\dot{\mathfrak{q}})\mathfrak{q} \\ &= J\dot{\omega} + \omega \times J\omega.\end{aligned}$$

The virtual work δW resulting from an infinitesimal displacement $\delta\mathfrak{q}$ is given by $\delta W = \delta\mathfrak{q}^\top F^{\mathfrak{q}}$, with $\delta\mathfrak{q} = \frac{1}{2}G(\mathfrak{q})^\top \delta s$, where $\delta s \in \mathbb{R}^3$ is the infinitesimal rotation vector. The virtual work is also given by $\delta W = \delta s^\top T$, therefore it holds that

$$\delta W = \delta s^\top \frac{1}{2} G(\mathfrak{q}) F^{\mathfrak{q}} = \delta s^\top \mathcal{P}_{\mathfrak{q}}(F^{\mathfrak{q}}) = \delta s^\top T,$$

hence $\mathcal{P}_{\mathfrak{q}}\left(F^{\mathfrak{q}}\right) = T$. □

Lemma 6.3. *The quaternion projection* (6.15) *satisfies* $\mathcal{P}_{\mathfrak{q}}\left(< Z, C_{\mathrm{r}} >\right) = 0$.

Proof. Using the definition of C_{r} and (6.17) one obtains

$$\mathcal{P}_{\mathfrak{q}}\left(\nabla_{\mathfrak{q}} Z(\mathfrak{q}^\top \mathfrak{q} - 1)\right) = \mathcal{P}_{\mathfrak{q}}\left(2Z\mathfrak{q}\right) = ZG(\mathfrak{q})\mathfrak{q} = 0.$$

□

Proof (Theorem 6.1). The proof follows directly from Lemma 6.2 and 6.3. □

Direction Cosine Matrix Representation

We consider the case $q_{\mathrm{r}} = R$ and define the DCM projection operator as

$$\mathcal{P}_R(A) = \mathcal{U}(R^\top A), \tag{6.19}$$

where $\mathcal{U}$ is the "unskew" operator

$$\mathcal{U}\left(\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\right) = \frac{1}{2}\begin{bmatrix} a_{32} - a_{23} \\ a_{13} - a_{31} \\ a_{21} - a_{12} \end{bmatrix}$$

Theorem 6.4. *Consider the Lagrangian* $\mathcal{L} = \mathcal{T}_{\mathrm{t}} + \mathcal{T}_{\mathrm{r}} - \mathcal{V} - \lambda^\top \mathcal{C} + \langle Z, \mathcal{C}_R \rangle$*, with* $\mathcal{T}_{\mathrm{r}} = \frac{1}{2}\omega^\top J\omega$ *and* $C_R = (R^\top R - I)$*. By projecting the rotational dynamics using* (6.19)*, one obtains the following dynamic equations*

$$\frac{\mathrm{d}}{\mathrm{d}t}\nabla_{\dot{q}_{\mathrm{t}}}\mathcal{L} - \nabla_{q_{\mathrm{t}}}\mathcal{L} = F^{q_{\mathrm{t}}}, \tag{6.20a}$$

$$J\dot{\omega} + \omega \times J\omega - 2\mathcal{P}_R(\nabla_R(\mathcal{V} + \lambda^\top \mathcal{C})) = T, \tag{6.20b}$$

$$\dot{R} = R\omega_\times, \qquad C = 0. \tag{6.20c}$$

Before proving the theorem, let us introduce some useful equivalences and lemmas

$$\nabla_{\dot{R}}\left(\mathcal{P}_R(\dot{R})^\top a\right) = Ra_\times, \qquad \mathcal{U}(a_\times b_\times) = \frac{1}{2}a\times b, \qquad \operatorname{tr}(a_\times^\top A) = 2a^\top \mathcal{U}(A),$$

$$\nabla_{R}\left(\mathcal{P}_R(\dot{R})^\top a\right) = -\dot{R}a_\times, \qquad \mathcal{U}(ab^\top) = -\frac{1}{2}a\times b, \quad \operatorname{tr}(Aa_\times) = -2\mathcal{U}(A)^\top a,$$

$$\nabla_{R}\left(a^\top Rb\right) = ab^\top, \qquad -\mathcal{U}(A^\top R) = \mathcal{P}_R(A). \tag{6.21}$$

From the DCM dynamics (6.11) and (6.19), one obtains that $\omega = \mathcal{P}_R(\dot{R})$. Then the rotational kinetic energy is given by

$$\mathcal{T}_{\mathrm{r}} = \frac{1}{2}\omega^\top J\omega = 2\,\mathcal{P}_R(\dot{R})^\top J\mathcal{P}_R(\dot{R}), \tag{6.22}$$

Lemma 6.5. *The following equalities hold*

$$2\,\mathcal{P}_R\left(\frac{\mathrm{d}}{\mathrm{d}t}\nabla_{\dot{R}}\mathcal{T} - \nabla_R\mathcal{T}\right) = J\dot{\omega} + \omega\times J\omega.$$

and

$$2\,\mathcal{P}_R\left(F^R\right) = T.$$

Proof. By using (6.22) and (6.21), one obtains

$$\nabla_{\dot{R}}\mathcal{T} = \frac{1}{2}R\left(J\omega\right)_\times,$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\nabla_{\dot{R}}\mathcal{T} = \frac{1}{2}R\left(J\dot{\omega}\right)_\times + \frac{1}{2}\dot{R}\left(J\omega\right)_\times,$$

$$\nabla_{R}\mathcal{T} = -\frac{1}{2}\dot{R}\left(J\omega\right)_\times.$$

Then

$$\begin{aligned}2\,\mathcal{P}_R\left(\frac{\mathrm{d}}{\mathrm{d}t}\nabla_{\dot{R}}\mathcal{T} - \nabla_R\mathcal{T}\right) &= J\dot{\omega} + 2\,\mathcal{U}\left(\omega_\times\left(J\omega\right)_\times\right)\\ &= J\dot{\omega} + \omega\times J\omega.\end{aligned}$$

The virtual work δW resulting from an infinitesimal displacement δR is given by $\delta W = \operatorname{tr}(\delta R^\top F^R)$, with $\delta R = R^\top \delta s_\times$, where $\delta s \in \mathbb{R}^3$ is the infinitesimal

rotation vector. The virtual work is also given by $\delta W = \delta s^\top T$, therefore it holds that

$$\delta W = \mathrm{tr}(\delta R^\top F^R) = \mathrm{tr}(\delta s_\times^\top R^\top F^R) = 2\delta s^\top \mathcal{U}(R^\top F^R) = 2\delta s^\top \mathcal{P}_R(F^R) = \delta s^\top T,$$

hence $2\mathcal{P}_R\left(F^R\right) = T$. □

Lemma 6.6. *The DCM projection* (6.19) *satisfies* $\mathcal{P}_R\left(\nabla_R \langle Z, C_\mathrm{r}\rangle\right) = 0$.

Proof. Using the definition of C_r and (6.21) one obtains

$$\mathcal{P}_R\left(\nabla_R \mathrm{tr}\left(Z(R^\top R - 1)\right)\right) = \mathcal{P}_R\left(R(Z^\top + Z)\right) = \mathcal{U}\left(\underbrace{R^\top R}_{I}(Z^\top + Z)\right) = 0.$$

The last equality is due to the symmetry of $Z^\top + Z$. □

Proof (Theorem 6.4). The proof follows directly from Lemma 6.5 and 6.6. □

The following theorem becomes useful for performing index reduction.

Theorem 6.7. *The following relation holds true for any function* $f(R)$

$$\frac{\mathrm{d}}{\mathrm{d}t} f(R) = 2\,\mathcal{P}_R\left(\nabla_R f(R)\right)^\top \omega. \tag{6.23}$$

Proof. Using (6.11) and (6.21), one obtains

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} f(R) = \mathrm{tr}\left(\nabla_R(f(R))\dot{R}^\top\right) = \mathrm{tr}\left(\nabla_R f(R)\omega_\times^\top R^\top\right) = \mathrm{tr}\left(\omega_\times^\top R^\top \nabla_R f(R)\right)\\ = 2\mathcal{U}\left(R^\top \nabla_R f(R)\right)^\top \omega = 2\,\mathcal{P}_R\left(\nabla_R f(R)\right)^\top \omega.\end{aligned}$$

□

6.3 Baumgarte Stabilisation

As observed in the previous sections, when a system is modelled using non-minimal coordinates, the obtained equations contain so called *invariants*, i.e. quantities that are preserved by the system dynamics. Typical examples of invariants for mechanical systems are the constraints $\mathcal{C}$, their time derivative $\dot{\mathcal{C}}$ and the rotation orthonormality condition $\mathfrak{q}^\top\mathfrak{q} - 1$ or $R^\top R - I$. The model has a physical meaning only if all the invariants $\mathcal{I}$ are satisfied, i.e. $\mathcal{I} = 0$.

When simulating the system over long time horizons, due to integrator inaccuracy, numerical errors cumulate and the invariants are typically not preserved. When solving optimal control problems, in some circumstances, this fact can be exploited by the optimiser in order to decrease the cost by altering the system dynamics in a favourable way: clearly, both situations need to be avoided. This section discusses *Baumgarte stabilisation* [11], a well-known technique which solves the aforementioned issues.

6.3.1 Tether Constraints

When performing index reduction of an index-3 DAE, the system constraints $\mathcal{C}$ are differentiated twice with respect to time and $\ddot{\mathcal{C}} = 0$ is imposed. Therefore, the time evolution of $\mathcal{C}$ is marginally stable and any numerical error will be integrated twice. The constraint dynamics are then described by the system

$$\dot{C} = A_C C = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} C, \qquad \text{where} \qquad C = \begin{bmatrix} \mathcal{C} \\ \dot{\mathcal{C}} \end{bmatrix}. \tag{6.24}$$

This system can be easily stabilised by modifying matrix A_C so as to make all its eigenvalues negative. A common choice is to set the same value for all its eigenvalues: $\text{eig}(A_C) = -\gamma_{\mathcal{C}},\ \gamma_{\mathcal{C}} > 0$, i.e.

$$A_C = \begin{bmatrix} 0 & I \\ -\gamma_{\mathcal{C}}^2 I & -2\gamma_{\mathcal{C}} I \end{bmatrix},$$

which results in

$$\ddot{\mathcal{C}} + 2\gamma_{\mathcal{C}}\dot{\mathcal{C}} + \gamma_{\mathcal{C}}^2 \mathcal{C} = 0. \tag{6.25}$$

6.3.2 Quaternion Invariant

In order to guarantee that the invariant $\mathfrak{q}^\top \mathfrak{q} - 1$ is preserved, the following stabilisation has been proposed in [51]

$$\dot{\mathfrak{q}} = \underbrace{\frac{1}{2} G(\mathfrak{q})^\top \omega}_{\dot{\mathfrak{q}}_{\mathrm{n}}} + \underbrace{\frac{1}{2}\gamma_{\mathfrak{q}} \mathfrak{q}((\mathfrak{q}^\top \mathfrak{q})^{-1} - 1)}_{\dot{\mathfrak{q}}_{\mathrm{c}}}, \tag{6.26}$$

with $\gamma_{\mathfrak{q}} > 0$.

A less nonlinear correction is obtained using the approximation

$$\left((\mathfrak{q}^\top \mathfrak{q})^{-1} - 1\right) \approx -\left((\mathfrak{q}^\top \mathfrak{q}) - 1\right),$$

which yields the system dynamics

$$\dot{\mathfrak{q}} = \underbrace{\frac{1}{2}G(\mathfrak{q})^\top \omega}_{\dot{\mathfrak{q}}_{\mathrm{na}}} - \underbrace{\frac{1}{2}\gamma_{\mathfrak{q}}\mathfrak{q}((\mathfrak{q}^\top\mathfrak{q}) - 1)}_{\dot{\mathfrak{q}}_{\mathrm{c}}}, \tag{6.27}$$

with $\gamma_{\mathfrak{q}} > 0$.

Note that, for vectors $\mathfrak{q}$ such that $\mathfrak{q}^\top\mathfrak{q} = 1$ holds, then the dynamics (6.26) $\dot{\mathfrak{q}} = \dot{\mathfrak{q}}_{\mathrm{n}}$. Moreover, the proposed corrections are orthogonal to the quaternion natural dynamics, i.e.

$$\dot{\mathfrak{q}}_{\mathrm{n}}^\top\dot{\mathfrak{q}}_{\mathrm{c}} = \frac{\rho}{4}\omega^\top G(\mathfrak{q})\mathfrak{q}((\mathfrak{q}^\top\mathfrak{q})^{-1} - 1) = 0,$$

$$\dot{\mathfrak{q}}_{\mathrm{na}}^\top\dot{\mathfrak{q}}_{\mathrm{c}} = -\frac{\rho}{4}\omega^\top G(\mathfrak{q})\mathfrak{q}((\mathfrak{q}^\top\mathfrak{q}) - 1) = 0.$$

Theorem 6.8. *If the quaternion dynamics are stabilised according to* (6.26) *or* (6.27)*, then the quaternion invariant dynamics are stable.*

Proof. The proof is given in [51] and we will recall it in the following. Equation (6.26) enforces the invariant dynamics

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right) = 2\mathfrak{q}^\top\dot{\mathfrak{q}} = \mathfrak{q}^\top G(\mathfrak{q})^\top\omega - \gamma_{\mathfrak{q}}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right)$$

$$= -\gamma_{\mathfrak{q}}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right).$$

Equation (6.27) enforces the invariant dynamics

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right) = 2\mathfrak{q}^\top\dot{\mathfrak{q}} = \mathfrak{q}^\top G(\mathfrak{q})^\top\omega - \gamma_{\mathfrak{q}}\mathfrak{q}^\top\mathfrak{q}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right)$$

$$= -\gamma_{\mathfrak{q}}\mathfrak{q}^\top\mathfrak{q}\left(\mathfrak{q}^\top\mathfrak{q} - 1\right).$$

Since $\mathfrak{q}^\top\mathfrak{q} \geq 0$, the invariant is stabilised for all initial quaternions $\mathfrak{q} \neq 0$. In particular, we are typically interested in maintaining $\mathfrak{q}^\top\mathfrak{q} \approx 1$, so that in practice the two approaches are expected to behave similarly. □

6.3.3 DCM Invariant

In order to guarantee that the invariant $R^\top R - I$ is preserved, the following stabilisation has been proposed in [52, 51].

$$\dot{R} = R\omega_\times + \frac{\gamma_R}{2}R\left(\left(R^\top R\right)^{-1} - I\right), \tag{6.28}$$

with $\gamma_R > 0$. A less nonlinear correction which does not need the matrix inversion is obtained using the first-order Neumann expansion of $(R^\top R)^{-1} - I$, which gives

$$\left((R^\top R)^{-1} - I\right) \approx -\left((R^\top R) - I\right),$$

which yields the system dynamics

$$\dot{R} = R\omega_\times - \frac{\gamma_R}{2} R\left(R^\top R - I\right). \tag{6.29}$$

Theorem 6.9. *If the DCM dynamics are stabilised using* (6.28) *or* (6.29)*, then the DCM invariant dynamics are stable.*

Proof. The proof is given in [51] and we will recall it here. First we prove that, for any matrix $R \in \mathbb{R}^{3\times 3}$, subject to the dynamics $\dot{R} = R\omega_\times$, the following holds:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = 0,$$

where $\|.\|_{\mathrm{F}}$ denotes the Frobenius norm.

This follows from the properties of the trace operator, and can be established as follows:

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = \\
&= \frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\mathrm{Tr}\left(\left(R^\top R - I\right)^\top \left(R^\top R - I\right)\right) = \mathrm{Tr}\left(\left(R^\top \dot{R} + \dot{R}^\top R\right)^\top \left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(R^\top R\omega_\times - \omega_\times R^\top R\right)^\top \left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(R^\top R\omega_\times - \omega_\times R^\top R\right)\left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(R^\top R\omega_\times R^\top R\right) - \mathrm{Tr}\left(R^\top R\omega_\times\right) - \mathrm{Tr}\left(\omega_\times R^\top R R^\top R\right) + \mathrm{Tr}\left(\omega_\times R^\top R\right) \\
&= \mathrm{Tr}\left(\omega_\times R^\top R R^\top R\right) - \mathrm{Tr}\left(\omega_\times R^\top R\right) - \mathrm{Tr}\left(\omega_\times R^\top R R^\top R\right) + \mathrm{Tr}\left(\omega_\times R^\top R\right) \\
&= 0.
\end{aligned}$$

We note that, for any symmetric matrix $M \in \mathbb{R}^{n\times n}$ the following holds:

$$2\mathrm{Tr}\left(M\right) - \mathrm{Tr}\left(M^2\right) - n = -\|M - I\|_{\mathrm{F}}^2.$$

This can be proven by observing that

$$2\mathrm{Tr}\left(M\right) - \mathrm{Tr}\left(M^2\right) - n = -\mathrm{Tr}\left(M^2 - 2M + I\right) =$$

$$-\mathrm{Tr}\left(\left(M-I\right)^2\right).$$

Then, because M is symmetric, we have

$$\mathrm{Tr}\left(\left(M-I\right)^2\right) = \left\|M-I\right\|_{\mathrm{F}}^2.$$

We prove now that, for any matrix $R \in \mathbb{R}^{3\times 3}$, subject to the dynamics (6.28), the invariant dynamics are exponentially stable under the Frobenius norm, i.e.

$$\left\|R^\top R - I\right\|_{\mathrm{F}} = e^{-\rho(t-t_0)} \left\|R_0^\top R_0 - I\right\|_{\mathrm{F}}, \tag{6.30}$$

where R_0 is the DCM matrix at time t_0 and R is the DCM matrix at time $t \geq t_0$.

First, we define $A = \frac{\rho}{2}\left(\left(R^\top R\right)^{-1} - I\right)$ and observe that

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = \\
&= \frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\mathrm{Tr}\left(\left(R^\top R - I\right)^\top \left(R^\top R - I\right)\right) = \mathrm{Tr}\left(\left(\dot{R}^\top R + R^\top \dot{R}\right)^\top \left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(\left(A^\top - \omega_\times\right) R^\top R + R^\top R\left(A + \omega_\times\right)\right)^\top \left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(A^\top R^\top R - \omega_\times R^\top R + R^\top R A + R^\top R \omega_\times\right)\left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(R^\top R \omega_\times - \omega_\times R^\top R\right)\left(R^\top R - I\right)\right) \\
&\qquad\qquad + \mathrm{Tr}\left(\left(A^\top R^\top R + R^\top R A\right)\left(R^\top R - I\right)\right) \\
&= \mathrm{Tr}\left(\left(A^\top R^\top R + R^\top R A\right)\left(R^\top R - I\right)\right).
\end{aligned}$$

We then turn to proving the negativity of

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = \mathrm{Tr}\left(\left(A^\top R^\top R + R^\top R A\right)\left(R^\top R - I\right)\right).$$

We observe that

$$\begin{aligned}
&\mathrm{Tr}\left(\left(A^\top R^\top R + R^\top R A\right)\left(R^\top R - I\right)\right) = \\
&\qquad = \mathrm{Tr}\left(\left(A^\top + A\right) R^\top R R^\top R\right) - \mathrm{Tr}\left(\left(A^\top + A\right) R^\top R\right),
\end{aligned}$$

and use

$$A^\top + A = \rho\left(\left(R^\top R\right)^{-1} - I\right),$$

such that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 &= \rho\left(2\mathrm{Tr}\left(R^\top R\right) - \mathrm{Tr}\left(R^\top R R^\top R\right) - n\right)\\ &= -\rho\left\|R^\top R - I\right\|_{\mathrm{F}}^2.\end{aligned}$$

It then follows that

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = -2\rho\left\|R^\top R - I\right\|_{\mathrm{F}}^2,$$

which yields

$$\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = e^{-2\rho(t-t_0)}\left\|R_0^\top R_0 - I\right\|_{\mathrm{F}}^2,$$

and hence (6.30).

Finally, we prove now that, for any matrix $R \in \mathbb{R}^{3\times 3}$, subject to the dynamics (6.29), the invariant dynamics are locally stable under the Frobenius norm, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = -\rho\left\|R\left(R^\top R - I\right)\right\|_{\mathrm{F}}^2. \tag{6.31}$$

We define $A \approx \tilde{A} = \frac{\rho}{4}\left(I - R^\top R\right)$ and reuse the previous developments

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = \mathrm{Tr}&\left(\left(\tilde{A}^\top + \tilde{A}\right)R^\top R R^\top R\right)\\ &- \mathrm{Tr}\left(\left(\tilde{A}^\top + \tilde{A}\right)R^\top R\right),\end{aligned}$$

where now $\tilde{A}^\top + \tilde{A}$ reads as:

$$\tilde{A}^\top + \tilde{A} = \frac{\rho}{2}\left(I - R^\top R\right).$$

This yields

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 &= \frac{\rho}{2}\mathrm{Tr}\left(\left(I - R^\top R\right)R^\top R R^\top R\right) - \frac{\rho}{2}\mathrm{Tr}\left(\left(I - R^\top R\right)R^\top R\right)\\ &= \rho\mathrm{Tr}\left(R^\top R R^\top R\right) - \frac{\rho}{2}\mathrm{Tr}\left(R^\top R R^\top R R^\top R\right) - \frac{\rho}{2}\mathrm{Tr}\left(R^\top R\right),\end{aligned}$$

hence

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = \\ &= -\frac{\rho}{2}\mathrm{Tr}\left(R^\top R R^\top R R^\top R - 2R^\top R R^\top R + R^\top R\right) \\ &= -\frac{\rho}{2}\mathrm{Tr}\left(R^\top R\left(R^\top R R^\top R - 2R^\top R + I\right)\right) \\ &= -\frac{\rho}{2}\mathrm{Tr}\left(R^\top R\left(R^\top R - I\right)^2\right),\end{aligned}$$

which entails

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\|R^\top R - I\right\|_{\mathrm{F}}^2 = -\rho\left\|R\left(R^\top R - I\right)\right\|_{\mathrm{F}}^2.$$

□

6.3.4 Numerical Simulations

In this section we illustrate our theoretical results by simulating the stabilised DCM and quaternion dynamics:

$$\dot{R} = R(A + \Omega), \qquad \dot{q} = \frac{1}{2}G(q)^\top \omega + \frac{1}{2}\rho q\left(\left(q^\top q\right)^{-1} - 1\right)$$

respectively, with $\Omega = \omega_\times$ and constant ω.

First, we ran 50 simulations for randomly generated perturbations of the initial quaternion or DCM invariant using a fixed stepsize Runge Kutta 4 (RK4) integrator with stepsize $T_\mathrm{s} = 0.05$ s. The results are shown in Figure 6.2, where the quaternion invariant and the Frobenius norm of the DCM invariant are plotted over time. As it can be seen, the proposed approach successfully reduces the invariant violation over time.

While integration schemes exist which guarantee to preserve the system invariants, in the context of optimal control one typically prefers to deploy simple integration schemes, so as to minimise the computational burden of carrying out the integration with sensitivity generation [94]. On short simulation horizons a good integration accuracy is generally sufficient to keep the invariant violation small. However, if the horizon is made long enough, the invariant violation can grow prohibitively large. The proposed stabilisation keeps the invariant violation small over arbitrarily long horizons.

There is no upper bound to constant ρ, and in principle a larger ρ would reduce the invariant violation to very small values. However, large values make the system stiff, which impedes the accuracy of the integrators. This effect is illustrated in Figure 6.3, where the maximum invariant violation over a long horizon simulation is plotted. It can be seen that as expected, when ρ is small, the invariant violation is not significantly reduced. As ρ increases the violation decreases, however it can be seen that when ρ becomes too large, the invariant violation increases again.

6.3.5 Application to optimal control

We illustrate next the proposed correction of the DCM representation of the SO(3) rotation group on the optimal control of a small satellite using magnetorquers interacting with the earth magnetic field, inspired from [74]. Both the de-tumbling and reorientation problems are considered.

Example 6.10 (Optimal re-orientation of a Satellite)**.** *The satellite equations are given by:*

$$\dot{R} = R\left(\omega_\times + \tilde{A}\right), \tag{6.32a}$$

$$J\dot{\omega} = T - \omega \times J\omega, \tag{6.32b}$$

$$T = u \times R^\top B(t). \tag{6.32c}$$

where $R \in \mathbb{R}^{3\times3}$ is the DCM, $\omega \in \mathbb{R}^3$ the angular speed and $\omega_\times \in \mathbb{R}^{3\times3}$ the resulting skew symmetric matrix. Vector T is the torque generated by the magnetorquers, $J \in \mathbb{R}^{3\times3}$ the satellite inertia in its own referential, and

$$B(t) = 2 \cdot 10^{-5} \left[\begin{array}{ccc} 0 & \cos(t/5400\text{ s}) & \sin(t/5400\text{ s}) \end{array}\right]^\top$$

is the earth magnetic field, which is rotating due to the satellite non-equatorial orbit. The control input $u \in \mathbb{R}^3$ is the dipole moment generated by the magnetorquers. In the following we will lump the states R, ω together in vector x and collect the system dynamics (6.32) in $\dot{x} = f(x, u)$.

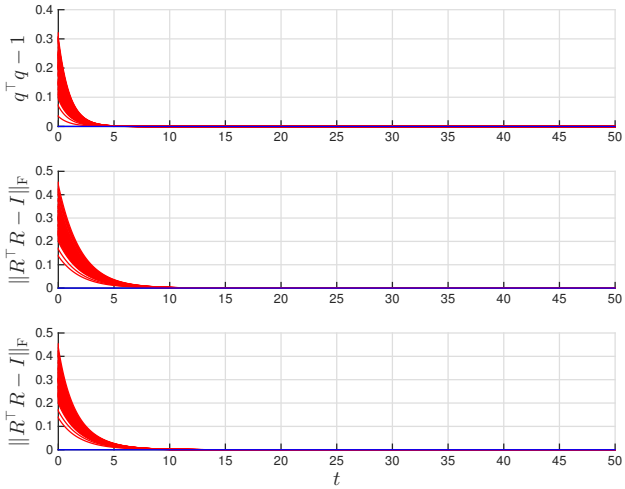


Figure 6.2: Invariant stabilisation: 50 randomly generated perturbations (red) of a nominal (blue) initial DCM or quaternion. Top Graph: quaternion invariant stabilisation. Middle graph: Frobenius norm of the DCM invariant. Bottom graph: Frobenius norm of the DCM invariant using the approximated stabilisation.

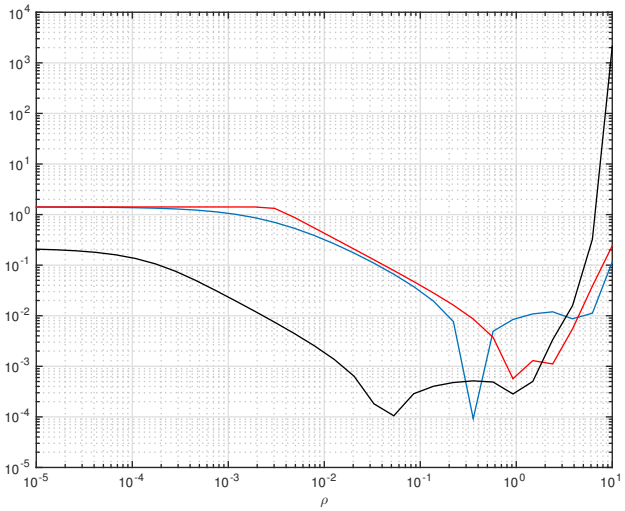


Figure 6.3: Maximum invariant violation as a function of ρ using a Runge-Kutta integrator of order 4, with a fixed stepsize $T_s = 0.05$, and a simulation time $T_f = 10000$. The maximum violation is displayed in black for the quaternion invariant, in blue for the Frobenius norm of the DCM invariant and in red for the Frobenius norm of the DCM invariant stabilised using the first order approximation.

In order to re-orient the satellite we formulate the following optimal control problem (OCP)

$$\min_{x(\cdot),u(\cdot)} \quad \int_0^{t_\mathrm{f}} 5\|R(t)-I\|_\mathrm{F}^2 + \frac{1}{2}\|\omega(t)\|_J^2 + \frac{1}{2}\|u(t)\|^2 \,\mathrm{d}t \tag{6.33a}$$

$$s.t. \quad x(0) = x_0, \tag{6.33b}$$

$$\dot{x} = f(x,u), \qquad t \in [0,T] \tag{6.33c}$$

$$-0.1 \text{ Am}^2 \leq u \leq 0.1 \text{ Am}^2, \tag{6.33d}$$

with the end time $t_\mathrm{f} = 18$ min *and an initial state given by*

$$R_0 = \mathrm{diag}\left(\begin{bmatrix} 1 & -1 & -1 \end{bmatrix}\right), \qquad \textit{and} \qquad \omega_0 = 0 \text{ rad/s}.$$

We solve OCP (6.33) *via direct collocation with Radau polynomials of order 1. The control inputs are discretised as piecewise constants, with a sampling time of* 3.6 s. *The problem is transcribed using the software tool CasADi [5], and the resulting NLP solved using the solver Ipopt [114].*

The trajectories resulting from solving (6.33) *for both* $\rho = 0$ *and* $\rho = 10$ *are reported in red and black, respectively, in Figures 6.4 and 6.5. The Frobenius norm of the DCM invariant is displayed in Figure 6.6, where it can be seen that the OCP formulated using the non-stabilised dynamics yields a solution with a large invariant violation. By violating the DCM invariant, the OCP gains the ability to steer the system in an unphysical way, as can be seen in Equation* (6.32c). *This results in different control and rotational speed trajectories, as it can be seen in Figures 6.5 and 6.4 respectively.*

In this example, the violation of the orthonormality condition has a limited impact on the OCP solution. This observation, however, cannot be generalised and if matrix R *has a strong impact on the dynamics, e.g. when they enter the construction of the aerodynamic forces [126], it can be crucial to maintain a good orthonormality.*

Example 6.11 (Optimal de-tumbling of a Satellite)**.** *We now turn to the de-tumbling of the same satellite, which can be formulated an OCP similar to* (6.33)*, but with different weights, i.e.:*

$$\min_{x(\cdot),u(\cdot)} \quad \int_0^{t_\mathrm{f}} 10^{-4}\|R(t)-I\|_\mathrm{F}^2 + 0.1\|\omega(t)\|_J^2 + \|u(t)\|^2 \,\mathrm{d}t \tag{6.34a}$$

$$s.t. \quad \text{(6.33b)} - \text{(6.33d)} \tag{6.34b}$$

with the end time $t_\mathrm{f} = 36$ min *and an initial state given by* $R_0 = I$*, and* $\omega_0 = \begin{bmatrix} 0.05 & 0 & 0 \end{bmatrix}^\top$ rad/s. *We turn to solving the OCP* (6.34) *via direct*

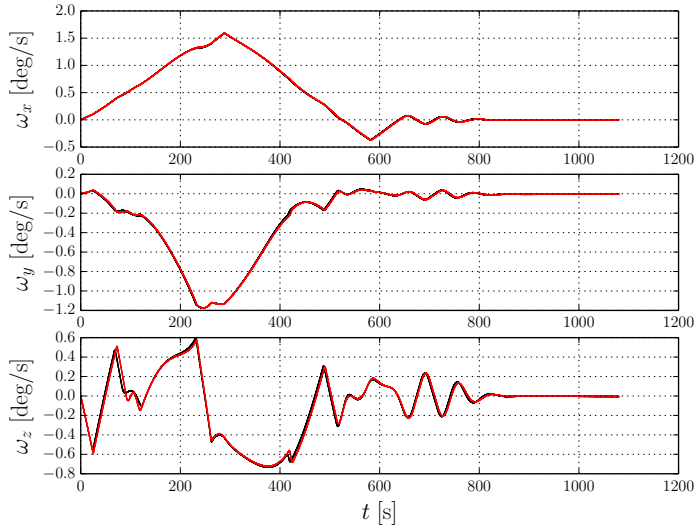


Figure 6.4: Rotational speed for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 10$.

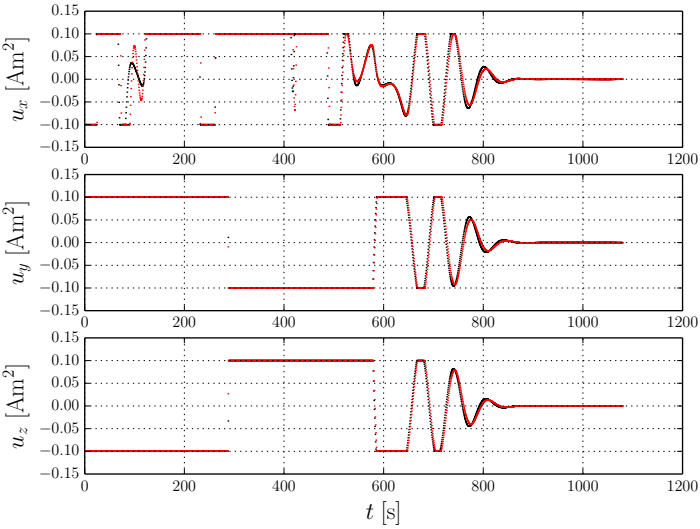


Figure 6.5: Controls for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 10$.

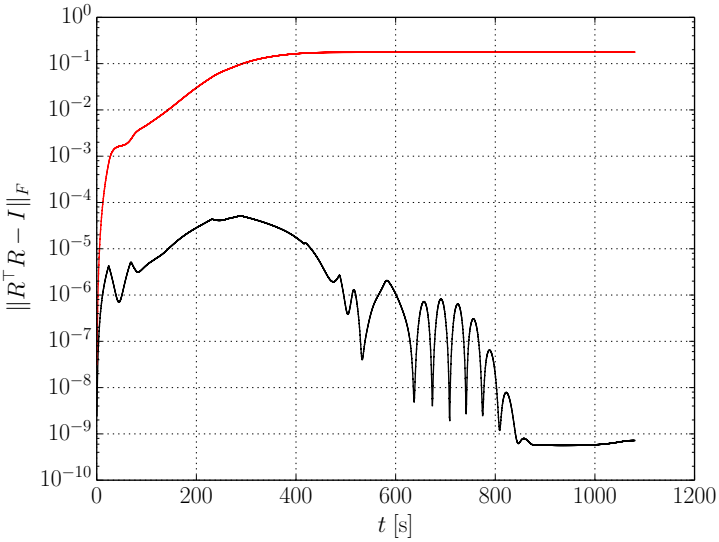


Figure 6.6: Collocation: Frobenius norm of the invariant for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 10$.

multiple shooting using $N = 200$ *control intervals. The control inputs are discretised as piecewise constant with a sampling time of* 3.6 s *and the system dynamics are discretised using a single step of a RK4 scheme using fixed step size. For the stabilised dynamics, the value* $\rho = 1e-2$ *has been chosen. As opposed to collocation, RK4 is an explicit integrator and therefore not suited for stiff systems. Higher values of* ρ *would deteriorate integration accuracy, as described in Section 6.3.4.*

The controls and rotational speed are displayed in Figure 6.9 and 6.7 respectively, where it can be seen that the non stabilised dynamics yield a different solution. In particular, the kinetic energy is reduced more when the dynamics are not stabilised, suggesting that the optimiser exploits the numerical inaccuracy in order to alter the system dynamics, by acting in particular on Equation (6.32c)*, and further decrease the cost.*

The invariant violation is displayed in Figure 6.8. It can be seen that the stabilisation of the system dynamics effectively reduces the invariant violation by keeping it to small values, while for the non stabilised dynamics the invariant violation increases over time. Note that it is hard in general to draw conclusions about the best value to choose for parameter ρ*. However, as we used a smaller value for* ρ *in comparison to collocation, the invariant violation is larger. Note*

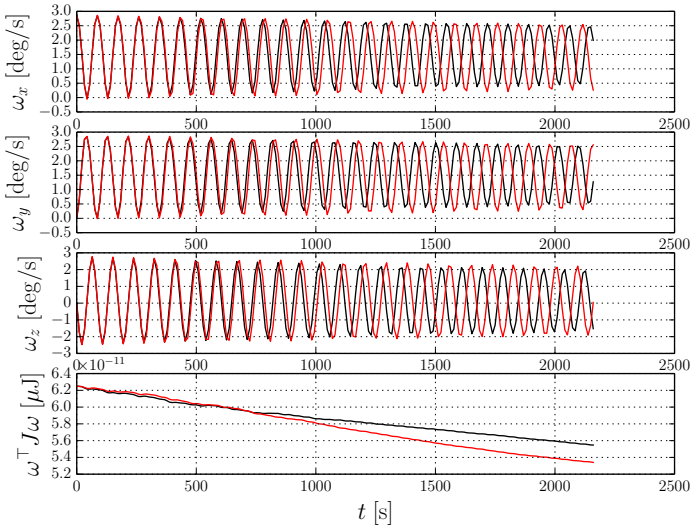


Figure 6.7: Multiple Shooting: Rotational speed and energy for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 0.01$.

that for the unstabilised dynamics, the invariant violation increases over time while in the collocation example they saturate quickly. This last fact stems from the way direct collocation handles dynamics in the OCP, which distributes the integration error over the time horizon rather than cumulating it sequentially.

6.4 Conclusions

In this chapter we have presented the Euler-Lagrange approach. In order to apply this approach to obtain efficient models in non-minimal coordinates, we extended the standard approach in order to be able to easily handle both the quaternion and DCM parametrisation for rotations, so as to reduce the complexity of the model. Furthermore, we proposed a Baumgarte-type stabilisation for the rotation invariants so as to stabilise them. We illustrated the effectiveness of the stabilisation with two simulation examples.

The modelling approach described in this chapter will be used in Chapter 7 in order to model tethered airplanes for energy harvesting.

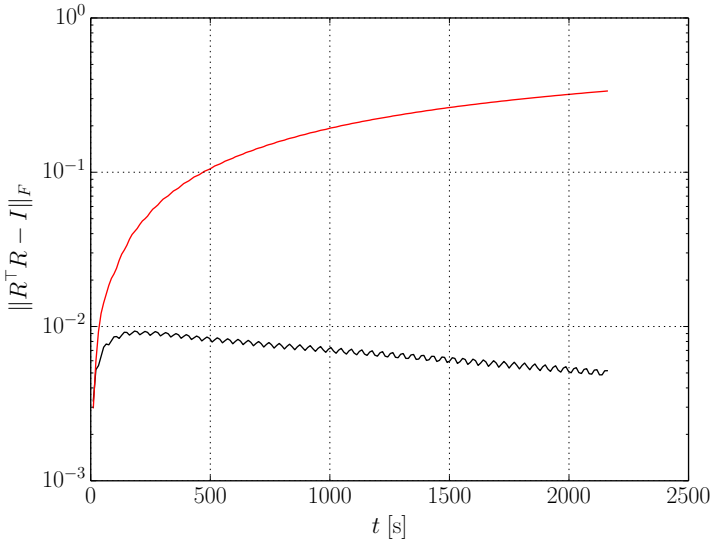


Figure 6.8: Multiple Shooting: Frobenius norm of the invariant for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 0.01$.

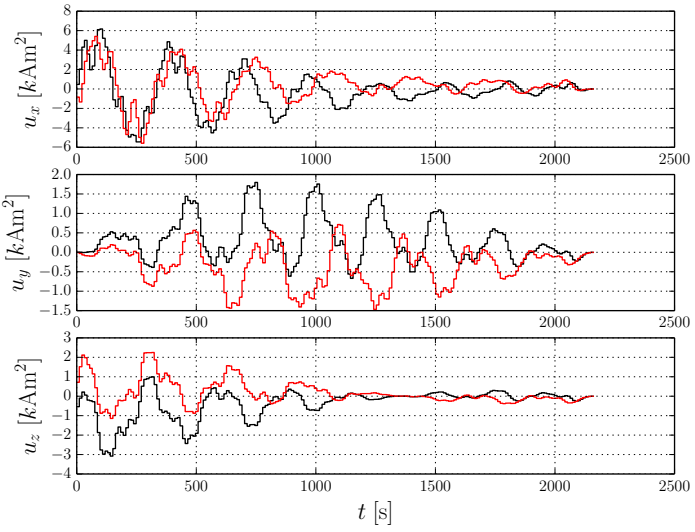


Figure 6.9: Multiple Shooting: Controls for the satellite without stabilisation (red) and stabilised using the approximate DCM stabilisation with $\rho = 0.01$.

Chapter 7

Airborne Wind Energy

In this chapter we focus on Airborne Wind Energy (AWE): not only a promising technology for renewable energy production, but also a challenging application for optimal control, MPC and MHE.

In Section 7.1 we present the technology and highlight the key role that optimal control, MPC and MHE can have for its success. In Section 7.2 the methods introduced in Chapter 6 are used to derive models of reduced complexity for tethered airplanes aimed at optimal control. A framework for a reliable formulation of the offline and online OCP formulations is given in Section 7.3.

Section 7.4 proposes a comparison of single and dual airfoils for energy harvesting. The comparison is done by means of a parametric study which compares the two systems for different wing areas. First we consider a simpler study in which the interaction between the airfoil and the airmass is neglected. Then, we develop a simple interaction model based on blade-element–momentum theory applied to AWE systems.

In Sections 7.5, and 7.6 the proposed approach is applied to control respectively single and dual airfoil AWE systems for power generation. In Section 7.7 the concept of rotational startup is introduced and both simulation and experimental results for a small scale prototype are presented.

In Section 7.8 most of the contributions of this thesis are combined together and applied to control a single airfoil for maximum power extraction by means of tuned periodic tracking NMPC.

Acknowledgement The first part of this chapter is mainly based on [49, 119, 124, 50, 126, 121, 42]. Section 7.8 is based on new material due to be published. The detail of the contributions is as follows:

- Section 7.2 is based on material presented in [49, 50, 121, 126, 121, 58, 59, 41]. Sébastien Gros and Mario Zanon are the main authors of the single and dual airfoil models. Boris Houska introduced the simple tether model which has been subsequently extended by Mario Zanon and Sébastien Gros. The aerodynamic model has been developed by Boris Houska, Kurt Geebelen, Joris Gillis, Sébastien Gros, Mario Zanon and Greg Horn.
- Section 7.3 is based mostly on [49, 48, 52]. Sébastien Gros is the main author of all three papers. He developed mainly the initialisation technique proposed in Section 7.3.1. Mario Zanon contributed some numerical results and served as a discussion partner.
- Section 7.4 is based on [119, 124]. Mario Zanon is the main author of both publications and developed the AWE models jointly with Sébastien Gros. Mario Zanon performed all numerical simulations and Sébastien Gros contributed the first version of the airfoil-airmass interaction, subsequently refined by Johan Meyers.
- Section 7.5 is based on [49, 50]. Sébastien Gros is the main author of both publications and contributed most of the developments. Mario Zanon contributed some numerical simulations and served as a discussion partner.
- Section 7.6 is based on [126]. Mario Zanon is the main author of the paper and contributed most of the developments. Sébastien Gros and Moritz Diehl served as valuable discussion partners and supervisors.
- Section 7.7 is based on [121, 42]. Mario Zanon is the main author of [121] and contributed most of the developments. Sébastien Gros and Moritz Diehl served as valuable discussion partners and supervisors. Kurt Geebelen is the main author of [42], which is based mostly on the contributions of Kurt Geebelen, Milan Vukov, Andrew Wagner, Hammad Ahmad, Mario Zanon and Sébastien Gros under the supervision of Dirk Vandepitte, Jan Swevers and Moritz Diehl.
- Section 7.8 is based on new contributions due to be published. Mario Zanon contributed most of the developments under the supervision of Moritz Diehl and Sébastien Gros.

All the work has been done under the supervision of Moritz Diehl.

7.1 Introduction

The importance of renewable energy has significantly increased over the last years and is expected to grow even more in the future. In particular, wind energy plays a very important role among renewable energy sources, as stated in the European energy policy, which plans wind energy to account for more than one third of all electricity production from renewable energy sources by 2020 and almost 40% by 2030.

Over the last years, conventional wind turbines have grown in size and mass up to a scale at which the major challenges are posed by the structural loads [73, 24]. In particular, this limitation prevents wind turbines from reaching higher altitudes, where the wind speed is typically higher and more consistent. In order to overcome these limitations, the main idea behind Airborne Wind Energy (AWE) is to eliminate all elements which are not essential to power extraction, resulting in a much lighter structure that only involves an airfoil tethered to the ground. In this configuration, higher altitudes can be reached and the swept area is not fixed by the structure of the system, but can be optimised so as to maximise the extracted power. The system is thus free to operate in previously inaccessible portions of the wind field, where higher wind resources can be found [9].

The potential of this technology has been established in theory [75]. Several architectures have been proposed for AWE systems, ranging from a single tethered airfoil to complex multi-airfoil structures. In this chapter we will focus on tethered rigid airfoils. Power can be extracted in two different configurations: in the so-called *drag mode* generators are present on board which extract energy from the airmass; in the so-called *pumping mode* the generator is attached to the winch on the ground and the energy is extracted by reeling the tether in and out. The power P that can be generated with a tethered airfoil operated either in drag or in lift mode under idealised assumptions has been estimated by Loyd [75] to be approximately given by

$$P = \frac{2}{27}\rho A v_{\mathrm{w}}^3 C^{\mathrm{L}} \left(\frac{C^{\mathrm{L}}}{C^{\mathrm{D}}}\right)^2, \tag{7.1}$$

where ρ is the air density, A the wing area, v_{w} the wind speed and C^{L} and C^{D} the lift and drag coefficients respectively. Note that the efficiency, or gliding ratio, $\mathcal{E} = \frac{C^{\mathrm{L}}}{C^{\mathrm{D}}}$, enters Equation (7.1) quadratically. Typical values for the gliding ratio for soft airfoils are below $\mathcal{E} = 13$, while for rigid wings they can get as high as $\mathcal{E} = 70$.

While the potential efficiency of the principle is established in theory, a major research effort is still required to address the many engineering difficulties posed by its implementation, and to achieve its industrial development. In particular,

it is widely recognised in the AWE community that the industrial viability of the technology will require the optimisation of the power generation.

Optimisation and optimal control are hence extremely powerful tools which can have a high impact on AWE. As opposed to conventional wind turbines, AWE systems have the freedom to follow trajectories of arbitrary shape. This freedom can be exploited in order to maximise the extracted power. Moreover, the system can be optimised together with the trajectory, so as to simultaneously design the plant and the trajectory. Clearly, optimising the full design of the system over a very large set of parameters while also optimising a power-extracting orbit is a very difficult problem to solve. By restricting to a smaller set of parameters, however, it is possible to obtain excellent results, where e.g. the tether length and diameter are optimally chosen, while the aerodynamics of the aircraft are fixed.

Tethered airfoils are unstable systems with fast dynamics: the rotational dynamics are nonlinear by nature, additional strong nonlinearities arise from the aerodynamic forces and torques, and the presence of the tether introduces further strong nonlinearities. Moreover, AWE systems are constrained, as the airfoil must operate inside the flight envelope, the tether must always be under tension and the input bounds can be significantly activated. MPC is a control technique able to address all these issues, as it is designed to control constrained nonlinear systems. MPC schemes are designed under the assumption that the model parameters and the state are perfectly known. In most real applications, a full state measurement is either not possible or not desirable, both for technical and economical reasons. Moreover, the model parameters and wind conditions can be uncertain or varying, and may therefore need to be estimated online. Moving Horizon Estimation (MHE) is an optimisation-based observer which accounts for system nonlinearities and constraints, while performing combined parameter and state estimation.

While both NMPC and MHE can efficiently deal with nonlinear constrained unstable and fast systems, they require the online solution of a dynamic optimisation problem. This can in general require too long computational times to allow for real-time implementations for most mechanical systems. A careful formulation of the optimisation problem, combined with the use of efficient algorithms based on direct multiple shooting, the Real Time Iteration (RTI) scheme and automatic tailored C code generation, allows to significantly reduce computational times to the scale of milliseconds.

Having the aim of maximising energy extraction from the airmass, tethered airfoils are a challenging application for economic MPC. The system being characterised by very complicated dynamics, any attempt at trying to prove strict dissipativity for ensuring stability will face extreme difficulties. Moreover,

the impossibility of applying the efficient MPC algorithms developed for tracking NMPC to economic NMPC might make it hard to implement economic MPC on a real system. In this framework, the developments of Chapter 5 can be used to tune a tracking NMPC scheme which locally behaves like economic NMPC up to first order.

In this chapter, we have mainly two aims: (a) demonstrating that optimal control, MPC and MHE can be applied even to very nonlinear and complicated systems like tethered airfoils and (b) using these control techniques in order to establish the potential of the technology using realistic and accurate studies. On the other hand, since the technology is still quite young and only a very small number of prototypes has been built, there are still many aspects which deserve being investigated and the work done so far is meant to be used as a starting point for further studies based on simulations, optimisations and experiments.

7.2 Modelling of Airborne Wind Energy Systems

Airborne wind energy systems are multi-body systems which result from the interaction of the winch, the tether(s) and the wing(s). Many different mathematical models have been derived ranging from very simple models which only capture the main dynamics to accurate models which include an aerodynamic airfoil model.

In this section, we present AWE models which account for the aerodynamics of the wing. First we will introduce the used reference frames and the aerodynamic model, which uses the aircraft translational and rotational velocities in order to compute the aerodynamic forces and torques. The use of Cartesian coordinates for describing the aircraft's centre of mass makes these expressions dramatically simpler than when polar coordinates are used, especially for multi-airfoil systems [47].

7.2.1 Reference Frames

In order to describe physical quantities, in this chapter we will make use of the following convention for the reference frames. A visualisation of the used reference frames is given in Figure 7.1.

An orthonormal right-hand inertial reference frame E is chosen such that vector E^z is opposed to the gravitational acceleration vector g and the attachment point of the tether to the ground is in the origin of E. For simplicity and

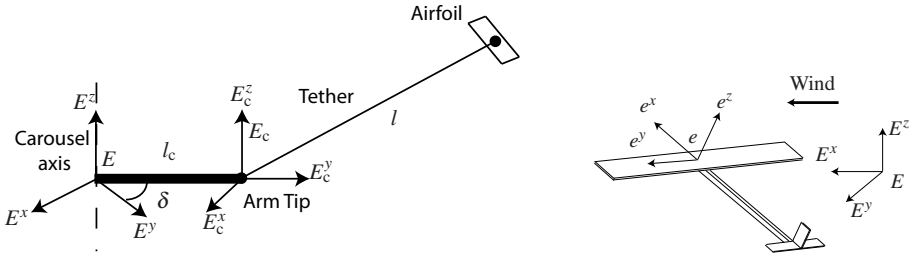


Figure 7.1: Scheme of the used reference frames. Without loss of generality, we assume that the wind blows in the E^x direction, unless differently specified.

without loss of generality, we assume throughout the chapter that the wind blows in the E^x direction, unless differently specified.

We parametrise the position and orientation of the airfoil using the coordinates of its centre of mass. A right-hand orthonormal reference frame e is attached to the airfoil centre of mass s.t. (a) the basis vector e^x spans the wing longitudinal axis, pointing in the forward direction and is aligned with the wing chord, (b) the basis vector e^z spans the vertical axis, pointing in the upward direction.

In case of rotational startup, we further define a non-inertial frame E_c attached to the carousel arm tip. Frame E_c is rotated with respect to frame E around the E^z-axis by an angle δ.

7.2.2 Multi-Airfoil System Architecture

A multi-airfoil system can be described as a set of N nodes $n \in \{0, ..., N\}$ with associated coordinate vectors $r_n \in \mathbb{R}^3$. The fixed node $r_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$ stands for the attachment point of the AWE system to the ground. The subset $\mathcal{A} \subset \{1, ..., N\}$ of the set of nodes describes the nodes associated to the airfoils. Assuming a tree structure, to each node $n \in \{1, ..., N\}$ a single tether $k = n$ is associated, and the parent node i to which the tether is attached is defined by the map $i = \mathcal{F}(n)$. The system architecture is then defined by the number of nodes N, the set $\mathcal{A}$, and the map $\mathcal{F}$. Figure 7.2 illustrates the architecture of a single and a dual-airfoil system.

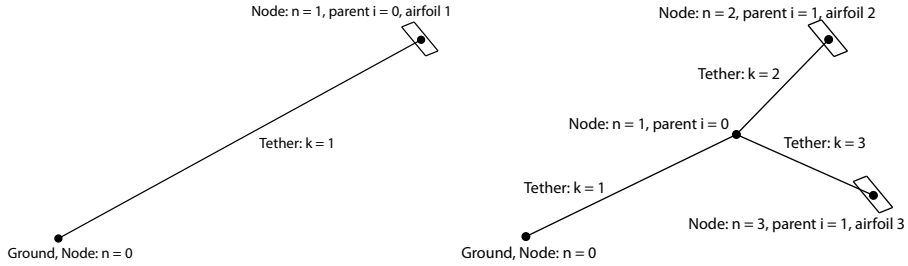


Figure 7.2: Left figure: schematic of the single-airfoil architecture, with $N = 1$, $\mathcal{A} = \{1\}$, $\mathcal{F}(1) = 0$. Right figure: schematic of the dual-airfoil architecture, with $N = 3$, $\mathcal{A} = \{2, 3\}$, $\mathcal{F}(1) = 0$, $\mathcal{F}(2) = 1$, $\mathcal{F}(3) = 1$.

7.2.3 Airfoil Aerodynamic Model

This section presents the computation of the aerodynamic forces and torques. In the following, we will express quantities relative to airfoil n, however, for the sake of simplicity, we will omit the subscript n. We define the airfoil relative velocity in frame E as

$$v = \dot{r} - w(r),$$

where w is the local wind velocity vector and r the airfoil position given in frame E. The relative velocity of the airfoil in frame e is then given by $V = R^\top v$, where $R = \begin{bmatrix} e^x & e^y & e^z \end{bmatrix}$ is the rotation matrix between frame e and frame E; unitary vectors $e^\bullet$ are aligned with axis $\bullet$ of frame e and $\bullet \in \{ x, y, z \}$. We define the components of V along axis $\bullet$ by $V^\bullet, \bullet \in \{ x, y, z \}$. The angle of attack (AoA) α and side-slip angle β are then given by

$$\alpha = -\tan^{-1}\left(\frac{V^z}{V^x}\right) = -\tan^{-1}\left(\frac{(e^z)^\top v}{(e^x)^\top v}\right), \tag{7.2a}$$

$$\beta = -\sin^{-1}\left(\frac{V^y}{\|v\|_2}\right) = -\sin^{-1}\left(\frac{(e^y)^\top v}{\sqrt{v^\top v}}\right). \tag{7.2b}$$

For small angles α and β, Equations (7.2) can be approximated as

$$\alpha \approx \frac{(e^z)^\top v}{(e^x)^\top v}, \qquad \beta \approx \frac{(e^y)^\top v}{(e^x)^\top v}. \tag{7.3}$$

Note that, in the context of optimal control or MPC, constraints on the angle of attack can be efficiently formulated as

$$\alpha \leq \overline{\alpha} \qquad \Leftrightarrow \qquad \left(\tan(\overline{\alpha})e^x + e^z\right)^\top v \geq 0, \tag{7.4}$$

where $\tan(\overline{\alpha})$ is a numeric value that can be precomputed and we avoid the unnecessary use of trigonometric functions and divisions.

The aerodynamic forces F^{A} expressed in frame E and torques T^{A} expressed in frame e are computed using a coefficient-based aerodynamic model as [82]

$$F^{\mathrm{A}} = \frac{1}{2}\rho A \left(\|v\|_2 (\underbrace{C^{\mathrm{L}} v \times e^y}_{\text{lift}} - \underbrace{C^{\mathrm{D}} v}_{\text{drag}}) - \underbrace{C^{\mathrm{S}}(e^y \times v) \times v}_{\text{side force}} \right),$$

$$T^{\mathrm{A}} = \frac{1}{2}\rho A \|v\|_2^2 \begin{bmatrix} b\, C^{\mathrm{R}} \\ c\, C^{\mathrm{P}} \\ b\, C^{\mathrm{Y}} \end{bmatrix},$$

where ρ is the local air density, A is the reference surface and b and c are the span and chord of the wing respectively.

In general, the aerodynamic coefficients are functions of the AoA α, side-slip angle β and control surface deflections a^{e}, a^{a}, a^{r} for the elevator, ailerons and rudder respectively. The torque coefficients moreover depend on the ratio of the rotational velocity and translational velocity $\frac{\omega}{\|v\|_2}$. In the following equation, all symbols in the right hand side which are not the specified arguments are fixed numbers which depend on the aerodynamic properties of the aircraft. The aerodynamic coefficients are then typically given by [82]

$$C^{\mathrm{L}} = C^{\mathrm{L}_0} + C^{\mathrm{L}\alpha}\alpha + C^{\mathrm{Le}} a^{\mathrm{e}},$$

$$C^{\mathrm{D}} = C^{\mathrm{D}_0} + C^{\mathrm{D}\alpha}\alpha + C^{\mathrm{D}\alpha_2}\alpha^2 + C^{\mathrm{D}\beta_2}\beta^2 + C^{\mathrm{D}_{\mathrm{e}}} a^{\mathrm{e}} + C^{\mathrm{De}_2}(a^{\mathrm{e}})^2 + C^{\mathrm{D}\alpha\mathrm{e}}\alpha a^{\mathrm{e}} + C^{\mathrm{D}_{\mathrm{a}}} a^{\mathrm{a}} + C^{\mathrm{Da}_2}(a^{\mathrm{a}})^2 + C^{\mathrm{D}\beta\mathrm{a}}\beta a^{\mathrm{a}} + C^{\mathrm{Dr}_2}(a^{\mathrm{r}})^2 + C^{\mathrm{D}\beta\mathrm{r}}\beta a^{\mathrm{r}},$$

$$C^{\mathrm{S}} = C^{\mathrm{S}\beta}\beta + C^{\mathrm{Sr}} a^{\mathrm{r}},$$

$$\begin{bmatrix} C^{\mathrm{R}} \\ C^{\mathrm{P}} \\ C^{\mathrm{Y}} \end{bmatrix} = \begin{bmatrix} C^{\mathrm{Ra}} a^{\mathrm{a}} + C^{\mathrm{R}\beta}\beta + C^{\mathrm{R}\alpha\beta}\alpha\beta \\ C^{\mathrm{P}_0} + C^{\mathrm{P}_{\mathrm{e}}} a^{\mathrm{e}} + C^{\mathrm{P}\alpha}\alpha \\ C^{\mathrm{Yr}} a^{\mathrm{r}} + C^{\mathrm{Y}\beta}\beta + C^{\mathrm{Y}\alpha\beta}\alpha\beta \end{bmatrix} + D \begin{bmatrix} b\,\omega^x \\ c\,\omega^y \\ b\,\omega^z \end{bmatrix} \frac{1}{2\|v\|_2},$$

Some simplified aerodynamic models might neglect some contributions and only consider a reduced aerodynamic model.

For AWE systems in drag mode, the generators are airborne and an additional drag force due to the turbines needs to be added

$$F^{\mathrm{G}} = -\frac{1}{2}\rho A \kappa \|v\|_2 v, \tag{7.5}$$

where κ is a controlled variable which indicates how much energy is extracted from the airmass. Instead of directly controlling κ, it might be preferable to instead control u^κ and introduce the additional equation $\dot{\kappa} = u^\kappa$ to the system dynamics.

7.2.4 Tether Model

The assumption of a rigid tether is clearly only valid in special cases, e.g. a light tether subject to a strong tension. If this assumption is not too inaccurate, the tether can be modelled as a rigid rod with mass [59]. Otherwise, it is necessary to model the tether more accurately in order to account for the tether dynamics appropriately. In order to account for tether sagging and flexibility, a finite element model can be used. In this model, the tether is split into N nodes connected by $N-1$ rigid links.

When discussing a full tethered airfoil system, we would need to identify each node of the tether as node j on tether k. For simplicity of notation, in this subsection we will drop the index relative to the tether so that the position $r_{k,j}$ of element j in tether k will be denoted as $r_j = \begin{bmatrix} x_j & y_j & z_j \end{bmatrix}^\top$ in frame E. Link j has length l_j and connects nodes j and $j-1$.

We denote the tether linear density by μ. Two models are possible, (a) a lumped mass model in which the tether mass is distributed on the nodes and the links are massless, and (b) a full model in which the nodes are massless and the tether mass is distributed over the links. Note that, by picking a full model with $N = 2$ nodes, one obtains the rigid rod model.

For the lumped mass model, the tether kinetic and potential energy are given by

$$\mathcal{T}^{\mathrm{T}} = \sum_{j=1}^{N-1} \frac{1}{2}\mu l_j \dot{r}_j^\top \dot{r}_j + \frac{1}{4}\mu l_0 \dot{r}_0^\top \dot{r}_0 + \frac{1}{4}\mu l_N \dot{r}_N^\top \dot{r}_N, \tag{7.6a}$$

$$\mathcal{V}^{\mathrm{T}} = \sum_{j=1}^{N-1} \mu l_j g z_j + \frac{1}{2}\mu l_0 g z_0 + \frac{1}{2}\mu l_N g z_N. \tag{7.6b}$$

For the full model, the tether kinetic and potential energy are given by

$$\mathcal{T}^{\mathrm{T}} = \sum_{j=1}^{N} \frac{1}{6}\mu l_j \left(\dot{r}_j^\top \dot{r}_j + \dot{r}_j^\top \dot{r}_{j-1} + \dot{r}_{j-1}^\top \dot{r}_{j-1}\right), \tag{7.7a}$$

$$\mathcal{V}^{\mathrm{T}} = \sum_{j=1}^{N} \mu l_j g \left(z_j + z_{j-1}\right). \tag{7.7b}$$

Both models include the following constraints:

$$\mathcal{C}^{\mathrm{T}} = \frac{1}{2}\begin{bmatrix} (r_1 - r_0)^\top (r_1 - r_0) - l_1^2 \\ \cdots \\ (r_N - r_{N-1})^\top (r_N - r_{N-1}) - l_N^2 \end{bmatrix}. \tag{7.8}$$

The main differences between the two approaches regard the accuracy and computational effort. The lumped mass model yields a diagonal mass matrix, while the full model yields a banded mass matrix. On the other hand, if the tether is discretised using the same amount of elements, the full model is more accurate. The decision on which of the two models to prefer therefore depends on several considerations regarding the needed accuracy and the numerical scheme used for simulation or optimal control. Note that modelling the tether as a rigid link with mass is equivalent to a full model using only one element in the discretisation.

For AWE systems, the tethers are typically made of Dyneema®, which is light and has an extremely high Elasticity Modulus. Therefore, modelling the tether sections as springs is to be avoided for two main reasons: (a) due to the high stiffness, the length of each tether section is constant and can be modelled by a constraint and (b) the spring model would be extremely stiff, making the simulation and optimisation of the system numerically hard and computationally intensive.

An accurate tether drag model computes the force acting on the nodes by evaluating the integral over the tether section of the infinitesimal force acting on each infinitesimal tether section, i.e.

$$F_{j,j-1}^{\mathrm{S}} = \int_0^1 \sigma F_j(\sigma)\,\mathrm{d}\sigma, \qquad F_{j,j}^{\mathrm{S}} = \int_0^1 (1-\sigma) F_j(\sigma)\,\mathrm{d}\sigma, \tag{7.9a}$$

$$F_j^{\mathrm{T}} = F_{j,j}^{\mathrm{S}} + F_{j+1,j}^{\mathrm{S}}, \tag{7.9b}$$

where $F_j(\sigma)$ is the aerodynamic force acting on the infinitesimal tether section located at position $\sigma r_{j-1} + (1-\sigma) r_j$. By assuming (a) that the wind is orthogonal

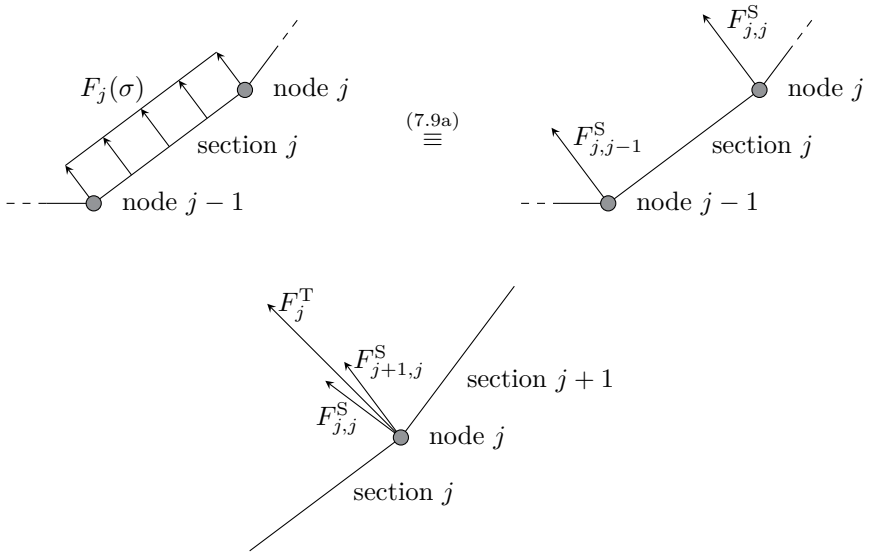


Figure 7.3: Tether drag computation. The force at each node is given by the sum the contributions of all tether sections connected to it, see Equation (7.9).

to the tether and (b) that the tether produces no lift, one obtains

$$F_j(\sigma)\mathrm{d}\sigma = -\frac{1}{2}\rho d l_j C^{\mathrm{D,S}} \|v_j\| v_j \,\mathrm{d}\sigma,$$

$$v_j = \sigma \dot{r}_{j-1} + (1-\sigma)\dot{r}_j + w(z), \qquad z = \sigma z_{j-1} + (1-\sigma) z_j,$$

where $C^{\mathrm{D,S}}$ is the tether drag coefficient, d the tether diameter and ρ the local air density.

Note that in the general case the integrals in Equation (7.9) do not have an analytical solution. They can therefore be evaluated numerically using a quadrature formula. Clearly, the numerical complexity of such a model increases significantly. However, it has been shown in [126] that such a drag model can still be used for real-time applications.

An approximate model for the drag force acting on each section has been proposed in [119]. It assumes a constant force on the whole tether section, equal to the force in the midpoint:

$$F_j^{\mathrm{S,m}} = -\frac{1}{2}\rho d l_j C^{\mathrm{D,S}} \|\bar{v}_j\| \bar{v}_j, \tag{7.10}$$

where d is the tether diameter and

$$\bar{v}_j = \frac{\dot{r}_{j-1} + \dot{r}_j}{2} - w\left(\frac{z_{j-1} + z_j}{2}\right).$$

The tether drag is then modelled by forces acting on the nodes, given by

$$F_j^{\mathrm{T}} = \frac{F_j^{\mathrm{S,m}} + F_{j+1}^{\mathrm{S,m}}}{2}. \tag{7.11}$$

Note that the approximated formula (7.10) is equivalent to applying the rectangle rule to (7.9) with one integration interval where the function is evaluated in the middle of the integration interval.

A more efficient model having the same order of accuracy can be obtained by evaluating the drag force directly at the nodes, i.e.

$$F_j^{\mathrm{T}} = -\frac{1}{2}\rho d l_j C^{\mathrm{D,S}} \|v_j\| v_j, \qquad v_j = \dot{r}_j - w\left(z_j\right). \tag{7.12}$$

This is equivalent to applying the rectangle rule to (7.9) with one integration interval where the function is evaluated at the end of the integration interval.

A more accurate model can be obtained by using

$$F_j^{\mathrm{S,t}} = -\frac{1}{4}\rho d l_j C^{\mathrm{D,S}} \left(\|v_j\| v_j + \|v_{j+1}\| v_{j+1}\right), \qquad v_j = \dot{r}_j - w\left(z_j\right). \tag{7.13}$$

Note that (7.13) is equivalent to applying the trapezoidal quadrature rule to (7.9) with one integration interval, which has a higher order of accuracy than (7.10).

While for drag-mode airfoils the tether length does not change, for pumping-mode airfoils the tether length varies and therefore mass enters and leaves the system with a nonzero speed. This fact should be taken into account by adding a correction term in the Lagrangian formulation as explained in [85]. Because the tether is very light and the tether velocity is not extremely high, it is reasonable to neglect the effect on the system dynamics due to mass entering or leaving the system.

7.2.5 Examples of Model Derivation

In this subsection we present the derivation of three models, all using a rigid rod for the tether, i.e. a full FEM model with only one element per tether. Clearly, FEM tether models with more than one element can also be derived within the proposed framework, but the full derivation is omitted for the sake

of brevity. In this derivation, we do not detail the specific aerodynamic model, as any aerodynamic model of the form presented in Section 7.2.3 can be used to compute the aerodynamic forces and torques.

Single Airfoil

In this section, we derive the model of a single airfoil where the tether is assumed to be a rigid rod. We make use of the methods described in Chapter 6 to obtain an index 1 DAE.

The tether is considered here as a rigid link whose length l can be variable when considering a pumping-mode system, in which case we assume that the tether acceleration $\ddot{l}$ is directly controlled. The generalised coordinates are given by $q = \{r, R\}$, where $r = \begin{bmatrix} x & y & z \end{bmatrix}$ denotes the position of the airfoil centre of mass in frame E, the DCM R denotes the relative orientation of frame e with respect to frame E, i.e. $R = \begin{bmatrix} e^x & e^y & e^z \end{bmatrix}$, where vectors $e^{x,y,z}$ are given in frame E. We do not include the tether length l in the generalised coordinates, since either (drag mode) it is constant or (pumping mode) its dynamics are trivially obtained by double integration of $\ddot{l}$.

The point at which the tether is attached to the airfoil in the body frame e is fixed and given by r_{T}. The tether constraint is then given by

$$\mathcal{C} = \frac{1}{2}\left((r + R\,r_{\mathrm{T}})^{\top}(r + R\,r_{\mathrm{T}}) - l^2\right),$$

and is the only constraint of the system. Then we obtain

$$2\,\mathcal{P}_R(\nabla_R \mathcal{C}) = 2\,\mathcal{P}_R(\nabla_R(r^{\top} R r_{\mathrm{T}})) = 2\,\mathcal{P}_R(r r_{\mathrm{T}}^{\top}) = 2\mathcal{U}(R^{\top} r r_{\mathrm{T}}^{\top}) = r_{\mathrm{T}} \times R^{\top} r,$$

$$\nabla_r \mathcal{C} = r + R\,r_{\mathrm{T}}.$$

The kinetic and potential energy are given by

$$\mathcal{T} = \frac{1}{2}\left(m_{\mathrm{A}} + \frac{1}{3}\mu l\right)\dot{r}^{\top}\dot{r} + \frac{1}{2}\omega^{\top} J \omega, \qquad \mathcal{V} = \left(m_{\mathrm{A}} + \frac{1}{2}\mu l\right) g z,$$

where m_{A}, μ, g and J denote respectively the airfoil mass, tether linear density, gravitational acceleration and airfoil moment of inertia in frame e. Defining

$\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda \mathcal{C}$ and using (6.20), after index reduction we obtain

$$\begin{bmatrix} \left(m_{\mathrm{A}} + \frac{1}{3}\mu l\right) I & 0 & r + R\, r_{\mathrm{T}} \\ 0 & J & r_{\mathrm{T}} \times R^{\top} r \\ (r + R\, r_{\mathrm{T}})^{\top} & (r_{\mathrm{T}} \times R^{\top} r)^{\top} & 0 \end{bmatrix} \begin{bmatrix} \ddot{r} \\ \dot{\omega} \\ \lambda \end{bmatrix} =$$

$$= \begin{bmatrix} F - \left(m_{\mathrm{A}} + \frac{1}{2}\mu l\right) g E^{z} - \frac{1}{3}\mu \dot{l} \dot{r} \\ T - \omega \times J\omega \\ -(\nabla_r \dot{\mathcal{C}})^{\top} \dot{r} - 2\,\mathcal{P}_R\left(\nabla_R \dot{\mathcal{C}}\right)^{\top} \omega + \dot{l}^2 + l\ddot{l} - 2\gamma_{\mathcal{C}}\, \dot{\mathcal{C}} - \gamma_{\mathcal{C}}^2 \mathcal{C} \end{bmatrix}, \tag{7.14a}$$

$$\dot{R} = R\omega_{\times} + \frac{\gamma_R}{2} RS, \qquad \dot{a} = \begin{bmatrix} u^{\mathrm{a}} & u^{\mathrm{e}} & u^{\mathrm{r}} \end{bmatrix}, \tag{7.14b}$$

where $a = \begin{bmatrix} a^{\mathrm{a}} & a^{\mathrm{e}} & a^{\mathrm{r}} \end{bmatrix}^{\top}$, a^{a}, a^{e}, and a^{r} denote the deflections of the ailerons, elevator and rudder respectively, S is chosen according to (6.28) or (6.29),

$$\dot{\mathcal{C}} = (r + Rr_{\mathrm{T}})^{\top} \dot{r} + (r_{\mathrm{T}} \times R^{\top} r)^{\top} \omega + l\dot{l},$$

$$\nabla_r \dot{\mathcal{C}} = \dot{r} + R(\omega \times r_{\mathrm{T}}),$$

$$2\,\mathcal{P}_R\left(\nabla_R \dot{\mathcal{C}}\right) = 2\,\mathcal{P}_R\left(r(\omega \times r_{\mathrm{T}})^{\top} + \dot{r} r_{\mathrm{T}}^{\top}\right) = -\left(R^{\top} r\right) \times (\omega \times r_{\mathrm{T}}) - \left(R^{\top} \dot{r}\right) \times r_{\mathrm{T}}.$$

Note that variables $\gamma_R, \gamma_{\mathcal{C}} \geq 0$ are arbitrarily chosen so as to stabilise the model invariants. For MPC, MHE and short horizon optimal control problems, invariant stabilisation might not be needed and the relative terms in the system equations can be removed so as to reduce the computational complexity.

In addition to the system dynamics, the consistency conditions

$$\mathcal{C} = 0, \qquad \dot{\mathcal{C}} = 0, \qquad R^{\top} R - I = 0 \tag{7.15}$$

must be enforced at some point of the trajectory.

Pumping vs Drag Mode In case of a pumping-type system, the mechanical power extracted by the system is given by $P = \dot{l} F^{\mathrm{t}} = l\dot{l}\lambda$, where $F^{\mathrm{t}} = l\lambda$ is the tether tension. Then, assuming that one directly controls the tether acceleration, the equation $\ddot{l} = u^{l}$ is added to the model. If instead, one models a drag-type system, the tether length l is fixed and the power extracted by the system is given by $P = v^{\top} F^{\mathrm{G}}$, with F^{G} from (7.5). Then the equation $\dot{\kappa} = u^{\kappa}$ is added to the model.

Single Airfoil for Rotational Startup

A model for rotational startup can easily be derived following the previously presented procedure. However, the coordinates parametrising the plane position can be chosen to be given (a) in the inertial frame E attached to the ground, or (b) in the non-inertial frame E_c attached to the carousel arm tip.

Clearly, both approaches yield valid model equations, however, approach (a) has the disadvantage that, even if the plane is flying at a steady configuration with respect to the carousel, the coordinates r and DCM R undergo periodic cycles at the frequency of the carousel rotation. This adds unnecessary complexity to the model equations and makes the numerical integration of the model more difficult. We will therefore consider approach (b).

A natural choice for describing the orientation of the carousel would be to use angle δ. For practical reasons, however, this is not advisable, as by integrating over time, the angle will keep increasing to very high values, which could cause numerical problems. Moreover, when measuring angle δ using an encoder, once the full scale is reached, the value is reset to 0 and special care needs to be taken. An alternative choice which solves these issues is to parametrise the carousel orientation with a 2 dimensional DCM, i.e.

$$R_\mathrm{c} = \begin{bmatrix} E_\mathrm{c}^x & E_\mathrm{c}^y & E_\mathrm{c}^z \end{bmatrix} = \begin{bmatrix} e_\mathrm{c}^x & -e_\mathrm{c}^y & 0 \\ e_\mathrm{c}^y & e_\mathrm{c}^x & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad (e_\mathrm{c}^x)^2 + (e_\mathrm{c}^y)^2 - 1 = 0, \tag{7.16a}$$

$$\begin{bmatrix} \dot{e}_\mathrm{c}^x \\ \dot{e}_\mathrm{c}^y \end{bmatrix} = \begin{bmatrix} -\dot{\delta} e_\mathrm{c}^y \\ \dot{\delta} e_\mathrm{c}^x \end{bmatrix}. \tag{7.16b}$$

The time derivative of the DCM R is given by

$$\dot{R} = R\tilde{\omega}_\times, \tag{7.17}$$

where ω is the aircraft angular velocity given in frame e, and $\tilde{\omega} = \omega - R^\top E^z \dot{\delta}$ is the angular velocity between the airfoil reference frame e and the carousel reference frame E_c, given in the airfoil reference frame e, so that

$$\omega = \mathcal{P}_R(\dot{R}) + R^\top E^z \dot{\delta}.$$

Note that Equation (7.16b) does not include the invariant stabilisation for the DCM R_c, but its stabilisation can be easily be obtained by applying the

methodology described in Chapter 6. Note that, when including the invariant stabilisation, Equation (7.16b) has the same form of Equation (5.50).

We define the carousel arm length l_c, the aircraft position in the absolute reference E frame

$$r_A = R_c \begin{bmatrix} l_c + x \\ y \\ z \end{bmatrix},$$

the carousel moment of inertia J_c, the airfoil mass m and moment of inertia J and the gravitational acceleration g. Then, the kinetic and potential energy are given by

$$\mathcal{T} = \frac{1}{2} J_c \dot{\delta}^2 + \frac{1}{2} m \dot{r}_A^\top \dot{r}_A + \frac{1}{2} \omega^\top J \omega, \qquad \mathcal{V} = mgz.$$

We moreover define the point at which the tether is attached to the airfoil in the body frame e as r_T and the tether length l. Then the constraint is given by

$$\mathcal{C} = \frac{1}{2} \left((r + R r_T)^\top (r + R r_T) - l^2 \right).$$

The derivation of the system dynamics is then obtained by applying the same framework used before and is omitted for the sake of brevity.

Dual Airfoils

In this section we present a model for a dual-airfoil AWE system in which two airfoils are attached to a Y-shaped tether. The tree-like structure of the system is composed of three nodes: node 0, where the secondary tethers are attached to the main tether and nodes $1, 2$, corresponding to the centre of mass of each airfoil. An orthonormal right-hand reference frame E is chosen such that vector E^z is opposed to the gravitational acceleration vector g.

We choose to parametrise the position and orientation of the airfoils using the coordinates of their centres of mass. A right-hand orthonormal reference frame e_i is attached to the centre of mass of airfoil i s.t. (a) the basis vector e_i^x spans the wing longitudinal axis, pointing in the forward direction and is aligned with the wing chord, (b) the basis vector e_i^z spans the vertical axis, pointing in the upward direction.

The generalised coordinates are given by $q = \{r_0, r_1, r_2, R_1, R_2\}$, where vectors $r_i = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}$ denote the position of node i in frame E, the DCM R_i

denote the relative orientation of frame e_i with respect to frame E, i.e. $R_i = \begin{bmatrix} e_i^x & e_i^y & e_i^z \end{bmatrix}$, where vectors $e_i^{x,y,z}$ are given in frame E.

The point at which the tether is attached to the airfoil i in the body frame e_i is fixed and given by r_{T}. The tether constraint is then given by We model the tethers as rigid rods with mass. The system constraint vector $\mathcal{C}$ has therefore three components, given by

$$\mathcal{C}_0 = \frac{1}{2}\left(r_0^\top r_0 - l_0^2\right),$$

$$\mathcal{C}_i = \frac{1}{2}\left((r_i + R_i r_{\mathrm{T}} - r_0)^\top (r_i + R_i r_{\mathrm{T}} - r_0) - l_i^2\right), \qquad i = 1, 2.$$

The kinetic and potential energies are given by

$$\mathcal{T} = \underbrace{\frac{1}{6}\mu_0 l_0 \dot{r}_0^\top \dot{r}_0}_{\text{Main tether}} + \underbrace{\frac{1}{6}\sum_{i=1}^{2} \mu_i l_i \left(\dot{r}_0^\top \dot{r}_0 + \dot{r}_i^\top \dot{r}_i + \dot{r}_0^\top \dot{r}_i\right)}_{\text{Secondary tethers}}$$

$$+ \underbrace{\frac{1}{2}\sum_{i=1}^{2} m_i \dot{r}_i^\top \dot{r}_i}_{\text{Airfoils, translation}} + \underbrace{\frac{1}{2}\sum_{i=1}^{2} \omega_i^\top J_i \omega_i}_{\text{Airfoils, rotation}},$$

$$\mathcal{V} = \underbrace{\frac{1}{2}\mu_0 l_0 g (E^z)^\top r_0}_{\text{Main tether}} + \underbrace{\frac{1}{2}\sum_{i=1}^{2} \mu_i l_i g (E^z)^\top (r_0 + r_i)}_{\text{Secondary tethers}} + \underbrace{\sum_{i=1}^{2} m_i g (E^z)^\top r_i}_{\text{Airfoils}},$$

where m_i, μ_i and J_i denote respectively the mass of airfoil i, linear density of tether i and the moment of inertia of airfoil i in frame e_i.

Defining $\mathcal{L} = \mathcal{T} - \mathcal{V} - \lambda^\top \mathcal{C}$ and using (6.20), after index reduction we obtain

$$\begin{bmatrix} M & G \\ G^\top & 0 \end{bmatrix} \begin{bmatrix} \ddot{r} \\ \dot{\omega} \\ \lambda \end{bmatrix} = F, \qquad \ddot{r} = \begin{bmatrix} \ddot{r}_0 \\ \ddot{r}_1 \\ \ddot{r}_2 \end{bmatrix}, \qquad \dot{\omega} = \begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \end{bmatrix}, \tag{7.18a}$$

$$a_i = \begin{bmatrix} a_i^{\mathrm{a}} & a_i^{\mathrm{e}} & a_i^{\mathrm{r}} \end{bmatrix}^\top, \qquad \dot{a}_i = \begin{bmatrix} u_i^{\mathrm{a}} & u_i^{\mathrm{e}} & u_i^{\mathrm{r}} \end{bmatrix}^\top, \qquad i = 1, 2, \tag{7.18b}$$

$$R_i(0)^\top R_i(0) - I = 0, \qquad \dot{R}_i = R_i\left((\omega_i)_\times + \frac{\gamma_R}{2} S_i\right), \quad i = 1, 2, \tag{7.18c}$$

where S_i is chosen according to (6.28) or (6.29) and we define $\xi_i = \frac{1}{3}\mu_i l_i I$ and

$$M = \begin{bmatrix} \sum_{i=0}^{3} \xi_i & \frac{1}{2}\xi_1 & \frac{1}{2}\xi_2 & 0 & 0 \\ \frac{1}{2}\xi_1 & \xi_1 + m_1 I & 0 & 0 & 0 \\ \frac{1}{2}\xi_2 & 0 & \xi_2 + m_2 I & 0 & 0 \\ 0 & 0 & 0 & J_1 & 0 \\ 0 & 0 & 0 & 0 & J_2 \end{bmatrix},$$

$$G = \begin{bmatrix} r_0 & \nabla_{r_0}\mathcal{C}_1 & \nabla_{r_0}\mathcal{C}_2 \\ 0 & \nabla_{r_1}\mathcal{C}_1 & 0 \\ 0 & 0 & \nabla_{r_2}\mathcal{C}_2 \\ 0 & 2\,\mathcal{P}_{R_1}\left(\nabla_{R_1}\mathcal{C}_1\right) & 0 \\ 0 & 0 & 2\,\mathcal{P}_{R_2}\left(\nabla_{R_2}\mathcal{C}_2\right) \end{bmatrix},$$

$$F = \begin{bmatrix} F_0 - \sum_{i=0}^{2} \frac{1}{2} g \mu_i l_i E^z \\ F_1 - \left(\frac{1}{2} g \mu_1 l_1 + g m_1\right) E^z \\ F_2 - \left(\frac{1}{2} g \mu_2 l_2 + g m_2\right) E^z \\ T_1 - \omega_1 \times J_1 \omega_1 \\ T_2 - \omega_2 \times J_2 \omega_2 \\ -\dot{r}_0^\top \dot{r}_0 - 2\gamma_{\mathcal{C}}\,\dot{\mathcal{C}}_0 - \gamma_{\mathcal{C}}^2 \mathcal{C}_0 + \dot{l}_0^2 + l_0 \ddot{l}_0 \\ -(\nabla_{r_0}\dot{\mathcal{C}}_1)^\top \dot{r}_0 - (\nabla_{r_1}\dot{\mathcal{C}}_1)^\top \dot{r}_1 - 2\,\mathcal{P}_{R_1}\left(\nabla_{R_1}\dot{\mathcal{C}}_1\right)^\top \omega_1 - 2\gamma_{\mathcal{C}}\,\dot{\mathcal{C}}_1 - \gamma_{\mathcal{C}}^2 \mathcal{C}_1 \\ -(\nabla_{r_0}\dot{\mathcal{C}}_2)^\top \dot{r}_0 - (\nabla_{r_2}\dot{\mathcal{C}}_2)^\top \dot{r}_2 - 2\,\mathcal{P}_{R_2}\left(\nabla_{R_2}\dot{\mathcal{C}}_2\right)^\top \omega_2 - 2\gamma_{\mathcal{C}}\,\dot{\mathcal{C}}_2 - \gamma_{\mathcal{C}}^2 \mathcal{C}_2 \end{bmatrix}.$$

The sum of the forces applied to node i is denoted by F_i. Similarly, the sum of the torques applied to airfoil i is denoted by T_i. In the above expressions, for $i = 1, 2$, it holds that

$$\dot{\mathcal{C}}_0 = r_0^\top \dot{r}_0, \qquad \dot{\mathcal{C}}_i = (r_i + R_i r_{\mathrm{T}} - r_0)^\top (\dot{r}_i + R_i \omega_i \times r_{\mathrm{T}} - \dot{r}_0),$$

$$-\nabla_{r_0}\mathcal{C}_i = \nabla_{r_i}\mathcal{C}_i = (r_i + R_i r_{\mathrm{T}} - r_0),$$

$$-\nabla_{r_0}\dot{\mathcal{C}}_i = \nabla_{r_i}\dot{\mathcal{C}}_i = (\dot{r}_i + R_i \omega_i \times r_{\mathrm{T}} - \dot{r}_0),$$

$$2\,\mathcal{P}_{R_i}(\nabla_{r_i}\mathcal{C}_i) = r_{\mathrm{T}} \times R_i^\top (r_i - r_0),$$

$$2\,\mathcal{P}_{R_i}(\nabla_{r_i}\dot{\mathcal{C}}_i) = R_i^\top (r_i - r_0) \times (\omega \times r_{\mathrm{T}}) + R_i^\top (\dot{r}_i - \dot{r}_0) \times r_{\mathrm{T}}.$$

In addition to the system dynamics, the consistency conditions

$$\mathcal{C}_i = 0, \quad \dot{\mathcal{C}}_i = 0, \quad i = 0, 1, 2, \quad \text{and} \quad R_i^\top R_i - I = 0, \quad i = 1, 2 \tag{7.19}$$

must be enforced at some point of the trajectory.

Pumping vs Drag Mode In case of a pumping-type system, the mechanical power extracted by the system is given by $P = \dot{l}_0 F^{\mathrm{t}} = l_0 \dot{l}_0 \lambda_0$, where $F^{\mathrm{t}} = l_0 \lambda_0$ is the tether tension. Then, assuming that one directly controls the tether acceleration, the equation $\ddot{l}_0 = u_0^l$ is added to the model. If instead, one models a drag-type system, the tether length l_0 is fixed and the power extracted by each airfoil is given by $P_i = v_i^\top F_i^{\mathrm{G}}$, with F^{G} from (7.5). Then the equation $\dot{\kappa}_i = u_i^\kappa$ is added to the model.

7.3 Offline and Online OCP Formulation

As anticipated, optimal trajectory planning, control using MPC and estimation using MHE require one to formulate and solve optimal control problems. When dealing with nonconvex optimisation problems, fast and reliable convergence depend on (a) using efficient and tailored algorithms, (b) deriving an optimisation-friendly model, (c) formulating the optimisation problem appropriately and (d) providing a good initial guess to the solver.

The trajectory planning, control and estimation OCPs have a rather different nature and are characterised by specific difficulties. The first can be performed offline but a good initial guess is not easy to construct. Sophisticated NLP solvers, e.g. interior point or SQP methods using a good globalisation strategy are typically necessary. Even when using excellent NLP solvers, a feasible initial guess needs to be pre-computed in order to guarantee reliable convergence.

The MPC and MHE OCPs need instead to be solved online in real-time and an excellent initial guess can be constructed using the solution at the previous time instant, which is the framework for which the RTI scheme has been designed. However, in the presence of state constraints feasibility issues might arise in a perturbed environment and a clever problem formulation is needed in order to guarantee feasibility of the NLP at all times.

Numerical algorithms for optimal control have been presented in Chapter 2 and optimisation-friendly models have been derived in Chapter 6. In this section we therefore focus on the problem formulation and initial guess computation. In the following we will propose strategies to deal with (a) ensuring a reliable convergence of the offline OCPs, (b) ensuring that MPC always remains

feasible, and (c) imposing the model invariants to guarantee that the trajectories predicted by the model have physical meaning.

7.3.1 A Relaxation Strategy for OCPs

Optimisation is currently used by the AWE community to address simple design problems. However, a significant research effort is still needed to develop tools that can be reliably used for the optimisation of AWE systems based on complex, high-fidelity models. Such models yield strongly nonlinear dynamics, resulting in strongly non-convex optimal control problems (OCPs). Solving non-convex OCPs using derivative-based optimisation techniques requires initial guesses that are close to feasibility. However, in practice, a good initial guess is seldom available.

Let us consider the problem of computing energy-optimal orbits, i.e.

$$E = \min_{x,u,z,p} \ \frac{1}{T}\int_0^T P \ \mathrm{d}t \tag{7.20a}$$

$$\text{s.t.} \ \ g_{\mathrm{p}}(x(0), x(T)) = 0, \tag{7.20b}$$

$$f^{\mathrm{c}}(\dot{x}, x, z, u, p) = 0, \tag{7.20c}$$

$$h(x, z, u, p) \geq 0, \tag{7.20d}$$

where P is the power extracted from the airmass, the final time T is included in the vector p of parameters to be optimised, and g_{p} is a function which enforces both the periodicity conditions and that the invariants must be at their equilibrium while guaranteeing that LICQ is not violated. Clearly, if both conditions are enforced simultaneously, some constraints are redundant because the invariant dynamics are not controllable and the invariants must always be at their equilibrium. Enforcing periodicity of all states would therefore also enforce periodicity of the invariants, which corresponds to enforcing that the invariants must be at their equilibrium a second time.

Problem (7.20) is nonconvex and needs a good initial guess in order for the numerical schemes to converge. Such a guess is, however, not readily available. To tackle that issue, one can first solve a modified problem, where the dynamic constraints resulting from the physical model are relaxed by the introduction of fictitious forces and moments at critical stages of the dynamics. This strategy allows to dramatically reduce the model nonlinearities and couplings. As a result, Newton-type techniques converge reliably for the relaxed problem even with a poor (infeasible) initial guess. The discrepancy between the dynamic

constraints of the relaxed problem and of the physical model can be adjusted via a single parameter. Starting from the fully relaxed problem, a homotopy procedure is then applied, where the relaxation parameter is gradually modified so that at the end of the homotopy the dynamic constraints of the relaxed problem match the original ones.

The system dynamics of AWE systems are typically a chain of four nonlinear integrators, with several feedback loops occurring at different stages of the chain, as depicted in Figure 7.4. The feedback loops result from the aerodynamic forces and moments, which are strongly nonlinear functions of the state of the system. The dynamics resulting from the interaction of the integrators with the aerodynamic feedback loops are highly coupled, and strongly nonlinear.

For a practitioner, the a priori knowledge of the optimal trajectory of a specific AWE system is likely to be limited to its topology (e.g. circular or eight) and an educated guess of some variables such as the wing velocity, the cable length, the turn radius, and the operational altitude. We therefore suggest in the following a procedure to compute an initial guess for Problem (7.20) based on such limited information only.

From an arbitrary guess of the wing position over time $X_{\mathrm{g}}(t)$, with $X_{\mathrm{g}}(0) = X_{\mathrm{g}}(T)$ and $C(X_{\mathrm{g}}(t)) = 0,\ \forall\, t \in [0, T]$ the states are initialised such that:

1. the wing longitudinal axis e_{g}^{x} is aligned with the wing absolute velocity $\dot{X}_{\mathrm{g}}$. The angle of attack α_{g} is yielded by the component of the wing relative velocity due to the wind only, and is therefore small if the absolute velocity is chosen reasonably high.
2. the wing vertical axis e_{g}^{z} is aligned with the tether, such that the lift mainly acts in the direction of the tether axis
3. the wing angular velocity ω_{g} is initialised by taking the numerical derivative of the pose of the wing R_{g}
4. the tether length is fixed, the wing control surfaces are neutral

These requirements can be formally stated as follows:

$$e_{\mathrm{g}}^{x} = \|\dot{X}_{\mathrm{g}}\|^{-1}\, \dot{X}_{\mathrm{g}}, \qquad e_{\mathrm{g}}^{z} = r_{\mathrm{g}}^{-1}\, X_{\mathrm{g}}, \qquad e_{\mathrm{g}}^{y} = e_{\mathrm{g}}^{z} \times e_{\mathrm{g}}^{x},$$

$$\omega_{\mathrm{g}} = \left(R_{\mathrm{g}}^{\top} \dot{R}_{\mathrm{g}}\right)_{\times^{-1}}, \qquad r_{\mathrm{g}} = r_0, \qquad \dot{r}_{\mathrm{g}} = 0, \qquad a_{\mathrm{g}} = 0,$$

where $a_{\mathrm{g}} = \begin{bmatrix} a_{\mathrm{g}}^{\mathrm{a}} & a_{\mathrm{g}}^{\mathrm{e}} & a_{\mathrm{g}}^{\mathrm{r}} & \kappa_{\mathrm{g}} \end{bmatrix}^{\top}$ and $A_{\times^{-1}} \in \mathbb{R}^3$ is the vector yielded by the skew symmetric matrix $A \in SO(3)$. Note that for drag mode, r_{g} is fixed and thus not part of the model and for pumping mode κ_{g} is not part of the model.

Attempts at solving Problem (7.20) with the proposed initial guess using both (a) a collocation-based [17] discretisation of Problem (7.20) and the NLP solver Ipopt [114] and (b) the software ACADO [60] based on Multiple-Shooting [22] and Sequential Quadratic Programming [23] have led to the failure of the NLP solvers at a point where feasibility cannot be improved. The failure occurred regardless of the accuracy of discretisation.

Because the nonlinearities of the model dynamics are mainly due to the feedback loops introduced by the aerodynamic forces and torques, a system where these feedback loops are open (see Fig. 7.5) is decoupled and mildly nonlinear. The aerodynamic forces and torques are then replaced by fictitious forces F^{F} and torques T^{F} and the computation of a feasible trajectory becomes a much simpler problem to solve.

This observation is exploited here for developing a relaxed dynamic model where the feedback loops can be initially fully open and subsequently closed progressively. In order to be able to implement this strategy, we introduce a parameter $\gamma \in [0,1]$, and introduce the forces and torques as a convex combination of the aerodynamic and fictitious ones (see Figure 7.6), i.e.

$$F = \gamma F^{\mathrm{A}} + (1-\gamma)F^{\mathrm{F}}, \qquad T = \gamma T^{\mathrm{A}} + (1-\gamma)T^{\mathrm{F}}, \qquad \gamma \in [0,1]. \tag{7.21}$$

With $\gamma = 0$ the loops are completely open, while with $\gamma = 1$ the original system dynamics are obtained.

In order to define the strategy for computing a feasible initial guess, we introduce the following OCP

$$\mathcal{P}(p_{\mathrm{H}}) = \underset{x,u,z,p,F^{\mathrm{F}},T^{\mathrm{F}},\gamma}{\text{minimise}} \quad \frac{1}{T}\int_0^T \|x - x_{\mathrm{g}}\|_Q^2 + \|u - u_{\mathrm{g}}\|_R^2 \,\mathrm{d}t \tag{7.22a}$$

$$\text{subject to} \quad g_{\mathrm{p}}(x(0) - x(T)) = 0, \tag{7.22b}$$

$$f^{\mathrm{c}}(\dot{x}, x, z, u, p, F^{\mathrm{F}}, T^{\mathrm{F}}, \gamma) = 0, \tag{7.22c}$$

$$F = \gamma F^{\mathrm{A}} + (1-\gamma)F^{\mathrm{F}}, \tag{7.22d}$$

$$T = \gamma T^{\mathrm{A}} + (1-\gamma)T^{\mathrm{F}}, \tag{7.22e}$$

$$\gamma = p_{\mathrm{H}}, \tag{7.22f}$$

$$p = p_{\mathrm{g}}, \tag{7.22g}$$

$$h(x, z, u, p) \geq 0. \tag{7.22h}$$

The homotopy parameter p_{H} is embedded [34] in problem $\mathcal{P}$ by introducing the decision variable γ and the constraint (7.22f). Once $\mathcal{P}(0)$ has been solved to a

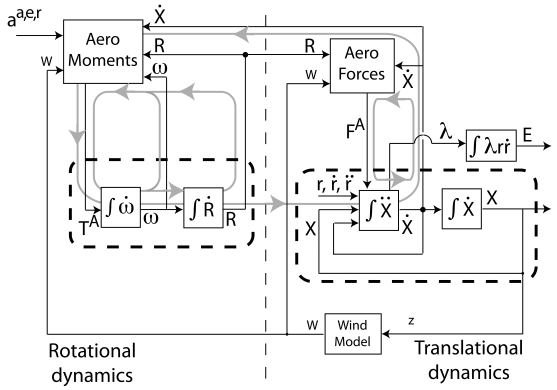


Figure 7.4: Architecture of the system dynamics. The feedback loops introduced by the aerodynamics are highlighted using the light-grey arrows.

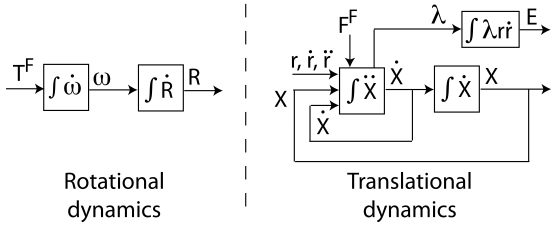


Figure 7.5: Dynamic model without aerodynamic feedback loops.

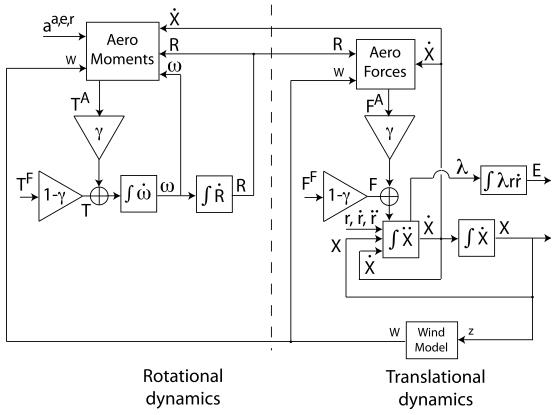


Figure 7.6: Dynamic model with the relaxation strategy: the feedback loops can be opened and closed by changing the value of parameter γ.

Algorithm 1: Relaxation Strategy

Input : Initial guess x_g, u_g, p_g. Tolerances $\delta_0 > \delta_\mathrm{f} > 0$. Number of homotopy steps N_H

Set $p_\mathrm{H} \leftarrow 0$

while $p_\mathrm{H} < 1$ **do**

  Solve $\mathcal{P}(p_\mathrm{H})$ to tolerance δ_0

  Set $p_\mathrm{H} \leftarrow p_\mathrm{H} + \frac{1}{N_\mathrm{H}}$

Solve $\mathcal{P}_\mathrm{E}$ to tolerance δ_f

reasonable degree of accuracy δ_0, for N_H chosen sufficiently large, a single full Newton step can be sufficient to update the solution. However, if the homotopy step size is not adjusted to ensure that the accuracy of the solution remains sufficient throughout the homotopy, more than one (not necessarily full) Newton step may be needed.

7.3.2 Reliable MPC and MHE Implementations

In this section, we consider two aspects which are extremely important in order to guarantee a reliable implementation of MPC and MHE for practical applications. We propose first a technique to guarantee feasibility of the MPC scheme and second a technique for enforcing state estimates which have a physical meaning.

Guaranteeing MPC Feasibility

Let us consider the MPC problem

$$\min_{x,u} \quad \sum_{k=0}^{N-1} \|x_k - x_k^\mathrm{r}\|_Q^2 + \|u_k - u_k^\mathrm{r}\|_R^2 + \|x_N - x_N^\mathrm{r}\|_P^2 \tag{7.23a}$$

$$\text{s.t.} \quad x_0 = \hat{x} \tag{7.23b}$$

$$x_{k+1} = f(x_k, u_k), \qquad k = 0, \dots, N-1, \tag{7.23c}$$

$$h(x_k, u_k) \geq 0, \qquad k = 0, \dots, N-1. \tag{7.23d}$$

As the initial state $\hat{x}$ is imposed by the real system, which is typically subject to uncontrollable perturbations, and the control authority is limited in the

beginning of the control horizon, it might be impossible for the system to satisfy the path constraints (7.23d). If such a situation occurs, Problem (7.23) becomes infeasible and the NMPC solution does not exist. In order to avoid such a situation, one can introduce slack variables s in order to relax the path constraints, i.e.

$$\min_{x,u,s} \sum_{k=0}^{N-1} \|x_k - x_k^{\mathrm{r}}\|_Q^2 + \|u_k - u_k^{\mathrm{r}}\|_R^2 + \|x_N - x_N^{\mathrm{r}}\|_P^2 + w_1^\top s \tag{7.24a}$$

$$\text{s.t.} \quad x_0 = \hat{x} \tag{7.24b}$$

$$x_{k+1} = f(x_k, u_k), \qquad k = 0, \dots, N-1, \tag{7.24c}$$

$$h(x_k, u_k) \geq -s, \qquad k = 0, \dots, N-1, \tag{7.24d}$$

$$s \geq 0. \tag{7.24e}$$

Provided that w_1 is chosen large enough, the term $w_1^\top s$ in the cost guarantees that, whenever possible, the slack variable s is 0 and therefore the path constraints are satisfied. In this case, Problem (7.24) delivers the same solution as Problem (7.23). If, instead, the path constraints (7.23d) cannot be satisfied, it is always possible to satisfy (7.24) and, due to the choice of the cost function, MPC will select the controls which ensure the least possible constraint violation.

As an example, let us consider a tethered airfoil operated in a turbulent wind. Because the maximum power is extracted by operating at maximum lift i.e. at the maximum angle of attack, turbulences leading to an increase of angle of attack are particularly dangerous for stalling. By taking some back-off from the stall condition, i.e. by reducing the maximum angle of attack allowed, stall will be avoided. However, if the path constraint on the angle of attack is not relaxed, even though stall is avoided, the MPC problem becomes infeasible and the controller fails at stabilising the airfoil. On the other hand, if the path constraint on the angle of attack is relaxed, the MPC problem remains feasible and, because the objective function is dominated by the term $w_1^\top s$, all available control authority will be used to bring the airfoil back to a safe operation.

All MPC schemes used in the remainder of the chapter are of the form (7.24), unless explicitly stated.

Enforcing a Physical State Estimate

Let us consider the MHE problem

$$\underset{x,u}{\text{minimise}} \quad \sum_{k=-E}^{0} \|y(x_k, u_k) - \tilde{y}_k\|^2_{\Sigma^{-1}} \tag{7.25a}$$

$$\text{subject to} \quad x_{k+1} = f(x_k, u_k), \qquad k = -E, \dots, -1, \tag{7.25b}$$

$$g(x_0) = 0. \tag{7.25c}$$

For tethered airfoils the model assumptions require that the airfoil flies with a positive velocity which is larger than the stall velocity and that it does not touch the ground. Because the system is operated far from these conditions, we do not need to impose any path constraint in order to ensure that the model has physical meaning. Moreover, we do not impose $-1 \leq R_{ij} \leq 1$: simulations have shown that, as expected, these constraints actually deteriorate the quality of the estimate.

In order for the model to have physical meaning, we instead need to impose Constraints (7.25c). They guarantee that both the tether invariant Equations (7.15) or (7.19) and the DCM or quaternion Invariant (6.10) or (6.12) respectively are enforced. Note that these constraints are implicitly enforced in the MPC scheme, through the initial state constraint, which relies on the state estimate provided by MHE [52].

All MHE schemes used in the following of the chapter are of the form (7.25).

7.4 Comparison of Dual and Single Airfoils

The drag due to the motion of the tether during crosswind flight has a significant impact on the system performance. To tackle this issue, the dual-airfoil design was first introduced in [84] and later investigated in e.g. [59, 86, 116]. The key idea of the dual-airfoil design is to fly two airfoils connected on a single main tether (see Figure 7.7) in a balanced manner. As a result, only the shorter secondary tethers move at a high velocity and generate drag, while the motion of the main tether is negligible.

While the dual-airfoil design has the potential to reduce the problem of tether drag for AWE systems, the system design and trajectory must be carefully selected so as to fully exploit the gains of reducing the tether drag. More precisely: a) the airfoil trajectories must balance the forces on the main tether

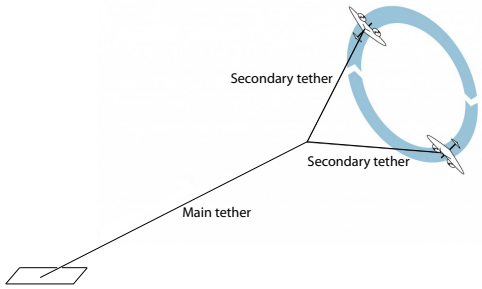


Figure 7.7: Schematic of a dual-airfoil AWE system (cf. [84], Fig. 3).

so as to minimise its motion, maintain the optimal airfoil velocities, and maintain an optimal angle between the secondary tethers, b) the aerodynamic forces yielded by on-board power generation must be appropriately chosen so as to maximise the system efficiency, c) the tether lengths must be chosen so as to achieve the best trade-off between reaching higher altitude and adding airborne mass, and d) the tether diameters must be selected so as to achieve the best trade-off between reducing the drag and withstanding the forces in the system.

In this section, we compare single and dual-airfoil systems of different sizes so as to investigate whether it is indeed more advantageous to use a dual-airfoil system as opposed to a single-airfoil system. We consider a wide range of wing areas by fixing all other aerodynamically relevant parameters, such as the aspect ratio and wing geometry. In order to have a fair comparison, we design the optimal system parameters and trajectory simultaneously by means of optimal control so as to harvest the most average energy.

First, we consider a simplified setting in which we do not account for the fact that the energy extracted by the airfoils from the airmass slows down the wind. Second, we introduce a simplified model of the airfoil-airmass interaction and account for the reduced wind speed due to the power harvesting.

7.4.1 System Model

In this section, we adopt the following modelling choices:

1. the airfoils are inertially modelled as point-masses;
2. each tether k is modelled with a lumped-mass finite element model with n_k elements;

3. the lift forces are orthogonal to the airfoil transversal axis;
4. the airfoils have a perfect yaw control, resulting in no side-slip;
5. the time-derivatives of the lift coefficient C^{L} and roll angle ψ are controlled and actuation is instantaneous;
6. the time-derivative of the onboard turbine drag coefficient κ is controlled and actuation is instantaneous;
7. the drag coefficient is given by the simplified model $C_i^{\mathrm{D}} = C_i^{\mathrm{D}_0} + C_i^{\mathrm{D}_\mathrm{I}} \left(C_i^{\mathrm{L}}\right)^2$.

A detailed derivation of the model can be found in [119]. The proposed model construction can straightforwardly accommodate different tether and airfoil models, e.g.: a 6-DOF airfoil description and more elaborate aerodynamic models. Yet a simple model was preferred, so as to reduce the complexity of the presentation.

The state and control vectors are then given by

$$X = \left[\begin{array}{ccccccccc} r_{1,1} & \cdots & r_{1,n_1} & \dot{r}_{1,1} & \cdots & \dot{r}_{1,n_1} & C_1^{\mathrm{L}} & \psi_1 & \kappa_1 \end{array} \right],$$

$$U = \left[\begin{array}{ccc} \dot{C}_1^{\mathrm{L}} & \dot{\psi}_1 & \dot{\kappa}_1 \end{array} \right],$$

for the single airfoil and

$$X = \left[\begin{array}{cccccccc} r_{1,1} & \cdots & r_{1,n_1} & \dot{r}_{1,1} & \cdots & \dot{r}_{1,n_1} & & \\ r_{2,1} & \cdots & r_{2,n_2} & \dot{r}_{2,1} & \cdots & \dot{r}_{2,n_2} & & \\ r_{3,1} & \cdots & r_{3,n_3} & \dot{r}_{3,1} & \cdots & \dot{r}_{3,n_3} & & \\ C_2^{\mathrm{L}} & \psi_2 & \kappa_2 & C_3^{\mathrm{L}} & \psi_3 & \kappa_3 & & \end{array} \right],$$

$$U = \left[\begin{array}{cccccc} \dot{C}_2^{\mathrm{L}} & \dot{\psi}_2 & \dot{\kappa}_2 & \dot{C}_3^{\mathrm{L}} & \dot{\psi}_3 & \dot{\kappa}_3 \end{array} \right],$$

for the dual airfoil. Note that indices k, j relative to the position of the nodes in the FEM tether model refer to element j of tether k, as explained in Section 7.2. For all other states and controls, index n refers to the node the quantity is associated with, as depicted in Figure 7.2.

The system dynamics are obtained by applying the approach detailed in Chapter 6 and we lump them in the index-1 DAE

$$f(\dot{X}, X, \lambda, U) = 0, \qquad g(X, \lambda) = 0, \tag{7.26}$$

where λ is the vector of algebraic states which, in the Euler-Lagrange approach are also the Lagrange multipliers associated with the tether constraints $\mathcal{C}$. Then, the tension in section j of tether k is readily given by $\Gamma_{k,j} = \lambda_{k,j} l_{k,j}$.

Wind and atmosphere model

Assuming a laminar wind flow with a logarithmic wind shear model blowing uniformly in the E^x-direction, the free-flow wind speed W_∞ at altitude z is given by [77]:

$$W_\infty(z) = W_0 \frac{\log(z/z_\mathrm{r})}{\log(z_0/z_\mathrm{r})}, \tag{7.27}$$

where $W_0 \in \mathbb{R}$ is the wind velocity at altitude z_0 and z_r is the ground roughness.

To account for the drop of density with altitude the following atmospheric model is introduced [7]:

$$T(z) = T_0 - T_\mathrm{L} z, \qquad P(z) = P_0 \left(1 - \frac{T_\mathrm{L} z}{T_0}\right)^{\frac{gM_\mathrm{a}}{RT_\mathrm{L}}}, \qquad \rho(z) = \frac{PM_\mathrm{a}}{RT},$$

where T_0 is the sea level standard temperature, T_L is the temperature lapse rate, P_0 is the pressure at sea level, M_a is the molar air density and R is the universal gas constant.

7.4.2 Optimisation Problem

In order to maximise the system average power generation, we formulate an optimal control and design problem, in which we simultaneously optimise the trajectory as well as some system parameters. The duration T_p of the orbit is an optimisation variable, as well as the tether lengths l_k and diameters d_k, so that

$$p = \begin{bmatrix} T_\mathrm{p} & l_1 & l_2 & d_1 & d_2 \end{bmatrix},$$

and we choose $l_3 = l_2$ and $d_3 = d_2$, i.e. the secondary tethers are the same. The periodicity of the system is enforced by the boundary conditions

$$X(0) - X(T_\mathrm{p}) = 0. \tag{7.28}$$

In order to avoid constraint qualification problems related to the presence of redundant constraints, we adopt the approach described in Section 7.3.1 which consists in removing the directions violating the consistency conditions from the set of periodicity conditions. Defining a matrix Z that forms a basis of the null-space of

$$J = \begin{bmatrix} \frac{\partial \mathcal{C}}{\partial X} \\ \frac{\partial \dot{\mathcal{C}}}{\partial X} \end{bmatrix}_{t=0},$$

i.e. $JZ = 0$, the set of consistency conditions $\mathcal{C}(0) = 0$, $\dot{\mathcal{C}}(0) = 0$ together with

$$Z^\top (X(0) - X(T_\mathrm{p})) = 0, \tag{7.29}$$

have no redundancy. The basis Z is non-unique, and can be chosen so as to limit its computational complexity.

In order to ensure that the tethers are always under tension but that their resistance is never exceeded, the constraints

$$\frac{\gamma}{f_\mathrm{s}} \frac{\pi}{4} d_k^2 \geq \underbrace{\lambda_{k,j} l_{k,j}}_{=\Gamma_{k,j}} \geq 0, \qquad \forall\ t,\ j = 1, \ldots, n_k,\ k = 1, 2, 3, \tag{7.30}$$

are imposed, where γ is the tether yield strength, f_s a safety factor and $\Gamma_{k,j}$ the tension in section j of tether k. Moreover, the following bounds are imposed:

$$\begin{aligned} 0 \leq C_\mathrm{L}^i \leq 1, \qquad & -5\ \mathrm{s}^{-1} \leq \dot{C}_\mathrm{L}^i \leq 5\ \mathrm{s}^{-1}, \\ -80^\circ \leq \psi_L^i \leq 80^\circ, \qquad & -5\ \mathrm{s}^{-1} \leq \dot{\psi}_L^i \leq 5\ \mathrm{s}^{-1}, \\ -1000\ \mathrm{kg/(ms)} \leq \dot{\kappa}_i \leq 1000\ \mathrm{kg/(ms)}, \qquad & \forall t,\ i \in \mathcal{A}. \end{aligned} \tag{7.31}$$

For convenience, we will summarise Constraints (7.30) and (7.31) in the function $h(X, U, p) \geq 0$.

The periodic power optimisation problem reads:

$$\bar{P} = \max_{U, X, \lambda, p, T_\mathrm{p}} \ \frac{1}{T_\mathrm{p}} \int_0^{T_\mathrm{p}} \sum_{i \in \mathcal{A}} P_i \ \mathrm{d}t, \tag{7.32a}$$

$$\text{s.t.} \quad f(\dot{X}(t), X(t), \lambda(t), U(t)) = 0, \qquad t \in [0, T], \tag{7.32b}$$

$$g(X(t), \lambda(t)) = 0, \qquad t \in [0, T], \tag{7.32c}$$

$$\mathcal{C}(0) = 0, \qquad \dot{\mathcal{C}}(0) = \left(C_X \dot{X}\right)_{t=0} = 0, \tag{7.32d}$$

$$Z^\top (X(0) - X(T_\mathrm{p})) = 0, \tag{7.32e}$$

$$h(X(t), U(t), p) \geq 0, \qquad t \in [0, T]. \tag{7.32f}$$

where Note that T_p is an optimisation variable, thus the duration of the orbit will be adapted by the optimiser so as to maximise the average power. In order to be able to treat this problem, a time transformation can be introduced, as outlined in Section 2.3.4.

Because the system dynamics (7.26) are unstable, a simultaneous optimal control technique is best suited to optimise the system model. The discretisation of the

Parameter	Symbol	Value	Unit
Air density at temperature T_0	ρ	1.23	$\mathrm{kg \cdot m^3}$
Tethers density	ρ_c	1450	$\mathrm{kg \cdot m^3}$
Airfoil parasitic drag coefficient	$C^{\mathrm{D},0}$	0.02	-
Airfoil induced drag coefficient	$C^{\mathrm{D,I}}$	0.02	-
Airfoil aspect ratio	A_R	10	-
Wind velocity at altitude z_0	W_0	10	m/s
Altitude of wind velocity W_0	z_0	100	m
Roughness factor	z_r	0.1	m
Sea level standard temperature	T_0	288.15	K
Temperature lapse rate	T_L	0.0065	K/m
Sea level pressure	P_0	101325	Pa
Molar air density	M_a	0.0289644	kg/mol
Universal gas constant	R	8.31447	J/(molK)
Tether drag coefficient	C^T	1	-
Tether yield strength	γ	$3.9 \cdot 10^9$	Pa
Safety factor	f_s	5	-

Table 7.1: Fixed model parameters

model dynamics (7.26) was based on a direct collocation approach [18], where the model simulation, constraints and optimisation are handled simultaneously in a large-scale sparse Nonlinear Program (NLP).

The problem transcription was performed using the open-source optimisation framework *CasADi* [6]. The resulting NLP was solved using the interior-point solver *IPOPT 3.10.1* [113] using *WSMP* as a linear solver.

For the dual-airfoil system, solving (7.32) on a full orbit yields quasi-identical trajectories for the two airfoils, hence Problem (7.32) was solved on a half orbit instead. For both the single and dual-airfoil problems, the control input profiles were discretised using a piecewise-constant parametrisation having 20 intervals per full orbit. One collocation element has been used per control interval.

7.4.3 Parametric Study without Airfoil-Airmass Interaction

In the proposed parametric study, we aim at assessing the relationship between the total airfoil surface and the average generated power, i.e. $\bar{P}\left(A_\mathrm{tot}\right)$ where $A_\mathrm{tot} = \sum_{n \in \mathcal{A}} A$. This study focused on assessing if, for a given total airfoil surface, a single or a dual-airfoil system should be preferred.

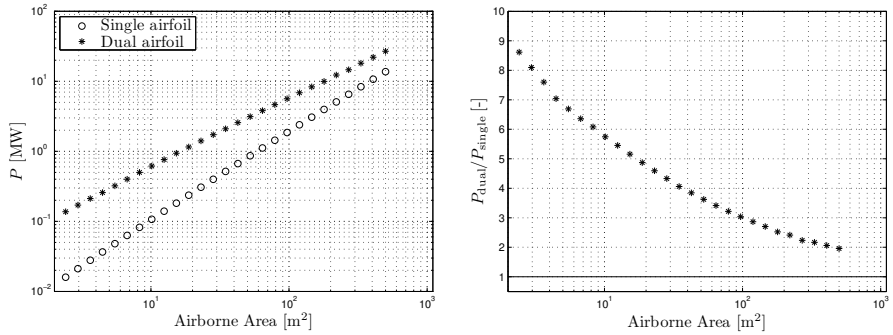


Figure 7.8: Dual vs. one large single airfoil: average power output for different wing surface A, keeping the wing loading constant. The comparison assumes that both systems have the same overall airfoil surface, i.e. $A_{\text{dual}} = \frac{1}{2}A_{\text{single}}$, hence assessing the advantage of having a dual-airfoil system with two smaller airfoils vs. having a single large airfoil.

The study was based on airfoils having a maximum gliding ratio $L/D = 25$. The tethers are assumed to be made of *Dyneema*®, which has a very high stiffness and yield strength for a low density. The fixed model parameters are summarised in Table 7.1.

For the single airfoil the tether has been discretised using 5 segments. For the dual airfoils the main tether has been discretised using 20 segments, while 5 segments were used for both secondary tethers. This results in 41 states for the single airfoil and 207 states for the dual airfoils. More refined tether discretisations have been tested and the obtained results don't show any relevant difference.

First, a solution to (7.32) for the single-airfoil system using parameter values $A = 500$ m^2 and $M_{\text{A}}/A = 20$ kg/m^2, and a solution for the dual-airfoil system using parameter values $A = 250$ m^2 and $M_{\text{A}}/A = 20$ kg/m^2 are computed.

Starting from these solutions, a homotopy with respect to the total airfoil area A_{tot} is applied to (7.32). Keeping the wing loading $M_{\text{A}}/A = 20$ kg/m^2 constant (i.e. the airfoil mass scales linearly with the total airfoil area), A_{tot} is gradually reduced and (7.32) repeatedly solved, using the solution from the previous step as an initial guess for the subsequent step. The average generated power for both the single and dual-airfoil systems are displayed in Figure 7.8, top graph. The ratio between the average generated power for the dual-airfoil system and the single-airfoil system is displayed in Figure 7.8, bottom graph. Note that the graphs in Figure 7.8 display the mechanical power dissipated by the onboard

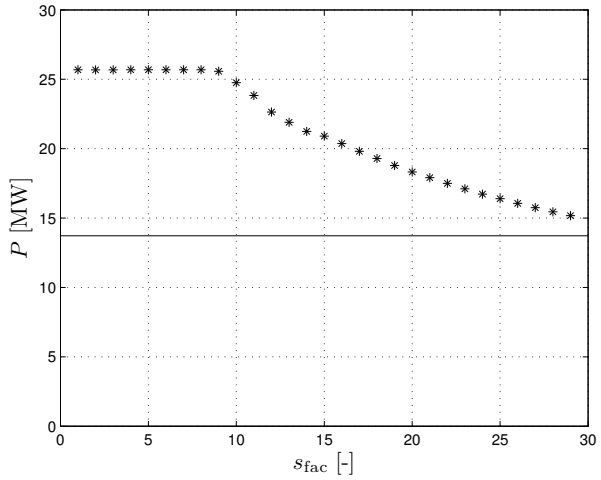


Figure 7.9: Study of the impact of the safety constraint on the extracted power P for the largest dual-airfoil system. The parameter s_{fac} expresses the distance between the airfoils, measured in wingspans. For comparison, the energy extracted by the single airfoil is displayed as a continuous line.

turbines. The actual electrical power depends on the generators and converters efficiency, whose value is arguably similar for the two systems. For the chosen parameters, the dual-airfoil system is always more advantageous than the single one. As the total airfoil surface increases, however, the ratio between the power extracted by the dual and single airfoil decreases significantly.

It should be observed that the required total airfoil surface for a desired amount of average generated power is also indirectly assessed through the proposed parametric study. Indeed, it can be seen from Figure 7.8 that for an average power generation of 10 MW the dual-airfoil system requires a total airfoil surface of approximately half the one of the single-airfoil system, but the dual airfoil system requires gradually less airfoil surface as the desired average generated power decreases.

For safety reasons, it is desirable that the dual-airfoil trajectories keep the airfoils far from each other, thus avoiding the risk of collisions. A second reason for having the airfoils flying large orbits, is to reduce the interaction with the airmass. In this study, no hypothesis is made on this complex interaction, which is assumed to be small and is thus neglected. This assumption might not hold if the airfoils fly too close to each other, as the interaction will be higher. A study

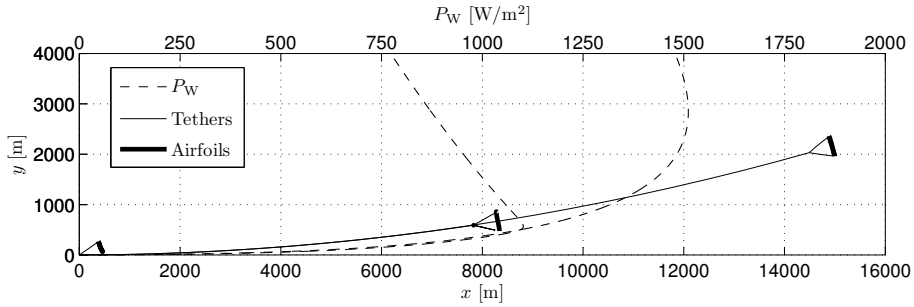


Figure 7.10: Trajectory comparison between the single and dual airfoils for a total wing surface $A_{\text{tot}} = 500\,\text{m}^2$. The trajectories are shown as thick lines. The available wind power $P_{\text{w}} = \rho W_{\infty}^3/2$ is plotted as a dashed line. For the dual airfoils, two wind profiles have been considered: a) the logarithmic profile (7.27) and b) the logarithmic profile saturated above $z = 500$ m.

has thus been done, to check how the extracted power is affected by imposing a safety constraint on the distance between the airfoils. This safety constraint is expressed as

$$(X_2 - X_3)^T (X_2 - X_3) \geq (s_{\text{fac}} w_{\text{s}})^2, \tag{7.33}$$

where the wingspan is given by $w_{\text{s}} = \sqrt{A_{\text{R}} A}$, with A_{R} the aspect ratio. For the simulations, the value $A_{\text{R}} = 10$ has been chosen. The results are displayed in Figure 7.9 for the largest dual-airfoil system, i.e. $A_{\text{tot}} = 500\ \text{m}^2$. It can be seen that allowing the airfoils to fly closer than 9 w_{s} does not lead to an increase in the extracted power. For larger orbits, the extracted power diminishes but the loss is not dramatic and, even for very large orbits, the dual airfoils still extract more power than the single airfoil.

The trajectory for the biggest systems considered, i.e. $S_{\text{tot}} = 500\ \text{m}^2$, is displayed in Figure 7.10. It can be noted that the dual airfoils operate at much higher altitude, approaching the peak of the available wind power formula (2850 m), also displayed in Figure 7.10. The proposed wind shear model (7.27) is valid only in the atmospheric boundary layer, which is typically lower than 2000 m. In this paper, the boundary layer was supposed to have an infinite thickness. The resulting optimal trajectories for the dual-kite systems reaches over 2000 m, which strongly suggests that the optimal altitude is always at the top of the boundary layer, regardless of its thickness. Arguably, in boundary layers that are not developed to the top altitude of 2000 m, the dual-airfoil system would loose some of its efficiency, and its advantage over the single-airfoil system would be reduced. In regard of these results, in practice, the optimisation of a

dual-airfoil AWE system should consider the average altitude of the boundary layer in the region of interest. As a term of comparison, a second scenario has been considered, where the wind profile saturates for $z \geq 500$ m. The resulting trajectory and related available wind power are also displayed in Figure 7.10. In this second case, the trajectory is still in proximity of the peak of available wind power, which occurs at much lower altitude. For a total wing surface $A_{\text{tot}} = 500\,\text{m}^2$, also in the case of a saturated wind profile, the power extracted by the dual airfoils exceeds the one extracted by the single airfoil by the ratio $P_{\text{dual}} = 1.54\ P_{\text{single}}$.

The tether length obtained for the dual-kite system is arguably extremely large. It was observed, however, that the sensitivity of the power generation to the tether length is rather small, i.e. constraining the tether to smaller length does not result in a large power loss. Arguably, economical factors such as the material cost and the electrical resistance of a very long tether would yield a system with a shorter tether. In this study, however, only the physics of the system were considered.

To check the precision of the collocation discretisation, an optimisation for the biggest system has been run with a refined collocation scheme having 4 times more collocation elements, resulting in NLPs with up to 47350 optimisation variables. No relevant difference has been noticed in the resulting trajectory, suggesting that the chosen collocation scheme is accurate enough.

The computed trajectories are only valid for the nominal case and, in a real application, wind perturbations and unmodelled dynamics will affect the performance of the system: this problem can be tackled within a robust optimisation framework. The resulting NLP will, though, be considerably more complex than the proposed one. In the context of a real application, performance is also affected by the choice of the controller. Both investigations are out of the scope of this study.

The proposed scenario assumes that the airfoils do not modify the wind field. The development of an accurate model needs extensive studies. Early results relying on simplified interaction models will be proposed in the following.

7.4.4 Airfoil-Airmass Interaction

The study presented in the previous section assumes no interaction between the airfoils and the airmass, i.e. the presence of the airfoils does not reduce the velocity of the airmass. It is well known [77, 15] that this hypothesis does not hold true for conventional wind turbines and the Betz law provides an upper bound to the energy that can be extracted from the airmass. That analysis does

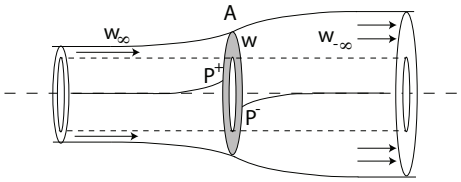


Figure 7.11: Schematic of the airflow through an actuator disk (grey area) of area A^{d}. W_∞ is the free flow velocity, W the flow velocity at the actuator disk, $W_{-\infty}$ the velocity after the actuator disk. P^+ and P^- stand for the pressure just before and after the actuator disk.

not directly apply to AWE systems, and, to the authors' knowledge, no study is available in the literature, which considers interaction between the airfoil and the airmass. In this section, the effect of such interaction is taken into account in the optimal control problem by introducing an interaction model which adapts the blade element-momentum theory to AWE systems.

Blade-Element–Momentum Theory for AWE Systems

The blade-element–momentum (BEM) theory for conventional wind turbines considers the interaction between a blade element and the airflow. The main assumption is that, given the annulus swept by the considered blade element, the change of momentum of the air which passes through it is due only to the force exerted by the blade element on the airmass [29].

In the following, the BEM theory is formulated for AWE systems. Note that, in this case, no analytic solution can be computed, as the annulus swept by the AWE system depends on the trajectory flown by the airfoil. The derivation will thus be included in the optimisation problem as a constraint which bounds the maximum power that the system can extract from the air mass by adapting the wind velocity at the swept annulus.

For ease of notation, in the following we drop the dependency of the wind speed on the height z, i.e. we will use W instead $W(z)$. As shown in Figure 7.11, the wind field velocity W at the swept annulus is less than the free flow velocity W_∞, and depends on the average thrust F^{d} exerted on the air mass by the actuator disk. This effect is conventionally represented by the axial induction factor a, defined through $W = (1-a)W_\infty,\ 0 < a < 1$.

The change of air mass momentum in the annular control volume (cf. Figure 7.11) results from the thrust force F^{d} acting on the annulus, such that

$$F^{\mathrm{d}} = \rho A^{\mathrm{d}} W \left(W_{\infty} - W_{-\infty}\right), \tag{7.34}$$

where $W_{-\infty}$ is the steady-state flow velocity in the far wake of the annulus, as displayed in Figure 7.11. Locally around the annular actuator disk, we can further express (since the velocity is continuous over the disk)

$$F^{\mathrm{d}} = A^{\mathrm{d}} \left(P^{+} - P^{-}\right), \tag{7.35}$$

such that the difference of pressure on the annulus is given by

$$P^{+} - P^{-} = \rho W \left(W_{\infty} - W_{-\infty}\right). \tag{7.36}$$

From Bernoulli's principle, the conservation of the air mass energy along streamlines before and after the actuator disk requires that

$$\frac{1}{2}\rho W^2 + P^{+} = \frac{1}{2}\rho W_{\infty}^2 + P_0, \qquad \frac{1}{2}\rho W^2 + P^{-} = \frac{1}{2}\rho W_{-\infty}^2 + P_0,$$

where P_0 is the pressure in the free flow, resulting in

$$P^{+} - P^{-} = \frac{1}{2}\rho \left(W_{\infty}^2 - W_{-\infty}^2\right). \tag{7.37}$$

By combining (7.36) and (7.37), it can be verified that

$$W_{-\infty} = 2W - W_{\infty} = (1 - 2a) W_{\infty}. \tag{7.38}$$

Using (7.34) and (7.38), the power extracted from the flow at is then given by:

$$P^{\mathrm{d}}(a) = F^{\mathrm{d}} W = 2\rho A^{\mathrm{d}} a (1 - a)^2 W_{\infty}^3. \tag{7.39}$$

This result is essentially the same as the classical Betz result for wind turbines (albeit there for a circular area instead of an annulus). The Betz limit occurs for $a = 1/3$, leading to a power coefficient $C_{\mathrm{p}} \equiv P/(\rho W_{\infty}^3/2) = 16/27$, but this presumes an ideal case with frictionless power extraction. In the context of an AWE system, drag forces on the wing, and the possibility to fly the wing at different radii R and speeds, leads to an optimisation problem in which $a = 1/3$ will not be recovered as an optimum.

For an AWE system, P^{d} is also given by summing up all the average power dissipated by the system P^{e}, i.e. the total average power resulting from the drag yielded by the tether, airfoil and turbines:

$$P^{\mathrm{e}} = \frac{1}{T_{\mathrm{p}}} \int_0^{T_{\mathrm{p}}} \left(\sum_{i \in \mathcal{A}} \left(P_i + v_i^{\top} F_i^{\mathrm{D}}\right) + \sum_{n=1}^{N} v_n^{\top} F_n^{\mathrm{T}} \right) \mathrm{d}t. \tag{7.40}$$

Variable P^{e} depends on the wind field velocity $W(z)$, taken as $W(z) = (1 - a)W_\infty(z)$ in this paper. We make this dependency explicit in the notation, and obtain $P^{\mathrm{e}}(W(z)) = P^{\mathrm{e}}\left((1-a)W_\infty(z)\right)$. As a result, the implicit equality

$$P^{\mathrm{d}}(a) = P^{\mathrm{e}}\left((1-a)W_\infty(z)\right), \tag{7.41}$$

must hold for some value of a. In the framework of optimal control a is treated as a decision variable and the implicit relationship (7.41) as an equality constraint.

Because of the wind shear, the wind velocity on the annulus swept by the AWE system is not uniform, hence it is proposed here to evaluate 7.39 using the integral:

$$P^{\mathrm{d}}(a) = 2\rho a(1-a)^2 \int_{A^{\mathrm{d}}} W_\infty^3(z)\ \mathrm{d}A^{\mathrm{d}}. \tag{7.42}$$

It can be verified that for a single airfoil, P^{d} can be computed using:

$$P^{\mathrm{d}}(a) = 2\rho a(1-a)^2 \int_0^{T_{\mathrm{p}}} z_{\mathrm{tip}}\dot{y}_{\mathrm{tip}} W_\infty(z_{\mathrm{tip}})^3\ \mathrm{d}t, \tag{7.43}$$

where the coordinate vector $r^{\mathrm{tip}} = \begin{bmatrix} x^{\mathrm{tip}} & y^{\mathrm{tip}} & z^{\mathrm{tip}} \end{bmatrix}^\top$ is the airfoil outer-tip position in frame E. For dual-airfoil systems, both airfoils fly the same trajectories, and the area A^{d} can be computed based on the trajectory of any of the two airfoils.

Numerical Results

In order to assess the impact of the airfoil-airmass interaction on the extracted power, Constraint (7.41) has been added to Problem (7.32) and the same study has been performed.

The computed trajectories for the dual and single airfoil are comparable to the ones obtained in the previous study (cf. Section 7.4.3). The most noticeable difference from the previous study can be seen in the power extracted from the airmass. In Figure 7.12 the average power extracted by the single and dual airfoil systems is compared. Two cases are considered: a) the airfoil-airmass interaction is modelled and b) the interaction is not modelled. In both cases, the dual airfoils are more advantageous, but the airfoil-airmass interaction has a stronger impact on the dual airfoils: the dual airfoils extract 39% less power when the airfoil-airmass interaction is modelled, while the power drop for the single airfoil is 27%.

The proposed framework for taking into account the interaction between the airfoils and the airmass involves a rather simple formulation and can fail to

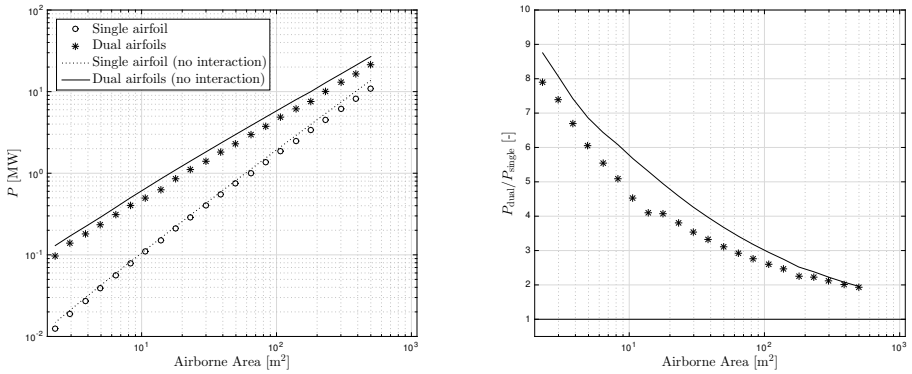


Figure 7.12: Dual vs. one large single airfoil accounting for the airfoil-airmass interaction. Average power output for different wing surface A, keeping the wing loading constant. The comparison assumes that both systems have the same overall airfoil surface, i.e. $A_{\text{dual}} = \frac{1}{2}A_{\text{single}}$, hence assessing the advantage of having a dual-airfoil system with two smaller airfoils vs. having a single large airfoil. The power output obtained when neglecting the interaction has also been reported for comparison.

accurately capture all aerodynamic phenomena, yet, it allows to introduce a model of the interaction in the optimisation problem. The improvement of the airfoil-airmass interaction model is the object of ongoing research.

7.5 Control of a Single Airfoil for Power Generation

In this section, we consider the problem of controlling a single airfoil system for power generation in pumping mode. First a power optimal periodic orbit is computed, then an MHE-MPC control scheme is used to track the power optimal orbit in a perturbed wind field.

System Model

In this section, we adopt the following modelling choices:

- the airfoil is inertially modelled as a rigid body with rotational and translational inertia J and m_{A} respectively;
- the tether is modelled as a rigid rod with mass;

- we consider a pumping-mode airfoil in which the tether acceleration $\ddot{l}$ is directly controlled;
- the time derivatives of the control surfaces (ailerons, elevators and rudder) deflections $\dot{a}^{\mathrm{a}}$, $\dot{a}^{\mathrm{e}}$, $\dot{a}^{\mathrm{r}}$ are directly controlled.

A detailed derivation of the model and all parameters can be found in [49, 50].

The local wind velocity is given by

$$w(r) = \begin{bmatrix} w^x(r) & w^y(r) & w^z(r) \end{bmatrix}. \tag{7.44}$$

Assuming a laminar wind flow with a logarithmic wind shear model blowing in the uniform x-direction, the wind velocity is given by [77]

$$w^x(r) = w_0 \frac{\log(z/z_{\mathrm{r}})}{\log(z_0/z_{\mathrm{r}})} + w^{x,\mathrm{t}}(r), \quad w^y(r) = w^{y,\mathrm{t}}(r), \quad w^z(r) = w^{y,\mathrm{t}}(r), \tag{7.45}$$

where $w_0 \in \mathbb{R}$ is the wind velocity at altitude z_0 and z_r is the ground roughness. The quantities

$$w^{\mathrm{t}}(r) = \begin{bmatrix} w^{x,\mathrm{t}}(r) & w^{y,\mathrm{t}}(r) & w^{z,\mathrm{t}}(r) \end{bmatrix}, \tag{7.46}$$

are the wind turbulences and will be assumed to be 0 for the computation of power optimal trajectories, but will be estimated by MHE when controlling the airfoil in the perturbed windfield.

Note that, in general, the perturbed windfield also has a rotational component, described by

$$\Omega^{\mathrm{t}}(r) = \begin{bmatrix} \Omega^{x,\mathrm{t}}(r) & \Omega^{y,\mathrm{t}}(r) & \Omega^{z,\mathrm{t}}(r) \end{bmatrix}. \tag{7.47}$$

Also these perturbations will be assumed to be 0 for the offline power optimisation, but we will attempt at estimating them online using MHE.

For control and estimation purposes, it is proposed here to use a rather simple turbulence model, given by

$$\dot{w}^{\mathrm{t}} = -\frac{w^{\mathrm{t}}}{\tau} + u_w^{\mathrm{t}}, \qquad \dot{\Omega}^{\mathrm{t}} = -\frac{\Omega^{\mathrm{t}}}{\tau} + u_\Omega^{\mathrm{t}}, \tag{7.48}$$

where u_w^{t}, u_Ω^{t} are the forcing term in this first-order differential equation, here modelled as pseudo-controls. While in the estimation problem the optimiser is free to choose the terms u_w^{t}, u_Ω^{t} so as to best fit the measurements, in the control problem, these modes are uncontrollable and, for the prediction horizon, the forcing term is assumed to be 0.

The state vector $X \in \mathbb{R}^{29}$ and the control vector $U \in \mathbb{R}^{10}$ are respectively given by

$$X = (r,\, \dot{r},\, e^x,\, e^y,\, e^z,\, \omega,\, a^{\mathrm{a}},\, a^{\mathrm{e}},\, a^{\mathrm{r}},\, l,\, \dot{l},\, w^{\mathrm{t}},\, \Omega^{\mathrm{t}}),$$

$$U = (\ddot{l},\, u^{\mathrm{a}}, u^{\mathrm{e}}, u^{\mathrm{r}},\, u_w^{\mathrm{t}},\, u_\Omega^{\mathrm{t}}).$$

Note that, when solving the trajectory planning optimal control problem we assume $w^{\mathrm{t}} = 0$, $\Omega^{\mathrm{t}} = 0$, $u_w^{\mathrm{t}} = 0$, while for MPC we assume $u_w^{\mathrm{t}} = 0$, $u_\Omega^{\mathrm{t}} = 0$.

Path Constraints

We use the following control input and state bounds:

$$-5\,\mathrm{m/s^2} \leq \ddot{l} \leq 5\,\mathrm{m/s^2}, \qquad -10\,\mathrm{m/s} \leq \dot{l} \leq 10\,\mathrm{m/s}.$$

In addition, in order to keep the system in the region where the model assumptions are valid, the following path constraints need to be considered:

$$-1 \leq C^{\mathrm{L}}(\mathcal{X}, W_0) \leq 1, \qquad \lambda(\mathcal{X}, W_0) \leq 0.$$

Constraint $\lambda \leq 0$ is required to keep the tether under tension, and constraint $-1 \leq C^{\mathrm{L}} \leq 1$ is required to keep the wing in the linear-lift region [82]. Note that the actual bounds on the linear-lift region depend on the wing used.

Power Optimal Trajectories

Two common types of trajectories for AWE systems are considered. The first is a circular trajectory, a topology that is often preferred in practice for its simplicity, but for which a swivel mechanism is required to avoid the winding of the tether. The second trajectory is a lying eight, for which the problem of tether winding is avoided, but which requires strong angular accelerations $\dot{\omega}$ to be performed.

The OCP was discretised using multiple-shooting [22], and the resulting NLP tackled via the SQP method implemented in the software ACADO [60]. The primal active set QP solver qpOASES [36] was used to solve the underlying Quadratic Problems.

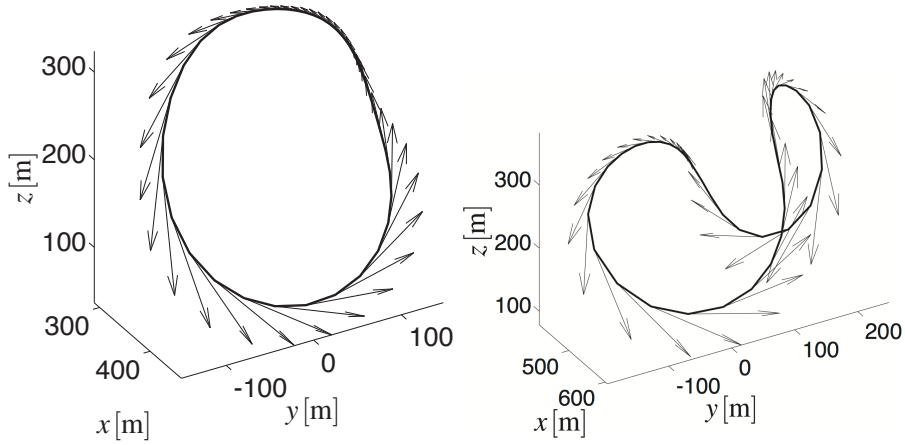


Figure 7.13: 3D trajectories for a circular and lying eight trajectory. The arrows indicate the wing velocity.

Circular Trajectory The relaxed optimisation problem was initialised using

$$\delta_k \in [0, 2\pi], \qquad \psi = \psi_{\max} \sin(\delta) + \psi_0, \qquad \theta = \theta_{\max} \cos(\delta), \tag{7.49a}$$

$$r(\psi, \theta, l) = l \begin{bmatrix} \cos(\psi)\cos(\theta) \\ \sin(\theta) \\ \cos(\theta)\sin(\psi) \end{bmatrix}, \tag{7.49b}$$

for $k = 0, ..., N-1$, where r, $\psi_{\max}$, ψ_0 and $\theta_{\max}$ must be chosen by the user. The optimal trajectories are reported in 3D in Figure 7.13.

Lying Eight Trajectory The relaxed optimisation problem was initialised using

$$\delta_k \in [0, 2\pi], \qquad \psi = \psi_{\max} \sin(2\delta) + \psi_0, \tag{7.50a}$$

$$\theta = \quad \theta_{\max} \left(\cos(2\delta) - 1\right), \quad \delta \in [0, \pi),$$

$$\theta = -\theta_{\max} \left(\cos(2\delta) - 1\right), \quad \delta \in [\pi, 2\pi], \tag{7.50b}$$

$$r(\psi, \theta, l) = l \begin{bmatrix} \cos(\psi)\cos(\theta) \\ \sin(\theta) \\ \cos(\theta)\sin(\psi) \end{bmatrix}, \tag{7.50c}$$

for $k = 0, ..., N-1$. The optimal trajectories are reported in 3D in Figure 7.13.

Sensor	Measurement	σ
IMU	Linear accelerations in frame e	$5\,\mathrm{cm}\cdot\mathrm{s}^{-2}$
IMU	Angular velocities ω	$1\,\mathrm{deg}\cdot\mathrm{s}^{-1}$
GPS	Absolute positions r	$0.1\,\mathrm{m}$
GPS	Absolute velocities $\dot{r}$	$0.6\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Variometer	Absolute vertical velocity $\dot{z}$	$0.5\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Tether encoder	Tether length & velocity l,$\dot{l}$	$5\,\mathrm{m}$, $1\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Tether gauge	Tether tension F_t	$500\,\mathrm{N}$
Pitot tube	Longitudinal relative velocity $(e^x)^\top v$	$1\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Air Probe	Angle of attack (AoA) α	$2.5\,\mathrm{deg}$
Air Probe	Side-slip angle β	$5\,\mathrm{deg}$
Surface encoders	Deflections a^a, a^e, a^r	$0.1\,\mathrm{deg}$

Table 7.2: Available sensors for MHE and standard deviation.

Feedback Control in a Perturbed Wind Field

The system is operated in a perturbed wind field with the unknown translational turbulences w^t and rotational turbulences Ω^t. It was observed in simulations that assuming a laminar wind flow in the MHE scheme yields inaccurate state estimations and eventually results in the failure of the MHE-NMPC scheme. Taking into account at least the translational turbulences is crucial for the robustness of the control scheme to wind turbulences.

MHE Setup Wind turbulences can be modelled as elaborate stochastic processes, which have been extensively studied in the literature. However, making strong assumptions on the statistical properties of the wind turbulences for the control of AWE system can yield poor control performances when such assumptions are not correct. In the proposed MHE formulation, it assumed that $\dot{w}^\mathrm{t}$ and $\dot{\Omega}^\mathrm{t}$ are Gaussian white noise, and introduced as fictitious control variables in the MHE formulation. Moreover, because of actuator noise and inaccuracy, the control policies computed by the NMPC scheme may not be perfectly implemented on the system, therefore the system inputs are also decision variables in the MHE scheme, and their deviation from the inputs provided by NMPC is also penalised in the MHE cost by introducing pseudo-measurements.

Following the argument developed in [52], the consistency conditions (7.15) are enforced in the MHE scheme at the current time t_i, i.e. at the end of the MHE horizon.

Variable	Description	Unit	Bounds
a^{a}	Aileron	deg	[-20,20]
a^{e}	Elevator	deg	[-20,20]
a^{r}	Rudder	deg	[-20,20]
u^{a}	Aileron rate	deg/s	[-12.5,12.5]
u^{e}	Elevator rate	deg/s	[-3.25,3.25]
u^{r}	Rudder rate	deg/s	[-2.25,2.25]

Variable	Weight
r	10^{-4}
$\dot{r}$	10^{-3}
$R,\ \omega$	10^{-1}
$a^{\cdot},\ u^{\cdot}$	1
$l,\ \dot{l}$	10^{-4}
$\ddot{l}$	10

Table 7.3: System constraints and MPC weights.

Data from IMU and GPS can already provide a reasonably good estimation of the wing attitude [46], however, the availability and fusion of more sensor data helps reducing the covariance of the state estimation. The list of sensors used is provided in Table 7.2, with the corresponding standard deviations σ. The sensors accuracy can vary by orders of magnitude, depending on the quality of the equipment chosen. The standard deviations were chosen to reasonably match values commonly observed in the industry.

For the sake of simplicity, we assume that all sensors are sampled at the same frequency Δt_{E}, and that the sensors have no dynamics. In the MHE formulation, the weights Σ^{-1} relative to the measurement vector $\bar{Y}_k \in \mathbb{R}^{26}$ are chosen diagonal, with their entries taken as the inverse of the covariance of the noise of the corresponding sensor, i.e. $\{\Sigma\}_i = \sigma_i^2$, as detailed in Table 7.2. Similarly, the weight for the pseudo-measurements relative to the controls was chosen as the inverse of the selected input noise $\sigma_u = 0.1\,\mathrm{deg}\cdot\mathrm{s}^{-1}$ introduced in the system. The turbulence weights T_{T} and T_{R} were chosen in a similar way, using the covariance of w^{t}, Ω^{t}, $\dot{w}^{\mathrm{t}}$ and $\dot{\Omega}^{\mathrm{t}}$, extracted form the wind data.

MPC Setup The latest estimation of the translational and rotational turbulences are used in the NMPC scheme, and assumed constant over the prediction horizon.

The MPC weights and constraints are summarised in Table 7.3, and the terminal cost matrix has been chosen as $P = Q$. Note that the input bounds have been artificially tightened to demonstrate the capability of the proposed scheme to handle limited actuator capabilities. In practice, such bounds are typically less tight, allowing the control scheme to take stronger control actions, while the constraints are less often activated.

Wind Model for Simulation It is assumed here that the wind is the superposition of a laminar, logarithmic wind shear model blowing in the uniform x-direction with a Von Karman [77] turbulence model. The Von Karman parameters are given in [50]. The resulting 10 min wind profiles used in the proposed simulations are displayed in Figure 7.17. Since it assumes a level flight at constant velocity, the Von Karman model is not the most appropriate turbulence model for AWE systems, and should not be used for certification purposes. However, it is arguably a simple mean to test control algorithms.

Simulation Results Figure 7.14 displays the trajectory of the wing. It can be observed that the control scheme exploits the cable length to cope with the turbulences. A better tracking performance can be obtained by relaxing the input weights, resulting, however, in a loss of generated power. Figure 7.15 displays the corresponding inputs. Figure 7.16 displays the energy for the reference trajectory and the energy obtained from the system subject to turbulences. It can be seen that the system extracts more power than the reference trajectory, hence using the extra power available due to the turbulences. Figure 7.17 displays the turbulences estimated by the MHE. The translational turbulences are well estimated, while the rotational turbulences are not accurately captured. The lack of accuracy in capturing the rotational turbulences is clearly the result of their limited observability. This difficulty is not a feature of the proposed MHE scheme but of the system itself, this observation will require further investigations to fully understand it. However, it was observed in simulations that, even though an estimation of the translation turbulences is crucial for obtaining good control performances, an estimation of the rotational turbulences did not impact significantly the behaviour of the control scheme. This may result from the controllability of the wing attitude, which is stronger and more direct than the wing velocity and position in space.

Figure 7.18 displays the coefficient C^{L} and side-slip angle β. Both variables remain within the domain of validity of the aerodynamic model, moreover, the tether remains constantly under tension. The simulations were run on a single CPU clocked at 2.66 GHz using ACADO Code Generation to export RTI-based solvers. The time needed to perform the estimation update and control policy update are always lower than the sampling time, with a reasonable safety margin.

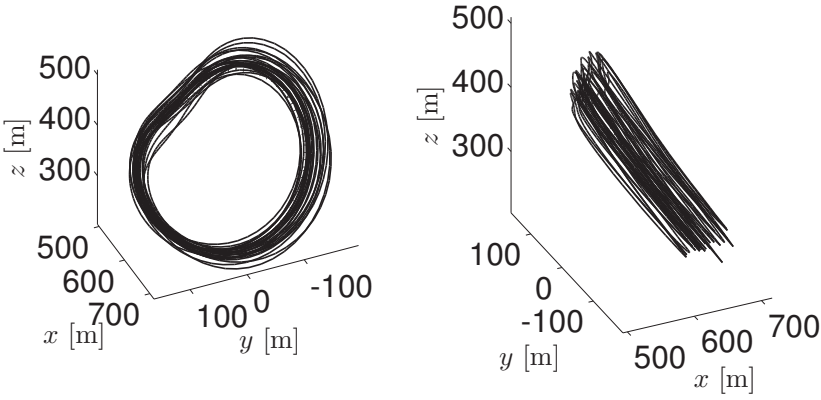


Figure 7.14: System trajectories.

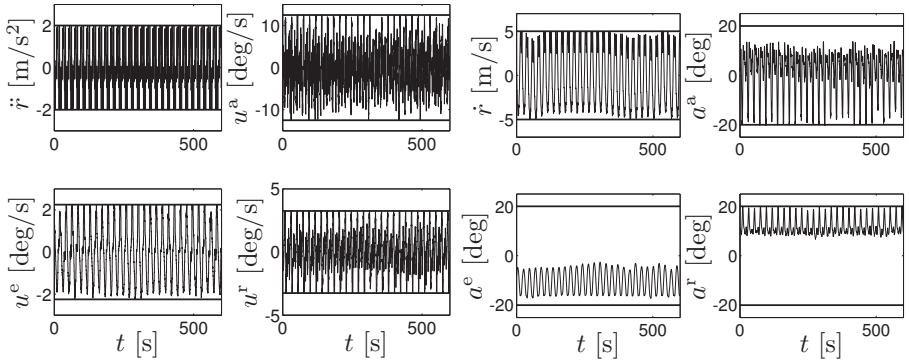


Figure 7.15: Left graph: tether reeling acceleration and deflexion rate of the control surfaces. Right graph: tether reeling speed and deflection of the control surfaces.

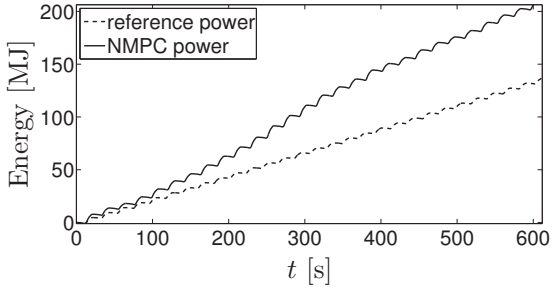


Figure 7.16: Generated power, reference (dotted line), actual (plain line). The extra energy is yielded by the turbulences.

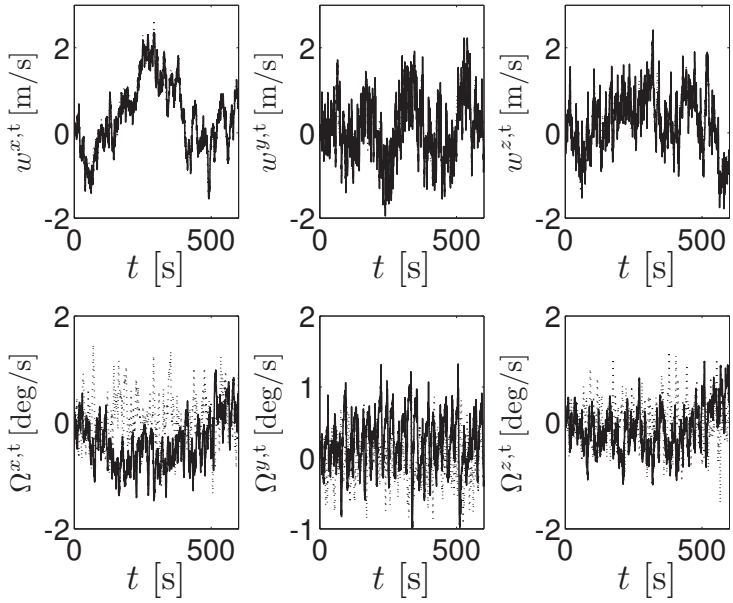


Figure 7.17: Von Karman wind turbulences (dotted line) and the MHE wind estimation (plain line).

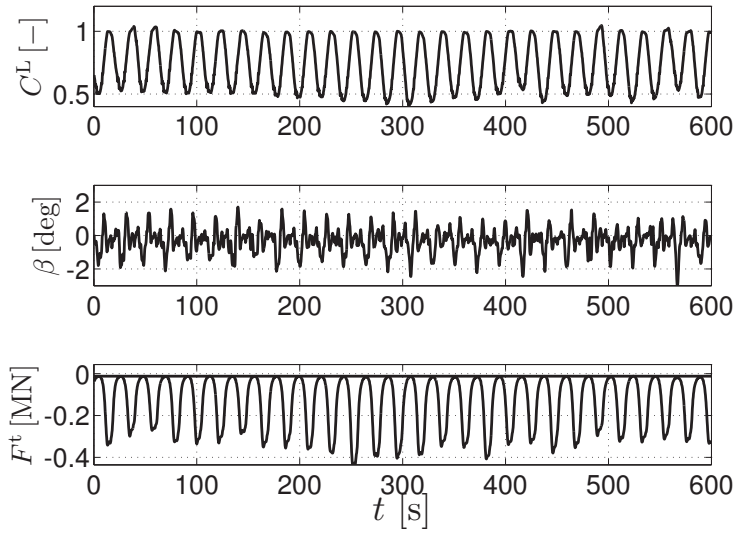


Figure 7.18: Lift coefficient C^{L}, side-slip angle β, and tether force F^{t}.

7.6 Control of Dual Airfoils for Power Generation

In this section, we consider the problem of controlling a dual airfoil system for power generation in drag mode. First we compute a power optimal periodic orbit by highlighting the implementation of the initialisation procedure described in Section 7.3.1. Then we adopt the MHE-MPC framework used in Section 7.5 to control the dual airfoil system in turbulent wind.

System Model

In this section, we adopt the following modelling choices:

- the airfoils are inertially modelled as rigid bodies with rotational and translational inertia;
- the tether is modelled as a rigid rod with mass, but the tether drag is evaluated evaluating Equation (7.9) numerically using a trapezoidal quadrature rule with 6 integration intervals per tether;
- we consider drag-mode airfoils in which the time-derivative of the onboard turbine drag coefficients $\kappa_\cdot$, is controlled and actuation is instantaneous;
- the time derivatives of the control surfaces (ailerons, elevators and rudder) deflections $\dot{a}_\cdot^{\mathrm{a}}$, $\dot{a}_\cdot^{\mathrm{e}}$, $\dot{a}_\cdot^{\mathrm{r}}$ are directly controlled.

A detailed derivation of the model and the parameters used can be found in [126]. The tree-like structure of the system is composed of three nodes: node 0, where the secondary tethers are attached to the main tether and nodes $1, 2$, corresponding to the centre of mass of each airfoil.

Analogously to the previous study on single airfoils, the wind turbulence model used in MHE is assuming a superposition of a laminar logarithmic windshear model in the x-direction and a turbulent wind field. For control and estimation purposes, it is proposed here to use a rather simple turbulence model, given by

$$\dot{w}_i^{\mathrm{t}} = -\frac{w_i^{\mathrm{t}}}{\tau} + u_i^{\mathrm{t}}, \qquad i = 1, 2, \tag{7.51}$$

where u_i^{t} is the forcing term in this first-order differential equation, here modelled as a pseudo-control. While in the estimation problem the optimiser is free to choose the term u_i^{t} so as to best fit the measurements, in the control problem, these modes are uncontrollable and, for the prediction horizon, the forcing term is assumed to be 0.

The state vector $X \in \mathbb{R}^{56}$ and the control vector $U \in \mathbb{R}^{14}$ are respectively given by

$$X = (r_0,\, \dot{r}_0,\, r_1,\, \dot{r}_1,\, r_2,\, \dot{r}_2,\, e_1^x,\, e_1^y,\, e_1^z,\, e_2^x,\, e_2^y,\, e_2^z,\, \omega_1,\, \omega_2, a_1,\, a_2,\, \kappa_1,\, \kappa_2,\, w_1^{\mathrm{t}}\, w_2^{\mathrm{t}}),$$

$$U = (u_1,\, u_2,\, \dot{\kappa}_1,\, \dot{\kappa}_2,\, u_1^{\mathrm{t}},\, u_2^{\mathrm{t}}).$$

Note that, when solving the trajectory planning optimal control problem we assume $w_k^{\mathrm{t}} = 0$, $u_k^{\mathrm{t}} = 0$, $k = 1, 2$, while for MPC we assume $u_k^{\mathrm{t}} = 0$, $k = 1, 2$.

Power Optimal Trajectories

The trajectory extracting the maximum amount of energy from the airmass is computed by solving a periodic optimal control problem (OCP) for a given reference wind speed w_0. The period T of the orbit, and some parameters such as the tether lengths l_k and the tether diameters d_k are treated as optimisation variables and lumped together in vector $p = \begin{bmatrix} T & l_1 & l_2 & d_1 & d_2 \end{bmatrix}^\top$. Note that, based on the same reasoning used in Section 7.4, we optimise over a half period.

The trajectories obtained with the proposed Algorithm 1 are displayed in Figures 7.19 and 7.20. It can be seen that the feasible initial guess obtained after the homotopy procedure tracks the selected reference, which is infeasible. The point at which the main and secondary tethers are joined (shown in blue) moves slightly in the guess obtained after the homotopy, but it moves much less in the power optimal trajectory. It can also be noted that the power optimal trajectory can be quite different from the initial guess.

The lift coefficient C^{L} and extracted power P are displayed in Figure 7.21. The initial guess is infeasible for the lift coefficient, which must satisfy $C^{\mathrm{L}} \leq 1$. As opposed to the feasible trajectory, the power optimal one pushes the lift coefficient close to its bound for the whole orbit. Moreover, the extracted power is always positive and much larger than the one extracted by the feasible trajectory. The power optimal trajectory also has a shorter duration, as the power optimal orbit time is $T \approx 5.02\,\mathrm{s}$, while the OCP had been initialised with an arbitrary orbit time $T = 10\,\mathrm{s}$.

Feedback Control in a Perturbed Wind Field

MPC Constraints and Tuning The control surfaces deflections a_i^{a}, a_i^{e}, a_i^{r} and their u_i^{a}, u_i^{e}, u_i^{r} rates are constrained by the mechanical design of the airfoil and the chosen actuators.

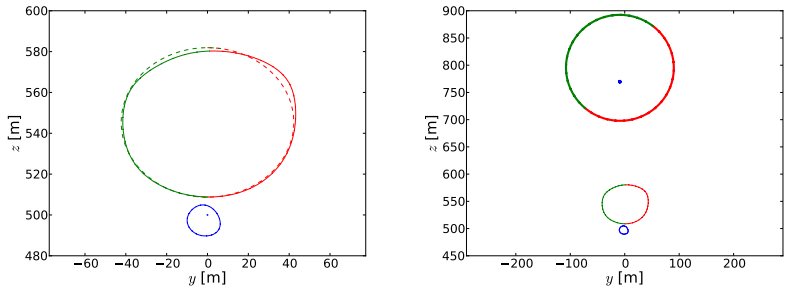


Figure 7.19: Initialisation procedure: central node connecting the three tethers in blue, kite 1 in red and kite 2 in green. Left figure: arbitrarily chosen 0-shaped reference (dashed line) and feasible initial guess (continuous line) obtained after the homotopy. Right figure: power optimal trajectory (thick line) and feasible initial guess (thin line).

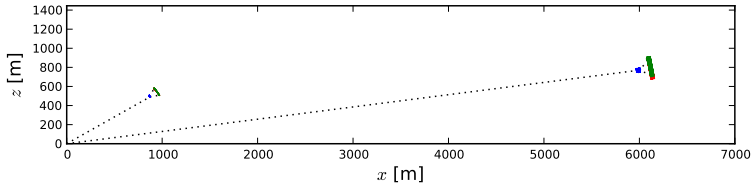


Figure 7.20: Initialisation procedure, side view: the kites are shown in red and green, the point joining the main and secondary tethers is shown in blue and the tethers in black (dotted lines). The power optimal trajectory is shown in thick line, and the feasible trajectory in thin line.

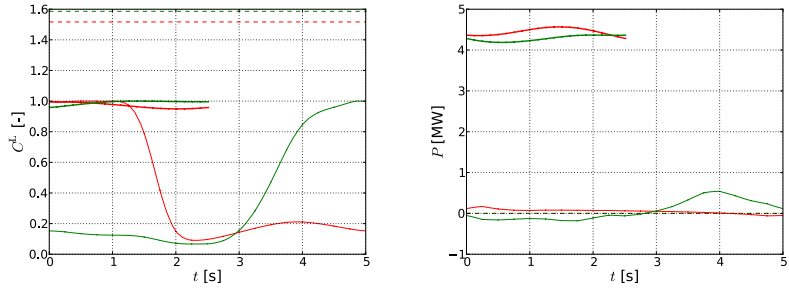


Figure 7.21: Initialisation procedure: kite 1 in red and kite 2 in green for the initial guess (dashed line), the feasible trajectory (continuous line) and the power optimal trajectory (thick line). Left figure: lift coefficient C^{L}. Right figure: extracted power P.

Variable	Description	Unit	Bounds
a_i^{a}	Aileron	deg	[-15,15]
a_i^{e}	Elevator	deg	[-18.3,18.3]
a_i^{r}	Rudder	deg	[-11.5,11.5]
u_i^{a}	Aileron rate	deg/s	[-13.2,13.2]
u_i^{e}	Elevator rate	deg/s	[-1.7,1.7]
u_i^{r}	Rudder rate	deg/s	[-11.5,11.5]
C_i^{L}	Lift coefficient	–	[-1,1]
F_i^{t}	Tether tension	N	$[0,\infty)$

Variable	Weight
r_i	8e-8
$\dot{r}_i$	1.5e-6
R_i	6e-8
ω_i	6e-8
$a_i^{\mathrm{a}},\ a_i^{\mathrm{e}},\ a_i^{\mathrm{r}}$	0.1
κ_i	9.4e-6
$u_i^{\mathrm{a}},\ u_i^{\mathrm{e}},\ u_i^{\mathrm{r}}$	9.4e-6
u_i^{κ}	9.4e-8

Table 7.4: System constraints and MPC weights.

Sensor	Measurement	Variable	σ
IMU	Accelerations in frame e_i	a_i^{IMU}	$5\,\mathrm{cm}\cdot\mathrm{s}^{-2}$
IMU	Angular velocities	ω_i	$1\,\mathrm{deg}\cdot\mathrm{s}^{-1}$
GPS	Absolute positions	r_i	$0.1\,\mathrm{m}$
GPS	Absolute velocities	$\dot{r}_i$	$0.6\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Variometer	Absolute vertical velocity	$\dot{z}_i$	$0.5\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Tether gauge	Tether tension	F_i^{t}	$500\,\mathrm{N}$
Pitot tube	Long. relative velocity	v_i^x	$1\,\mathrm{m}\cdot\mathrm{s}^{-1}$
Air Probe	AoA	α_i	$2.5\,\mathrm{deg}$
Air Probe	Side-slip angle	β_i	$5\,\mathrm{deg}$
Encoders	Control surfaces deflections	$a_i^{\mathrm{a}},\ a_i^{\mathrm{e}},\ a_i^{\mathrm{r}}$	$0.1\,\mathrm{deg}$

Table 7.5: Available sensors for MHE, with the corresponding noise standard deviation σ.

In addition to the previous constraints, in order to keep the system in the region where the model assumptions are valid, further path constraints need to be added on the lift coefficients C_i^{L} to avoid stall and on the tether tension F_i^{t}, to make sure that the tethers are never in compression.

All constraints are summarised in Table 7.4. MPC has been discretised using a sampling time $T_{\mathrm{s}} = 125$ ms and a prediction horizon of $N_{\mathrm{NMPC}} = 20$ elements. The MPC weighting matrices Q and R were chosen diagonal, with each entry specified in Table 7.4, accordingly with the state it corresponds to. Note that the units of the weights are defined consistently with the variables they correspond to, so as to yield a dimensionless cost.

MHE & Available Sensors The airfoils are assumed to be both equipped with GPS, an inertial measurement unit (IMU), a Pitot tube, a variometer, an air probe and control surface encoders. The IMU onboard each airfoil i measures the linear accelerations $a_i^{\text{IMU}} = R_i^\top \ddot{r}_i + \begin{bmatrix} 0 & 0 & g \end{bmatrix}^\top$ and the rotational velocities ω_i, given in frame e_i.

All available sensors are listed in Table 7.5 with the associated standard deviation σ. Because of actuator noise and inaccuracy, the control inputs computed by the controller, may not be perfectly implemented by the system and are also subject to noise. They are thus added as pseudo-measurements to the cost function.

MHE has been discretised using a sampling time $T_\text{s} = 125$ ms and an estimation horizon of $N_\text{MHE} = 10$ elements. The MHE matrix Σ^{-1} was chosen so as to match the variance of the measurement noise, given in Table 7.5. Based on the same variance, Gaussian noise has been added to all measurements in the proposed simulations.

Simulation Results The proposed scenario considers tracking a power optimal periodic reference trajectory in a turbulent wind. For the simulated wind turbulence and the MHE wind turbulence model the same approach used for the single airfoil has been used. The reference trajectory and the simulated trajectory are displayed in Figure 7.22. The direction of the nominal wind speed w^ℓ is assumed to be known and is also displayed in Figure 7.22. It can be seen that the airfoil is well stabilised around the reference even in the presence of strong turbulences.

The wind turbulences estimated by the MHE are displayed in Figure 7.23, where it can be seen that the estimates closely match the real turbulences. In particular, as shown in Figure 7.23, the power spectral density (PSD) of the estimation error is much lower than the one of the original signal at low frequencies. The signal-to-noise ratio (SNR) is thus high at low frequencies. At higher frequencies, where the SNR decreases, also the PSD of the turbulences is low.

For the purpose of illustrating the ability of NMPC to deal with constraints, the bounds on the control surfaces and their rates have been artificially tightened so as to activate them often in the simulation scenario. The resulting trajectories are displayed in Figure 7.22, together with the bounds. It can be seen that the trajectories of the control surfaces are far from being periodic as they are used to react to wind turbulences and stabilise the system.

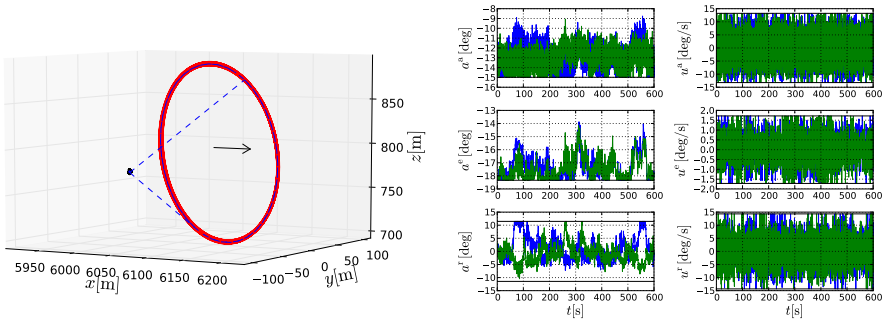


Figure 7.22: Left figure: airfoil trajectories (red line), joint position (thick black line), reference trajectory (dashed blue line) and the nominal wind direction (black arrow). Right figure: control surfaces and their time rates.

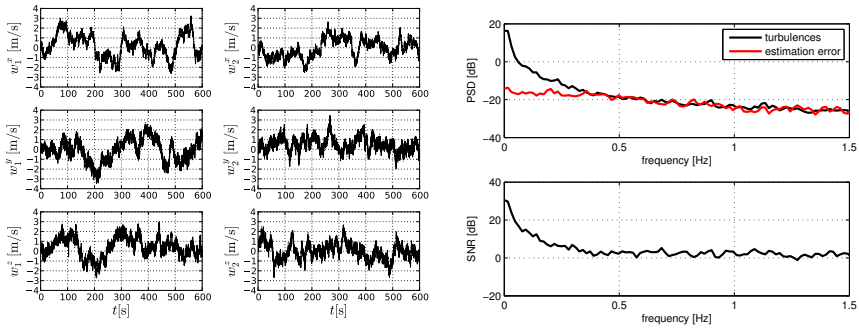


Figure 7.23: Left figure: real wind turbulence in dashed line and turbulence wind speed estimated by the MHE in thick line for the two airfoils; as the estimation error is small, the dashed line is covered by the continuous line. Right figure: power spectral density of the turbulences and of the estimation error (top graph); signal-to-noise ratio for the estimated signal (bottom graph).

The simulations were run on a 2.8 GHz CPU. The computational times are consistently below $T_s = 125$ ms, allowing for a real-time implementation.

7.7 Rotational Startup

The studies on power extracting orbits established the potential of AWE, however, a strategy for safe launch and landing of AWE systems is fundamental in order to ensure applicability of this technology in practice. For AWE systems in drag mode, vertical take off and landing (VTOL) has been proposed and successfully applied in practice [76]. The main idea is to use the on board generators as propellers in the take off and landing phases and the airfoil becomes a quadcopter-like system. For AWE systems in pumping mode, several options have been investigated, one of which is rotational startup and landing. The main idea is to attach the airfoil to a rotating arm, or carousel, and to use some engine on the ground in order to add momentum to the system, until the airfoil is able to reach an altitude at which it can use wind power in order to remain airborne and transition to a power generating orbit.

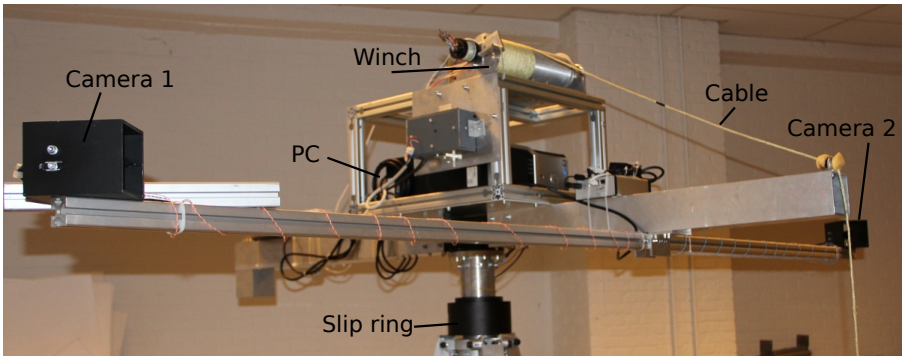
In this section we present both simulations and experiments based on a single airfoil in a rotational startup configuration. Both in simulations and experiments we used the model derived in Section 7.2.5. The details of the model derivation and all model parameters can be found in [121]

7.7.1 Experimental Setup

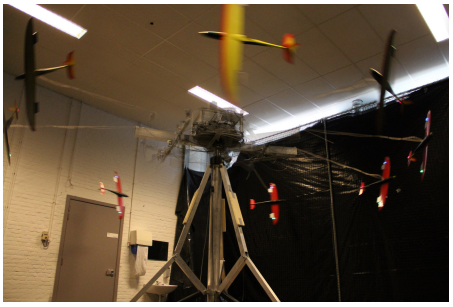
For the experiments, we used the carousel setup which has been built at KU Leuven for investigating rotational startup in the framework of the ERC HIGHWIND project. We remark that the main scope of this test setup was not only to investigate rotational startup, but also to test the hardware, software and control algorithms in a controlled and safe environment.

Hardware

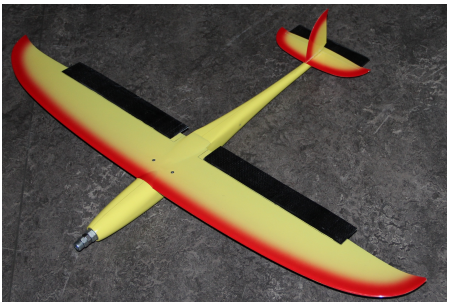
The carousel test setup is small enough to operate indoors, allowing the system to be tested before moving to a more unpredictable (windy) outdoors environment. This also allows experiments to be performed year-round, independently of the weather conditions. The test setup is shown in Figure 7.24a and 7.24b. While a production system might use a lighter structure, the test setup design has high stiffness and high rotating mass. As a result, the forces of the airplane have



(a) Close-up of top of carousel



(b) The rotating carousel



(c) Ariane P5

Figure 7.24: KU Leuven test setup

negligible influence on the structure of the carousel, and only minimal influence on the rotation speed of the arm. The carousel can rotate at a maximum speed of 60 rotations per minute with the airplane attached to it. A winch with a power of 400 W is placed on the carousel to control the tether length.

The airplane used for the experiments is an Ariane P5, shown in Figure 7.24c. This model is typically used for F5D class radio controlled aircraft pylon racing. These airplanes have to take sharp turns at high velocities, and are therefore built from carbon fibre, giving them high strength and stiffness. The Ariane P5 has a wingspan of 1 m, a surface area of 0.1 m^2 and a weight of 0.6 kg when fully equipped with all sensors, actuators and electronics. It has ailerons and elevators to control roll and pitch dynamics, but lacks a rudder. To increase the control authority of the airplane and make the control experiments more interesting, the ailerons and elevators are enlarged with carbon plates. This is needed because the tether is attached to the fuselage 2.5 cm under the centre of

mass, resulting in a restoring torque when the airplane rolls or pitches. This torque has a stabilizing effect, but the original control surfaces cannot provide enough torque to overcome this effect at short tether lengths.

The tether used in this AWE setup contains 3 pairs of insulated copper wires. Two pairs are used for the communication between the airplane and the ground computer. The third pair is used to provide power to the electronics and servomotors in the airplane. The copper pairs are enclosed in a sheath of braided Kevlar that takes the mechanical tension. The outer diameter of the cable is 3.6 mm.

Sensors

The sensors available on the test setup include an encoder for measuring the carousel angle, a stereo vision system, and an Inertial Measurement Unit (IMU). The measurements coming from the different sensors will be used by MHE to compute a state estimate. Since in the current implementation MHE assumes the measurements to come at prescribed times, particular care was taken to synchronize the different sensors.

The *stereo vision system* consists of two Point Grey Flea3 cameras with a resolution of 1600 by 1200 pixels that can take images at a maximum sampling frequency of 15 Hz. The cameras are mounted on the carousel and observe three markers (red, green and blue LEDs) mounted on the airplane. This provides information to estimate the position and orientation of the airplane.

The *six degrees of freedom IMU* (Analog Devices ADIS16367) measures linear acceleration and angular velocities of the airplane in all directions. The maximum sampling frequency of the IMU is 800 Hz, which is fast enough to capture the major mechanical modes of the tethered aircraft.

7.7.2 Simulation Results

For a tether length $r = 0$ the system is in a singular configuration and the proposed model is not valid. In practice such a configuration is undesirable and the airfoil can be docked to a support at $r \neq 0$.

At very short tether lengths, the system is open loop stable. The control strategy can thus consist in reeling out the tether until a minimal length is reached. By keeping the system in the open loop stable region, MPC and MHE can be warm started. After this phase of initialization, the control algorithms can safely control the process. A similar strategy can be used for landing: the controller

Time	z_r [m]	r_r [m]	$\dot{\delta}_\mathrm{r}$ [rad/s]
$5 \leq t < 11$	0	1.2	2π
$11 \leq t < 23$	2	6.0	1.5π
$23 \leq t < 35$	0	1.2	2π

Table 7.6: Reference trajectory for the NMPC.

needs to bring the system into the open loop stable region and, from there, the tether can be reeled in until the airfoil docks to the carousel.

Scenario The simulation scenario consists in tracking a static reference trajectory, obtained by computing the equilibrium for given height $z = z_\mathrm{r}$, tether length $r = r_\mathrm{r}$, rotation speed $\dot{\delta} = \dot{\delta}_\mathrm{r}$ and no wind ($w = 0$).

The detailed derivation of the model is given in [121]. The model parameters are relative to the small scale setup at KU Leuven and are reported in [45]. The proposed scenario considers a turbulent wind, displayed in Figure 7.27 (solid line). After 12 s an abrupt change in the wind speed and direction is introduced. At every time step t_i a static reference, reported in Table 7.6, is provided.

MPC and MHE Setup The NMPC prediction horizon was set to $T_\mathrm{P} = 1\,\mathrm{s}$ and the problem was discretised using $N = 10$ elements of uniform duration. The system dynamics are discretised over the shooting intervals via an implicit Gauss-Legendre Runge-Kutta integration method of order 2. The NMPC sampling time $T_\mathrm{s} = t_i - t_{i-1}$ was chosen as $T_\mathrm{s} = T_\mathrm{P}/N$. The MPC weights and constraints are summarised in Table 7.7. Note that the units of the weights are defined consistently with the variables they correspond to, so as to yield a dimensionless cost.

The MHE estimation horizon was set to $T_\mathrm{E} = 1\,\mathrm{s}$ and the problem was discretised using $N = 10$ elements of uniform duration. The MHE weights were chosen so as to match the variance of the measurement noise, given in Table 7.8. Based on the same variance, Gaussian noise has been added to all measurements obtained from simulations.

Numerical Results In the first 5 s the tether is reeled out to $r = 1.2$ m. After this phase, the control scheme starts to control the plane by keeping it in the stable region. After 11 s, the reference is changed and the controller elongates the tether and steers the airfoil to an altitude of $z = 2$ m. In the rest of

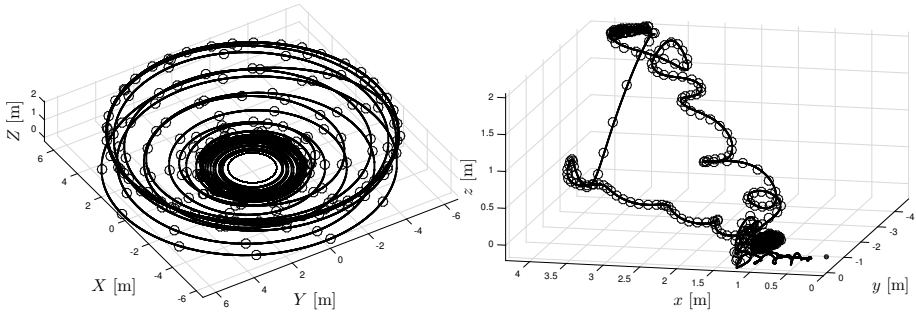


Figure 7.25: Aircraft trajectory 3D plot (thick line) with turbulent wind. Left figure (system seen from a fixed point of view): the MHE estimates are shown as circles and the carousel armtip trajectory as a thin line. Right figure (system seen from the carousel arm tip): the MHE estimates are shown as circles and the carousel armtip as a black dot.

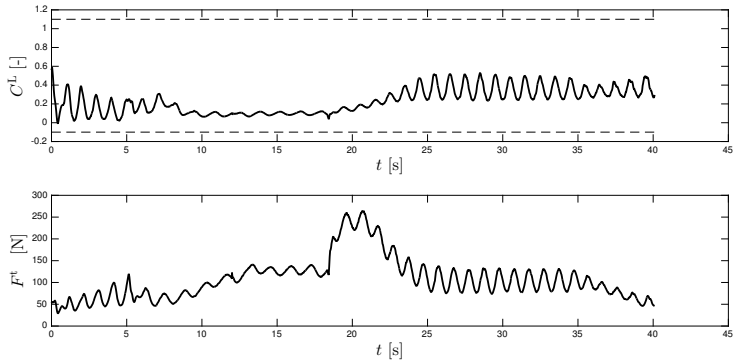


Figure 7.26: Lift coefficient C^{L} and tether tension F^{t} for the given trajectory. The path constraints on those variables are never active.

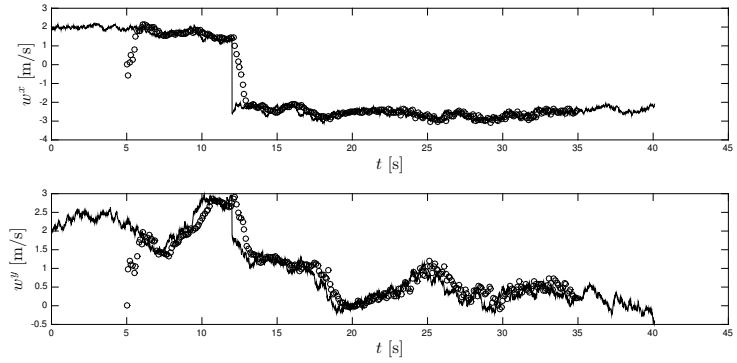


Figure 7.27: Real wind profile (x and y components) in continuous line and wind speed estimated by the MHE (x and y components) in circles.

Variable	Description	Unit	Bounds	Variable	Weight
a^{a}	Aileron	rad	[-0.3,0.3]	$r, \dot{r}$	10^{-3}
a^{e}	Elevator	rad	[-0.3,0.3]	R	10^{-2}
u^{a}	Aileron rate	rad/s	[-20,20]	ω	10^{-2}
u^{e}	Elevator rate	rad/s	[-20,20]	$a^{\mathrm{a}},\ a^{\mathrm{e}}$	10^{-2}
$\ddot{\delta}$	Carousel acceleration	deg/s^2	[-30,30]	$u^{\mathrm{a}},\ u^{\mathrm{e}}$	10^{-2}
C^{L}	Lift coefficient	–	[-0.1,1.1]	$l,\ \dot{l},\ \ddot{l}$	10^{-2}
F^{t}	Tether tension	N	$[0,\infty)$	$\delta,\ \dot{\delta},\ \ddot{\delta}$	10^{-2}

Table 7.7: System constraints and weights.

Sensor	Measurements	Standard deviation
IMU	Linear acceleration	0.25 m/s^2
IMU	Angular velocity	0.025 rad/s
Cameras	LED position	25 px
Encoder	Tether length	$2.5 \cdot 10^{-3}$ m
Encoder	Carousel angle	$2.5 \cdot 10^{-3}$ rad
Encoder	Control surface angle	$2.5 \cdot 10^{-4}$ rad

Table 7.8: Available measurements

the simulation, the airfoil is controlled back to the stable initial configuration where the tether can be reeled in. The proposed procedure is reversible and the controller is able to safely bring the airfoil to landing.

The trajectory obtained for the given scenario is displayed in Figure 7.25 as a thick line, together with the state estimates in circles. It can be seen that in the initial and final part of the trajectory, the airfoil is close to the armtip and no estimate is shown, as the system is in the open-loop stable region and doesn't need any feedback control. The lift coefficient and the tether tension are displayed in Figure 7.26. Note that the proposed path constraints are never active for the proposed scenario.

The x and y components of the wind speed are introduced as parameters to be estimated by the MHE scheme. Their estimates are displayed in Figure 7.27 in circles. Though no direct measurement of the wind speed is available, it can be seen that the proposed MHE scheme provides a good estimate to both wind speed and direction. The introduction of adequate wind models in the MHE scheme is subject of ongoing research at the KU Leuven.

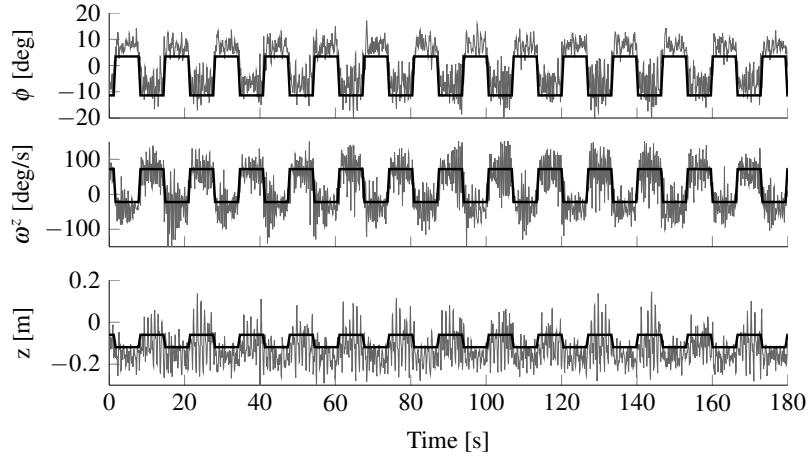


Figure 7.28: Closed loop tracking behavior for roll ϕ, angular velocity ω^z and height z. Reference in black, estimated variables in grey [42].

The simulations were run on a 2.8 GHz CPU and the computational times are consistently below $T_s = 100$ ms, allowing for a real-time implementation.

7.7.3 Experimental Results

Several closed-loop experiments were performed on the KU Leuven test setup. An experiment was performed at a constant carousel velocity of 60 rpm for a duration of 2.6 hours, during which the setpoint for the controller was periodically toggled between two values. The setpoints are computed by finding the steady state of the airplane, given a certain feasible height z. The heights were chosen here to be 0.118 m and 0.06 m below the arm level. The lower setpoint height is the height the airplane would have when all control surfaces are neutral. The Trajectory generator holds one setpoint for 6 s at a time and then transitions to the other setpoint. The transition from one setpoint to the other is done by a linearly interpolated ramp in the height state with a duration of 0.5 s.

Figure 7.28 shows the closed-loop tracking performance for the roll angle ϕ, angular velocity around the z-axis and for the height z. The roll is defined here to be 0° when the wing is vertical. From the figure, it is clear that there is an oscillation that cannot be controlled. The plane is flying through its own wake, which is not included in the model. Therefore, it is not surprising that the controller cannot suppress all perturbations. Moreover, since there is no

direct yaw control via a rudder and the airplane has a small vertical stabilizer, there is little yaw damping. This makes the control of these oscillations hard or even impossible, however, suppressing these oscillations is not the goal of the current experiments. The figures show that, except for the oscillations described before, the tracking behaviour for the roll and angular velocity is reasonable. The tracking behaviour for the height is worse. This is related to the fact that the height is only indirectly controlled via the orientation and therefore more sensitive to unmodelled dynamics and unmeasured perturbations.

7.8 Towards Tuned Tracking MPC for Power Maximisation

When tuning the MPC schemes presented in Section 7.5 and 7.6, it became clear that there was a large potential for increasing the energy harvested by the system depending on how the tuning was chosen. For instance, when considering a single airfoil in pumping mode, reducing the cost relative to the position and tether length let the system adapt to the wind turbulences in a way which harvested significantly more energy. This observation served as a motivation for investigating systematic tuning techniques. In this section, we apply the tuning technique proposed in Chapter 5 in order to design a tracking MPC scheme which approximately solves the economic problem, i.e. maximising the following performance index:

$$\ell(X,U) = -\frac{E(X,U)}{T_\mathrm{s}} + 10\|U\|_2^2,$$

$$E(X,U) = \int_0^{T_\mathrm{s}} v^\top F^\mathrm{G} \mathrm{d}t = \int_0^{T_\mathrm{s}} \frac{1}{2}\rho A \kappa\, v^\top v \|v\|_2 \,\mathrm{d}t,$$

where X and U denote the states and controls respectively, E denotes the extracted energy and the second term in the cost, which penalises using the controls too much, is added to reduce fatigue of the actuators and hence maintenance costs. Note that, because we formulate the problem as a minimisation problem, the energy has a minus sign.

In order to achieve this goal, we combine most of the contributions of the thesis: we apply the tuning technique proposed in Chapter 5 to an AWE single-airfoil system modelled by a DAE with invariants (see Chapter 6 and Section 7.2). We make use of the techniques presented in Section 7.3 in order to initialise the periodic OCP for computing the maximum power orbit.

In this section, we adopt the following modelling choices:

- the airfoil is inertially modelled as a rigid body with rotational and translational inertia;
- the tether is modelled as a rigid rod with mass;
- we consider a drag-mode airfoil in which the the time-derivative of the onboard turbine drag coefficient κ, is controlled and actuation is instantaneous;
- the time derivatives of the control surfaces (ailerons, elevators and rudder) deflections $\dot{a}^{\mathrm{a}}$, $\dot{a}^{\mathrm{e}}$, $\dot{a}^{\mathrm{r}}$ are directly controlled.

In summary, the system is the drag-type equivalent of the pumping-type model used in Section 7.5. The only difference between the two models is that instead of using a tether of variable length and a generator on ground, we make use of on-board generators.

Similarly to the previous sections, we formulate the power optimal trajectory planning problem as an optimal control problem. However, in order to obtain the correct Lagrange multipliers, and hence the correct Hessian, starting from the optimal solution, we formulate the periodicity and invariant conditions in a slightly different way. Because we stabilise the invariants we do not impose any condition on them and rather impose periodicity of all states. Note that this choice does not violate LICQ; moreover, it also enforces the invariant conditions: because the invariants are uncontrollable and have asymptotically stable dynamics, the only possible periodic solution is to always satisfy the invariants.

Using the block diagonal Hessian from the periodic optimal control problem, we formulate the periodic SDP (5.49) for convexifying the stage cost, as proposed in Chapter 5. We then use the obtained positive definite stage cost in tracking nonlinear MPC. Note that all nonlinear path constraints are imposed in their slack version, i.e. like in Problems (5.37) and (5.38).

We tested this approach on a single airfoil system. As a preliminary result, we investigated the MPC feedback control for a series of initial conditions and compared economic MPC, tuned tracking MPC and standard tracking MPC. Note that, for both tracking MPC schemes we also used the gradient of the Lagrangian of the rotated problem in the stage cost, as explained for the steady-state case in Section 5.4.1. We moved the initial condition by starting on the periodic optimal trajectory and varying the height $z_0 = z_0^{\mathrm{p}} + \Delta z$ while keeping $y_0 = y_0^{\mathrm{p}} = 0$. In order to keep a consistent initial condition, we used the following formula

$$x_0 = \sqrt{z_0^2 - r^2}, \qquad \dot{z}_0 = -\frac{x_0 \dot{x}_0^{\mathrm{p}}}{z_0}.$$

We implemented all MPC controllers with a horizon $N = 30$ and a terminal constraint.

The results are displayed in Figure 7.29, where the controls are displayed as the increment over the periodic optimal control, i.e. $\Delta U = U_0^{\mathrm{MPC}} - U_0^{\mathrm{p}}$. It can be seen that the tuned tracking MPC scheme approximates the economic MPC scheme very well, while the standard MPC scheme does not. The periodic optimal controls are displayed in Figure 7.30. It is interesting to note that the MPC feedback controls are very sensitive even to very small perturbations of the initial state.

The cost of the MPC open-loop predicted trajectory is displayed in Figure 7.31, where we define $J = \sum_{k=0}^{N-1} \ell(X_k, U_k)$. It can be seen that, as the perturbation of the initial state becomes larger, the economic cost delivered by the standard tracking NMPC formulation quickly increases, while when using the tuned tracking NMPC scheme, one obtains a cost similar to that obtained using economic NMPC.

The 3D plot of the periodic optimal trajectory is displayed in Figure 7.32 together with the MPC open-loop predictions starting from the perturbed initial condition for all three MPC schemes. It can be seen that the standard tracking MPC scheme brings the airfoil close to the periodic optimal trajectory much quicker than the economic and tuned tracking MPC schemes.

These results are preliminary, but demonstrate that the techniques developed in Chapter 5 can be applied also to complicated mechanical systems and, even in the presence of strong nonlinearities, they can perform very well. Future work will include investigating this application more in detail and trying to use the proposed approach in the RTI framework.

One extension of the proposed scheme consists in including the wind turbulences using the simple model (7.51) and assuming zero turbulences in the power optimal periodic trajectory design. By doing so, the tuned tracking MPC scheme will also take into account the sensitivity of the economic MPC scheme to wind turbulences and is therefore expected to have improved performance when operated in a turbulent wind.

7.9 Conclusions

In this chapter we have applied the modelling techniques developed in Chapter 6 in order to derive optimisation-friendly models of reduced complexity. We have proposed problem formulations and initialisation strategies which make it

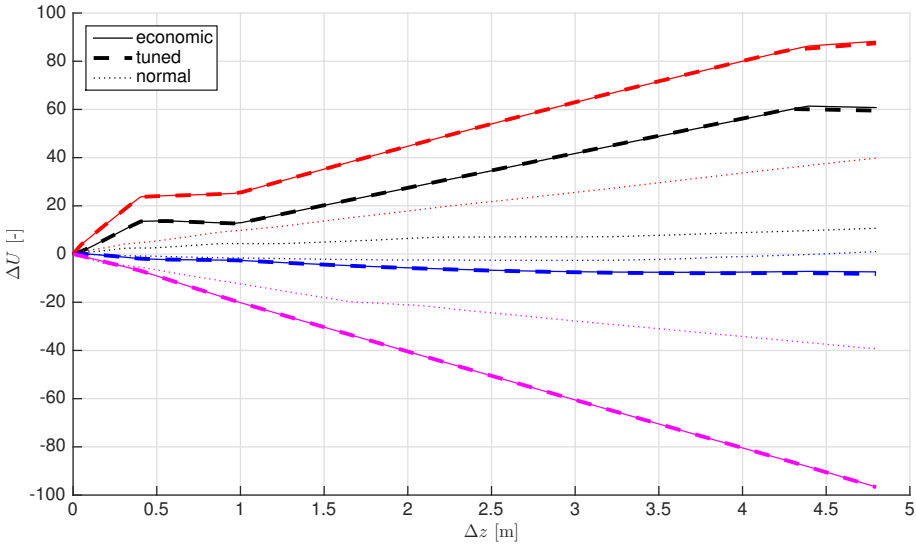


Figure 7.29: MPC feedback control law for varying initial conditions. The controls are displayed as follows: u^{a} in black, u^{e} in red, u^{r} in blue, u^{κ} in magenta.

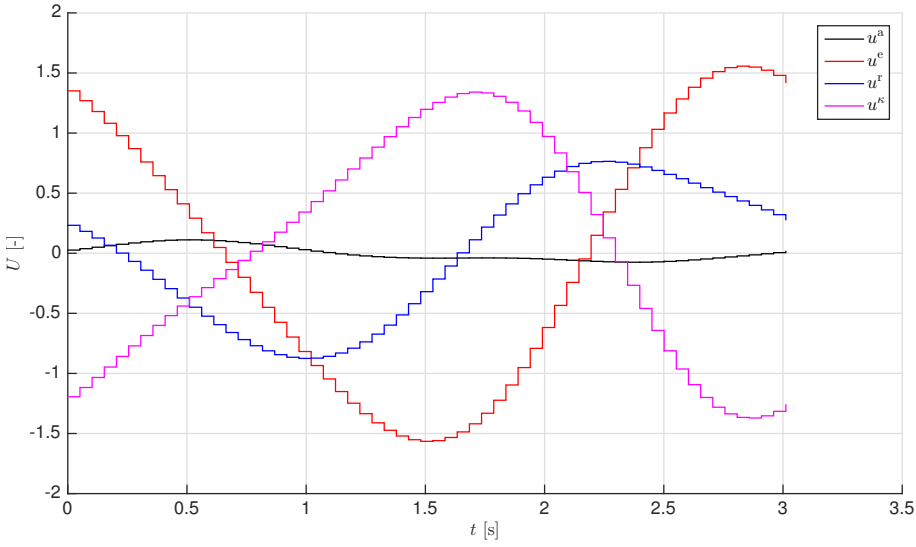


Figure 7.30: Periodic optimal controls.

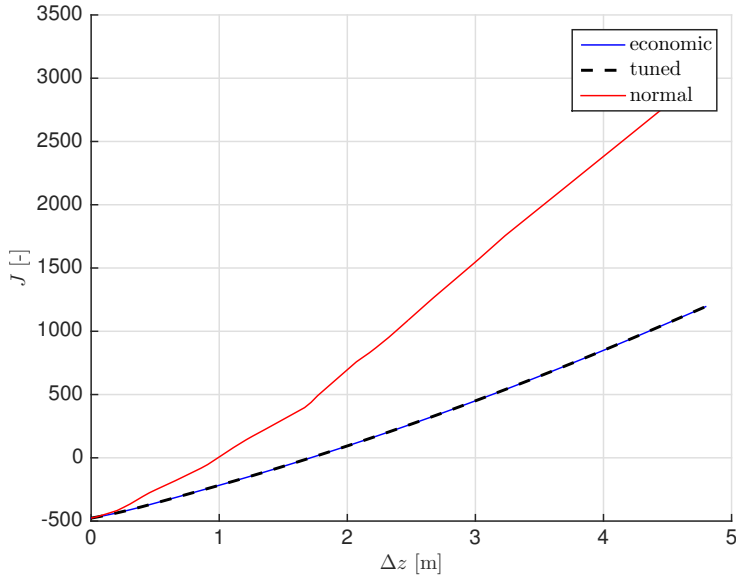


Figure 7.31: Cost of the open-loop NMPC predicted trajectory for the three NMPC schemes.

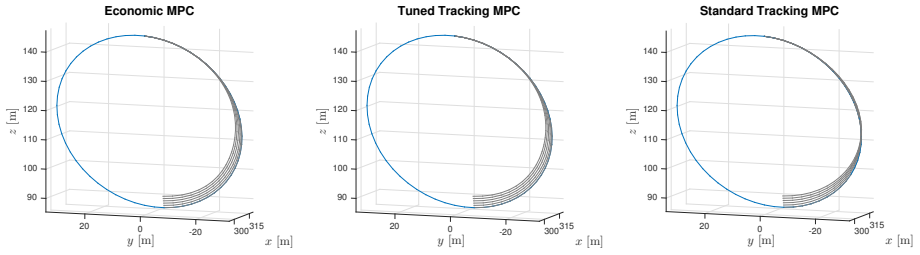


Figure 7.32: 3D plot of the optimal periodic trajectory and the MPC predictions starting from the perturbed initial condition for the three MPC controllers. Blue: periodic optimal trajectory, grey: open-loop MPC predictions, black: tether position at the beginning of the periodic optimal trajectory.

possible for the algorithms to converge in a fast and reliable way to the optimal solution.

We have applied the optimal control techniques to the comparative study of single and dual airfoils in drag mode while also modelling the interaction between the airfoil and the airmass. This has allowed us to conclude that dual airfoils are expected to be more efficient than single airfoils for both very small and rather large wing areas. The study was conducted using a simplified model and future studies should aim at further investigating this comparison by considering (a) more realistic models, (b) conducting the study for pumping-mode systems, (c) more advanced models of the airfoil-airmass interaction, (d) jointly designing the optimal trajectory and airfoil shape that maximise the harvested energy and (e) validating the results in experiments.

We have then used MPC and MHE in order to estimate the state and wind turbulences and control both single and dual airfoils in a perturbed wind field. For the case of rotational startup we also conducted a first experimental validation of the controller. While simulations have yielded promising results, there are still many aspects to be investigated including (a) designing a strategy for the transition from the startup phase to the power generation phase and vice versa, (b) conducting an extensive experimental campaign in order to test the robustness of the controlled system in a perturbed wind field.

Finally, pushed by the observation that the harvested energy strongly depends on the MPC tuning, we have used the tuning procedure proposed in Chapter 5 in order to design a tracking NMPC scheme which locally behaves like economic MPC. Simulation results suggest that the tuned tracking MPC scheme is expected to deliver a control law very close to that delivered by economic MPC in a reasonably large region around the periodic optimal orbit.

Chapter 8

Conclusions and Outlook

The focus of this thesis has been on optimisation-based control and estimation with the aim of applying them in practice. Key elements for reliable implementations are (a) tailored algorithms, (b) efficient problem formulations, (c) adequate mathematical models.

In Chapter 2, we have introduced an algorithmic framework developed in the past literature which makes a fast and reliable implementation of optimisation-based control and estimation in real time possible. This chapter is a milestone on which the rest of the thesis is based.

Autonomous Driving

In Chapter 3, we have applied the framework of optimal control, MPC and MHE to autonomous driving: a challenging and nontrivial application. We have proposed a high-fidelity vehicle model which accurately models the tire contact forces and includes a suspension model. By means of MHE we have estimated the state and the unmeasured road-tire friction coefficient simultaneously and we have tested the MHE-MPC-based controller in simulations in an obstacle avoidance scenario on a low friction road. Finally, we have proposed a first step towards time-optimal MPC for small-scale race cars and we have tested the algorithm in experiments.

Future research directions include testing the control scheme on a real vehicle, further investigating the estimation of the road-tire friction coefficient and developing a framework for an arrival cost update also for L^1 and Huber penalties. On the side of time-optimal MPC for race cars, current research is

aiming at using more advanced vehicle models and algorithms for enhanced performance.

MPC Stability Theory

In Chapter 4, we have studied the stability properties of MPC: because optimality does not imply stability, this latter is not always guaranteed. It has been proven that, while for tracking MPC the requirements for guaranteeing stability are rather mild, for economic MPC closed-loop stability hinges on the satisfaction of a strict dissipativity condition. Stability theory for economic MPC is a young research topic and is still under development. Most available results consider the case in which the system is best operated at steady state. However, for many cases of interest, it is more advantageous to operate the system periodically. We have contributed two ways of extending currently available results to the periodic case. Using some examples, we have illustrated the theoretical developments.

Future research directions include further investigating time-varying periodic economic MPC under milder assumptions than those considered in current work. Moreover, though recent research greatly increased the understanding of economic MPC, there are still many aspects that deserve further investigations. Because the main advantage of economic MPC with respect to tracking MPC becomes evident in transients, economic gains are expected to be due to perturbations acting on the system. Therefore, it would be extremely interesting to investigate the robustness of economic MPC with respect to model uncertainties and perturbations both in terms of stability and closed-loop cost. Problem formulations which take into account explicitly the presence of perturbations are expected to deliver better closed-loop performance.

Tuning of Tracking MPC based on Economic Criteria

Because it is in general hard to prove strict dissipativity and it is computationally more demanding to solve economic MPC problems in real time, we have investigated in Chapter 5 the possibility of tuning tracking MPC such that it locally approximates economic MPC. In order to achieve this goal, we have analysed in depth the case of linear economic MPC, therefore also deepening the understanding of stability theory of economic MPC. We have proven that tuning tracking linear and nonlinear MPC with a positive definite stage cost such that it locally approximates economic NMPC up to first order both for the steady-state and the periodic case under some stability assumption. We have developed a technique for computing the stage cost of the tracking MPC scheme and we have applied it to some examples in order to demonstrate the

effectiveness of the approach. The advantages of the proposed approach consist in the fact that the obtained tracking NMPC scheme:

1. has a positive definite stage cost, so that ensuring stability of the closed-loop system is not harder than for any standard tracking MPC scheme;
2. accounts for the nonlinearities present both in the system dynamics and path constraints;
3. delivers a feedback control law which is proven to be a local first-order approximation of the economic NMPC feedback law;
4. allows one to use the efficient algorithms available for standard tracking NMPC, e.g. the RTI scheme based on generalised Gauss-Newton.

This research opens new research directions that include the following.

1. The nonlinear cost rotation can be used in its time-varying form (5.46) for online Hessian regularisation when using exact Hessian algorithm. This topic is already being investigated at the University of Freiburg.
2. The obtained positive definite stage cost matrix can be used to construct a Hessian approximation to be used in a new economic MPC algorithm.
3. Because every system is operated in a perturbed environment, the impact of perturbations should be taken into account in a stochastic or robust framework. Given the complexity of stochastic and robust problems, future research could aim at extending the framework proposed here in order to also account for the impact of perturbations on the cost in an approximate way.

Modelling of Mechanical Systems Aimed at Optimal Control

When deploying optimal control, MPC and MHE for fast mechanical systems, one can greatly benefit from using mathematical models of the system which are optimisation friendly. In Chapter 6, we have developed a framework for multibody systems based on the Euler-Lagrange approach and nonminimal coordinates. We have proposed an approach which delivers models of reduced complexity, in the form of index-1 DAEs with invariants. Direct methods for optimal control can straightforwardly handle such systems as we have shown both in Chapter 2 and Chapter 7.

Future research directions include investigating more optimisation-friendly problem formulations in order to handle the constraints imposed on the invariants in a systematic way.

Airborne Wind Energy

Airborne Wind Energy proposes to use tethered airfoils to extract wind energy. While its potential has been established in theory, its success hinges, among others, on a reliable orbit planning and control strategy. In Chapter 7, we have used the methods developed throughout this thesis in order to both achieve the goal and further study the potential of multi-airfoil systems. In particular:

1. we have provided strategies for having reliable implementations of optimal control, MPC and MHE;
2. we have compared single-airfoil and dual-airfoil systems by computing power optimal trajectories and we have concluded that dual airfoils can be expected to outperform single airfoils for all wing sizes;
3. we have introduced a simple model of the interaction between the airfoil and the airmass in order to investigate the reduction in the extracted power for both single and dual-airfoil;
4. we have used optimal control to perform trajectory and system design aimed at maximum energy harvesting;
5. we have applied MHE and MPC to single and dual airfoils for power generation in simulation and we have demonstrated that such control techniques are indeed able to stabilise AWE systems around a trajectory of maximum average energy harvesting;
6. we have applied MHE and MPC to a small-scale single-airfoil system with rotational startup both in simulations and experiments further demonstrating the capability of MPC and MHE to control AWE systems in real time;
7. we have applied the tuning strategy proposed in Chapter 5 to a single-airfoil and we have demonstrated its effectiveness in simulations.

The studies presented in this thesis constitute a starting point for further use of optimal control, MPC and MHE in order to analyse AWE systems more in depth. Future research directions include:

1. developing fatigue-reducing control schemes for large-scale systems;
2. improving the results obtained in simultaneous trajectory planning and system design by also adapting the aerodynamic properties of the wing;
3. investigating the interaction of AWE systems and the power grid;

4. further investigating the interaction of AWE systems and the airmass in order to develop more accurate interaction models;
5. further studying the comparison between single and dual airfoils by also considering pumping-mode systems;
6. applying the techniques used to design optimal trajectories and track them on a real system;
7. studying the behaviour of the tether by using accurate models and possibly deriving reduced models for control;
8. developing strategies for takeoff and landing based on robust optimal control;
9. studying the possibility of creating wind farms of AWE systems.

Bibliography

[1] Albersmeyer, J. *Adjoint-based algorithms and numerical methods for sensitivity generation and optimization of large scale dynamic systems*. PhD thesis, Ruprecht-Karls-Universitität Heidelberg, 2010.

[2] Albersmeyer, J., and Diehl, M. The Lifted Newton Method and its Application in Optimization. *SIAM Journal on Optimization 20*, 3 (2010), 1655–1684.

[3] Amrit, R., Rawlings, J., and Angeli, D. Economic optimization using model predictive control with a terminal cost. *Annual Reviews in Control 35* (2011), 178–186.

[4] Amrit, R., Rawlings, J. B., and Biegler, L. T. Optimizing process economics online using model predictive control. *Computers & Chemical Engineering 58* (2013), 334 – 343.

[5] Andersson, J. *A General-Purpose Software Framework for Dynamic Optimization*. PhD thesis, Arenberg Doctoral School, KU Leuven, October 2013.

[6] Andersson, J., Åkesson, J., and Diehl, M. CasADi – A symbolic package for automatic differentiation and optimal control. In *Recent Advances in Algorithmic Differentiation* (Berlin, 2012), S. Forth, P. Hovland, E. Phipps, J. Utke, and A. Walther, Eds., Lecture Notes in Computational Science and Engineering, Springer, pp. 297–307.

[7] Andrews, D. *An Introduction to ATMOSPHERE PHYSICS*. Cambridge University Press, 2010.

[8] Angeli, D., Amrit, R., and Rawlings, J. On Average Performance and Stability of Economic Model Predictive Control. *IEEE Transactions on Automatic Control 57* (2012), 1615 – 1626.

[9] Archer, C., and Caldeira, K. Global assessment of high-altitude wind power. *Energies 2* (2009), 307–319.

[10] Bär, V. Ein Kollokationsverfahren zur numerischen Lösung allgemeiner Mehrpunktrandwertaufgaben mit Schalt– und Sprungbedingungen mit Anwendungen in der optimalen Steuerung und der Parameteridentifizierung. Master's thesis, Universität Bonn, 1984.

[11] Baumgarte, J. Stabilization of Constraints and Integrals of Motion in Dynamical Systems. *Computer Methods in Applied Mechanics and Engineering 1* (1972), 1–16.

[12] Bellman, R. *Dynamic programming*. Princeton University Press, 1957.

[13] Bertsekas, D. *Dynamic Programming and Optimal Control*, 3rd ed., vol. 2. Athena Scientific, 2007.

[14] Betts, J. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd ed. SIAM, 2010.

[15] Bianchi, Fernando D. and Battista, Hernán de and Mantz, Ricardo J. *Wind Turbine Control Systems*. Springer, 2007.

[16] Biegler, L. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers and Chemical Engineering 8* (1984), 243–248.

[17] Biegler, L. *Nonlinear Programming*. 2010.

[18] Biegler, L. T. *Nonlinear Programming*. MOS-SIAM Series on Optimization. SIAM, 2010.

[19] Bittanti, S., and Colaneri, P. *Periodic Systems - Filtering and Control*. Springer, 2008.

[20] Bock, H. Recent advances in parameter identification techniques for ODE. In *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, P. Deuflhard and E. Hairer, Eds. Birkhäuser, Boston, 1983, pp. 95–121.

[21] Bock, H. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, vol. 183 of *Bonner Mathematische Schriften*. Universität Bonn, Bonn, 1987.

[22] Bock, H., and Plitt, K. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings 9th IFAC World Congress Budapest* (1984), Pergamon Press, pp. 242–247.

[23] Boggs, P., and Tolle, J. Sequential Quadratic Programming. *Acta Numerica* (1995), 1–51.

[24] Bossanyi, E. A. Further Load Reductions with Individual Pitch Control. *Wind Energy 8* (2005), 481–485.

[25] Boyd, S., and Vandenberghe, L. *Convex Optimization*. University Press, Cambridge, 2004.

[26] Braghin, F., Cheli, F., Melzi, S., and Sabbioni, E. Race driver model. *Computers & Structures 86* (July 2008), 1503–1516.

[27] Broyden, C. The convergence of a class of double rank minimization algorithms, part I and II. *J. Inst. Maths. Applns. 6* (1970), 76–90 and 222–231.

[28] Bryson, A., and Ho, Y.-C. *Applied Optimal Control*. Wiley, New York, 1975.

[29] Burton, T., Sharpe, D., Jenkins, N. and Bossanyi, E. *Wind Energy Handbook*. John Wiley and Sons, Ltd, Chichester, UK, 2001.

[30] Butcher, J. C. *Numerical Methods for Ordinary Differential Equations*. 2003.

[31] Diehl, M. *Real-Time Optimization for Large Scale Nonlinear Processes*. PhD thesis, Universität Heidelberg, 2001.

[32] Diehl, M., Amrit, R., and Rawlings, J. A Lyapunov Function for Economic Optimizing Model Predictive Control. *IEEE Trans. of Automatic Control 56*, 3 (March 2011), 703–707.

[33] Diehl, M., Bock, H., Schlöder, J., Findeisen, R., Nagy, Z., and Allgöwer, F. Real-time optimization and Nonlinear Model Predictive Control of Processes governed by differential-algebraic equations. *Journal of Process Control 12*, 4 (2002), 577–585.

[34] Diehl, M. and Bock, H.G. and Schlöder, J.P. Real-Time Iterations for Nonlinear Optimal Feedback Control. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on* (2005), pp. 5871 – 5876.

[35] Domahidi, A., Zgraggen, A., Zeilinger, M., Morari, M., and Jones, C. Efficient Interior Point Methods for Multistage Problems Arising in Receding Horizon Control. In *IEEE Conference on Decision and Control (CDC)* (Maui, HI, USA, Dec. 2012), pp. 668 – 674.

[36] Ferreau, H. J., Kirches, C., Potschka, A., Bock, H. G., and Diehl, M. qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation 6*, 4 (2014), 327–363.

[37] Fletcher, R. A new approach to variable metric algorithms. *Computer J. 13* (1970), 317–322.

[38] Frasch, J. V., Gray, A. J., Zanon, M., Ferreau, H. J., Sager, S., Borrelli, F., and Diehl, M. An Auto-generated Nonlinear MPC Algorithm for Real-Time Obstacle Avoidance of Ground Vehicles. In *Proceedings of the European Control Conference* (2013), pp. 4136–4141.

[39] Frasch, J. V., Sager, S., and Diehl, M. A Parallel Quadratic Programming Method for Dynamic Optimization Problems. *Mathematical Programming Computations 7*, 3 (2015), 289–329.

[40] Frison, G., Sorensen, H., Dammann, B., and Jorgensen, J. High-performance small-scale solvers for linear model predictive control. In *Proc. 2014 European Control Conference (ECC)* (June 2014), pp. 128–133.

[41] Geebelen, K., and Gillis, J. Modelling and control of rotational start-up phase of tethered aeroplanes for wind energy harvesting. Master's thesis, KU Leuven, June 2010.

[42] Geebelen, K., Vukov, M., Wagner, A., Ahmad, H., Zanon, M., Gros, S., Vandepitte, D., Swevers, J., and Diehl, M. An experimental test setup for advanced estimation and control of an airborne wind energy systems. In *Airborne Wind Energy*, U. Ahrens, M. Diehl, and R. Schmehl, Eds. Springer, 2013, pp. 459–471.

[43] Goldfarb, D. A family of variable metric methods derived by variational means. *Maths. Comp. 17* (1970), 739–764.

[44] Gray, A., Zanon, M., and Frasch, J. Parameters for a Jaguar X-Type. `http://www.mathopt.de/RESEARCH/obstacleAvoidance.php`, 2012.

[45] Gros, S., Ahmad, H., Geebelen, K., and Diehl, M. In-flight estimation of the aerodynamic roll damping and trim angle for a tethered aircraft based on multiple-shooting. In *System Identification Conference* (2012), pp. 1407–1412.

[46] Gros, S., and Diehl, M. Attitude estimation based on inertial and position measurements. In *Conference on Decision and Control* (2012), pp. 1758–1763.

[47] Gros, S., and Diehl, M. *Airborne Wind Energy*. Springer, 2013, ch. Modeling of Airborne Wind Energy Systems in Natural Coordinates.

[48] Gros, S., Zanon, M., and Diehl, M. Orbit Control for a Power Generating Airfoil Based on Nonlinear MPC. In *American Control Conference* (2012), pp. 137–142.

[49] Gros, S., Zanon, M., and Diehl, M. A Relaxation Strategy for the Optimization of Airborne Wind Energy Systems. In *European Control Conference* (2013), pp. 1011–1016.

[50] Gros, S., Zanon, M., and Diehl, M. Control of Airborne Wind Energy Systems Based on Nonlinear Model Predictive Control & Moving Horizon Estimation. In *European Control Conference* (2013), pp. 1017–1022.

[51] Gros, S., Zanon, M., and Diehl, M. Baumgarte Stabilisation over the SO(3) Rotation Group for Control. (accepted).

[52] Gros, S., Zanon, M., Vukov, M., and Diehl, M. Nonlinear MPC and MHE for Mechanical Multi-Body Systems with Application to Fast Tethered Airplanes. In *Proceedings of the 4th IFAC Nonlinear Model Predictive Control Conference, Noordwijkerhout, The Netherlands* (2012), pp. 86–93.

[53] Grüne, L. Economic receding horizon control without terminal constraints. *Automatica 49* (2013), 725–734.

[54] Grüne, L., and Pannek, J. *Nonlinear Model Predictive Control*. Springer, London, 2011.

[55] Guddat, J., Vasquez, F. G., and Jongen, H. *Parametric Optimization: Singularities, Pathfollowing and Jumps*. Teubner, Stuttgart, 1990.

[56] Hairer, E., Nørsett, S., and Wanner, G. *Solving Ordinary Differential Equations I*, 2nd ed. Springer Series in Computational Mathematics. Springer, Berlin, 1993.

[57] Hairer, E., Nørsett, S., and Wanner, G. *Solving Ordinary Differential Equations II – Stiff and Differential-Algebraic Problems*, 2nd ed. Springer Series in Computational Mathematics. Springer, Berlin, 1996.

[58] Horn, G., Gros, S., and Diehl, M. *Airborne Wind Energy*. Springer, 2013.

[59] Houska, B., and Diehl, M. Optimal Control for Power Generating Kites. In *Proc. 9th European Control Conference* (Kos, Greece„ 2007), pp. 3560–3567. (CD-ROM).

[60] Houska, B., Ferreau, H. J., and Diehl, M. ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization. *Optimal Control Applications and Methods 32*, 3 (2011), 298–312.

[61] Houska, B., Ferreau, H. J., and Diehl, M. An Auto-Generated Real-Time Iteration Algorithm for Nonlinear MPC in the Microsecond Range. *Automatica 47*, 10 (2011), 2279–2285.

[62] Huang, R., Biegler, L. T., and Harinath, E. Robust stability of economically oriented infinite horizon NMPC that include cyclic processes. *Journal of Process Control 22* (2012), 51–29.

[63] Jiang, Z.-P., and Wang, Y. A converse lyapunov theorem for discrete-time systems with disturbances. *Systems & Control Letters 45*, 1 (2002), 49–58.

[64] Joshi, A. *Elements of Group Theory for Physicists*. New Age International Publishers, 1997.

[65] Kehrle, F. Optimal Control of Vehicles in Driving Simulators. Master's thesis, Ruprecht–Karls–Unversität Heidelberg, 2010.

[66] Kehrle, F., Frasch, J. V., Kirches, C., and Sager, S. Optimal control of formula 1 race cars in a VDrift based virtual environment. In *Proceedings of the 18th IFAC World Congress* (2011), S. Bittanti, A. Cenedese, and S. Zampieri, Eds., pp. 11907–11912.

[67] Khalil, H. *Nonlinear Systems*, second ed. Prentice Hall, Upper Saddle River, NJ, 1996.

[68] Kiencke, U., and Nielsen, L. *Automotive Control Systems*. Springer, 2005.

[69] Kirches, C., Sager, S., Bock, H., and Schlöder, J. Time-optimal control of automobile test drives with gear shifts. *Optimal Control Applications and Methods 31*, 2 (2010), 137–153.

[70] Klintberg, E., and Gros, S. A Primal-dual Newton Method for Distributed Quadratic Programming. In *Conference on Decision and Control* (2014), pp. 5843–5848.

[71] Kozma, A., Klintberg, E., Gros, S., and Diehl, M. An improved distributed dual Newton-CG method for convex quadratic programming problems. In *American Control Conference* (2014), pp. 2324–2329.

[72] Kühl, P., Diehl, M., Kraus, T., Schlöder, J. P., and Bock, H. G. A real-time algorithm for moving horizon state and parameter estimation. *Computers and Chemical Engineering 35*, 1 (2011), 71–83.

[73] Laks, J., Pao, L., and Wright, A. Control of Wind Turbines: Past, Present, and Future. In *American Control Conference* (2009), pp. 2096–2103.

[74] Lovera, M., and Astolfi, A. Spacecraft attitude control using magnetic actuators. *Automatica* (2004), 1405–1414.

[75] Loyd, M. Crosswind Kite Power. *Journal of Energy 4*, 3 (July 1980), 106–111.

[76] Makani Power. Makani Power Website. `http://www.google.com/makani/`. Accessed: 19–01–2015.

[77] Manwell, J. F., McGowan, J. G. and Rogers, A. L. *Wind Energy Explained: Theory, Design and Application, Second Edition*. John Wiley & Sons, Ltd, Chichester, UK, 2009.

[78] Molinari, B. P. The Stabilizing Solution of the Discrete Algebraic Riccati Equation. *IEEE Transactions on Automatic Control 20* (1975), 396–399.

[79] Müller, M. A., Angeli, D., and Allgöwer, F. On necessity and robustness of dissipativity in economic model predictive control. *IEEE Transactions on Automatic Control 60*, 6 (2015), 1671–1676.

[80] Nocedal, J., and Wright, S. *Numerical Optimization*, 2 ed. Springer Series in Operations Research and Financial Engineering. Springer, 2006.

[81] Pacejka, H. B. *Tyre and Vehicle Dynamics*. Elsevier, 2006.

[82] Pamadi. *Performance, Stability, Dynamics, and Control of Airplanes*. American Institute of Aeronautics and Astronautics, Inc., 2003.

[83] Patrinos, P., and Bemporad, A. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *Automatic Control, IEEE Transactions on 59*, 1 (Jan 2014), 18–33.

[84] Payne, P., and McCutchen, C. Self-Erecting Windmill. United States Patent 3987987, Oct. 26 1976.

[85] Pesce, C. P. The Application of Lagrange Equations to Mechanical Systems With Mass Explicitly dependent on Position. *Journal of Applied Mechanics 70* (September 2003), 751–756.

[86] Podgaets, A., and Ockels, W. Flight control and stability of a multiple kites tethered system. In *Proceedings of Renewable Energy Conference* (2006).

[87] Pontryagin, L., Boltyanski, V., Gamkrelidze, R., and Miscenko, E. *The Mathematical Theory of Optimal Processes*. Wiley, Chichester, 1962.

[88] Quirynen, R., Gros, S., and Diehl, M. Efficient NMPC for nonlinear models with linear subsystems. In *Proceedings of the 52nd IEEE Conference on Decision and Control* (2013), pp. 5101–5106.

[89] Quirynen, R., Gros, S., and Diehl, M. Fast auto generated ACADO integrators and application to MHE with multi-rate measurements. In *Proceedings of the European Control Conference* (2013), pp. 3077–3082.

[90] Quirynen, R., Gros, S., and Diehl, M. Inexact Newton based Lifted Implicit Integrators for fast Nonlinear MPC. In *Proceedings of the 5th IFAC Conference on Nonlinear Model Predictive Control* (2015), pp. 32–38.

[91] Quirynen, R., Gros, S., and Diehl, M. Lifted implicit integrators for direct optimal control. In *Conference on Decision and Control* (2015).

[92] Quirynen, R., Houska, B., Vallerio, M., Telen, D., Logist, F., Van Impe, J., and Diehl, M. Symmetric Algorithmic Differentiation Based Exact Hessian SQP Method and Software for Economic MPC. In *Conference on Decision and Control* (2014), pp. 2752–2757.

[93] Quirynen, R., Vukov, M., and Diehl, M. Auto Generation of Implicit Integrators for Embedded NMPC with Microsecond Sampling Times. In *Proceedings of the 4th IFAC Nonlinear Model Predictive Control Conference* (2012), M. Lazar and F. Allgöwer, Eds., pp. 175–180.

[94] Quirynen, R., Vukov, M., Zanon, M., and Diehl, M. Autogenerating Microsecond Solvers for Nonlinear MPC: a Tutorial Using ACADO Integrators. *Optimal Control Applications and Methods 36* (2014), 685–704.

[95] Quirynen, R., Zanon, M., Kozma, A., and Diehl, M. A Compression Algorithm for Real-Time Distributed Nonlinear MPC. In *Proceedings of the European Control Conference* (2015).

[96] Ran, A. C. M., and Trentelman, H. L. Linear Quadratic Problems with Indefinite Cost for Discrete Time Systems. *SIAM Journal on Matrix Analysis and Applications 14* (1993), 776–797.

[97] Rao, C. *Moving Horizon Estimation of Constrained and Nonlinear Systems*. PhD thesis, University of Wisconsin–Madison, 2000.

[98] Rawlings, J., and Ji, L. Optimization-based State Estimation: Current Status and Some New Results. *Journal of Process Control 22*, 8 (2012), 1439 – 1444.

[99] Rawlings, J., and Mayne, D. *Model Predictive Control: Theory and Design*. Nob Hill, 2009.

[100] Rawlings, J. B., and Amrit, R. Optimizing Process Economic Performance using Model Predictive Control. In *Proceedings of NMPC 08 Pavia*. 2009, pp. 119–138.

[101] Rucco, A., Notarstefano, G., and Hauser, J. Computing minimum lap-time trajectories for a single-track car with load transfer. In *51st IEEE Conference on Decision and Control* (2012), pp. 6321–6326.

[102] Savorgnan, C., Kozma, A., Andersson, J., and Diehl, M. Adjoint-Based Distributed Multiple Shooting for Large-Scale Systems. In *18th IFAC World Congress* (2011), vol. 18, pp. 410–415.

[103] Savorgnan, C., Romani, C., Kozma, A., and Diehl, M. Multiple shooting for distributed systems with applications in hydro electricity production. *Journal of Process Control 21* (2011), 738–745.

[104] Schulz, V. Solving discretized optimization problems by partially reduced SQP methods. *Computing and Visualization in Science 1* (1998), 83–96.

[105] Schulz, V., Bock, H., and Steinbach, M. Exploiting invariants in the numerical solution of multipoint boundary value problems for DAEs. *SIAM Journal on Scientific Computing 19* (1998), 440–467.

[106] Shabana, A. A. *Dynamics of Multibody Systems*, 3 ed. Cambridge University Press, 2005.

[107] Shanno, D. Conditioning of quasi-Newton methods for function minimization. *Maths. Comp. 24* (1970), 647–656.

[108] Shuster, M. D. A Survey of Attitude Representations. *The Journal of the Astronautical Science 41*, 4 (1993), 439–517.

[109] Tsang, T., Himmelblau, D., and Edgar, T. Optimal control via collocation and non-linear programming. *International Journal on Control 21* (1975), 763–768.

[110] Verscheure, D., Demeulenaere, B., Swevers, J., Schutter, J. D., and Diehl, M. Time-Optimal Path Tracking for Robots: a Convex Optimization Approach. *IEEE Transactions on Automatic Control 54* (2009), 2318–2327.

[111] Verschueren, R. Design and Implementation of a Time-Optimal Controller for Model Race Cars. Master's thesis, KU Leuven, 2014.

[112] Verschueren, R., Bruyne, S. D., Zanon, M., Frasch, J. V., and Diehl, M. Towards Time-Optimal Race Car Driving using Nonlinear MPC in Real-Time. In *Proceedings of the 53rd Conference on Decision and Control (CDC)* (2014), pp. 2505–2510.

[113] Wächter, A., and Biegler, L. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming 106*, 1 (2006), 25–57.

[114] Wächter, A., and Biegler, L. IPOPT - an Interior Point OPTimizer. https://projects.coin-or.org/Ipopt, 2009.

[115] Willems, J. C. Least Squares Stationary Optimal Control and the Algebraic Riccati Equation. *IEEE Transactions on Automatic Control AC-16*, 6 (1971), 621–634.

[116] Williams, P. and Lansdorp, B. and Ockels, W.J. Modeling of Optimal Power Generation using Multiple Kites. In *AIAA Modelling and Simulation Technologies Conference and Exhibit, Honolulu, Hawaii, USA* (2008).

[117] Zanon, M., Frasch, J., and Diehl, M. Nonlinear Moving Horizon Estimation for Combined State and Friction Coefficient Estimation in Autonomous Driving. In *Proceedings of the European Control Conference* (2013), pp. 4130–4135.

[118] Zanon, M., Frasch, J. V., Vukov, M., Sager, S., and Diehl, M. Model Predictive Control of Autonomous Vehicles. In *Proceedings of the Workshop on Optimization and Optimal Control of Automotive Systems*. 2014, pp. 41–57.

[119] Zanon, M., Gros, S., Andersson, J., and Diehl, M. Airborne Wind Energy Based on Dual Airfoils. *IEEE Transactions on Control Systems Technology 21* (July 2013), 1215–1222.

[120] Zanon, M., Gros, S., and Diehl, M. A Lyapunov Function for Periodic Economic Optimizing Model Predictive Control. In *Proceedings of the 52nd Conference on Decision and Control (CDC)* (2013), pp. 5107–5112.

[121] Zanon, M., Gros, S., and Diehl, M. Rotational Start-up of Tethered Airplanes Based on Nonlinear MPC and MHE. In *Proceedings of the European Control Conference* (2013), pp. 1023–1028.

[122] Zanon, M., Gros, S., and Diehl, M. Indefinite Linear MPC and Approximated Economic MPC for Nonlinear Systems. *Journal of Process Control 24* (2014), 1273–1281.

[123] Zanon, M., Gros, S., and Diehl, M. A Tracking MPC Formulation that is Locally Equivalent to Economic MPC. (to be submitted), 2015.

[124] Zanon, M., Gros, S., Meyers, J., and Diehl, M. Airborne Wind Energy: Airfoil-Airmass Interaction. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control* (2014), pp. 5814–5819.

[125] Zanon, M., Grüne, L., and Diehl, M. Periodic optimal control, dissipativity and MPC. *IEEE Transactions on Automatic Control* (2015). (submitted).

[126] Zanon, M., Horn, G., Gros, S., and Diehl, M. Control of Dual-Airfoil Airborne Wind Energy Systems Based on Nonlinear MPC and MHE. In *European Control Conference* (2014), pp. 1801–1806.

Curriculum Vitae

Mario Zanon

born November 08, 1985 in Rovereto, Italy.

2011 – 2015	PhD studies in Electrical Engineering, KU Leuven, Belgium
2014 – 2015	Research Assistant (E13) with Prof. Moritz Diehl, Freiburg University, Germany
2014	Visiting Researcher with Ass. Prof. Sébastien Gros, Chalmers University, Sweden
2013 – 2014	Visiting Researcher with Prof. Lars Grüne, Bayreuth University, Germany
2011 – 2013	Marie-Curie ESR in ITN SADCO, KU Leuven, Belgium
2010	Researcher in control engineering with Prof. Alberto Bemporad, University of Trento, Italy
2008 – 2010	Master of Science in Mechatronics, University of Trento, Italy
2006 – 2008	Master of Science in General Engineering, Ecole Centrale Paris, France
2004 – 2008	Bachelor of Science in Mechatronics, University of Trento, Italy
1999 – 2004	Diploma di Maturità (university entrance qualification), Liceo Scientifico G. Galilei, Trento, Italy

List of Publications

Journal Articles (submitted / in planning)

1. M. Zanon, S. Gros and M. Diehl. *A Tracking MPC Formulation that is Locally Equivalent to Economic MPC*. (To be submitted)
2. S. Gros, M. Zanon and M. Diehl. *Penalty Functions for Handling Large Deviation of Quadrature States in NMPC*. (To be submitted)
3. S. Gros, M. Zanon, R. Quirynen, A. Bemporad and M. Diehl. *From Linear to Nonlinear MPC: bridging the gap via the Real-Time Iteration*. (To be submitted)
4. M. Zanon, L. Grüne and M. Diehl. *Periodic optimal control, dissipativity and MPC*. IEEE Transactions on Automatic Control, 201x (In revision)
5. K. Worthmann, M. W. Mehrez, M. Zanon, G. K. I. Mann, R. G. Gosine, M. Diehl. *Model Predictive Control of Nonholonomic Mobile Robots without Stabilizing Constraints and Costs*, IEEE Transactions on Control Systems Technology, 201x (In revision)

Journal Articles (published)

1. M. Zanon, S. Gros and M. Diehl. *Indefinite Linear MPC and Approximated Economic MPC for Nonlinear Systems*. Journal of Process Control, 2014
2. R. Quirynen, M. Vukov, M. Zanon and M. Diehl. *Autogenerating Microsecond Solvers for Nonlinear MPC: a Tutorial Using ACADO Integrators. Optimal Control Applications and Methods*, 2014
3. M. Zanon, S. Gros, J. Andersson and M. Diehl. *Airborne Wind Energy Based on Dual Airfoils*. IEEE Transactions on Control Systems Technology, 2013

Conference Proceedings

1. K. Worthmann, M. W. Mehrez, M. Zanon, G. K. I. Mann, R. G. Gosine, M. Diehl. *Regulation of Differential Drive Robots using Continuous Time MPC without Stabilizing Constraints or Costs.* Proceedings of the Nonlinear Model Predictive Control Conference, 2015

2. A. Mohammadi, M. Diehl and M. Zanon. *Estimation of Uncertain ARX models with Ellipsoidal Parameter Variability.* Proceedings of the European Control Conference, 2015

3. R. Quirynen, M. Zanon, A. Kozma and M. Diehl. *A Compression Algorithm for Real-Time Distributed Nonlinear MPC.* Proceedings of the European Control Conference, 2015

4. M. Zanon, S. Gros and M. Diehl. *Local Properties of Economic NMPC, Dissipativity and Dynamic Programming.* Proceedings of the Conference on Decision and Control, 2014

5. R. Verschueren, S. De Bruyne, M. Zanon, J. Frasch and M. Diehl. *Towards Time-Optimal Race Car Driving using Nonlinear MPC in Real-Time.* Proceedings of the Conference on Decision and Control, 2014

6. M. Zanon, G. Horn, S. Gros and M. Diehl. *Control of Dual-Airfoil Airborne Wind Energy Systems Based on Nonlinear MPC and MHE.* Proceedings of the European Control Conference, 2014

7. M. Zanon, S. Gros, J. Meyers and M. Diehl. *Airborne Wind Energy: Airfoil-Airmass Interaction.* Proceedings of the 19th World Congress of the International Federation of Automatic Control, 2014

8. M. Zanon, S. Gros and M. Diehl. *A Lyapunov Function for Periodic Economic Optimizing Model Predictive Control.* Proceedings of the 52nd Conference on Decision and Control (CDC), 2013

9. M. Zanon, S. Gros and M. Diehl. *Rotational Start-up of Tethered Airplanes Based on Nonlinear MPC and MHE.* Proceedings of the European Control Conference, 2013

10. J. V. Frasch, A. Gray, M. Zanon, H. J. Ferreau, S. Sager, F. Borrelli and M. Diehl. *An Auto-generated Nonlinear MPC Algorithm for Real-Time Obstacle Avoidance of Ground Vehicles.* Proceedings of the European Control Conference, 2013

11. M. Zanon, J. V. Frasch and M. Diehl. *Nonlinear Moving Horizon Estimation for Combined State and Friction Coefficient Estimation in*

Autonomous Driving. Proceedings of the European Control Conference, 2013

12. S. Gros, M. Zanon, and M. Diehl. *Control of Airborne Wind Energy Systems Based on Nonlinear Model Predictive Control & Moving Horizon Estimation*. European Control Conference, 2013

13. S. Gros, M. Zanon, and M. Diehl. *A Relaxation Strategy for the Optimization of Airborne Wind Energy Systems*. European Control Conference, 2013

14. S. Gros, M. Zanon, M. Vukov and M. Diehl. *Nonlinear MPC and MHE for Mechanical Multi-Body Systems with Application to Fast Tethered Airplanes*. Proceedings of the Nonlinear Model Predictive Control Conference, 2012

15. S. Gros, M. Zanon and M. Diehl. *Orbit Control for a Power Generating Airfoil based on Nonlinear MPC*. American Control Conference, 2012

Book Chapters

1. M. Zanon, J. V. Frasch, M. Vukov, S. Sager and M. Diehl. *Model Predictive Control of Autonomous Vehicles*. Proceedings of the workshop on Optimization and Optimal Control of Automotive Systems, Springer, 2013

2. M. Zanon, S. Gros and M. Diehl. *Control of Rigid-Airfoil Airborne Wind Energy Systems*. Airborne Wind Energy, Springer, 2013

3. K. Geebelen, M. Vukov, A. Wagner, H. Ahmad, M. Zanon, S. Gros, D. Vandepitte, J. Swevers, M. Diehl. *An experimental test setup for advanced estimation and control of an airborne wind energy system*. Airborne Wind Energy, Springer, 2013

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF ELECTRICAL ENGINEERING
ESAT - STADIUS
Kasteelpark Arenberg 10 box 24462
B-3001 Heverlee

