Feedback Optimizing Model Predictive Control with Power Systems Applications

Amba Asuk

Department of Automatic Control and Systems Engineering
University of Sheffield

Thesis submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

September 2023

*Dedicated to the loving memory of Miss **Ruth Ogar**.*

Acknowledgements

I am grateful to the Federal Government of Nigeria for providing the financial sponsorship for this research through the Petroleum Technology Development Fund (PTDF). I would also like to thank my supervisor Dr. Paul Trodden for his availability, readiness to help, guidance and understanding. He gave me the creative freedom to pursue my interest and showed optimism towards my often unpolished ideas.
Finally, I am grateful to my family and loved ones for their love and support. Special thanks goes to my mum Mrs. Beatrice Asuk for her unconditional love, to my (late) father Barr. Richard Asuk (LL.B) for his guidance, provision and love, and to my wife Mrs. Gift Amba for her love and support.

Abstract

Feedback optimization (FO) is a control paradigm that is gaining popularity for the optimal steady-state operation of complex systems through the use of optimization algorithms in closed-loop control. FO controllers are capable of addressing control objectives beyond simply regulating setpoints and are often used to track solution trajectories of time-varying optimization problems that are not known in advance. Previous research in this area has typically utilized simplified control dynamics, ignored model uncertainties, and has not adequately addressed constraints or transient performance. Additionally, traditional optimal control approaches often require prior knowledge of the desired equilibrium point. In this thesis, we approach the FO problem from an optimal control and model predictive control (MPC) perspective. Specifically, we propose MPC schemes that can steer the steady-state of a linear dynamical system to the solution of a defined static optimization problem without numerically solving the problem or relying on external setpoints. We accomplish this by formulating the cost functional in MPC to embed an optimization algorithm for the steady-state optimization problem, which is driven to convergence by the implicit feedback inherent in MPC. This allows for the system to be driven to an optimal equilibrium point following a disturbance, without explicit knowledge of the disturbance or setpoints, while also achieving improved transient performance. Compared to direct online economic optimization (e.g., economic MPC), our approach offers improved computational efficiency, and robustness to model uncertainty and unmeasured disturbances. Additionally, the algorithms we develop are only slightly more complex than conventional linear tracking MPC, so theoretical guarantees of stability and performance can be readily derived from standard tracking MPC results without additional assumptions. To demonstrate the effectiveness of the proposed MPC schemes, we present several numerical examples and an application to the challenging problem of real-time economic dispatch in load-frequency control of power system networks. The results obtained show that our proposed MPC schemes are indeed feedback optimizing, with good robustness properties and optimal transient performance.

Table of contents

List of figures

List of tables

Nomenclature

0.1 Notation

Sets

$\emptyset$	The empty set
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Set of positive real numbers
$\mathbb{I}$	Set of positive integers
$\mathbb{I}_{[1,n]}$	Set of positive integers from 1 to n
$\mathbb{R}^n$	Set of n-dimensional vectors with real entries
$x \in \mathbb{X}$	x is an element of $\mathbb{X}$
$\mathbb{X} \subset \mathbb{Y}$	$\mathbb{X}$ is a subset of $\mathbb{Y}$
$\lvert\mathbb{X}\rvert$	Cardinality of the set $\mathbb{X}$

Matrices

I_n	n by n identity matrix
$I_{n\times m}$	n by m diagonal matrix containing ones in the diagonal entries
$\mathbf{0}_{p\times m}$	p by m matrix containing all zeros
$\mathbb{1}_{p\times m}$	p by m matrix containing all ones
$\text{blkdiag}(A_1,\ldots,A_n)$	block-diagonal matrix with $A_1,\ldots,A_n$ in the diagonal entries
$v=(v_i)_{i\in\{1,\ldots,n\}}$	column vector with the entries $\{v_1,\ldots,v_n\}$
$A^\top$	transpose of matrix A
A^{-1}	inverse of matrix A
$A \succeq 0$	A is positive semidefinite, i.e $x^\top Ax \geq 0,\ \forall x \in \mathbb{R}^n$
$A \preceq 0$	A is negative semidefinite, i.e $x^\top Ax \leq 0,\ \forall x \in \mathbb{R}^n$
$A \succ 0$	A is positive definite, i.e $x^\top Ax > 0,\ \forall x \in \mathbb{R}^n$
$A \prec 0$	A is negative definite, i.e $x^\top Ax < 0,\ \forall x \in \mathbb{R}^n$
$\|x\|_2$	Euclidean norm of x, where $\|x\|_2 \coloneqq \sqrt{x^\top x}$
$\|x\|_P$	Weighted Euclidean norm of x, where $\|x\|_P \coloneqq \sqrt{x^\top Px}$

Set operators

$\mathbb{X}\oplus\mathbb{Y}$	Minkowski sum, $\mathbb{X}\oplus\mathbb{Y} \coloneqq \{x+y \mid x\in\mathbb{X}, y\in\mathbb{Y}\}$
$\mathbb{X}\ominus\mathbb{Y}$	Pontryagin difference, $\mathbb{X}\ominus\mathbb{Y} \coloneqq \{z \mid z+y\in\mathbb{X}, \forall y\in\mathbb{Y}\}$
$\mathbb{X}\times\mathbb{Y}$	Cartesian product, $\mathbb{X}\times\mathbb{Y} \coloneqq \{(x,y) \mid x\in\mathbb{X}, y\in\mathbb{Y}\}$
$\bigoplus_j^N \mathbb{X}_j$	Sum of sets, $\bigoplus_j^N \mathbb{X}_j = \mathbb{X}_1 \oplus \ldots \mathbb{X}_j \ldots \oplus \mathbb{X}_N$
$\Pi_j^N \mathbb{X}_j$	Product of sets, $\Pi_j^N \mathbb{X}_j = \mathbb{X}_1 \times \ldots \mathbb{X}_j \ldots \times \mathbb{X}_N$
$\pi_j(\mathbb{X})$	Projection of set $\mathbb{X}$ onto the space $\mathbb{X}_j$, $\pi_j(\mathbb{X}) = x_j \in \mathbb{X}_j$ where $\mathbb{X} = \Pi_j^N \mathbb{X}_j$
$\mathbb{X}\cup\mathbb{Y}$	Set union, $\mathbb{X}\cup\mathbb{Y} \coloneqq \{x \mid x\in\mathbb{X} \text{ or } x\in\mathbb{Y}\}$
$\mathbb{X}\cap\mathbb{Y}$	Set intersection, $\mathbb{X}\cap\mathbb{Y} \coloneqq \{x \mid x\in\mathbb{X} \text{ and } x\in\mathbb{Y}\}$
$\mathbb{Y}\backslash\mathbb{X}$	Relative complement of X in Y, $\mathbb{Y}\backslash\mathbb{X} \coloneqq \{x \mid x\in\mathbb{Y} \text{ and } x\notin\mathbb{X}\}$
$\mathbb{X}'$	Complement of X, $\mathbb{X}' \coloneqq \mathscr{U}\backslash\mathbb{X}$, where $\mathscr{U}$ = Universal set

0.2 Acronyms/Abbreviations

MPC	Model Predictive Control
PID	Proportional Integral Derivative
FOMPC	Feedback Optimizing Model Predictive Control
GTMPC	Generalized Tracking Model Predictive Control
LQC	Linear Quadratic Control
LQ	Linear Quadratic
ACE	Area Control Error
LFC	Load Frequency Control
FOMPLFC	Feedback Optimizing Model Predictive Load Frequency Control
CA	Control Area
DAPI	Distributed Averaging Proportional Integral
ED	Economic Dispatch
OCP	Optimal Control Problem
RTO	Real-time Optimization

Chapter 1

Introduction

Automatic feedback control is ubiquitous in today's society. It is an enabling technology behind many critical infrastructure systems such as electric power networks, chemical processing plants, building automation systems, and communication networks, to name but a few. For most systems, reliable and efficient operation is achieved via feedback control and in this thesis, we develop model predictive control strategies that guarantee stable and economically optimal steady-state operation.

In this chapter, we describe the motivation and goals of this thesis, and also set the stage for the research presented in subsequent chapters. Firstly, we introduce the concepts of feedback optimizing control, model predictive control and optimal load frequency control. We then present a motivation for the research. The research aims and objectives are then stated. Finally, the thesis outlines, contributions and the list of publications are given.

1.1 Feedback Optimizing Control

To remain competitive, practical systems must be operated cost-effectively while being responsive to changes that affect the system performance. Such cost-optimal operation is for most systems associated with a steady-state (economic) optimization problem [104]. It is therefore essential to operate at such optimal steady-states despite uncertainties and time-varying disturbances.

Conventionally, optimal steady-state operation is achieved via a hierarchical control approach (Fig. 1.1). Here, a system's efficiency (e.g., economic optimality) and reliability (e.g., stability, robustness) goals are handled separately at different timescales. At a slow timescale, a steady-state optimization problem is numerically solved by a real-time optimizer based on the systems operational costs, constraints, steady-state model, and disturbance estimates to generate optimal setpoints/references. These setpoints are then tracked in real-time by a dynamic controller acting over faster timescales to reject the influence of disturbances. A typical example is the management and control of power system networks [221]. Here, power balance/frequency stability is maintained using load-frequency control (LFC) at the faster timescales (i.e., at the (sub)-second level). Economic dispatch then acts a much slower timescale (i.e., every 15 minutes) to generate optimal setpoints tracked by LFC. The economic dispatch problems are solved based on forecasts of loads and renewable generation, a grid model, and security constraints.

Although practically convenient, hierarchical approaches can have significant limitations for systems with uncertainty and fast-changing disturbances. Some of these limitations include but are not limited to the following:

1. **Lack of robustness**
   More often than not, real-world systems lack precise models and are plagued by unknown disturbances. Hence, the numerical solution to the steady-state optimization problem using such imprecise data may produce setpoints that are suboptimal or, worse infeasible for the actual physical system [150]. Although robust [21] and stochastic [31] optimization techniques exist and could be used to alleviate the impact of these uncertainties, they often result in overly conservative solutions and may be computationally expensive for most real-time applications.

2. **The need for timescale separation**
   In hierarchical approaches, a timescale separation between the real-time optimizer and dynamic controller is necessary to guarantee a stable and near-optimal system operation [82, 88]. To ensure timescale separation, the real-time optimization setpoints are only updated after the system settles at a new steady-state following a disturbance [118]. Also, the dynamic controller must respond faster to disturbances relative to the frequency of setpoint updates from the real-time optimization [88]. However, maintaining a timescale separation between the real-time optimizer and dynamic controller becomes practically impossible for systems with dynamics that span multiple timescales. Such systems are usually characterised by high model uncertainty, and frequent steady-state transitions due to fast-changing disturbances.
   For instance, future power networks will have a high penetration of distributed/renewable

generation, which are by nature highly variable and uncertain. Therefore, large and fast disturbances will be introduced into the power network, inducing dynamics across the real-time optimization and dynamic control layers [37, 103]. The conventional assumptions of timescale separation will, as a result, not be valid, leading to a breakdown of hierarchical control [56] with the possible loss of efficiency and closed-loop stability [37, 104].

3. **Higher computational complexity**
   Most real-time optimization algorithms are based on an explicit numerical solution of a steady-state optimization problem to generate the setpoints. For systems with frequent steady-state transitions and high uncertainty, it becomes necessary to solve the steady-state optimization problem faster and regularly, placing a higher computational burden on the real-time optimization layer and creating some practical challenges. Firstly, timely convergence of the numerical optimization algorithms to the optimal solution may not be achievable under limited computational resources, leading to suboptimal, or at worse, infeasible setpoints [117]. Also, a certain level of expertise and competence may be required to implement and maintain numerical optimization solutions in real-time making it challenging to cost-effectively maintain these applications in the real world. As discussed in [117, 77], the performance degradation due to lack of maintenance and support often leads to most real-time optimization applications being turned off by operators who are often more confident working with control applications instead. For this reason, most traditional process industries ignore real-time optimization and instead rely on simple feedback control tools to optimize their operations [78].

4. **Need for external setpoints and references**

To mitigate the limitations of hierarchical control, it is imperative to design feedback controllers capable of implicitly tracking the optimal solution of a steady-state optimization problem in closed-loop, without external setpoints/references using only input/output measurements and minimal knowledge of the steady-state model [101, 87, 129]. Feedback optimizing or self-optimizing control is the framework for achieving this autonomous steady-state optimization.
As an alternative or complement to hierarchical control, self-optimizing [199] or feedback optimizing [117] control (Fig. 1.2) is a promising approach to regulate dynamical

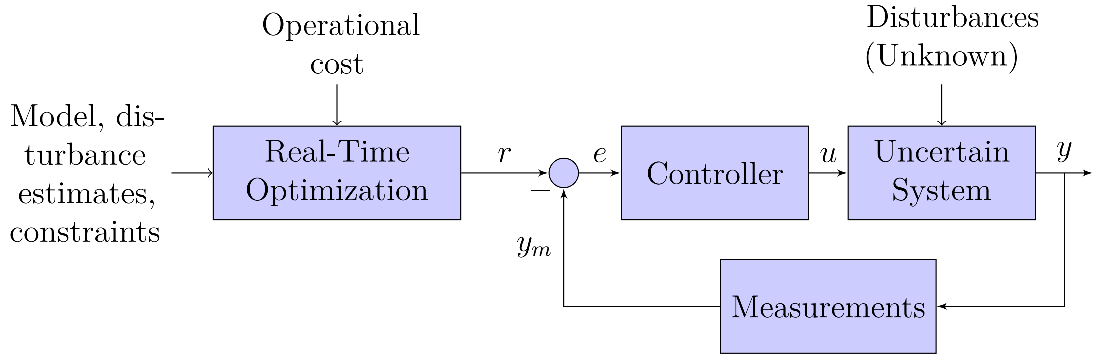

Fig. 1.1 Hierarchical control

systems to economically optimal operating points, which are the solution of a defined steady-state optimization problem. In feedback optimizing control, steady-state optimality is achieved by directly translating the steady-state (economic) performance objectives into robust dynamic control goals. This is nicely captured by the following statement adapted from [162]:

> in attempting to synthesize a feedback optimizing control structure, our main objective is to translate the economic into process control objectives. In other words, we want to find a function c of the process variables which when held constant, leads automatically to the optimal adjustments of the manipulated variables, and with it, the optimal operating conditions

The main idea behind feedback optimizing control is therefore to find a self-optimizing variable or function, which implicitly achieves optimal steady-state performance for a defined static optimization problem when driven by a dynamic controller to a constant value. The optimality conditions associated with a steady-state optimization problem are a good choice for the self-optimizing control function. Because these conditions take a constant value of zero at optimality, they can be regulated to the origin using a dynamic feedback controller to implicitly achieve steady-state optimality. Indeed, the steady-state gradient from cost to input, an early self-optimizing control variable proposed in [198] is also an optimality condition for a first-order optimization problem.

A critical aspect of feedback optimizing control is that, rather than relying on a complete model of the physical system, the controller uses measurements of the system outputs to drive the system towards efficient operating points autonomously. As a result, feedback optimizing control has inherent robustness against unmeasured disturbances and model uncertainties compared to conventional feedforward-based numerical optimization, which relies on accurate knowledge or estimates of the system uncertainty.

For several decades, the synthesis of feedback optimizing control structures has been of interest to researchers mainly due to its ability to address some of the limitations of conventional hierarchical control with applications historically targeted towards process control [162] and communication systems [110, 140]. With the current efforts towards the integration of more renewable energy sources in power system networks, there have been renewed interest in the development and application of novel feedback optimizing control algorithms in power systems [160, 87]. In particular, feedback optimizing control algorithms for real-time economic dispatch has attracted a lot of interest in power system as efficient and fast-acting control capabilities are required to handle the increased volatility and randomness of future power system demand.

Recently, there have been several feedback optimizing control proposal based on the idea of directly implementing optimization algorithms as feedback controllers. This idea which originated from the work done by economists in [107] has formed the basis for most modern feedback optimizing control strategies. This is because this approach opens up new opportunities for the optimization and control of dynamic systems. In particular, it now becomes possible with this approach to study and analyse the disturbance rejection and robustness properties of optimization systems and also design feedback controllers with self optimizing properties using just tools from control theory. Also, it eliminates the need for external setpoints or references. Instead, the system is autonomously driven to steady-states that are optimal for a defined static optimization problem. This feature is particularly powerful as it has the potential to completely address the problem of infeasible or unreachable setpoints associated with setpoint based control schemes.

Because feedback optimizing control translates an optimization problem into a dynamic control problem, it opens up opportunities for applying the wealth of rigorous design tools from control theory. This allows the benefits of feedback control design such as dynamic optimality and robust stability to be applied to optimization problems. Indeed, the goal of this thesis is to apply the tools from model predictive control design in the feedback optimizing control of uncertain dynamic systems. In the next section, we introduce the concept of model predictive control.

[1] Model may not always be necessary or may be very simplified

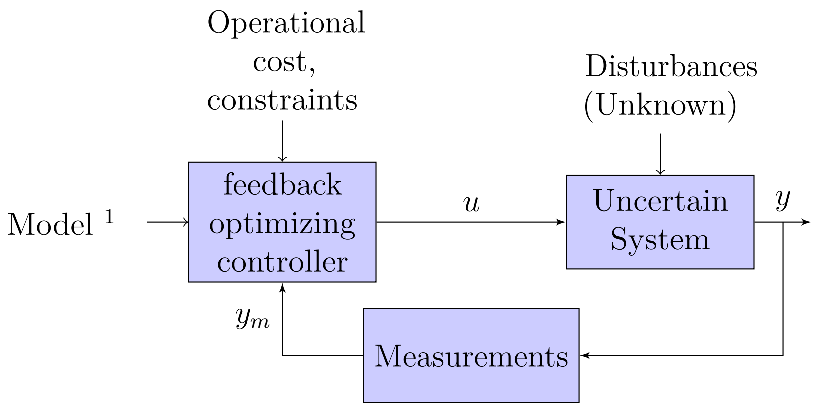

Fig. 1.2 Feedback optimizing control.

1.2 Model Predictive Control

In this thesis, we develop feedback optimizing control algorithms using a model predictive control (MPC) framework. MPC is one of the few advanced control methods that has had significant impact in control engineering practice. This has been due to the unique ability of MPC to systematically deal with constraints and optimize performance. MPC was first developed in the late seventies for the control of complex, constrained multi-variable systems in the process control industry [177]. The term Model Predictive Control is not a specific control strategy but rather a collection of control methods which make explicit use of a model of a system to obtain the control actions by minimizing a performance objective in real-time. The basic idea behind MPC is to predict the behaviour of the plant to be controlled N steps into the future using the plant model and the currently measured state. Then using the $N-$horizon system predictions, the optimal control inputs required to steer the predicted plant trajectories towards a desired reference is determined by solving online, an optimal control problem (OCP). The control action on the plant is then derived from the first element of the optimal control sequence. At the next sampling instant, the current state is measured and the entire process repeated to compute the next control action. This way of implementing the optimal control input is called a receding horizon implementation and it generates an implicit feedback in the closed-loop system. Thanks to the fact that an explicit dynamic model is used for prediction and a performance objective is minimized, MPC gives good control accuracy. Consequently, MPC unlike other control algorithms such as PID can be successfully applied to plants which are difficult to control. The core advantages of MPC over other control methods are [152]: (i) its explicit and systematic constraint handling capabilities (ii) its intuitive appeal

(iii) its ability to optimize arbitrary performance criteria, and (iv) its ability to yield performance benefits of both feed-forward and feedback control.
Over the years, tremendous progress has been made in the theory and application of MPC. Successful applications have been reported in many industrial systems such as petrochemical, power/energy and automotive plants etc.,[180, 177]. The theory of MPC has also been substantially developed to address a wide range of control problems such as: setpoint tracking, economic optimization, distributed control and robust/stochastic control.

Traditionally, economic objectives are considered in MPC via a two layer hierarchical control approach where the economically optimal setpoints are computed by a RTO and sent as targets/references to be tracked in real time by MPC [65, 4]. However, inconsistency between the RTO model and the dynamic MPC model can result in steady-state offsets. Also, in the presence of plant-model mismatch and unknown disturbances, the RTO setpoints might not correspond to feasible operating points for the plant. Also, uncertainty in the RTO model may results in suboptimal setpoints [199]. To address these challenges, an offset free MPC design has been developed. Combined with modifier-adaptation methods, offset-free MPC can be made to track the true optimal setpoints even for systems with plant-model mismatch and unknown disturbances. This however adds substantial complexity to the MPC algorithm. As an alternative to offset-free MPC, the steady-state economic cost can be directly minimized in an economic MPC formulation. Despite the recent progress in design and analysis of economic MPC schemes, the handling of uncertainties and the asymptotic tracking of the true optimal steady-state setpoint under plant-model mismatch still poses open problems [17]. In this thesis, we address some of these challenges by designing feedback optimizing offset-free MPC schemes to autonomously track the optimal RTO setpoints for a linear dynamical system.

1.3 Optimal Load-Frequency Control

At the heart of power system operation is the load-frequency control (LFC) problem, which involves maintaining the frequency of a power system close to its nominal value while ensuring economically optimal steady-state operation, despite fluctuating loads and generation. A key objective of LFC is the optimal allocation of generation and network resources to minimize the cost of power balancing. Conventionally, tertiary control or economic dispatch achieves this objective by solving steady-state optimization problems at a slower timescale to obtain set points, tracked in real-time by fast-acting primary (droop) and secondary (Automatic Generation Control (AGC)) frequency controllers. The economic efficiency of this approach depends on the predictability of demand, the absence of multi-time scale dynamics, and the efficient coordination between economic dispatch and primary and secondary frequency controllers. However, due to increased integration of renewable energy generation, more responsive demand, and the deregulation of power systems, the net demand (uncontrollable load minus uncontrollable generation) is becoming less predictable. It fluctuates faster and larger across both the slow and fast time scales. Consequently, economically efficient LFC operations may be challenging to achieve via the conventional multi-layered control approach [89]. Because the setpoints obtained from a feedforward-based economic dispatch operation may become sub-optimal after a disturbance, the resulting load-frequency control operations may become inefficient.

The optimal load-frequency control problem [160, 55] is concerned with the design of load frequency control algorithms capable of achieving economic dispatch autonomously (without external setpoints) while also guaranteeing frequency stability. When the demand fluctuations are unpredictable, LFC must be designed to be self-optimizing rather than rely on inefficient economic dispatch operations. Unlike traditional LFC, Optimal Load-Frequency Control (OLFC) is not prediction-based. Instead, it relies on input-output measurements to drive the power system to economically optimal operating points while guaranteeing frequency stability despite uncertainty and variability in the power system. As a result, optimal load-frequency control algorithms achieve a reliable and efficient integration of renewable generation and flexible demand into the power system. The optimal load frequency control (OLFC) problem is a much-researched topic in the recent literature on load frequency control, and we provide a detailed review of this problem in subsequent chapters. One of the goals of this research is to apply the algorithms developed for feedback optimizing control to the OLFC problem. The algorithms developed in this thesis have the potential to improve the dynamic performance of OLFC for power systems with a rapid variation of their optimal steady-state operating point due to the high integration of renewable generation.

1.4 Research Motivation

As discussed in the previous sections, driving a dynamic system to economically optimal steady states is conventionally realized via a two-layer architecture comprising an RTO and dynamic control layer. Although this approach may work well for systems with infrequent setpoint changes, it may fail to guarantee or achieve closed-loop stability and convergence to the optimal economic setpoints for systems with rapidly changing equilibria caused by unknown, time-varying disturbances. A practical example occurs in the operation of power systems under high penetration of intermittent energy sources such as wind and solar. Here, rapid fluctuations in the net demand resulting from the intermittency of wind and solar results in economically sub-optimal power system operation.

The paradigm of feedback optimizing control that integrates real-time optimization and dynamic control has recently re-emerged to overcome the limitations of hierarchical control. Several methods for achieving feedback optimizing control have been proposed, differing according to the assumptions made on the problem and the features and limitations of the resulting scheme. Most recently proposed feedback optimizing control strategies rely on the direct implementation of the optimization algorithms for solving an RTO problem as feedback controllers [87, 160, 55]. This approach reduces the timescale separation between RTO and dynamic control [88] and also possesses inherent robustness to uncertainty in the plant model due to its feedback nature [173, 46]. However, most feedback optimizing control schemes proposed in the literature ignore the system's dynamic performance, assuming the system to be pre-stabilized [128, 129]. Therefore dynamic performance is left as an offline design exercise [87, 55]. The problem with this approach is that it becomes challenging to guarantee performance and constraint satisfaction when the dynamic system changes in unexpected ways. As a result, the design space of these feedback optimizing control schemes are unnecessarily restricted, thereby making it difficult to consider much wider control objectives such as dynamic optimality, constraint feasibility and robust control. Also, results on the robust stability of these controllers have been difficult to establish even though experiments have been conducted to verify their inherent robustness to uncertainty [169, 46].

To address these limitations, in this thesis the framework of model predictive control is utilized to synthesize novel feedback optimizing control schemes. Due to a control law obtained via online optimization, MPC optimizes the dynamic performance of the closed-loop system in real-time while also ensuring systematic constraint satisfaction in the transient phase. It is also possible to achieve optimal steady-state operation under changing disturbances with offset-free MPC schemes. Modifier adaptation approaches have recently been combined with offset-free MPC to track the optimal solution to an

RTO problem under plant-model mismatch. However, modifier adaptation techniques often result in optimal control problems that are significantly more complicated to design, analyze, and implement, making them less appealing in practice. To address the shortcomings of modifier adaptation-based offset-free MPC while also tackling the limitations of conventional feedback optimizing control design, we present novel MPC schemes that integrate the design philosophy of conventional feedback optimizing control with offset-free MPC. We propose simple and computationally efficient control algorithms that are guaranteed to converge to the optimal solution of an RTO problem with optimal dynamic performance and guaranteed constraint satisfaction without needing an external setpoint or explicit online solution of the RTO problem. The control schemes we develop in this thesis are only moderately more complex than conventional MPC, computationally efficient, and submit to standard theoretical analysis already established in the MPC literature. Also, the design framework presented in this thesis, unlike conventional feedback optimizing control, can be easily adapted to design and analyze robust feedback optimizing control solutions.

1.5 Research Aim and Objectives

The aim of this thesis is to develop computationally efficient feedback optimizing control algorithms based on an MPC framework to regulate uncertain linear systems to efficient steady-state equilibria for a defined steady-state optimization problem without explicitly solving the problem. The control algorithms should utilize information about the steady-state optimization problem and the dynamic model of the plant to generate in a computationally efficient and straightforward manner model predictive control laws that are self-optimizing, dynamically optimal, and recursively feasible. Also, our proposed solutions should have quantifiable robustness to plant-model mismatch and unmeasured disturbances.

The specific objectives of the thesis are:

1. Formulate the feedback optimizing MPC (FOMPC) problem as a generalization of tracking MPC, where the steady-state tracking error is not available apriori but instead derived from input/output measurements and information about the steady-state optimization problem.

2. Analyze the conditions for which a computable and stabilizable solution exists to the FOMPC problem by leveraging controller existence results available for tracking MPC.

3. Develop and implement elegant and computationally efficient solutions to the nominal FOMPC problem, in which the plant is a disturbed linear system with known system coefficients and the steady-state optimization problem is a quadratic program.

4. Develop and implement computationally efficient and constructive solutions to the robust FOMPC problem for uncertain linear systems with polytopic model uncertainty and a quadratic steady-state optimization problem.

5. Develop and analyze scalable and computationally efficient solutions to the distributed FOMPC problem for decomposable large-scale systems with a separable steady-state quadratic cost.

6. Analyze and characterize the convergence, recursive feasibility, and robust stability of the developed FOMPC solutions under both nominal and uncertain conditions.

7. Apply the proposed FOMPC solutions to the optimal load-frequency control problem for real-time economic dispatch in future power systems with high penetration of variable generation.

1.6 Thesis Outline and Contributions

This thesis is comprised of 10 chapters. A summary of each chapter and the associated contributions is presented next.

1.6.1 Chapter 2: Background and literature review

This chapter presents a detailed review of the current literature on feedback optimizing control, economics optimizing MPC, and the optimal load-frequency control problem. The review revealed the limitation of conventional feedback optimizing control and economics optimizing MPC and sheds light on new opportunities to develop control schemes that integrate both approaches into a single framework that builds on the strengths of both approaches to address their combined limitations.

1.6.2 Chapter 3: The feedback optimizing model predictive control problem

In this chapter, the feedback optimizing control problem is formulated in the framework of model predictive control. Here, the steady-state optimization problem is defined for uncertain linear systems. Then a formal statement of the feedback optimizing MPC (FOMPC) problem is stated. The Karush-Kuhn-Tucker (KKT) conditions for optimality of the steady-state optimization problem are then stated. To construct an optimality error that is a function of the input and output measurements rather than a pre-computed setpoint, we express the KKT conditions in subspace form by eliminating the equality constraint. Using this result, the FOMPC problem is then expressed as a generalized tracking MPC problem (GTMPC) and from this we derive conditions for solvability of the FOMPC problem.

1.6.3 Chapter 4: Deterministic feedback optimizing linear quadratic control

In this chapter, we take the first step towards solving the FOMPC problem. To simplify the problem, we assume the inequality constraints are inactive at all times (i.e. during the transient phase and in steady-state). With this assumption, the FOMPC problem reduces to a feedback optimizing linear quadratic control (FOLQC) problem. Using the results from Chapter 3, the FOLQC problem is then translated to an equivalent tracking linear quadratic control problem with a tracking error that is derived from the input/output measurements. To solve the FOLQC problem, we formulate the resulting tracking linear quadratic control problem in velocity form, and using the techniques of dynamic programming and linear matrix inequalities, we derive explicit FOLQC laws for the case of a steady-state quadratic program. We then analyze the robust stability of the obtained FOLQC laws using standard results on the robust stability of the linear quadratic regulator. Simulation results demonstrating the performance of FOLQC are also presented.

1.6.4 Chapter 5: Robust feedback optimizing linear quadratic control

This chapter extends the results obtained in Chapter 4 to the case of uncertain linear systems. We consider linear dynamical systems with coefficients that are unknown and may be time-varying but take values from a known compact polytopic set. Using a linear matrix inequality approach, we synthesize robust FOLQC laws by formulating and solving semi-definite programs for a velocity form of the system dynamics. Numerical simulation of the robust FOLQC laws are also provided to illustrate the effectiveness of the approach.

1.6.5 Chapter 6: Nominal feedback optimizing model predictive control

In this chapter, we develop centralized MPC algorithms to solve the FOMPC problem for quadratic steady-state cost and a nominal model of the linear system (i.e. system coefficients are known and the additive disturbance is piecewise constant). We assume the inequality constraints are only active in the transient phase and therefore avoid the important but difficult situation of unreachable setpoints. Using an input-to-state stability approach, we analyse the (inherent) robust stability of the nominal FOMPC. Results obtained show that the nominal FOMPC is robust to un-modelled dynamics if the uncertainty satisfies a given bound. Also, it was realized that intuitively, nominal FOMPC is more robust under stronger regulatory action i.e. higher penalty on the control error compared to the control input. Finally, we present numerical examples that demonstrate the efficacy of nominal FOMPC.

1.6.6 Chapter 7: Robust feedback optimizing model predictive control

This chapter develops robust FOMPC algorithms that extend the results obtained in Chapter 6 to the case of linear systems with unknown coefficient matrices that take values from a known bounded polytopic set. We present two approaches to the design of robust FOMPC algorithms. The first approach is based on the computationally efficient but conservative tube approach to robust MPC, while the second approach uses the less conservative but computationally inefficient min-max approach to robust MPC. We also analyze the convergence of robust FOMPC and present numerical examples to illustrate the performance.

1.6.7 Chapter 8: Distributed feedback optimizing model predictive control

In this chapter, we solve the nominal FOMPC problem for the case of large-scale linear time-invariant systems with decomposable dynamics and separable steady-state quadratic costs. We present distributed FOMPC algorithms that approximate the solution to the centralized nominal FOMPC presented in Chapter 6. We adopt a tube-based MPC approach to design the associated control laws. We also present a detailed analysis of the convergence of the proposed algorithms to a neighbourhood of the optimal steady-state equilibrium.

1.6.8 Chapter 9: Feedback optimizing predictive load-frequency control for real-time economic dispatch

This chapter completes the contribution of this thesis by applying the FOMPC algorithms developed in the thesis to solve the vital problem of real-time economic dispatch in power systems with high penetration of intermittent generation sources. Specifically, we formulate the optimal load-frequency control problem in the framework of FOMPC. We solve the optimal load-frequency control problem using the algorithms developed in the thesis for FOMPC. Numerical simulation of the algorithm shows that FOMPC gives superior performance to conventional algorithms developed for solving the optimal load-frequency control problem that ignores the dynamic performance objective in the control formulation.

1.6.9 Chapter 10: conclusion and future work

This chapter contains concluding remarks and future research directions and generalizations of the results presented in this thesis.

1.7 List of Publications

1. A. Asuk and P. Trodden, "Feedback Optimzing Linear Quadratic Control", in 2021 American Control Conference (ACC), pp. 3800-3805, IEEE, 2021.

2. A. Asuk and P. Trodden, "Feedback-Optimizing Model Predictive Control for Constrained Linear Systems", in IFAC-PapersOnLine, pp. 43-49, Elsevier, 2021.

3. A. Asuk and P. Trodden, "Feedback Optimizing MPC for Load Frequency Control and Economic Dispatch", in IFAC-PapersOnLine, pp. xx, Elsevier, 2023 (submitted).

Chapter 2

Background and Literature Review

The aim of this chapter is to present a theoretical background on control and optimization of dynamic systems and also give a detailed review of feedback optimizing control and its application to power systems. The chapter will primarily discuss control schemes that solve steady-state optimization problems within their feedback loop. The outline of the chapter is as follows: Section 2.1 presents a brief introduction to a standard centralized model predictive control algorithm. Section 2.2 reviews tracking model predictive control formulations. Section 2.3 briefly reviews steady-state optimizing model predictive control algorithms while distributed model predictive control algorithms are discussed in Section 2.4. In Section 2.5, the concept of feedback optimization is introduced and a detailed review of the available techniques for achieving feedback optimizing control is given. Section 2.6 gives a brief introduction to electric power systems with a brief review of power system frequency control presented in Section 2.7. Section 2.8 presents a review of economically optimal frequency control schemes. Finally, in Section 2.9, a summary of the chapter is given with conclusions.

2.1 Model Predictive Control

Model predictive control (MPC) was introduced in Chapter 1. Its basic principles and unique advantages were explained. In this section, a detailed review of MPC is presented. MPC was first developed for the control of complex, constrained multi-variable systems in the process industry. However, tremendous progress has been made both in the theory and application of MPC. There have been successful industrial applications of MPC to the control of petrochemical, power/energy, auto-mobiles plants etc. [177, 180]. The advantages of MPC includes [182]: explicit and systematic constraint handling capability; intuitive appeal; ability to optimize arbitrary performance objectives and the ability to yield the performance benefits of both feed-forward and feedback control. MPC provides an effective and generalized approach to designing controllers for a wide range of practically relevant control problems. The control action in MPC is obtained in real-time by repeatedly solving online, a constrained optimal control problem and implementing the solution for the current sampling time as the control law. Due to a control law obtained by online optimization, MPC is computationally intensive and therefore relies on the availability of decent computing power. Fortunately, the recent advances in computing hardware/software has greatly reduced the cost and improved the reliability of implementing MPC.

In [39], an interesting analogy of MPC was given where the algorithm is likened to driving a car. While MPC represents driving based on the information gathered by looking ahead through the wind-shield, classical feedback control such as PID is similar to driving based on information gathered from the past like bumping into other cars in order to figure out a safe direction to drive. A real driver uses an MPC-like approach to safely steer a car, since the driver looks forward (i.e. makes predictions) and chooses an ideal action (i.e. optimizes the control input) based on possible future outcomes, taking the real characteristics (or system model) of the car into consideration. A driver using classical control will likely steer the car blindly, basing their actions on the feedback from the outcome of previous decisions made. Moreover, with classical control, the driver cannot take into consideration the constraints in the system.
We can take the driver analogy further to understand how MPC actually works. Because the driver has a mental image of how the car operates (i.e., a simple mathematical model), in good visibility, the driver may see far ahead in the horizon and therefore drive much faster without risking safety. However, if the visibility is bad, the horizon in front of the driver is also short and the driver may easily make an unsafe decision. The portion of the road the driver can see is akin to the prediction horizon in MPC. If the prediction horizon is too short, instability might occur; this is equivalent to crashing the car because the driver could not see far ahead of the road and was driving too fast. In the next subsection, the formulation of a standard centralized MPC algorithm is presented in more detail.

2.1.1 Centralized Model Predictive Control

Consider a time-invariant, linear, discrete-time system described by the model

$$x(k+1) = Ax(k) + Bu(k), \;\; y(k) = Cx(k) \tag{2.1}$$

where $x(k) \in \mathbb{R}^{n_x}$ is the state vector, $u(k) \in \mathbb{R}^{n_u}$ is the input vector and the pair (A, B) are the system coefficients assumed to be reachable. The states and inputs are both subject to the following inequality constraints:

$$x(k) \in \mathbb{X} \subseteq \mathbb{R}^{n_x} \text{ and } u(k) \in \mathbb{U} \subseteq \mathbb{R}^{n_u}. \tag{2.2}$$

Let the goal be to find at time step k, the controller $u(k)$ that regulates the states $x(k)$ to the origin, $x = 0$, as $k \to \infty$, while satisfying the inequality constraints (2.2) at all times. A systematic way to synthesize this controller would be by solving the following constrained optimal control problem (OCP):

$$\begin{aligned} \min_{u(k+i)} \quad & \sum_{i=0}^{\infty} l(x(k+i), u(k+i)) \\ \text{s.t.} \quad & \\ & x(k) = x(0), \\ & x(k+i+1) = Ax(k+i) + Bu(k+i) \\ & x(k+i) \in \mathbb{X}, \;\; u(k+i) \in \mathbb{U} \end{aligned} \tag{2.3}$$

where $x(0)$ is the measured or estimated state at time step k, $x(k+i)$ the state prediction at time step $k+i$, $u(k+i)$ the corresponding control input, and $l(x(k+i), u(k+i))$ a stage cost capturing the performance objectives of the closed-loop system.
Due to an infinite number of decision variables and the presence of inequality constraints, solving problem (2.3) for an explicit control law of the form

$$u(k) = -Kx(k) \tag{2.4}$$

is generally difficult and may even be intractable.

Remark 1 (Linear-Quadratic Control (LQC)). *[133] In the absence of inequality constraints i.e.,*

$$\mathbb{X} = \emptyset, \quad \mathbb{U} = \emptyset, \tag{2.5}$$

and with a stage cost that is quadratic i.e.

$$l(x(k+i), u(k+i)) = \frac{1}{2}\Big(\|x(k+i)\|_Q^2 + \|u(k+i)\|_R^2\Big), \tag{2.6}$$

the OCP (2.3) *reduces to the LQC problem:*

$$\begin{aligned} \min_{u(k+i)} \quad & V(x(k)) = \sum_{i=0}^{\infty} l(x(k+i), u(k+i)) \\ s.t. \quad & \\ & x(k) = x(0) \\ & x(k+i+1) = Ax(k+i) + Bu(k+i) \end{aligned} \tag{2.7}$$

which admits an explicit solution of the form (2.4) *where*

$$K = (R + B^\top PB)^{-1}B^\top PA, \tag{2.8}$$

P is the solution of the Lyapunov equation

$$(A - BK)^\top P(A - BK) - P = -(Q + K^\top RK), \tag{2.9}$$

and

$$V^*(x(k)) = \frac{1}{2}\|x(k)\|_P^2 \tag{2.10}$$

is the optimal value of the infinite horizon cost function, $V(x(k))$, also known as the value function.

The main objective of MPC is to obtain an approximate solution to problem (2.3) by:

- approximating the problem with a tractable finite horizon OCP,
- solving the finite horizon OCP online to obtain the optimal control action at the current time step, k, and
- implementing the optimal control action in a receding horizon fashion (i.e. implementing only the current optimal control action and re-solving the OCP at the next time step) to induce an implicit feedback control law.

Given a prediction horizon equal to $N \in \mathbb{I}$ time steps, the objective of an MPC controller at time step k is to compute the following $N-$horizon control sequence

$$\mathbf{u}_N = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N-1) \end{bmatrix} \tag{2.11}$$

that minimizes the finite horizon objective function

$$V_N(x(k), \mathbf{u}_N) = V_f(x(k+N)) + \sum_{i=0}^{N-1} l(x(k+i), u(k+i)) \tag{2.12}$$

subject to the following constraints for all $i \in \mathbb{I}_{[0,N-1]}$

$$x(k) = x(0), \tag{2.13a}$$

$$x(k+i+1) = Ax(k+i) + Bu(k+i), \tag{2.13b}$$

$$x(k+i) \in \mathbb{X}, \quad u(k+i) \in \mathbb{U}, \tag{2.13c}$$

and the terminal constraint,

$$x(k+N) \in \mathbb{X}_f \subseteq \mathbb{X}. \tag{2.14}$$

Here, $l(x(k+i), u(k+i))$ is the stage cost at time step $k+i$, commonly defined as the quadratic function (2.6) which penalizes the predicted state $x(k+i)$ and input $u(k+i)$ at the time instant $k+i$ with $Q \succeq 0$ and $R \succ 0$ the respective penalties on the state and input. The function $V_f(x(k+N))$ is a terminal cost used to guarantee convergence of the MPC algorithm [153]. The terminal constraint $\mathbb{X}_f$ is also used to guarantee constraint satisfaction beyond the prediction horizon i.e., $\forall i > N$ [153]. If the cost functions $l(\cdot,\cdot)$, $V_f(\cdot)$ are quadratic and the sets $\mathbb{X}, \mathbb{U}$ and $\mathbb{X}_f$ are polytopic, then given the current value of the state $x(k) = x(0)$, the MPC optimization problem can be formulated as a quadratic program of the form:

$$\begin{aligned} \min_{\mathbf{u}_N} \quad & \tfrac{1}{2}\mathbf{u}_N^\top H \mathbf{u}_N + f^\top \mathbf{u}_N + c \\ \text{s.t.} \quad & P_u \mathbf{u}_N \leq q_u + S_x x(k). \end{aligned} \tag{2.15}$$

The MPC action, $u(k)$, applied to the actual plant at time step k is then derived as the first element of the optimal control sequence $\mathbf{u}_N$ obtained from the online solution of the quadratic program (2.15) at the current time step. i.e.,

$$u(k) = \mathbf{u}_N(0). \tag{2.16}$$

At the next time step, $k+1$ the quadratic program (2.15) is solved again using the current state measurement $x(k+1)$ and the MPC controller is derived in a similar way as previously described.
Because the control input is derived from the online solution of an optimization problem at each time step k, the MPC control law is an implicit state-feedback controller.
When analysing an MPC scheme, three properties are of importance [13]:

1. ***Recursive Feasibility:*** This is the property that guarantee the existence of a solution to the MPC optimization problem (*i.e., minimize* (2.12) *subject to* (2.13) *and* (2.14)) at all time steps $i > 0$ if there exists a solution to the MPC optimization problem at $i = 0$.
2. ***Stability:*** This is the property that guarantee convergence of the output, $y(k)$, of the closed-loop system under the MPC controller, $u(k)$, to the desired reference, r as $k \to \infty$.
3. ***Performance:*** This is the property that ensures the closed-loop dynamics has the desired transient properties.

To guarantee recursive feasibility and stability of MPC, the terminal constraint (2.14) and the terminal cost $V_f(x(k+N))$ are used [153]. The set $\mathbb{X}_f$ in (2.14) guarantees satisfaction of the inequality constraints beyond the prediction horizon and is designed such that:

$$x(k+N) \in \mathbb{X}_f \implies x(k+N) \in \mathbb{X},\ \ u(k+N) = -Kx(k+N) \in \mathbb{U} \text{ and } x(k+N+1) \in \mathbb{X}_f \tag{2.17}$$

where K is such that the terminal cost $V_f(x(k+N))$ is a Lyapunov function for the closed-loop dynamics:

$$x(k+N+1) = Ax(k+N) + Bu(k+N). \tag{2.18}$$

The terminal cost $V_f(x(k+N))$ is generally chosen to approximate the infinite horizon optimal cost for the unconstrained OCP i.e.,

$$\begin{aligned}
&\min_{u(k+N)} \quad \sum_{i=N}^{\infty} l(x(k+i), u(k+i)) \\
&\quad \text{s.t.} \\
&\qquad x(k+i+1) = Ax(k+i) + Bu(k+i).
\end{aligned} \tag{2.19}$$

Remark 2. *Due to the recursive feasibility of the terminal constraint* (2.14)*, the MPC problem (i.e., minimize* (2.12) *subject to* (2.13) *and* (2.14)*) automatically satisfies the inequality constraints beyond the prediction horizon i.e.* $i > N$ *and can therefore be treated as an unconstrained OCP equivalent to the LQC problem.*

From the solution to the LQC problem in Remark 1, the terminal cost can be chosen as the value function for the LQC problem (2.19) as,

$$V_f(x(k+N)) = \frac{1}{2}\|x(k+N)\|_P^2 \tag{2.20}$$

and P is given by (2.9). The dynamic performance of the closed loop system under MPC depends on the choice of the weights/penalties on the state and inputs i.e., Q and R respectively.

Remark 3. *Although the set* $\mathbb{X}_f$ *and the function* V_f *are required to obtain theoretical guarantees of recursive feasibility and stability in a MPC algorithm, they can reduce the domain of attraction of the MPC controller thereby introducing some degree of conservatism. However, practical asymptotic stability and feasibility can still be obtained without these terminal ingredients (i.e.,* $\mathbb{X}_f = \emptyset$ *and* $V_f = 0$*) if the prediction horizon is long enough and the tuning parameters* Q *and* R *are chosen appropriately. However, this can come at the expense of computational efficiency as longer prediction horizon lead to bigger online optimization problems which may be undesirable for systems with higher dimensions and real-time computing limitations.*

Putting the conceptual ideas above into an algorithm yields the MPC implementation given in Algorithm 1.

Algorithm 1 : Basic MPC Algorithm

1: **Initialize MPC**: with Prediction horizon(N), Plant Model ($x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k)$, Tuning Parameters (Q, P, R), Constraints etc.
2: **for** $k = 1$ to N **do**
Input: Current Measured/Estimated Plant Behaviour/State
3: Use Plant Model to **Predict system behaviour N steps into the future**
4: **Solve the $N-$horizon Optimal Control Problem (OCP) online** to obtain the N optimal control input sequence that gives the best predicted plant performance according to a predefined cost/objective function
Output: Apply the first of the N optimal control input sequence from (4) to the plant
5: **return** to Step 2

More details on the theory and design of MPC can be found in [182].

2.1.2 Robustness in MPC

The MPC algorithm previously described assume an accurate model of the system dynamics. In reality, the true model of the system dynamics is uncertain and the model in (2.1) is only approximate. For very small values of this uncertainty, the MPC algorithm above can guarantee some degree of robustness [151]. However, when the uncertainty becomes significant, a robust MPC design must be adopted. Uncertainty in the system dynamics is commonly modelled as [40]:

- an additive disturbance, $w(k)$, on the state equation i.e.,

$$x(k+1) = Ax(k) + Bu(k) + w(k), \tag{2.21}$$

- an uncertainty in the system parameters, δ , i.e.,

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k). \tag{2.22}$$

- or both i.e.,

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k) + w(k). \tag{2.23}$$

The most common approaches for dealing with uncertainty in MPC includes:

1. An inherent robustness approach where the MPC is designed as previously described for a nominal model without the uncertainty. The system is then analyzed to obtain stability guarantees under model uncertainty. This approach works if the uncertainty satisfies a given bound which is often very restrictive and sometimes difficult to compute. Details of this approach can be found in [182, 151]. In Subsection 6.3.2 of this thesis, this idea is used to design robust feedback optimizing MPC algorithms.

2. A "min-max" MPC approach where the MPC control law is computed such that it minimizes the worst case cost over all possible realizations of the uncertainty. This approach is computationally expensive and can result in poor dynamic performance. Details of this approach can be found in [40, 15]. In Sections 5.5 and 7.3 of this thesis, we apply this technique in the design of robust feedback optimizing linear quadratic and model predictive control algorithms respectively.

3. A feedback MPC approach where a sequence of control laws are computed which minimizes the cost function for a known nominal model of the system while satisfying the state and input constraints for every possible realization of the uncertainty. A popular example of this approach is the tube-based MPC algorithm originally proposed in [154] where a linear state feedback control law is assumed and a robust positive invaraint set is used to satisfy the constraints for every realization of the uncertainty. Here, the online computational burden is significantly reduced compared to the min-max approach as most of the computations are performed offline. In Section 7.2 of this thesis, robust feedback optimizing MPC algorithms are designed using this approach.

2.2 Tracking MPC

In a tracking MPC formulation, the goal is to steer the outputs, $y(k)$ of a dynamic system to a reference/setpoint, r while stabilizing the system and satisfying the constraints. Because the setpoint/reference in a tracking MPC problem can change in unexpected ways, the tracking MPC problem is inherently uncertain. In most cases, tracking MPC can treated as a deterministic problem by making certain assumptions on the future evolution of the reference. In the literature, there are two main approaches for solving the tracking MPC problem under constant or slowly varying references. These are the two-layer offset-free MPC and the velocity MPC algorithms [171, 144].

2.2.1 Two-layer Offset-free MPC

In a two-layer offset-free MPC algorithm, a steady-state target optimizer (SSTO) computes a target equilibrium point (x_r, u_r) for the system (2.1) that corresponds with the reference, y_{sp}. For systems where the inequality constraints (2.2) are inactive in steady-state, (x_r, u_r) is obtained from the solution of the following steady-state equations:

$$x_r = Ax_r + Bu_r \tag{2.24a}$$

$$Cx_r = y_{sp}. \tag{2.24b}$$

Solving (2.24), the target equilibrium (x_r, u_r) is computed as,

$$\begin{bmatrix} x_r \\ u_r \end{bmatrix} = S_r^{-1} \begin{bmatrix} \mathbf{0} \\ y_{sp} \end{bmatrix}, \tag{2.25}$$

where the matrix, S_r is given by

$$S_r = \begin{bmatrix} A - I & B \\ C & \mathbf{0} \end{bmatrix} \tag{2.26}$$

and must have full row rank for (2.25) to have a unique solution. For systems where the inequality constraints (2.2) are active in steady-state, the following optimization problem is solved to obtain (x_r, u_r):

$$\min_{(x_r, u_r) \in (\mathbb{X} \times \mathbb{U})} \quad \frac{1}{2}\Big(\|e_r\|^2_{Q_r} + \|u_r\|^2_{R_r}\Big) \quad \text{s.t. (2.24a).} \tag{2.27}$$

where $e_r = Cx_r - y_{sp}$ is the steady-state tracking error. The problem in (2.27) is called the steady-state target optimization problem [181]. The steady-state target optimization problem must be solved alongside the MPC problem if the reference changes between sample times. After computing (x_r, u_r), the system dynamics (2.1) is then shifted to the new equilibrium point (x_r, u_r), obtaining the following translated dynamics,

$$\tilde{x}(k+1) = A\tilde{x}(k) + B\tilde{u}(k) \tag{2.28}$$

where $\tilde{x}(k) = x(k) - x_r$ and $\tilde{u}(k) = u(k) - u_r$. A stabilizing MPC algorithm (similar to the algorithm in Subsection 2.1.1) is then designed to stabilize the translated dynamics, (2.28) at the origin and as a result tracking the reference, y_{sp} for the actual system

(2.1). To guarantee stability and recursive feasibility in a two-layer tracking MPC algorithm will require the use of a terminal cost function, V_f and a terminal set, $\mathbb{X}_f$ in a similar fashion to standard stabilizing MPC. However, for tracking MPC problems with changing references or additive uncertainty in the system dynamics, computing a terminal set can be challenging. This is because with changing references, the target equilibrium (x_r, u_r) also changes and therefore the terminal set $\mathbb{X}_f$ will need to be recomputed online alongside the MPC optimization problem. This will ultimately lead to a computationally expensive MPC problem which is undesirable. Several workarounds to this problem have been proposed, see for example the review in [135]. More details on the two-layer tracking MPC algorithm can be found in [144, 181]. In Sections B.2 and B.3, the idea behind this two-layer MPC algorithm will be extended to design tracking MPC controllers that are feedback optimizing.

2.2.2 Velocity MPC

The need to explicitly compute the steady-state targets complicates the formulation and implementation of two-layer tracking MPC algorithms. Also, with additive disturbances/uncertainty, two-layer MPC always require the use of disturbance observers even with piecewise constant disturbances. The velocity MPC algorithm first proposed in [172] addresses these limitations. Velocity MPC does not require disturbance estimation (if the disturbance is piecewise constant) and also does not rely on the explicit computation of a steady-state target in order to track the reference [26]. The velocity MPC algorithm works by expressing the system dynamics (2.1) in the so-called "velocity-form", where the new state variable is composed of the state increments, $\delta x(k) = x(k) - x(k-1)$, and the tracking error, $e(k) = y(k) - y_{sp}$. The control variable used in formulating the velocity MPC is the increment in the control action i.e. $\delta u(k) = u(k) - u(k-1)$. For the system dynamics (2.1), the following is the corresponding velocity form:

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k) \tag{2.29a}$$

$$e(k) = \mathcal{C}\epsilon(k), \tag{2.29b}$$

where

$$\epsilon(k) := \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x(k) &:= x(k) - x(k-1), \\ \delta u(k) &:= u(k) - u(k-1), \\ e(k) &:= y(k) - y_{sp}, \end{aligned} \tag{2.30}$$

and

$$\mathcal{A} = \begin{bmatrix} A & \mathbf{0}_{n_x \times n_y} \\ C & I_{n_y} \end{bmatrix}, \qquad \mathcal{B} = \begin{bmatrix} B \\ \mathbf{0} \end{bmatrix}, \qquad \mathcal{C} = \begin{bmatrix} C & I_{n_y} \end{bmatrix}. \tag{2.31a}$$

Given the velocity form (2.29), the velocity MPC controller is given by

$$u(k) = u(k-1) - \kappa_N(\epsilon(k)), \tag{2.32}$$

where $\kappa_N(\epsilon(k))$ is the first control move to the optimal control sequence $\delta\mathbf{u}^*(k)$ computed at time k by solving the following OCP,

$$\mathbb{P}(\epsilon(k))\colon \min_{\delta\mathbf{u}(k)\in\mathscr{U}_N} V_N(\epsilon(k), \delta\mathbf{u}(k)) \tag{2.33}$$

where the decision variable is the sequence of control increments over the N-step prediction horizon

$$\delta\mathbf{u}(k) \coloneqq \Big\{\delta u(k), \delta u(k+1) \ldots, \delta u(k+N-1)\Big\}, \tag{2.34}$$

and the feasible region $\mathscr{U}_N(\epsilon(k))$ is defined as

$$\mathscr{U}_N(\epsilon(k)) \triangleq \left\{ \delta\mathbf{u}(k) \,\middle|\, \begin{array}{r} \epsilon(k+i+1) = \mathcal{A}\epsilon(k+i) + \mathcal{B}\delta u(k+i),\ \forall i \in \mathbb{I}_{[0,N_p-1]} \\ (\epsilon(k+i), \delta u(k+i)) \in \mathbb{G},\ \forall i \in \mathbb{I}_{[0,N_p-1]} \\ \epsilon(k+N) \in \mathbb{G}_f \end{array} \right\}. \tag{2.35}$$

The performance objective is defined as

$$V_N(\epsilon(k)) = \frac{1}{2}\|\epsilon(k+N)\|_{\mathcal{P}}^2 + \frac{1}{2}\sum_{i=0}^{N-1}\Big(\|e(k+i)\|_{Q_e}^2 + \|\delta u(k+i)\|_{\mathcal{R}}^2\Big), \tag{2.36}$$

where $Q_e \succeq 0$, $\mathcal{R} \succ 0$ and $\mathcal{P} \succ 0$ is the stabilizing positive definite solution to discrete-time algebraic Riccati equation

$$\mathcal{P} = \mathcal{A}^\top\mathcal{P}\mathcal{A} + \mathcal{C}^\top Q_e\mathcal{C} - \mathcal{A}^\top\mathcal{P}\mathcal{B}\Big(\mathcal{R} + \mathcal{B}^\top\mathcal{P}\mathcal{B}\Big)^{-1}\mathcal{B}^\top\mathcal{P}\mathcal{A}. \tag{2.37}$$

The set $\mathbb{G}$ ensures that the velocity state δx and control increment δu both satisfy the inequality constraints (2.2) on x and u, that is,

$$\mathbb{G} := \Big\{(\delta u, \epsilon) | (u, x) \in \mathbb{U} \times \mathbb{X}\Big\}. \tag{2.38}$$

The set $\mathbb{G}_f$ is a terminal set constructed such that

$$\begin{aligned} \epsilon(k+N) \in \mathbb{G}_f \implies \Big(\epsilon(k+N), \delta u(k+N)\Big) \in \mathbb{G} \text{ and} \\ (\mathcal{A} - \mathcal{B}\mathcal{K})\epsilon(k+N) \in \mathbb{G}_f, \end{aligned} \tag{2.39}$$

where

$$\delta u(k+N) = -\mathcal{K}\epsilon(k+N), \tag{2.40}$$

and $\mathcal{K}$ is the infinite horizon solution to the MPC problem (2.33) without the inequality constraints $\mathbb{G}$ and $\mathbb{G}_f$, and is given by the linear quadratic control law (see Remark 1)

$$\mathcal{K} = (\mathcal{R} + \mathcal{B}^\top \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^\top \mathcal{P} \mathcal{A}. \tag{2.41}$$

A major challenge with velocity MPC algorithms is that it is very difficult to obtain theoretical guarantees of closed-loop stability and recursive feasibility [26]. This is due to the lack of systematic and generalized techniques for computing the set $\mathbb{G}$ from the sets $\mathbb{X}$ and $\mathbb{U}$. Also, except for the work done in [26, 28], results on how to compute the terminal set $\mathbb{G}_f$ in a velocity MPC algorithm are lacking. One of the contributions of this thesis is that we present a systematic approach for computing the set $\mathbb{G}$ by simplifying the technique pioneered in [28]. Finally, most of the feedback optimizing MPC algorithms developed in this thesis are based on the velocity MPC algorithm.

2.3 Steady-state (Economics) Optimizing MPC

In the following discussion, we will review model predictive control schemes that directly consider economic optimization objectives in their problem formulation. We will refer to these MPC formulations as economics optimizing MPC. It is important to emphasize here that the economics optimizing MPC schemes reviewed in this section are not inclusive of stabilizing or tracking MPC schemes that are merely tuned for improved economic performance via the penalty weights.

Recently, researchers have been focused on improving the economic performance of MPC by including economic performance objectives in the formulation of the optimal control problem [75, 63, 64]. In the economic MPC approach, the economic objectives are used directly as the objective function of the MPC problem [73]. Here, the controller optimizes directly, in real time, the economic performance of the system, rather than tracking a set-point. Economic MPC is computationally expensive and it's optimal solution is not a-priori guaranteed to be a steady-state optimum of the plant. This makes the stability analysis of economic MPC more involved. Also, economic MPC is highly model-based and not guaranteed to achieve optimality under plant-model mismatch. We refer the reader to the following references [75, 63, 73] for detailed review of economic MPC. One problem with economic MPC schemes is that convergence to the optimal steady-state is not guaranteed without explicit knowledge of the steady-state set-points [73]. Although alternative economic MPC formulations that address this limitation have been developed (see for example [91]), they are very restrictive and cannot deal with unknown disturbances.

To achieve asymptotic convergence to the economically optimal steady-state equilibrium, without a-priori knowledge of the steady-state set-points, novel MPC techniques have been developed [75]. These MPC schemes take a different approach from economic MPC. Unlike economic MPC, these techniques either retain the standard tracking MPC formulation with modifications to accommodate the steady-state economic optimization [75] or combine the design philosophy of tracking and economic MPC [74].
In the literature, two major approaches have been used to formulate these novel economics optimizing MPC schemes. The first approach is commonly referred to as the one-layer MPC [75] while the second approach is called a modifier adaptation MPC. A review of both approaches will now follow.

2.3.1 Modifier-Adaptation (economic) MPC

A limitation of standard economic MPC is that asymptotic convergence to the optimal steady-state is not guaranteed, especially under model uncertainty [170]. To address this, modifier-adaptation economic MPC schemes have been developed that combine the idea of modifier-adaption [148, 150] proposed for real-time optimization, with offset-free (economic) MPC schemes [215, 74, 170]. Modifier-adaptation [148, 150] provides a method to steadily reduce the discrepancy between a model-based solution and the actual (unknown) solution to an optimization problem by modifying the problem at every iteration to incorporate plant measurements from the previous iteration. Modifier-adaptation iteratively corrects the optimization problem by adding adaptation terms to the cost and constraint functions. In [170], modifier-adaptation and offset-free

MPC are combined to obtain an economic MPC scheme that achieves asymptotic convergence to the optimal steady-state equilibrium under plant-model mismatch. This idea was further developed in [74] where the need for a dedicated computation of the steady-state set-points was removed by exploiting the turnpike property of economic MPC schemes.
Major limitations of modifier-adaptation economic MPC are:- they require real-time estimation of the true plant gradients, and disturbances; they result in unconventional formulations of the optimal control problem; they are computationally expensive as a dynamic economic optimization problem is still solved online; and, they rely on the optimal control problem exhibiting turn-pike properties in order to guarantee convergence to the optimal steady-state. Also, no formal convergence guarantees have been presented for these algorithms.

2.3.2 One-layer MPC

In one-layer MPC schemes, the steady-state economic performance objective is included in the tracking MPC formulation as a terminal cost. As a result, the steady-state optimization problem is explicitly solved at the terminal stage of the tracking MPC problem.

To illustrate the one-layer economics optimizing MPC schemes, consider the system

$$x(k+1) = Ax(k) + Bu(k) + Ew(k), \tag{2.42a}$$

$$y(k) = Cx(k). \tag{2.42b}$$

Here: k denotes the sample time; $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$, $y(k) \in \mathbb{R}^{n_y}$, and $w(k) \in \mathbb{R}^{n_w}$ denote the system states, inputs, outputs and exogenous disturbances respectively. Let the optimal steady-state operating point of the system be determined by the solution of the following real-time optimization (RTO) problem

$$\begin{aligned}(u_s, y_s) = \arg\min_{u,y} \quad & \Phi(u, y) \\ \text{s.t.} \quad & x = Ax + Bu + Ew, \\ & y = Cx,\end{aligned} \tag{2.43}$$

where $\Phi(u, y)$ is the steady-state economic performance objective. We assume the problem (2.43) has a unique solution. To formulate the OCP for the one-layer MPC scheme, the following dynamic performance cost is defined:

$$V_N(x,\mathbf{u}(k)) = \Phi(y(N-1),u(N-1)) + \sum_{k=0}^{N-1}\left(\|x(k)-x(N-1)\|_Q^2 + \|u(k)-u(N-1)\|_Q^2\right) \tag{2.44}$$

where $\mathbf{u}(k) = \{u(0),\ldots,u(N-1)\}$.
The economics optimizing MPC control law is then derived by solving the following OCP in real-time

$$\min_{\mathbf{u}(k)} V_N(x,\mathbf{u}(k)) \tag{2.45}$$

subject to:
$\forall k \in \mathbb{I}_{[0,N-1]}$,

$$x = x(0), \tag{2.46a}$$

$$x(k+1) = Ax(k) + Bu(k) + Ew(k), \tag{2.46b}$$

$$y(k) = Cx(k), \tag{2.46c}$$

$$y(N-1) = Cx(N-1), \tag{2.46d}$$

$$x(N) = x(N-1). \tag{2.46e}$$

In the MPC problem above, it can be observed that the steady-state set-points (x_s, u_s, y_s) are not required in the problem formulation. Therefore, the MPC controller is self-optimizing. This control formulation has found wide applications in the process control literature and a rigorous study of the algorithm is presented in [76] for undisturbed linear systems. A detailed review can be found in [75] and the references therein.

The one-layer economic MPC controller above has two main limitations. Firstly, the modified terminal cost increases the complexity of the MPC problem even for a quadratic steady-state cost, but more so for non-linear steady-state economic objectives. This makes the MPC problem computationally more expensive to solve. Also, stability analysis can become more complicated due to the unconventional form of the terminal cost. Secondly, knowledge of the unknown disturbance is required to compute the optimal control law. Although this can be addressed via state estimation, it increases the model dependence of the algorithm making it less robust to uncertainty. Modifications of the one-layer approach to uncertain systems have been presented in [59, 53] using the tube MPC design from [154].

2.4 Distributed MPC

Due to the online solution of an optimal control problem, the application of MPC in real systems depends to a large extent, on the accuracy and speed of computing the solution to the OCP. To formulate the OCP, system states, outputs and inputs are required in real-time. Also, online computational resources (e.g. memory and processing power) are required. For certain systems, all the information and computational resources required to solve the OCP are readily accessible from a single plant. For such systems, it is possible and also advantageous for a single plant to solve the MPC problem and generate all control actions for the entire system. This approach to MPC is called centralized MPC (CMPC) and is often adopted when controlling a collection of plants with efficient inter-plant communication and sufficient computational resources. The access to system-wide information makes it possible to achieve the most optimal performance in CMPC assuming no communication and computational difficulties.

There exists however a different class of dynamic systems that consists of a collection of subsystems which are sometimes geographically dispersed with no localized access to the dynamic information of all subsystems. These systems are called large scale systems (LSS). A well known example of a LSS is an interconnected power network. For large scale systems, a centralized computation of the MPC law will require the aggregation of information from all subsystems into a single system/node and the subsequent computation of the MPC law for the LSS from this node. Although such a centralized solution may yield the most optimal performance obtainable, it may not be the most feasible approach to the control of LSS. For examples, some LSS may have very weak subsystem interactions that can be ignored without negatively impacting the overall performance of the controller. In such a scenario, a CMPC solution may not bring any justifiable improvements in performance relative to the added communication and computational expense. Also, a CMPC solution may not meet the reliability needs of a LSS as it has a single point of failure. To address these challenges, distributed MPC (DMPC) schemes have been developed [72]. In DMPC, the LSS is decomposed to a collection of say M subsystems, each controlled by M local controllers. For example, the system (2.1) can be decomposed into the following M interconnected subsystems, each one described by the state-space model

$$\mathcal{S}_i : x_i(k+1) = A_{ii}x_i(k) + B_{ii}u_i(k) + \sum_{j=1, j\neq i}^{M} \{A_{ij}x_j(k) + B_{ij}u_j(k)\}, \tag{2.47a}$$

$$y_i(k) = C_{ii}x_i(k), \tag{2.47b}$$

where $x_i \in \mathbb{R}^{n_{xi}}, y_i \in \mathbb{R}^{n_{yi}}$ and $u_i \in \mathbb{R}^{n_{ui}}$ are respectively the states, outputs and inputs of the i^{th} subsystem. We assume that each pair (A_{ii}, B_{ii}) is stabilizable and each subsystem state, x_i is locally measurable. For each subsystem $\mathcal{S}_i$, the following local constraints must be satisfied:

$$u_i \in \mathbb{U}_i \text{ and } x_i \in \mathbb{X}_i, \tag{2.48}$$

where $\mathbb{U}_i$ and $\mathbb{X}_i$ are polyhedral sets containing the origin in their interior. Also, we consider the following stage cost,

$$l_i(x_i, u_i) \triangleq \frac{1}{2}\Big(\|x_i\|^2_{Q_i} + \|u_i\|^2_{R_i}\Big), \tag{2.49}$$

and terminal cost

$$V_{fi}(x_i) \triangleq \frac{1}{2}\|x_i\|^2_{P_i} \tag{2.50}$$

for each subsystem $\mathcal{S}_i$. The matrix $Q_i \in \mathbb{R}^{n_{xi}\times n_{xi}}$ is positive semi-definite while $R_i \in \mathbb{R}^{n_{ui}\times n_{ui}}$ and $P_i \in \mathbb{R}^{n_{xi}\times n_{xi}}$ are both positive definite matrices.
Given the subsystem dynamics (2.47), the distributed MPC (DMPC) problem is formulated via the optimal control problem

$$\mathbb{P}_i(x_i(k), v_{-i}(k))\colon \min_{\mathbf{u}_i(k)\in\mathscr{U}_{N,i}} V_{Ni}(x_i(k), \mathbf{u}_i(k)), \tag{2.51}$$

where the feasible region $\mathscr{U}_{Ni}(x_i(k), v_{-i}(k))$ is defined as

$$\mathscr{U}_{Ni}(x_i(k), v_{-i}(k)) \triangleq \left\{ \mathbf{u}_i(k) \left| \begin{array}{r} x_i(k+t+1) = A_{ii}x_i(k+t) + B_{ii}u_i(k+t) + \\ \displaystyle\sum_{j=1, j\neq i}^{M} \Big\{A_{ij}x_j(k+t) + B_{ij}u_j(k+t)\Big\},\ \forall t \in \mathbb{I}_{[0,N-1]} \\ x_i(k+t) \in \mathbb{X}_i,\ u_i(k+t) \in \mathbb{U}_i\ \forall t \in \mathbb{I}_{[0,N-1]} \\ x_i(k+N) \in \mathbb{X}_{fi} \end{array} \right. \right\} \tag{2.52}$$

and $v_{-i}(k+t) = (x_j(k+t), u_j(k+t))$ is state and input from the neighbouring subsystems. In this problem, the decision variable is the sequence of control inputs over the N-step prediction horizon for subsystem i i.e.,

$$\mathbf{u}_i(k) \coloneqq \Big\{u_i(k), u_i(k+1) \ldots, u_i(k+N-1)\Big\} \tag{2.53}$$

chosen to minimize the local performance objective, V_{Ni}. The set $\mathbb{X}_{fi}$ is a local terminal constraint set for subsystem i. To solve $\mathbb{P}_i(x_i(k), v_{-i}(k))$ will require complete knowledge of the system dynamics (2.47) at subsystem i. Due to the interaction terms $v_{-i}(k+t)$, this information will not be available locally at subsystem i, and the feasible set $\mathscr{U}_{N,i}(x_i(k), v_{-i}(k))$ will therefore be uncertain. As a result, a solution to $\mathbb{P}_i(x_i(k), v_{-i}(k))$ cannot be precisely computed.
To recover as much of the performance guarantees of a CMPC solution as possible, coordination between the local MPCs of each subsystems may be required and the mechanism and degree of this coordination is a major feature that distinguishes various DMPC algorithms [72]. In the following paragraphs, several DMPC schemes will be discussed.

2.4.1 Decentralized MPC (DeMPC)

Decentralized MPC (DeMPC) is a very primitive form of DMPC where the local MPC controllers are designed using only the locally available information about each subsystems (i.e. x_i and u_i only), without any form of coordination/communication and ignoring all subsystem interactions $v_{-i}(k+t)$. This is based on the assumption that the inter-plant interactions are negligible i.e.,

$$\| \sum_{j=1, j\neq i}^{M} \{A_{ij}x_j(k) + B_{ij}u_j(k)\} \| \ll 0 \tag{2.54}$$

and therefore can be ignored without having significant impact on the overall performance of the LSS. The DeMPC algorithm for each subsystem is then formulated as the standard MPC algorithm in Section 2.1.1 using the local subsystem dynamics

$$x_i(k+1) = A_{ii}x_i(k) + B_{ii}u_i(k), \tag{2.55a}$$

$$y_i(k) = C_{ii}x_i(k). \tag{2.55b}$$

Each local MPC is designed to solve the following finite-horizon OCP:

$$\mathbb{P}_i^{De}(x_i(k)) \colon \min_{\mathbf{u}_i(k) \in \mathscr{U}_{N,i}^{De}} V_{Ni}(x_i(k), \mathbf{u}_i(k)) \tag{2.56}$$

where the feasible region $\mathscr{U}_{N,i}^{De}$ is defined as

$$\mathscr{U}_{Ni}^{De}(x_i(k)) \triangleq \left\{ \mathbf{u}_i(k) \left| \begin{array}{r} x_i(k+t+1) = A_{ii}x_i(k+t) + B_{ii}u_i(k+t) \ \forall t \in \mathbb{I}_{[0,N-1]} \\ x_i(k+t) \in \mathbb{X}_i, \ u_i(k+t) \in \mathbb{U}_i \ \forall t \in \mathbb{I}_{[0,N-1]} \\ x_i(k+N) \in \mathbb{X}_{fi} \end{array} \right. \right\}. \tag{2.57}$$

Here $V_{Ni}(x_i(k), \mathbf{u}_i(k))$ is defined as

$$V_{Ni}(x_i(k), \mathbf{u}_i(k)) = V_{fi}(x_i(k+N)) + \sum_{t=0}^{N-1} l_i(x_i(k+t), u_i(k+t)). \tag{2.58}$$

and depends only on local inputs u_i and states x_i because the subsystems interaction term $v_{-i}(k+t)$ is assumed to be negligible. Therefore each DeMPC problem, $\mathbb{P}_i^{De}(x_i(k))$ is solved without relying on computations from neighbouring subsystems and as a result no iterations are needed.

Large scale systems with strong inter-plant interactions (e.g., coupling in the dynamics, inputs, constraints or objectives of local plants) may experience significant degradation in stability and performance under DeMPC [146], creating a need for coordination among the decentralized local controllers. DeMPC algorithms have been proposed in [146, 184, 3]. A challenge with most DeMPC algorithms is that obtaining theoretical guarantees of stability and feasibility can be very difficult as the standard techniques used in centralized MPC are not directly applicable. For example, the DeMPC algorithm proposed in [146] impose very conservative contractive constraints on the local DeMPC problems in order to guarantee stability while the DeMPC algorithm proposed in [184] adopted a robust control approach to reject dynamic interactions and guarantee closed-loop stability. Despite its limitations, DeMPC is a very scalable, robust and flexible way to control LSS when the dynamic interactions are negligible.

When the dynamic interactions between subsystems cannot be ignored, it becomes necessary to coordinate the control actions of the DeMPC controllers and recover as much of the performance guarantees of CMPC as possible. To achieve this, several DMPC schemes have been proposed [145, 72, 187] and can be grouped into the following categories:

- Non-cooperative DMPC
- Cooperative DMPC

2.4.2 Non-cooperative DMPC

In non-cooperative DMPC, each local MPC minimizes a local objective function V_{Ni} without considering the objectives of neighbouring subsystems i.e. $V_{Nj},\ j \in \mathcal{N}_i$, but while taking into account the dynamic interactions between the subsystems i.e. $v_{-i} = (\hat{x}_j, \hat{u}_j)$, where $\mathcal{N}_i$ is the set of all subsystems directly interacting with subsystem i. Given a known value of the neighbours' control input sequences, $\{\hat{\mathbf{u}}_j\}_{j\in\mathcal{N}_i}$ and state predictions $\{\hat{\mathbf{x}}_j\}_{j\in\mathcal{N}_i}$, the non-cooperative DMPC controller is obtained by solving the following OCP locally:

$$\mathbb{P}_i^{NCDi}(x_i(k), v_{-i}(k))\colon \min_{\mathbf{u}_i(k)\in\mathscr{U}_{N,i}} V_{Ni}(x_i(k), \mathbf{u}_i(k)), \tag{2.59}$$

where the feasible region $\mathscr{U}_{Ni}(x_i(k), v_{-i}(k))$ is defined as in (2.52) with $v_{-i} = (\hat{x}_j, \hat{u}_j)$ and the objective function, V_{Ni} is given by (2.58).

Here $(\hat{x}_j, \hat{u}_j)$ are the planned input and states obtained from the local solution to the MPC problem (2.59) of neighbouring subsystems and communicated to subsystem i at each sample time. Non-cooperative DMPC can be iterative or non-iterative. In iterative schemes, local MPC controllers exchange information (such as planned state and input trajectories) several times within a sampling interval in order to improve the local solutions. These schemes often result in excessive communication but can guarantee improved performance. Non-iterative schemes exchange information between local MPCs only once in each sampling interval. As a result, these schemes trade performance for reduced communication and often resort to robust control techniques to compensate for the reduced access to global information. In this thesis, robust non-iterative DMPC algorithms will be proposed for the feedback optimization of large scale systems due to the reduced communication and computational needs of these algorithms.

Robust DMPC

Robust DMPC algorithms are non-iterative, non-cooperative DMPC algorithms that make use of robust control techniques to handle the unknown interactions between subsystems. These algorithms treat the unplanned interactions (i.e, the difference between the actual trajectories and the communicated trajectories of neighbouring subsystems) as disturbances and resort to several robust control techniques to handle these disturbances. In [178], the unplanned interactions are handled using an input-to-state stability and small gain approach. In [102], the unplanned interactions are handled using the computationally expensive min-max approach. In [213, 71, 185, 94], the computationally efficient tube-based robust MPC approach originally proposed in [154] was adopted to deal with the unplanned interaction-induced disturbances. Tube-based robust DMPC algorithms are one of the most widely adopted robust DMPC approaches due to their simplicity (very similar in design to standard nominal MPC) and computational efficiency. In Chapter 8 of this thesis, a tube-based robust DMPC algorithm will be developed for achieving feedback optimization in LSS systems using the velocity MPC algorithm.

2.4.3 Cooperative DMPC

In cooperative DMPC [204], each local MPC minimizes a common system-wide objective function V_N that is composed of the sum of all local objective functions, i.e.,

$$V_N(x(k), \mathbf{u}(k)) \triangleq \sum_{i=1}^{M} \alpha_i V_{Ni}(x_i(k), \mathbf{u}_i(k), \{\mathbf{u}_j(k)\}_{j\in\mathcal{N}_i}) \tag{2.60}$$

while taking into account the dynamic interactions between the subsystems i.e. $v_{-i} = (\hat{x}_j, \hat{u}_j)$, where $\mathcal{N}_i$ is the set of all subsystems directly interacting with subsystem i. Here $\alpha_i > 0$, for all i, are given scalar weights and $\mathbf{u} = (\mathbf{u}_1, \ldots, \mathbf{u}_M)$ is the overall control sequence for the LSS. Given a known value of the neighbours' control input sequences, $\{\hat{\mathbf{u}}_j\}_{j\in\mathcal{N}_i}$ and state predictions $\{\hat{\mathbf{x}}_j\}_{j\in\mathcal{N}_i}$, the cooperative DMPC controller is obtained by solving the following OCP locally:

$$\mathbb{P}_i^{CDi}(x_i(k), v_{-i}(k)) \colon \min_{\mathbf{u}_i(k)\in\mathscr{U}_{N,i}} V_N(x(k), \mathbf{u}(k)) \tag{2.61}$$

where the feasible region $\mathscr{U}_{Ni}(x_i(k), v_{-i}(k))$ is defined as in (2.52) with $v_{-i} = (\hat{x}_j, \hat{u}_j)$. Cooperative DMPC can be iterative or non-iterative where local MPC controllers exchange information with neighbouring MPC controllers to improve their solutions. A benefit of cooperative DMPC is that the local MPC actions are computed with considerations given to the objectives and dynamics of neighbouring subsystems. This makes it possible to recover the performance of centralized MPC in a cooperative DMPC algorithm at the cost of excessive communication and computational requirements.

2.5 Feedback Optimizing Control

Feedback optimizing control is the class of feedback control algorithms that use input-output measurements to drive a closed-loop system to optimal equilibria defined by a steady-state optimization problem. These control algorithms are also referred to as feedback optimization [156, 117, 85], autonomous optimization [55], and self-optimizing control [101]. The defining features of most feedback optimizing control algorithms are:

1. They do not require a-priori knowledge of the optimal steady-state set-points or the unknown disturbances to track the optimal equilibrium [87].
2. They rely on feedback as opposed to feed-forward computations i.e. advance information are not explicitly used to numerically compute the optimal steady-state set-points.
3. They require little to no model information [45].
4. They rely on input-output measurements to solve the steady-state optimization [45, 87].

Traditionally, to achieve optimal steady-state control, a feed-forward approach is often adopted [65]. In this approach, the steady-state optimization problem is periodically solved using advanced knowledge of the disturbances and a detailed system model to generate steady-state set-points [87]. A tracking type controller is then designed to track these set-points in real-time. Recently, there has been increased interest in adopting feedback instead of a feed-forward approach to the steady-state optimization of dynamic systems. The motivation for this paradigm shift has mainly resulted from the following unique benefits of feedback over feed-forward optimization [87, 88, 55, 160, 200]:

1. It is inherently more robust against unmeasured disturbances and model uncertainties.
2. It is computationally more efficient.
3. It does not require advance knowledge of the optimal set-points or disturbances.
4. It has lesser model dependence and can be made model-free.
5. It inherently enforces the steady-state constraints thus avoiding the problem of unreachable set-points common in feed-forward optimization .

In the following subsections, existing approaches in the literature for achieving feedback optimization of dynamic systems will be discussed. The focus here will be on algorithms that operate in real-time and make use of input-output measurements to track the *(previously unknown*) optimum of a steady-state optimization problem without explicit measurement of the disturbances. Also, the strengths and weakness of the various approaches will be compared. To illustrate each of these approaches, the following example problem will be considered.

Problem 1 (Feedback optimizing control (FOC) problem)**.** *Consider the scalar continuous time linear system*

$$\dot{x} = ax + bu + w, \tag{2.62a}$$

$$y = cx, \tag{2.62b}$$

$$(u, y) \in \mathbb{Z}$$

where x, u, y *and* w *are the state, input, output and disturbance respectively, and* $\mathbb{Z}$ *are the constraints on* (u, y)*. At a steady-state, the following equations are satisfied*

$$0 = ax + bu + w, \tag{2.63a}$$

$$y = cx. \tag{2.63b}$$

From (2.63) *the steady-state input-output map for* (2.62) *is computed as*

$$y = h(u, w) = g_u u + g_w w \tag{2.64}$$

where

$$g_u = -ca^{-1}b \text{ and } g_w = -ca^{-1}. \tag{2.65}$$

Although a more general convex optimization problem can be handled, for simplicity the quadratic steady-state optimization problem

$$(u^*(w), y^*(w)) = \arg\min_{u,y} \Phi(u, y) = qu^2 + ry^2 + su + ty \quad \text{subject to (2.63)}, \tag{2.66}$$

will be considered. The goal of feedback optimzing control is to find the control law $u(x)$*, that regulates the system* (2.62) *in closed-loop, to steady-states that are the solution to the problem* (2.66)*, without measuring the disturbance* w *or knowing the optimal set point* $(u^*(w), y^*(w))$*.*

We now describe various approaches in the literature available for solving the above problem.

2.5.1 Extremum seeking control

Extremum seeking control (ESC) is a data-driven approach to feedback optimization of stable or pre-stabilized systems. The basic idea behind ESC is to inject a dither/probing signal (usually a sinusoid) to the system in order to locally explore the objective function and estimate its gradient in real-time [5]. In the ESC literature, it is generally assumed that a static input-output relation exists between the optimizing parameter e.g the inputs and the optimized parameter e.g the outputs. This steady-state input-output map often characterizes the performance of the system in a steady-state equilibrium. To illustrate the ESC design, we consider Problem 1. By replacing the system dynamics (2.62a) with the steady-state input-output map (2.64), the optimization problem (2.66) can be expressed as :

$$u^*(w) = \arg\min_u \ \hat{\Phi}(u, w) \tag{2.67}$$

where $\hat{\Phi}(u, w)$ is obtained by substituting the steady-state input-output map, $y = h(u, w)$, into $\Phi(u, y)$. By first order optimality condition, the optimal solution to the above problem can be obtained by solving the equation

$$\nabla\hat{\Phi}(u, w)) = \nabla h(u, w)\nabla\Phi(u, h(u, w)) = 0. \tag{2.68}$$

To compute the value of u that satisfies the equation 2.68 above, a precise knowledge of the gradient of the input-output map, $\nabla h(u, w)$, and the objective function, $\Phi(u, h(u, w))$ are required. Due to the uncertainty w, accurate values of $\nabla h(u, w)$ and $\Phi(u, h(u, w))$ are difficult if not impossible to compute numerically.

In ESC, instead of computing $\nabla h(u, w)$ and $\Phi(u, h(u, w))$ numerically, a sinusoidal perturbation is added to the input to generate data from which these parameters can be learned/estimated in real time. For the problem under consideration, let the perturbed input $\tilde{u}$ be given as

$$\tilde{u} = u + 2\alpha \sin\phi t. \tag{2.69}$$

To obtain the extreme seeking controller, we connect the following control dynamics in closed-loop with the system:

$$\dot{u} = F(u,t) \coloneqq -\frac{\epsilon}{\alpha}\Phi(u, h(u + 2\alpha \sin \phi t, w)) \sin \phi t. \tag{2.70}$$

Here α, ϕ and ϵ are tuning parameters.

Remark 4. *A very useful fact about the extremum seeking control algorithm is that with small α and large ϕ, the control dynamics in* (2.70) *behaves approximately like the gradient descent dynamics [41]*

$$\dot{u} = F(u,t) \approx -\epsilon \nabla \Phi(u, y), \tag{2.71}$$

which is feedback optimizing [47]. To implement the extremum seeking control dynamics (2.70)*, a knowledge of $\Phi(u, h(u + 2\alpha \sin \phi t, w))$ is not needed, only measurements of the inputs and outputs, (u, y) are required. For a detailed understanding of this approximation, please refer to the excellent discussion in Section 3.2.2 of [41].*

From the remark above, it is clear that ESC only relies on and does not require knowledge of h or ∇h to achieve feedback optimization. Therefore, it is model free and generally robust against uncertainty and unmeasured disturbances.
Although there is a large literature on extreme seeking control, it has mostly found application in the adaptive control of dynamic systems [8, 5]. The application of extreme seeking control to the problem of feedback optimization for general linear and non-linear systems have only recently gained traction. In [158], extreme seeking controllers are proposed for feedback optimization of linear systems using output measurements. [51] presented extremum seeking controllers that achieve feedback optimization in constrained non-linear systems. Application of extreme seeking control to the feedback optimization of enzyme production under cellular fitness constraints was made in [86]. In the following, we summarize the pros and cons of an extremum-seeking control approach to feedback optimization. Refer to the articles [127, 87] for details.

Pros of extremum seeking control

- Model free and highly robust against un-modelled dynamics.
- Strong theoretical guarantees exists.
- Can incorporate inequality constraints.

- Simple and computationally efficient.

Cons of extremum seeking control

- Only optimizes the steady-state input or output but not both [126, 128].
- Theoretical analysis can get very complicated.
- Control design is challenging for high dimensional systems [88].
- Control design ignores dynamic performance objectives [126].
- Only applicable to stable or pre-stabilized systems.

2.5.2 Optimization algorithms as dynamic (control) systems

To solve problem (2.66) numerically, the set of feasible steady-states are first computed by solving the steady-state equation (2.63) for an estimated value of the disturbance, w. This step usually requires complete knowledge of the system model and the additive disturbances. It also adds to the computational expense. Upon computing the feasible set, optimization algorithms are then designed to iteratively search through this set for the most optimal candidate, i.e., the one that minimizes the objective function, $\Phi(u, y)$. Most optimization algorithms rely on optimality conditions to advance through the feasible set until convergence to the optimal solution is achieved.

However, new applications are emerging where this numerical approach may no longer be appropriate [217]. These are applications where the computational capability is severely limited and the system is plagued by uncertainty. Recently, the idea of directly implementing numerical optimization algorithms online as dynamic (control) systems in closed-loop with the plant, as opposed to a direct online numerical optimization has gained traction in the optimization and control literature [87]. The main advantage of this approach results from its feedback nature where input-output measurements are used to drive the closed-loop system to the unknown optimizer of the steady-state optimization in response to unmeasured disturbance changes. Thanks to their feedback implementation, these methods are generally more robust against model uncertainty and unmeasured disturbances compared to feed-forward numerical optimization. Also, these methods when designed properly do not result in infeasible set-points as they implicitly drive the system to feasible steady-states for constrained optimization problems [156].

The idea of implementing optimization algorithms as dynamic systems was first popularized by economists in the seminal work [107]. After largely being ignored, problems in internet congestion control, distributed resource allocation and optimal frequency regulation motivated the re-discovery of this idea in the recent literature on feedback-based optimization. In the following, we review feedback optimizing control schemes based on the closed-loop implementation of optimization algorithms as dynamical (control) systems. We also discuss the pros and cons of the different approaches and illustrate their application to the solution of Problem 1.

Primal-dual control

Primal-dual dynamics are well-known continuous-time algorithms for solving constrained optimization problems [107]. These algorithms seek the saddle points of the Lagrangian associated with a constrained optimization problem. Under weak technical assumptions, these saddle points coincide with the solutions of optimization problems [33]. Primal-dual control are feedback optimizing control schemes derived from the closed-loop implementation of primal-dual dynamics as controllers. When properly designed, connecting the primal-dual dynamics for a defined optimization problem in feedback with a *stable* plant will result in a closed-loop system that autonomously seeks the steady-state optimizer of the associated optimization problem.

To illustrate the primal-dual control algorithm, we consider the steady-state optimization (2.66) here recalled as,

$$u^*(w), y^*(w) = \arg\min_{u,y} \; \Phi(u,y) = qu^2 + ry^2 + su + ty \tag{2.72a}$$

$$\text{subject to:} \quad y - g_u u - g_w w = 0. \tag{2.72b}$$

We define the Lagrangian for the above problem as:

$$L(u,y,\lambda) = \Phi(u,y) + \lambda(y - g_u u - g_w w) \tag{2.73}$$

where λ is a Lagrange multiplier. Assuming Φ is strictly convex and strong duality holds, then the trajectories of the following primal-dual dynamics (also called saddle-point dynamics)[107, 42]

$$\dot{y} = -k_y \nabla_y L(u, y, \lambda) = -k_y(2ry + t + \lambda),\ k_y > 0, \tag{2.74a}$$
$$\dot{u} = -k_u \nabla_u L(u, y, \lambda) = -k_u(2qu + s - \lambda g_u),\ k_u > 0, \tag{2.74b}$$
$$\dot{\lambda} = k_\lambda \nabla_\lambda L(u, y, \lambda) = k_\lambda(y - g_u u - g_w w),\ k_\lambda > 0 \tag{2.74c}$$

converges to a saddle point of $L(u, y, \lambda)$, i.e., a point (u^*, y^*, λ^*) such that $L(u^*, y^*, \lambda) \leq L(u^*, y^*, \lambda^*) \leq L(u, y, \lambda^*)$ for all (u, y, λ). These saddle points correspond to the KKT points (i.e., the optimal solution) of the optimization (2.66)[33]. The primal-dual controller is obtained from the feedback interconnection of the primal-dual dynamics (2.74) with the dynamic system (2.62). To design the controller, the gains k_y, k_u and k_λ are chosen such that the resulting closed-loop system is stable. A Lyapunov stability analysis is often used to certify the stability of the closed-loop system.
To implement the primal-dual control law in (2.74), a knowledge of the unknown disturbance, w is often required as it explicitly appears in (2.74c). Most primal-dual algorithms use the dynamic model (2.62) and its parameters to estimate w, see for example the primal-dual controllers presented in [134, 228, 147] for details. This disturbance estimation is a major limitation of primal-dual control.

The primal-dual controller is a pure feedback-based controller and does not require explicit solution of the optimization problem to converge to the optimal steady-state. Therefore, the controller is robust against small changes in the optimization problem [36]. Although the above example was developed for unconstrained optimization, primal-dual controllers can also be similarly designed for constrained optimization problems.

In the feedback optimization literature, primal-dual control is one of the most popular algorithm for achieving feedback optimization of dynamic systems. This is because, the primal-dual control design is very general and can be used to handle both constrained and unconstrained optimization problems for both linear and non-linear dynamics. Also, primal-dual control can sometimes be easier to implement as, for most systems the plant dynamics are themselves part of the control algorithm [134, 226]. Indeed, in the primal-dual dynamics (2.74), the equation (2.74c) can be seen to replicate the system dynamics (2.62) and therefore does not have to be explicitly implemented. Also, primal-dual dynamics are easily implementable as decentralized [218, 157] or distributed [37, 228, 231] feedback optimization controllers for large-scale systems under sparsity assumptions.

One of the earliest primal-dual control design for feedback optimization of dynamic systems can be found in [104, 36]. Both works considered constrained optimization problems in the output variable. In [104], the primal-dual controller is designed using the so-called 'complementary integrators' while [36] adopted a back-stepping approach instead. In [48], dual sub-gradient methods are leveraged to design primal-dual controllers that steer the output of a dynamical system towards the solution of semidefinite programs. In [227, 228], distributed primal-dual controllers are designed for the feedback optimization of linear network systems using a reverse and forward engineering approach. In [206, 233] distributed primal-dual controllers are presented for achieving optimal steady-state regulation in multi-agent systems. Finally, several applications of primal-dual control designs to problems in power systems can be found in [160, 55] and the references therein.

In conclusion, primal-dual control is a versatile and general approach to designing feedback optimizing control algorithms for convex optimization problems. It works for both constrained and unconstrained optimization problems, and also applies to both linear and non-linear dynamics. Also, it can be implemented as distributed controllers for sparse systems. Stability guarantees are easily established using standard Lyapunov arguments. Despite these advantages, the primal-dual control design is not without limitations. Firstly, it requires the dynamic system to be asymptotically stable or pre-stabilized offline [228]. As a result, the dynamics of the system are not considered in the control design. This makes it very challenging to tune primal-dual controllers for optimal transient response. Most primal-dual controllers produce severely under- or over-damped transient response [87] and are prone to constraint violations during the transient phase [85]. Finally, the need for disturbance estimation makes primal-dual control approaches less robust against model uncertainties. Refer to the survey articles [160, 55, 87, 127] for more details. We summarize the pros and cons of a primal-dual control approach to feedback optimization below.

Pros of a primal-dual control

- Applies to linear, non-linear, constrained, unconstrained and distributed settings.
- Asymptotically enforces inequality constraints by design.
- Computationally efficient.
- Convergence easily established using Lyapunov arguments.

- Plug and play capabilities.

Cons of a primal-dual control

- Optimizes the steady-state input or output but not both.
- Convergence guarantees rely on accurate knowledge of plant model.
- Knowledge of unknown disturbances required.
- Difficult to tune, oscillatory transient response [85].
- Requires system to be stable or pre-stabilized.
- Transient constraint violations, no dynamic performance guarantees [125].

Gradient flow control

The gradient descent is an optimization algorithm that finds the (local) minima of a function by following the direction that leads to a decrease in the gradient of the function. Gradient descent algorithms have been widely studied and applied to a wide range of optimization problems. Implementing gradient algorithms as dynamical systems results in gradient flows [87].

Consider the optimization problem (2.72), by eliminating the constraint (2.72b), we obtain the unconstrained optimization problem

$$u^*(w) = \arg\min_u \; \hat{\Phi}(u, w) \tag{2.75}$$

where $\hat{\Phi}(u, w) = \Phi(u, g_u u + g_w w)$. Assuming $\hat{\Phi}(u, w)$ to be differentiable in u for all $u \in \mathbb{R}$ and $w \in \mathbb{R}$, then by chain rule, we have

$$\nabla_u \hat{\Phi}(u, w) = (2qu + s) + g_u[2r(g_u u + g_w w) + t]. \tag{2.76}$$

To design the controller that steers the system (2.62) to the solution of the optimization problem (2.75), we consider the following gradient descent flow

$$\dot{u} = -k_u \nabla_u \hat{\Phi}(u, w) \tag{2.77}$$

where $k_u > 0$ is a tuning gain to adjust the convergence rate. To avoid disturbance estimation, we can replace $g_u u + g_w w$ in $\nabla_u \hat{\Phi}(u, w)$ by y following the equation (2.72b) to obtain the output-feedback gradient flow

$$\dot{u} = -k_u \nabla_u \hat{\Phi}(u, w) = -k_u G \nabla \Phi(u, y) = -k_u[(2qu + s) + g_u(2ry + t)] \tag{2.78}$$

where $G = \begin{bmatrix} 1 & g_u \end{bmatrix}$.
It is well known that the strict minima of $\hat{\Phi}(u, w)$ are stable equilibria of the gradient flow dynamics (2.78), and that, if the level sets of $\hat{\Phi}(u, w)$ are bounded, then the trajectories of (2.78) converge asymptotically to the set of critical points of $\hat{\Phi}(u, w)$ [47]. Under suitable timescale separation [156, 88], the interconnection of the gradient flow controller (2.78) in feedback with the *stable* dynamic system (2.62), will result in a closed-loop system that is guaranteed to converge to the minimizers of the optimization problem (2.75) in steady-state. Also if the matrix G is sparse, then the sparsity pattern will induce algebraic structures that can be exploited to achieve a distributed implementation of the gradient flow controller [87]. This has indeed been exploited in the class of distributed averaging proportional integral (DAPI) controllers commonly used to achieve distributed economic dispatch in power system load-frequency control [160, 188].

Although the gradient flow control above is designed to solve an unconstrained optimization problem, constraints can be easily incorporated using penalty functions, barrier functions or projection operators [85, 156, 87]. The stability of gradient flow control interconnected with a stable linear time-invariant dynamics has been shown in [156]. This analysis was generalized to non-linear dynamical systems in [88]. Here, the timescale separation needed to guarantee the stable interconnection of gradient flow controllers with a stable non-linear dynamical system is quantified using singular perturbation analysis. [85] incorporated input and output constraints in gradient flow control design using a discrete-time version of projected gradient flows. The inherent robustness of the gradient flow control algorithm is investigated in [46]. It was unsurprisingly found that gradient control algorithms are very robust against model uncertainty and unmeasured disturbances. Application of gradient flow control for autonomous optimization in power systems is discussed in [55]. The optimal load frequency control problem and various economic optimization problems in power systems are solved using gradient flow type controllers in [87].

One of the advantages of gradient flow control is their simplicity both in design and implementation. Also, gradient flow controllers have very minimal model dependence requiring only a knowledge of the input-output sensitivity. As a result, these control algorithms have been shown both in theory [46] and experimentally [169, 173] to have very impressive robustness against model uncertainties and unmeasured disturbances. We summarise the pros and cons of gradient flow controllers below.

Pros of gradient-flow control

- Simple to design and implement.
- Very minimal model knowledge required.
- Very robust against model uncertainty and unmeasured disturbances.
- Can incorporate steady-state constraints.

Cons of gradient-flow control

- Optimizes the steady-state input or output but not both.
- Incorporation of constraints complicates the controller.
- No transient performance guarantees.
- Requires system to be stable or pre-stabilized.

Integrated control and optimization

The feedback optimization strategies presented so far all assume the dynamic system to be stable or pre-stabilized. The focus has therefore been the design of feedback optimization dynamics that do not de-stabilize the otherwise stable open-loop system when connected in feedback with the plant. This design approach is very elegant and practical as it does not require a redesign of the existing controllers already in place. However, a major disadvantage of this approach is that it limits the design scope of the feedback optimization controllers [129]. The implication of this is that important control functionalities such as the optimization of transient performance, the enforcement of constraints during the transients, a robustness based control design and the dynamic stabilization of unstable systems cannot be achieved.

To address these limitations, attempts have been made in [129, 128, 167, 9, 10, 45] to design controllers that simultaneously stabilize the system dynamics while also guaranteeing the optimization dynamics are stable and convergent. Two distinct approaches have been followed with these controllers. In the first approach, the open-loop system dynamics is no longer assumed stable. The stability of closed-loop system is analyzed considering both the optimization dynamics and the open-loop system dynamics. Conditions guaranteeing the stability and convergence of the closed-loop system are then derived by resorting to a robust control analysis. These approaches often obtain stability guarantees that are based on the solution of linear matrix inequalities [87]. An example of this approach is presented in [45].

In the second approach, the optimization dynamics are integrated with the system dynamics and a single controller is designed to simultaneously stabilize both dynamics resulting in a closed-loop system that is both stable and convergent to optima of the optimization problem. Examples can be found in [167, 129, 128, 9, 10]. In [167], the control design is split into three distinct components. The first component estimates the system states from the output measurements. The second component uses the estimated state to compute a drift direction based on an optimization dynamics. The third component designs a feedback controller to drive the system towards the optimal steady-state. The stability and convergence of the closed-loop system is analyzed in the presence of constant disturbances. Sufficient conditions for global exponential stability of the closed-loop system are established based on linear matrix inequalities (LMI). In [128, 129] an internal model approach is used. Here, the optimization dynamics (also termed optimality model in the paper) is augmented with the system dynamics as an additional integral state. Tracking controllers are then designed using the augmented model and the theory of output regulation to obtain feedback optimizing controllers that guarantee a stable and optimizing closed loop system. Unlike in [167], stability guarantees do not rely on the solution of complex LMIs in the nominal case (i.e., assuming no model uncertainty). In [9, 10], the optimization dynamics is used to define a tracking error which is then used in a tracking linear quadratic/model predictive control formulation to regulate the closed-loop system to the optimal steady-state. Unlike the approaches in [167] and [128, 129], the techniques developed in [9, 10] consider transient performance objectives in the control design.

2.6 Electric power systems

Electric power systems are one of the largest and most expensive engineered machines in the world. They are arguably one of the most important infrastructure systems in today's society. Their importance stems from a myriad of reasons of technical and

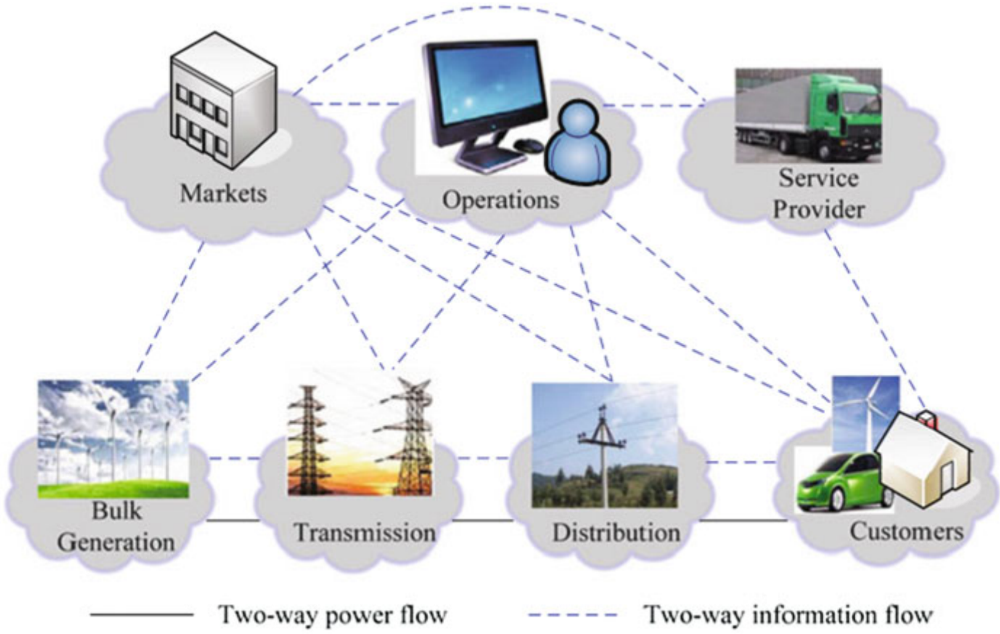

Fig. 2.1 The power system network [2].

socio-economic natures. Technically, power systems are interesting as they deal with electrical energy which requires continuous balancing and cannot be stored efficiently. Socially, power systems provide electricity which has become an essential commodity to the life of everyone in the planet. Economically, the electrical energy provided by power systems are essential to the operation of major industries and firms. For these reasons, power systems have been the subject of scientific enquiry for decades.

Structurally, power systems are complex, interconnected dynamical systems physically made up of generation systems, transmission, and distribution systems, and loads [121] (see Fig. 2.1). Power is produced by generation systems which are mostly synchronous generators with large rotating parts. Transmission and distribution systems transport the electricity from generators over long distance networks to consumers (loads) usually at high voltages [84]. Also, modern power systems have a cyber-physical layer that consists of decision support systems such as sensing, communication and computation that aid in the efficient operation of power systems and support control centres to provide essential services such as power markets, regulation and control [98]. Most power systems consists of networks of independently managed control areas [1]

[1]A control area is defined as a collection of generation, transmission, and loads within metered boundaries for which a utility company integrates resource plans for that area ahead of time, maintains the area's demand-supply power balance, and supports the area's interconnection frequency in real time [20]

interconnected by high voltage, long-distance tie-lines. Operations within a control area are managed in real-time by utilities, often referred to as the Balance Responsible Party (BRP). Also, to guarantee network-wide stability and economic efficiency, an independent/transmission system operator (ISO/TSO) coordinates and manages the interconnection of control areas [98].

Power systems are undergoing a rapid evolution, driven mostly by economic and environmental factors. With increased demand for affordable energy, the monopoly in traditional vertically integrated utilities (VIU) is being dismantled through deregulation and liberalization. Access to the transmission system is now open to all utilities within and outside a control area. This has given consumers the freedom to purchase the most affordable power from generating systems both within and outside their control area. The large amount of random inter-area power flows that occur as a result has made it more challenging for transmission system operators to manage and regulate the power network.

Global concerns about the environment is driving an increased penetration of renewable generation and the electrification of transportation. Due to the variability and uncertainty of renewable generation, and the random charge and discharge cycles of electric vehicles, future power systems will become more inefficient and unstable [166]. This will make it challenging for utilities to reliably and efficiently operate the power system.

On a positive note, power systems are evolving into smart grids with improved sensing, communication and computation capabilities. These advancements will allow the adoption of more advanced algorithms that are needed to deal with the challenges of an evolving power system. Also, smart grids will provide an enabling environment for the deployment of novel technologies such as real-time pricing that are needed to cost-effectively manage a more complex and uncertain power network.

2.7 Review of Frequency Control

A fundamental objective in a power system is the supply of uninterrupted power at a rated quality and the least possible operating cost/dis-utility [84]. Whenever there is an imbalance between the supply and demand of power, the following occurs:

1. the electrical frequency deviates from its rated/nominal value
2. the inter-area power exchange deviates from its scheduled economic value

3. the loading of controllable generators and controllable loads changes from their economically dispatched values

Frequency control is therefore necessary to stabilize the power system and counter the above impacts of power imbalance by [123, 122, 121, 124]:

1. restoring the frequency to its nominal/rated value
2. restoring the inter-area power exchanges to economic values
3. asymptotically stabilize the frequency deviation and minimize inadvertent inter-area power exchanges with good dynamic performances, and
4. minimize the operating costs of power balancing across the interconnected power network, returning the loading of controllable nodes (generators and controllable loads) to their economic values.

To achieve these goals, conventional frequency control is designed and implemented in a layered way, comprising primary, secondary and tertiary control [22]. Here, the objectives of economic optimality and frequency stability are handled separately across different time scales. This temporal decomposition is based on the following important characteristics of traditional power system demand [98, 96, 99, 209]:

1. the aggregate load/demand is decomposable into:
   (a) a predictable slowly-varying hourly demand variation, $P^L[h]$
   (b) a predictable near real-time (5-10 minutes) demand fluctuation, $P^L[min]$ and
   (c) an unpredictable real-time (seconds) demand fluctuations with zero mean, $P^L[s]$.
2. there is a clear timescale separation between each components of the above demand.

In the following paragraphs we present the main components of frequency control in conventional power systems.

2.7.1 Tertiary Frequency Control

Tertiary frequency control comprises the unit commitment (UC) and economic dispatch (ED) operations. The UC determines the economically optimal combination of power generating units to keep online, in order to meet the predicted hourly demand variations ($P^L[h]$). Unit commitment set-points are dispatched once, every 24 hours (daily) and consists of 24 hourly generator on/off selection profiles. Closer to real-time (5-10 minutes), steady-state economic dispatch problems are solved using predictions of the near real-time (5-10 minutes) demand fluctuations, $P^L[min]$, to efficiently allocate the actual power generation between the committed units and the inter-area power exchange schedules between control areas. The ED operation sends load-reference set-points for committed generating units, and tie-line power exchange schedules for each control area at intervals of 5mins to less than an hour.

A fundamental assumption underlying tertiary control operations is a power system in steady state at the predicted demand. This justifies the use of static optimization processes to determine tertiary control set-points as formulated in the multi-area economic dispatch problem (2.79) [205]:

$$\min_{P^{tie},P^m} \quad \Phi(P^m) = \sum_{i\in\mathcal{N}} C_i(P_i^m) \tag{2.79a}$$

$$\text{s.t. } \forall i \in \mathcal{N} : \tag{2.79b}$$

$$P_i^m - P_i^{tie} - P_i^L = \mathbf{0}, \ \forall i \in \mathcal{N}, \tag{2.79c}$$

$$\underline{P}_i^m \leq P_i^m \leq \overline{P}_i^m, \tag{2.79d}$$

$$\underline{P}_i^{tie} \leq P_i^{tie} \leq \overline{P}_i^{tie} \tag{2.79e}$$

where P_i^{tie} is the tie-line power flow in control area i, $P^{tie} = \{P_i^{tie}\}_{\forall i\in\mathcal{N}}$, P_i^m is the aggregate power generation in control area i, $P^m = \{P_i^m\}_{\forall i\in\mathcal{N}}$, P_i^L is the near real-time predicted demand of control area i and $\mathcal{N}$ is the index set of control areas. The capacity constraints of the generators is enforced by the constraints in (2.79d) while (2.79e) enforces the thermal limits on the interconnecting tie-lines. $C_i(P_i^m)$ is the variable cost of generating P_i^m amount of aggregate power by all generators in control area i and it depends on variables such as the fuel cost. $C_i(P_i^m)$ is often approximated by a quadratic function of P_i^m[22].

The solution to the multi-area economic dispatch problem above yields the economically optimal power generation $P_i^{m,*}$ and the tie-line power schedule $P_i^{tie,*}$ to meet the predicted near real-time power demand $P_i^L[min]$ for a pool of control areas, $\mathcal{N}$.$P_i^{m,*}$

is dispatched as the load reference set-point, P_i^c to the participating Governor-Turbine-Generator (GTG) systems in the power network every 5-15 minutes [186].

2.7.2 Primary & Secondary Frequency control

Between economic dispatch intervals, the actual instantaneous demand may deviate from its near real-time predicted value. This could result from unplanned load changes or fluctuations in generation (e.g. from renewable generation, or generation/power line outages). To ensure the matching of generation with demand, and also restore economically optimal operation of the power system in real time, load-frequency control is required. Load-frequency control (comprising primary and secondary control) tracks the set-points from the most recent ED operation while rejecting the disturbances from unpredictable components of the demand in real-time. Primary frequency control provides the first and the fastest automated response to real-time power fluctuations. Primary control acts at a timescale of seconds to adjust the power of generators and controllable demand based on local frequency deviations. The control action in primary control are localised at the generation or demand level and may stabilize the frequency deviation at non zero values owing to its proportional action.
To eliminate the non-zero deviation of frequency from its nominal value and restore the power system to its economically dispatched operating points, secondary frequency control is activated after primary control. Secondary frequency control, also known as Automatic Generation Control (AGC) acts at a timescale of tens-of-seconds to minutes to restore the frequency and tie-line power interchanges to their respective nominal/scheduled values. Traditional secondary control or AGC is centralized with respect to the generators in a control area but area-wise decentralized [225].

Both primary and secondary control are feedback loops with control actions derived from actual system measurements (frequency and tie-line power flow deviation), while tertiary control are feed-forward loops with control derived from demand predictions/estimates. Conventional secondary control (or AGC) actions are computed from measurements of the area control error (ACE). The ACE is a weighted combination of the deviation in frequency, f_i and deviation in scheduled tie-line power flow, P_i^{tie} of each control area, $i \in \mathcal{N}$ i.e.,

$$ACE_i = \beta_i f_i + P_i^{tie} \tag{2.80}$$

where β_i is the frequency bias coefficient of control area i, obtained from numerical studies of the frequency response characteristics of a control area.

Secondary frequency control acts to provide real-time corrections to the load-reference set-points, P_i^c fixed by a previous economic dispatch command in order to account for unplanned disturbances. Secondary frequency control is one of the most important control problems of interconnected power system design and operation, and is becoming even more significant today due to the increasing size, changing structure, emerging renewable energy sources and new uncertainties, environmental constraints, and complexity of modern power systems [29].

2.7.3 Conventional Load-Frequency Control (LFC)

Pioneering ideas in AGC were developed in the 1950s with N. Cohn introducing the ingenious tie-line bias control for meeting the frequency control objectives of interconnected power networks [44]. Tie line bias control works by driving the area control error (ACE), defined by (2.80) of individual control areas to zero. This is conventionally achieved via the proportional integral (PI) control algorithm:

$$u_i = -K_i e_i, \tag{2.81a}$$

$$\dot{e}_i = ACE_i \tag{2.81b}$$

where u_i is the control command given to the generators in control area i that are participating in LFC. The design of the control gain K_i was based on classical frequency domain design techniques such as the Bode and Nyquist plots. Due to its simplicity and decentralized implementation, conventional LFC based on tie-line bias control has been and is still widely adopted in practice. Although simple and easy to implement, conventional LFC algorithms have poor dynamic performance (long overshoots and settling times) [62] especially for constrained power systems with non-linear dynamics and uncertain parameters [190, 121]. Also, classical control techniques which are used to design conventional LFC algorithms cannot effectively deal with multi-variable power systems dynamics. Furthermore, they are mainly designed under nominal conditions and without considering the system constraints which makes these techniques ineffective during a change in operating point.

We summarize the pros and cons of conventional LFC algorithms below.

Pros of conventional LFC

- Simple intuitive design.
- Easy to implement, computationally cheap.
- Decentralized.

Cons of conventional LFC

- Poor dynamic performance (long overshoots and settling times).
- Cannot handle multi-variable power systems dynamics.
- **Constraints** and non-linearities not directly considered.
- Cannot guarantee steady-state economic operation i.e., economic dispatch.

2.7.4 Model-Predictive Load Frequency Control

To improve dynamic performance, modern control techniques were applied to the LFC problem with the linear quadratic regulator (LQR) developed in [62, 79, 123] for achieving dynamically optimal frequency regulation in multi-area power systems. Although LQR showed improved transient performances, and could deal with multi-variable systems dynamics, it was mostly criticised for requiring the complete state measurements, lacking constraint handling capabilities and lacking in robustness. The deficiencies of LQR led to the application of more advanced modern control techniques such as adaptive, robust, predictive and intelligent control approaches. Detailed surveys of these and many other control methodologies to the LFC problem are reviewed in [190, 120].

Recently, the interest in applying model predictive control to the LFC problem has increased dramatically. This is due to the ability of MPC to systematically handle power system constraints, as well as optimize the performance. Furthermore, developments in smart grid has increased the communication and computational capabilities of modern power systems, motivating the application of MPC to the LFC problem. Most MPC based LFC algorithms are designed as tracking MPC controllers (see Section 2.2) with the tracking error defined as the area control error (ACE) as given by (2.80).
In [195, 115, 191, 155, 137, 66, 136, 196, 194], centralized MPC schemes have been applied to the LFC problem. The use of centralized MPC for LFC is justified under the

assumption that system-wide information from all control areas is readily accessible by single control centre using for example, telemetry systems (e.g., Wide Area Measurement Systems (WAMS)). In [195], a centralized MPC algorithm that considers the Generation Rate Constraints (GRC) for multi-area LFC was presented. [115] proposed a centralized MPC-based LFC scheme that achieves nominal and robust stability using state contractive constraints. In [191], the economic cost of generation and short term load predictions are used to formulate an economically efficient centralized MPC algorithm for a single area power system.
In [66], a pioneering application of centralized MPC to a real world Nordic power system is demonstrated in simulation. The algorithm is designed using a simplified model of the real power system with complete operational constraints such as tie-line congestion limits, generation capacity limits and GRC included in the control formulation. Also, cost based tuning of the input weights is used to obtain economically efficient generation control. LFC was achieved using hydro rather than steam turbines. The performance of the MPC algorithm was shown to be better than PID in terms of the frequency response, constraints satisfaction, cost efficiency and disturbance rejection under uncertainty from renewable generation.
A further step towards testing MPC on real world systems was taken by [155] where centralized MPC is applied to the AGC of a 1479-bus high resolution simulation model of the single area Irish power system with significant wind generation, stochastic loads and communication delay. A velocity MPC formulation is used for disturbance rejection with GRC constraints considered. Also the MPC weights are tuned using the generation cost to achieve economically efficient AGC. Effects of significant renewable generation, stochastic loads and control delay are considered. The MPC is shown to outperform PID and minimize generation cost. However, PID shows more robustness to time delays. It is evident from this study that control communication plays a significant role in the successful real-world application of MPC to the LFC problem.

Centralised MPC using system-wide information may not be feasible for large-scale power systems due to increased communication and computational requirements; the need for independent management of control areas by different utilities with information sharing concerns; and the single point of failure associated with relying on one central controller. To overcome these challenges, decentralized MPC (DeMPC) and distributed MPC (DMPC) have been applied to the LFC of multi-area power systems. DeMPC-based LFC schemes were studied in [159, 163, 11, 184]. A key motivation for the use of DeMPC techniques is a weak tie-line interaction allowing the local MPC regulators to be based on local control area information. This significantly reduces the burden of communication and computation.

Load frequency control using DeMPC performs better than PI control but poorer than CMPC and DMPC based approaches [93]. This is because DeMPC uses an incomplete/partial prediction model since information about the subsystem interactions are ignored or rejected as disturbances. Furthermore, guaranteeing the stability of DeMPC based LFC is difficult for systems with strong interactions and the use of restrictive constraints to guarantee closed-loop stability further degrades the control performance [161, 93]. To improve the performance of DeMPC, distributed MPC (DMPC) schemes [216, 142, 161, 139, 143, 92, 138] have been applied for the LFC of power networks. DMPC schemes are based on the exchange of dynamic interaction information between control areas, which is then included in the design of each local MPC for improved performance. Different DMPC schemes have been applied to the LFC problem, differentiated mostly by the information communication requirements and the approach used for solving the MPC problem for each control area.

We summarize the pros and cons of MPC-based LFC algorithms below.

Pros of MPC-based LFC

- Works for multivariable systems.
- Systematic constraint handling capabilities.
- Can optimize a user defined performance objective.

Cons of MPC-based LFC

- Cannot guarantee steady-state economic operation i.e., economic dispatch.
- Computationally more demanding.
- Requires knowledge of system states and disturbances.

A key contribution of this thesis is to develop novel MPC-based LFC algorithms that can guarantee steady-state economic operation under significant variation in the unknown demand and without the need for explicit measurement or estimation of the demand. In the following paragraphs, we present a detailed review of LFC schemes proposed in the literature for achieving economic dispatch (i.e., steady-state economic optimization) in load-frequency control.

2.8 (Steady-state) Economics Optimizing Load Frequency Control

One of the objectives of load frequency control (LFC) is the optimal allocation of generation and network resources to minimize the cost of power balancing [120]. Conventionally, this function is achieved by Economic Dispatch (ED) operating at slower time scales using prediction-based steady-state optimization to obtain the optimal set-points and tie-line schedules tracked by LFC at faster timescales [121, 124]. The economic efficiency of this approach depends on the predictability of demand, absence of multi-timescale dynamics and the efficient coordination between the ED and LFC operations; all of which are becoming increasingly difficult to guarantee with the increased variability and loss of inertia in modern power systems. The integration of ED into LFC is commonly achieved via participation factors calculated using the most recent ED results or cost parameters of the generators participating in LFC [121, 221]. Although participation factors can allocate generation optimally between generators, it is still possible to obtain economically suboptimal LFC operations. This observation has for a long time motivated studies aimed at improving the efficiency of participation factors in ensuring economic power allocation in LFC. In [122], the need to coordinate ED and LFC in the same timescale as the LFC operation was emphasized for power systems with rapidly changing loads. In [12], Optimal Power Flow (OPF) rather than ED calculations are used to compute participation factors which ensure an economic LFC operation within the thermal limits of the transmission lines. More recently, the participation factor approach to coordinating ED and LFC has been improved in [130] to ensure economic and secure LFC operations under uncertainty from loads and renewable generation using the concept of bus dependent participation factors.

Participation factors may be suboptimal in real-time due to the inability to revise tie-line schedules and load-reference setpoints for highly uncertain power networks. This challenge has been taken up by recent frequency control studies [160], leading to the development of several techniques for integrating ED into the frequency regulation problem to ensure real-time economically efficient LFC for highly uncertain, low-inertia power systems.

Most of these techniques are designed to achieve tertiary economic optimization and inter-area coordination, within frequency regulation, with the objective of obtaining optimal allocation of resources across control area boundaries in the time scale of frequency regulation. Detailed reviews of (economically) optimal LFC studies can be found in [160, 120, 165]. A brief review of emerging techniques in the literature for obtaining economically optimal frequency control will now follow.

2.8.1 Modified Hierarchical Control

These techniques retain the timescale separation of hierarchical control with modifications to improve economic efficiency. Common changes include the design of improved interaction between the control layers and/or the remodelling of hierarchical control to include dynamics across multiple timescales. In [224, 225], a hierarchical scheme that unifies ED, AGC and load forecasting while retaining the area-wise decentralized nature of tie-line bias control was proposed. In [61], ED is designed to operate at a slower timescale and interacts with AGC to reschedule the entire generation in a way that minimizes the operating cost. In [97, 223], the concept of minimal regulation was introduced which minimizes the operating cost of AGC by rescheduling the entire system generation and separating tie-line control from frequency regulation. A more recent study in [209], redesigned conventional hierarchical ED to include information about the real time imbalance of power (i.e. frequency deviation) in the steady-state ED operations. The control design claimed to improve the economic efficiency of frequency control, compared to conventional ED. However, power system dynamics were ignored, which can adversely affect performance. In [19], the primary, secondary and tertiary control layers were remodelled as dynamical systems with a so-called transactive control architecture designed to coordinate tertiary level market transactions with the primary and secondary frequency control using price-like signals. Reduced LFC cost and improved social welfare was claimed compared to conventional hierarchical control. In [37] the joint problem of ED and frequency regulation was studied and conditions under which they can be decomposed into independent control layers without loss of optimality are derived. It was also shown that for ED based on a DC power flow model, the global problem of joint ED and frequency regulation is decomposable if an estimate of the difference between the average marginal cost of power generation in the ED and frequency regulation time-scales is available. Other notable modifications of hierarchical control and extensions of the methods discussed can be found in [193, 202, 54, 1].

Controllers based on a modified hierarchical design still rely on timescale separation assumptions and therefore require steady states between control layers. Consequently, real-time economic optimality under variable power fluctuations is still not guaranteed, due to the possibility of a loss of time scale separation for large and fast disturbances.

2.8.2 Optimal Load Frequency Control

Optimal load frequency control techniques dynamically stabilize the power system at equilibrium points corresponding to the (unknown) solutions of the economic dispatch problem. These controllers are often designed as online implementations of the optimization algorithms for solving the economic dispatch problem. The economic dispatch problem is often formulated to encapsulate the control design goals such as economic optimality, frequency regulation and constraint/congestion management. A review of optimization algorithms being implemented as dynamic control laws have been thoroughly discussed in a previous section. In [105], optimal load frequency control is achieved by the feedback implementation of dual decomposition algorithms for solving a defined economic dispatch problem. Here, the Lagrange multiplier (interpreted as dynamic nodal prices in the paper) were directly used as control signals to achieve economically optimal frequency control with tie-line congestion management. In [134], conventional Automatic Generation Control (AGC) was modified using a partial primal-dual algorithm to economically dispatch generation in secondary frequency control. By implementing a significant part of the controller via the power system dynamics, very little modification was required to the conventional integral-control-based AGC. Insights obtained from [134] were utilized in [157] to develop a practically oriented approach to achieving ED in conventional AGC. In [147], a more complicated but similar design approach to [58] was adopted in achieving economically optimal primary and secondary frequency control with tie-line congestion management using demand response. The work in [147] was extended to include both generation and demand response loads in [230]. In [219], primal-dual feedback optimizing controllers that achieve optimal load frequency control in power networks while also guaranteeing constraint satisfaction both in steady-state and during the transient are presented. However, the controller made use of saturation to enforce constraints which can lead to poor control performance and integrator windup problems.

An alternative to primal-dual control that has been widely used to achieve optimal load frequency control is the distributed averaging proportional integral (DAPI) controller. This controller is a variant of gradient-based feedback optimizing control for sparse systems. They can therefore be easily implemented as distributed load frequency controllers. DAPI controllers work by exchanging marginal (generation or demand response) cost information between traditional proportional-integral (PI) secondary frequency controllers in order to achieve economically optimal frequency control [160]. Economic optimality of DAPI is due to convergence via consensus to an equal incremental control cost, satisfying the unconstrained ED criteria in real-time. DAPI controllers although optimizing and computationally efficient can only guarantee asymptotic constraint satisfaction with the possibility of violating the constraints in the transient phase. They have also been shown to exhibit a trade-off between performance and robustness to communication and measurement noise [220, 189]. DAPI control has been applied to the secondary LFC of bulk power systems using simplified [229, 188] and complex non-linear models [203]. Micro grid secondary control applications have been shown in [56] for simplified models and [50] with more complex micro grid models. Other distributed averaging based methods for economically optimal frequency control include, a discrete-time consensus based approach in [164] and an approach based on output regulation theory in [210] for time-varying disturbances.

A major limitation of the above optimal load frequency controllers is the inability to guarantee performance and constraint satisfaction during transient operation. This can be detrimental to power systems with sustained operation away from the optimal steady states due to the occurrence of persistent and time varying disturbances. Partly addressing this limitation, optimal load frequency control algorithms are presented in [55, 89, 87] which enforce power system constraints both in steady-state and during transients using projected gradient descent algorithms. In [49], a projected primal-dual algorithm is used to develop feedback optimizing algorithms that solve the optimal power flow problem for AC power systems. Although optimal load frequency control algorithms with projection can achieve both transient and steady-state feasibility, they do not yield optimal transient performance as the power system is still assumed to be stable or pre-stabilized offline.

2.8.3 Economics Optimizing Model Predictive LFC

For most MPC based LFC schemes, economic considerations are treated as secondary objectives and are mostly achieved indirectly via economics based tuning of the MPC controller [155, 66, 191, 196]. This approach to secondary frequency control is justified for small power fluctuations (a common assumption in most LFC studies). However, when the disturbances are persistent, time-varying and large, the need to include economic objectives in the design of LFC schemes becomes vital. This is due to the significant deviation from previously determined optimal economic loading of generators, as well as the frequentness of LFC from expensive fast-ramping sources of generation. The increase in renewable generation and inter-area power transactions in future power grids will most likely result in power systems with large, time-varying and persistent disturbances, operating away from their predetermined economic equilibria for most of the time. Therefore, there is need for LFC schemes to restore economic operation rather than assume a power system close to its economic equilibrium.

An early decentralized MPC scheme with a cost minimization objective for LFC was proposed in [183]. Economic operation was achieved through the minimization of generator manoeuvring and reversals during LFC, rather than a direct use of generator cost functions in choosing control actions. A reduction in the LFC cost was claimed for a 3-area power system compared to the use of conventional MPC. The controller however is an equilibrium stabilizing MPC, with economic improvements achieved using a heuristic economic logic rather than the direct and systematic inclusion of generator cost information in the MPC design. Economic reallocation of generation and tie-line power flows is therefore not possible with the algorithm. In [201] an economic optimizing MPC (EMPC) which directly incorporates the LFC cost in the objective of the controller is proposed. A trade off between economic efficiency and set-point stabilization is achieved by the MPC via a multi-objective function definition. The economic LFC objective of the MPC scheme is defined to include the cost of wear and tear on generators due to the input rate of change; the incremental cost of power generation; and the cost of large frequency deviations which trigger uneconomical critical actions such as load shedding. The superior economic and dynamic performance of the EMPC scheme over a set-point stabilizing MPC was shown in simulation. The study was however conducted for a single area power system and therefore does not consider the economic benefits, nor the dynamic influence of inter-area interconnections on control performance. Also, operational constraints such as generator capacity limits and Generator Rate Constraints (GRC) are not directly considered in the control formulation.

In deregulated power networks, separately managed control areas may compete, rather than cooperate in order to optimize a local as opposed to global economic objective. Therefore, [57] proposed a distributed economic MPC (DEMPC) scheme for competing multi-area power networks. The controller was designed to optimize local economic objectives while achieving convergence to a desired global network steady-state e.g zero frequency deviation. Cooperation was required to achieve global network control objectives such as managing tie-line power flows. An iterative dual-decomposition optimization technique was adopted for network coordination. Also, the sharing of confidential economic information is not required by the algorithm. Although the control performance is locally optimal and globally stabilizing, the network-wide

economic performance may be suboptimal due to lack of economic cooperation.
A recent application of distributed EMPC to the LFC of multi-area power systems was demonstrated by [113, 112] where the sum of local and neighbouring subsystem's economic objectives are optimized using an iterative dual decomposition technique based on Alternating Direction Method of Multipliers (ADMM) algorithm. MPC regulators exchange local state predictions and control iterates with neighbouring controllers in order to solve the distributed optimization problem. Premature termination of the algorithm is possible, to meet real-time requirements with performance guarantees. Also, tie-line coupling constraints and generation capacity limits are considered. Improved economic performance compared to a state-of-the-art approach for achieving economically optimal LFC was shown in simulation. The controller however is iterative and based on the slow ADMM algorithm. It may therefore not be feasible for power systems with limited communication capabilities.

The previously reviewed economics optimizing MPC schemes for LFC are based on centralized [201], decentralized [183, 57] or iterative distributed [113] MPC approaches and may therefore be infeasible for power systems of large scale with limited communication capabilities. Economics optimizing model predictive LFC schemes that are computationally efficient, scalable and have reduced communication requirements have not been reported in the literature, yet such schemes are desired for the economic operation of future interconnected power systems with stronger inter-area coupling due to the increased need for cooperation to reject larger, persistent and uncertain disturbances.

2.9 Summary and Conclusion

This chapter presented theoretical background in optimization and control, and a detailed review of the current literature on feedback optimization, economics optimizing MPC, and optimal load-frequency control. From this review, it was found that conventional hierarchical control cannot guarantee a stable and economically optimal steady-state operation under uncertainty and time-varying disturbances. Alternatives to conventional hierarchical control have also been discussed. It was observed that most of these alternatives have limitations such as poor dynamic and constraint handling capabilities which can be detrimental for some important applications problems such as power system frequency control. Although MPC algorithms exists for achieving optimal steady-state control, most MPC schemes are computationally expensive and rely on disturbance estimation. Also, the idea of feedback optimizing control has been thoroughly discussed with a detailed review of existing algorithms. Feedback optimization has been found to be computationally efficient, robust and not reliant on disturbance measurements. However, most schemes for feedback optimization unlike MPC lack transient performance guarantees and only enforce constraints asymptotically.

From this review, it is clear that the strength of MPC is the weakness of feedback optimization and vice versa. Therefore, there is a strong incentive to combine the design philosophy of both feedback optimization and model predictive control to achieve the advantages of both while also addressing their combined limitations. In the following chapters of this thesis, we will do exactly that by combining the design philosophy of both control schemes to formulate and design MPC algorithms that achieve feedback optimization in closed-loop.

Chapter 3

The Feedback Optimizing Model Predictive Control Problem

3.1 Introduction

In this chapter, we expand the design scope of traditional feedback optimization to include transient performance objectives and recursive constraint satisfaction. To achieve this goal, the feedback optimization problem will be formulated using a Model Predictive Control (MPC) framework. This will allow additional performance objectives such as dynamic optimization and constraint enforcement during transients to be achieved. Towards this objective, the feedback optimization problem will be posed as a feedback optimizing model predictive control (FOMPC) problem. The benefit of this new way of posing the feedback optimization problem is that the advantages of a model predictive control based design are carried over to the feedback optimization problem. This however comes at the added expense of increased model dependence and increased computational complexity. However, one can reduce the impact of this trade-off by designing robust and computationally efficient FOMPC algorithms.
We begin the chapter by characterizing the feasible forced equilibrium set of the linear system and defining the steady-state optimization problem. We then define the FOMPC problem and examine the necessary conditions for optimality of the steady-state optimization problem. Defining the steady-state optimality error as the residual of the Karush-Kuhn-Tucker optimality conditions for the steady-state optimization, we show that the FOMPC problem can be reformulated as a generalized tracking model predictive control problem, allowing conventional tracking MPC algorithms to be adapted to solve the FOMPC problem.

3.2 Dynamical System

Consider a constrained discrete-time uncertain linear time-invariant system described by

$$\mathscr{P}(\delta)\colon \begin{cases} x(k+1) = A(\delta)x(k) + B(\delta)u(k) + Ew(k), \\ y(k) = Cx(k), \\ (u(k), y(k)) \in \mathbb{Z} := \mathbb{U} \times \mathbb{Y}, \end{cases} \tag{3.1}$$

where

- $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{U} \subseteq \mathbb{R}^{n_u}$, and $w(k) \in \mathbb{W} \subseteq \mathbb{R}^{n_w}$ are the state, input, and additive uncertainty (disturbance) vectors respectively. The variable $y(k) \in \mathbb{Y} \subseteq \mathbb{R}^{n_y}$ represents the selection of measurable states to be optimized and will hereafter be referred to as the output.

- $A(\delta) \in \mathbb{R}^{n_x \times n_x}$, $B(\delta) \in \mathbb{R}^{n_x \times n_u}$, $E \in \mathbb{R}^{n_x \times n_w}$, $C \in \mathbb{R}^{n_y \times n_x}$ are the uncertain system coefficient matrices.

- $\delta = (\delta_1, \ldots, \delta_i, \ldots, \delta_l) \in \mathbb{R}^l$ is a time-invariant (parametric) uncertainty vector which takes values from the compact set Δ_l.

The sets $\mathbb{U}$, $\mathbb{Y}$, and $\mathbb{W}$ are compact and convex sets of constraints on the input, output and exogenous disturbance vectors respectively.
Consider the system represented by (3.1) and let the following assumptions be satisfied.

Assumption 1 (Basic assumptions on the system)**.**

1. *The system* (3.1) *is reachable and observable for every realization of* δ*.*

2. *The state* $x(k)$ *is measurable at every sampling instant.*

3. $n_y \leq n_u$.

3.2.1 The steady-state optimization problem

Let the steady-state value of the unknown disturbance at time k be given by $w(k \to \infty) = \bar{w}$. Then from the system dynamics (3.1), a controlled equilibrium point $(\bar{x}, \bar{u})$

of the system satisfies

$$\begin{bmatrix} I_{n_x} - A(\delta) & -B(\delta) \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{u} \end{bmatrix} = E\bar{w}, \tag{3.2a}$$

$$\bar{y} = \begin{bmatrix} C & \mathbf{0}_{n_y \times n_u} \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{u} \end{bmatrix}. \tag{3.2b}$$

The steady-state input–output map can then be expressed as the affine function,

$$\bar{y} = G_u(\delta)\bar{u} + \tilde{y}(\delta, \bar{w}) \tag{3.3}$$

where $G_u(\delta)$ is the steady-state gain of the system (3.1) assumed finite (as steady-state optimization will be meaningless otherwise), and $\tilde{y}(\delta, \bar{w})$ is an (unknown) offset vector. We denote the set of controlled equilibria, parametrized by the disturbance $\bar{w}$ and the uncertainty δ as

$$\mathbb{F}(\delta, \bar{w}) \coloneqq \Big\{(\bar{x}, \bar{u}) \colon \ \text{(3.2a), (3.2b) are satisfied}\Big\} = \Big\{\bar{z} \colon G(\delta)\bar{z} = \tilde{y}(\delta, \bar{w})\Big\}, \tag{3.4}$$

where

$$G(\delta) = \begin{bmatrix} -G_u(\delta) & I_{n_y} \end{bmatrix}, \quad \tilde{y}(\delta, \bar{w}) = C(I_{n_x} - A(\delta))^{-1}E\bar{w}, \quad \text{and} \quad z = \begin{bmatrix} u \\ y \end{bmatrix}. \tag{3.5}$$

Lemma 3.2.1. *For the linear system* (3.1)*,*

$$G_u(\delta) = \begin{bmatrix} C & \mathbf{0}_{n_y \times n_u} \end{bmatrix} Z(\delta) \tag{3.6}$$

where $Z(\delta) \in \mathbb{R}^{(n_x+n_u) \times n_u}$ is the matrix whose column vectors are the unique basis for $\mathcal{N}\Big(\begin{bmatrix} I_{n_x} - A(\delta) & -B(\delta) \end{bmatrix}\Big)$ obtained from the reduced row echelon form.

Proof. Assuming $\begin{bmatrix} \bar{x}_0 \\ \bar{u}_0 \end{bmatrix}$ is a particular solution of (3.2a), then

$$\begin{bmatrix} I_{n_x} - A(\delta) & -B(\delta) \end{bmatrix} \begin{bmatrix} \bar{x}_0 \\ \bar{u}_0 \end{bmatrix} = E\bar{w}, \tag{3.7}$$

and (3.2a) minus (3.7) yields,

$$\begin{bmatrix} I_{n_x} - A(\delta) & -B(\delta) \end{bmatrix} \left(\begin{bmatrix} \bar{x} \\ \bar{u} \end{bmatrix} - \begin{bmatrix} \bar{x}_0 \\ \bar{u}_0 \end{bmatrix} \right) = \mathbf{0}. \tag{3.8}$$

$$\therefore \begin{bmatrix} \bar{x} \\ \bar{u} \end{bmatrix} = \begin{bmatrix} \bar{x}_0 \\ \bar{u}_0 \end{bmatrix} + Z(\delta)v \tag{3.9}$$

where $v \in \mathbb{R}^{n_u}$ is the unique coefficient vector for the rational basis of

$$\mathcal{N}\Big(\begin{bmatrix} I_{n_x} - A(\delta) & -B(\delta) \end{bmatrix} \Big). \tag{3.10}$$

Hence,

$$\bar{y} = G_v(\delta)v + \tilde{y}(\delta, \bar{w}) \tag{3.11}$$

where,

$$G_v(\delta) = \begin{bmatrix} C & \mathbf{0}_{n_y \times n_u} \end{bmatrix} Z(\delta) \text{ and } \tilde{y}(\delta, \bar{w}) = \begin{bmatrix} C & \mathbf{0}_{n_y \times n_u} \end{bmatrix} \begin{bmatrix} \bar{x}_0 \\ \bar{u}_0 \end{bmatrix}. \tag{3.12}$$

Because equation (3.2a) is assumed consistent (i.e. always admits a solution), there exist a unique v for every u hence,

$$G_v(\delta) = G_u(\delta). \tag{3.13}$$

□

Remark 5 (Invertible dynamics)**.** *If* $(I_{n_x} - A(\delta))$ *is non-singular, then* $G_u(\delta) = C(I_{n_x} - A(\delta))^{-1}B(\delta)$.

The steady-state optimization problem for the system involves minimization of a stationary economic performance objective $\Phi(\cdot)$ subject to the constraint sets $\mathbb{F}(\delta, \bar{w})$ and $\mathbb{Z}$. We define the problem as

$$\mathcal{RTO}(\delta, \bar{w})\colon \quad \bar{z}^*(\delta, \bar{w}) = \arg\min_{\bar{z}} \Big\{ \Phi(\bar{z})\colon \bar{z} \in \mathbb{F}(\delta, \bar{w}),\ \bar{z} \in \mathbb{Z} \Big\}, \tag{3.14}$$

where $\Phi\colon \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}$ is continuous.

The $\mathcal{RTO}(\delta, \bar{w})$ problem is a static optimization problem whose solution varies according to the uncertain parameters δ and $\bar{w}$. Therefore, solving $\mathcal{RTO}(\delta, \bar{w})$ explicitly with a nominal (i.e. assumed or estimated) value of δ and estimates of the disturbance $\bar{w}$, may generate setpoints ($\bar{z}^*$) which when tracked by a dynamic feedback controller converges to a sub-optimal equilibrium with steady-state errors [65]. Also, to guarantee closed-loop stability, the setpoints ($\bar{z}^*$) must be updated at a timescale slower than the system dynamics, consequently resulting in a slow response to set-point changes [37] .

In the remainder of this chapter, we propose a general framework that allows the $\mathcal{RTO}(\delta, \bar{w})$ problem to be solved in closed-loop while simultaneously tracking the optimal steady-state equilibrium for changing disturbances $\bar{w}$, with an inherent (quantifiable) robustness to the uncertainty δ. Rather than solving $\mathcal{RTO}(\delta, \bar{w})$ numerically, we adopt a feedback-based optimization approach [45, 23, 128, 156] where the $\mathcal{RTO}(\delta, \bar{w})$ problem is implicitly solved, without knowledge or estimates of the disturbance $\bar{w}$ and with inherent robustness to uncertainty in δ. To also regulate the system in an optimal and admissible way with respect to a transient performance criterion and constraints, we adopt an optimal control approach where the $\mathcal{RTO}(\delta, \bar{w})$ problem is solved implicitly via the feedback loop of the optimal controller.
To develop our control framework, the following assumptions are made about the $\mathcal{RTO}(\delta, \bar{w})$ problem.

Assumption 2 (Properties of the steady-state problem)**.**

1. *The cost function, $\Phi(\cdot)$, is differentiable and strictly convex.*

2. *The sets of admissible uncertainties, Δ_l and $\mathbb{W}$, are such that, for all $\bar{w} \in \mathbb{W}$, and $\delta \in \Delta_l$*

   (a) the set $\mathbb{F}(\delta, \bar{w})$ has a non-empty relative interior;

   (b) the minimizer $\bar{z}^*(\delta, \bar{w})$ is unique.

   (c) the minimizer $\bar{z}^*(\delta, \bar{w})$ exists.

Remark 6 (**About the assumptions**)**.** *1. Assumption 2(2a) is required for Slater's constraint qualification to be satisfied implying that strong duality holds for the optimization problem. This is a requirement for the KKT optimality conditions to be applicable.*

2. The strict convexity of $\Phi(\cdot)$, ensures the satisfaction of Assumption 2(2b) [33]. For a quadratic program, strict convexity is easily achieved by imposing positive definiteness on the quadratic cost.

3. Finally, Assumption 2(2c) although strong is necessary to ensure the dynamic system is driven to a unique steady-state equilibrium. One way to ensure satisfaction of Assumption 2(2c) is to assume the sets $\mathbb{F}(\delta, \bar{w})$ and $\mathbb{Z}$ are both compact *sets [100]. Also, choosing the cost function, $\Phi(\cdot)$ to be coercive (see Definition 1) can guarantee existence of a minimizer.*

Please refer to Chapter 2 of [100] for details on existence and uniqueness conditions of optimization problems.

Definition 1 (Coercive function)**.** *The continuous cost function, $\Phi(\bar{z})$ defined on all of $\mathbb{R}^{n_u+n_y}$ is coercive if*

$$\lim_{\|z\| \to \infty} \Phi(\bar{z}) = +\infty. \tag{3.15}$$

That is for any constant $M > 0$ there exists a constant $R_M > 0$ such that $\|\Phi(\bar{z})\| > M$ whenever $\|z\| > R_M$.

3.3 The Feedback Optimizing Model Predictive Control Problem

The main objective of feedback optimizing model predictive control (FOMPC) is to regulate the inputs and/or outputs of a disturbed linear time-invariant system to an equilibrium point that is the solution to a steady-state optimization problem. This regulation should be achieved without knowledge of the optimal steady-state set-points or explicit solution of the steady-state optimization problem, while also guaranteeing optimal dynamic performance between steady-states. We define the FOMPC problem precisely as follows.

Problem 2 (The FOMPC Problem)**.** *Design for the linear discrete-time uncertain system* (3.1) *a state feedback control law*

$$u(k) = \kappa_N(x(k), u(k-1)) \tag{3.16}$$

obtained from the solution to an optimal control problem, such that for any admissible $\bar{w} \in \mathbb{W}$ and $\delta \in \Delta$:

1. *For all feasible $x(0) = x_0$, the point $\bar{z}^*(\delta, \bar{w})$ is an asymptotically stable equilibrium for the closed-loop system, satisfying*

$$\lim_{k \to \infty} (u(k), y(k)) = \bar{z}^*(\delta, \bar{w}). \tag{3.17}$$

2. *The feedback policy $\kappa_N(\cdot, \cdot)$ minimizes a transient performance criterion.*

3. The constraints $(u(k), y(k)) \in \mathbb{Z}$ are satisfied at all times.

Remark 7 (Disturbance assumptions)**.** *Problem 2 is a generalized formulation of the FOMPC problem with the only restriction on the disturbance being its boundedness to the set $\mathbb{W}$ i.e., $\bar{w} \in \mathbb{W}$. Developing MPC solutions for this generalized problem might be very challenging and sometimes additional simplifying assumptions on the disturbance are necessary to obtain a more tractable control problem. In the subsequent chapters of this thesis, the disturbance will be assumed to be constant (i.e. slowly varying) in addition to being bounded, allowing the design of MPC solutions to solve the FOMPC problem above.*

3.4 Conditions for Steady-state Optimality

To design the FOMPC law that solves the optimization problem $\mathcal{RTO}$ in feedback, we first examine the necessary conditions for optimality of the problem.

3.4.1 KKT Optimality Conditions

To handle the inequality constraints, we reformulate the $\mathcal{RTO}$ problem in (3.14) as the equality constrained problem,

$$\mathcal{RTO}_p(\delta, \bar{w}): \quad \bar{z}_p^*(\delta, \bar{w}) = \arg\min_{\bar{z}} \left\{ \tilde{\Phi}(\bar{z}) \colon \bar{z} \in \mathbb{F}(\delta, \bar{w}) \right\}, \tag{3.18}$$

where $\tilde{\Phi} : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is a modified performance index defined as

$$\tilde{\Phi}(\bar{z}) = \Phi(\bar{z}) + p(\bar{z}). \tag{3.19}$$

The function $p(\bar{z}) : \mathbb{R}^{n_c} \to \mathbb{R}$, where n_c is the total number of output and input inequalities, adds a penalty to the cost function $\Phi(\bar{z})$ [141] on violation of the constraints that define the set $\mathbb{Z}$. We define a penalty function $p(\bar{z})$ as follows [141],

Definition 2 (Penalty function)**.** *A function $p(\bar{z}) : \mathbb{R}^{n_c} \to \mathbb{R}$ is a penalty function for $\mathcal{RTO}(\delta, \bar{w})$ if:*

1. $p(\bar{z}) = 0$, iff $\bar{z} \in int(\mathbb{Z})$, and

2. $p(\bar{z}) > 0$, $\forall \bar{z} \notin int(\mathbb{Z})$.

A penalty function for the inequality constraints $\mathbb{Z} := \mathbb{U} \times \mathbb{Y} := \left\{\bar{z} \colon g(\bar{z}) \leq \mathbf{0}_{n_c}\right\}$ that is guaranteed to be differentiable is the Courant-Beltrami penalty [43],

$$p(\bar{z}) = \mathbf{max}\Big(\mathbf{0}_{n_c}, g(\bar{z})^\top \Gamma g(\bar{z})\Big) = \frac{1}{2} g^+(\bar{z})^\top \Gamma g^+(\bar{z}) \tag{3.20}$$

where the constant

$$\Gamma = \Gamma^\top = \begin{bmatrix} \Gamma_u & \mathbf{0} \\ \mathbf{0} & \Gamma_y \end{bmatrix} \in \mathbb{R}^{n_c \times n_c} \succ \mathbf{0} \text{ is a penalty matrix,} \tag{3.21a}$$

$$g^+(\bar{z}) = \mathbf{max}\Big(\mathbf{0}_{n_c}, g(\bar{z})\Big) = \begin{bmatrix} \max(0, g_1(\bar{z})) \\ \vdots \\ \max(0, g_{n_c}(\bar{z})) \end{bmatrix}, \text{ and} \tag{3.21b}$$

$$\max(0, g_i(\bar{z})) = \begin{cases} 0 & g_i(\bar{z}) \leq 0, \\ g_i(\bar{z}) & g_i(\bar{z}) > 0. \end{cases} \tag{3.21c}$$

Remark 8 (Exact Penalty)**.** *The Courant-Beltrami penalty function is not exact, therefore the solution $\bar{z}_p^*$ to the equality constrained optimization problem $\mathcal{RTO}_p$ may not be exactly equal to the solution $\bar{z}^*$ to the inequality constrained problem $\mathcal{RTO}$ (that is the true solution). However it is well known that $\bar{z}_p^*$ will be a good approximation to the actual solution, $\bar{z}^*$ as the value of Γ increases [43].*

Remark 9 (Hard constraints)**.** *The non-exact penalty method only guarantees asymptotic constraint satisfaction and therefore cannot handle hard inequality constraints which must be enforced at all times. We will rely on model predictive control to enforce these hard constraints in the transient phase of the FOMPC controller.*

Remark 10 (Differentiability of the Courant-Beltrami penalty)**.** *Although* $\max(\mathbf{0}_{n_c}, g(\bar{z}))$ *is not differentiable, the Courant-Beltrami penalty function has a continuous gradient because*

$$\mathbf{max}\Big(\mathbf{0}_{n_c}, g(\bar{z})^\top \Gamma g(\bar{z})\Big) \tag{3.22}$$

has a continuous first derivative: at $\mathbf{0}_{n_c}$, the derivative is $\mathbf{0}_{n_c}$ from both sides. In general the gradient of $\nabla p(\bar{z})$ can be obtained using backward chain rule [214] as

$$\nabla p(\bar{z}) = \nabla g(\bar{z}) \Gamma g^+(\bar{z}). \tag{3.23}$$

By virtue of Assumption 2 and the properties of $p(\bar{z})$, problem $\mathcal{RTO}_p(\delta, \bar{w})$ in (3.18) is a convex optimization problem and strong duality holds [33]. Therefore, the Karush–Kuhn–Tucker (KKT) conditions are necessary and sufficient for its optimality. We obtain the KKT conditions for $\mathcal{RTO}_p$ by forming the corresponding Lagrangian,

$$\mathcal{L}(\bar{z}, \lambda) := \tilde{\Phi}(\bar{z}) + \lambda^\top \Big(G(\delta)\bar{z} - y(\delta, \bar{w})\Big) \tag{3.24}$$

where λ is a multiplier of appropriate dimension. The corresponding KKT optimality conditions are

$$\nabla \mathcal{L}(\bar{z}, \lambda) = \begin{bmatrix} \nabla \tilde{\Phi}(\bar{z}) + G(\delta)^\top \lambda \\ G(\delta)\bar{z} - y(\delta, \bar{w}) \end{bmatrix} = \mathbf{0}_{n_u + 2n_y}. \tag{3.25}$$

3.4.2 Subspace Formulation of the KKT Conditions

By solving the KKT system of equation (3.25), we obtain the optimal solution to the problem $\mathcal{RTO}_p$. However, it is impossible to obtain precisely an explicit solution $(\bar{z}^*, \lambda^*)$ to (3.25) due to the uncertain terms $G(\delta)$ and $y(\delta, \bar{w})$. The goal therefore is to obtain optimality conditions independent of these uncertain terms. We achieve this by expressing (3.25) in the following equivalent subspace form [24]:

$$\nabla \mathcal{L}(\bar{z}, \lambda) = \mathbf{0}_{n_u + 2n_y} \iff \begin{bmatrix} \nabla \tilde{\Phi}(\bar{z}) \in \mathcal{R}\Big(G(\delta)^\top\Big) \\ G(\delta)\bar{z} = y(\delta, \bar{w}) \end{bmatrix}. \tag{3.26}$$

By a fundamental theorem of linear algebra, $\mathcal{R}\Big(G(\delta)^\top\Big) = \mathcal{N}\Big(G(\delta)\Big)^\perp$, and therefore

$$\nabla \tilde{\Phi}(\bar{z}) \in \mathcal{R}\Big(G(\delta)^\top\Big) \iff \nabla \tilde{\Phi}(\bar{z}) \in \mathcal{N}\Big(G(\delta)\Big)^\perp. \tag{3.27}$$

Therefore, let $\tilde{G}(\delta)$ be any full-rank matrix such that,

$$\tilde{G}(\delta) G(\delta)^\top = \mathbf{0} \quad \text{or} \quad \mathcal{R}\Big(\tilde{G}(\delta)^\top\Big) = \mathcal{N}\Big(G(\delta)\Big) \quad \text{for all } \delta \in \Delta. \tag{3.28}$$

Remark 11. *For the linear model* (3.1) *with the (finite) steady-state input-output map* (3.3)*, the matrix*

$$\tilde{G}(\delta) = \Big[\Big(G_u(\delta)^\top\Big)^\dagger \quad I_p\Big] \tag{3.29}$$

satisfies (3.28)*.*

With this, the KKT optimality condition (3.25) becomes

$$\nabla \mathcal{L}(\bar{z}, \lambda) = \begin{bmatrix} \tilde{G}(\delta)\nabla\tilde{\Phi}(\bar{z}) \\ G(\delta)\bar{z} - y(\delta, \bar{w}) \end{bmatrix} = \mathbf{0}_{n_u+2n_y}. \tag{3.30}$$

It follows that $\bar{z}$ is optimal with respect to problem (3.18) if, and only if, it satisfies (3.30). This establishes the following result, which—similar to [128]—allows the steady-state equilibrium optimization problem to be posed as a stabilization problem.

Proposition 1. *Part (1) of the FOMPC problem is solved if, from any feasible initial state x_0, any uncertainty $\delta \in \Delta_l$ and any admissible disturbance $\bar{w} \in \mathbb{W}$, the control law*

$$u(k) = \kappa_N(x(k), u(k-1)) \tag{3.31}$$

is such that $z(k) = (u(k), y(k))$:

1. *is regulated to a steady-state equilibrium,* ***and,***
2. *satisfies* $\lim_{k\to\infty} \left\{ e(k) := \tilde{G}(\delta)\nabla\tilde{\Phi}(z(k)) \right\} = \mathbf{0}_{n_u+n_y}$.

Proof. Condition (1) is satisfied if and only if $G(\delta)\bar{z} = y(\delta, \bar{w})$, which is necessary and sufficient for equilibrium. Condition (2) implies, and is implied by, the KKT conditions (3.30) being met in the limit, which is necessary and sufficient for optimality. □

3.5 Main Result

3.5.1 Reduction of FOMPC to Generalized Tracking MPC

Given Proposition 1, we define the following generalized tracking MPC (GTMPC) problem.

Problem 3 (Generalized Tracking Model Predictive Control (GTMPC) Problem)**.** *For the linear discrete-time uncertain system* (3.1)*, design if possible a state feedback controller of the form* (3.31) *such that for all $\bar{w} \in \mathbb{W}$ and $\delta \in \Delta_l$, the closed-loop system has the following properties:*

1. *For all feasible $x(0)$, the closed-loop dynamics*

$$x(k+1) = A(\delta)x(k) + B(\delta)\kappa_N(x(k), u(k-1)) + Ew(k) \tag{3.32}$$

are asymptotically stable with respect to the (unknown) steady-state optimizers of (3.14)*.*

2. *All trajectories of the closed loop system are such that*

$$\lim_{k\to\infty} e(k) = \mathbf{0} \text{ and } \Big(u(k), y(k)\Big) \in \mathbb{Z} \quad \forall k \geq 0 \tag{3.33}$$

where $e(k)$ is a steady-state tracking error.

3. *The feedback policy $\kappa_N(\cdot,\cdot)$ minimizes a transient performance criterion.*

With the GTMPC problem defined, we now state our main result.

Theorem 3.5.1 (Reduction of FOMPC to GTMPC)**.** *A solution to the FOMPC problem exists for $\mathcal{RTO}$ if the GTMPC problem with the tracking error*

$$e(k) := \tilde{G}(\delta)\nabla\tilde{\Phi}(z(k)) \tag{3.34}$$

has a solution.

Proof. Existence of a solution to Problem 3 (i.e., the GTMPC problem) implies that a control law

$$u(k) = \kappa_N(x(k), u(k-1)) \tag{3.35}$$

can be found such that, as $k \to \infty$,

$$x(k) \to \bar{x}, \quad y(k) \to \bar{y}, \quad u(k) \to \bar{u}, \tag{3.36}$$

and

$$\begin{aligned} \bar{x} &= A(\delta)\bar{x} + B(\delta)\bar{u} + E\bar{w}, \\ \bar{y} &= C\bar{x}, \\ \tilde{G(\delta)}\nabla\tilde{\Phi}(z(k)) &= \mathbf{0}, \end{aligned} \tag{3.37}$$

where $\bar{x}$, $\bar{u}$, $\bar{y}$ are the steady-state values of the respective variables.
Because the steady-state equations in (3.37) satisfy the conditions of Proposition 1, it implies that Part 1 of the FOMPC problem is solved. By implication, the existence of a solution to the GTMPC problem also solves Part 2 and 3 of the FOMPC problem which concludes the proof. □

The tracking error in (3.34) may be computed directly from measurements of the input-output vector $z(k)$, provided the objective $\tilde{\Phi}$ and the steady-state gain matrix $G_u(\delta)$ are available a priori. This choice therefore obviates the need for knowledge of the optimal equilibrium $\bar{z}^*(\delta, \bar{w})$ and the additive uncertainty $\bar{w}$. However, a knowledge of the multiplicative (parametric) uncertainty δ may still be required to compute $G_u(\delta)$. To circumvent this limitation, $G_u(\delta)$ can be estimated online at the current operating point using past input-output measurements [215]. This estimation problem is non-trivial and an in-depth discussion of this topic is out of scope for this thesis. We refer the reader to [35, 149, 119] for details regarding the steady-state gain estimation problem within the context of feedback-based optimization. Where online estimation of the steady-state gain may not be feasible, a nominal value of $G_u(\delta)$ (that is $G_u(\delta) \approx G_u(0)$) can be adopted for all values of the parametric uncertainty, $\delta \in \Delta_l$ [46, 173]. Both approaches cannot guarantee convergence to the true optimum without some error (i.e. estimation and approximation errors).

Some linear systems may possess a steady-state gain $G_u(\delta)$ that is independent of the multiplicative uncertainty δ. For these systems, the tracking error (3.34) is invariant to both the additive and multiplicative uncertainty. We will associate such systems with a robust steady-state gain property as defined next.

Definition 3 (Robust steady-state gain)**.** *The uncertain linear system* (3.1) *is said to posses the robust steady-state gain (RSG) property if there exists a fixed matrix* G_u *such that*

$$G_u(\delta) = G_u \quad \forall \delta \in \Delta_l. \tag{3.38}$$

Lemma 3.5.2 (RSG Property)**.** *The RSG property is satisfied for the uncertain linear system* (3.1) *if* $\begin{bmatrix} C(\delta) & \mathbf{0}_{n_y \times n_u} \end{bmatrix}$ *is independent of* δ *and* rank $Z(\delta)$ *is* δ*-invariant.*

Proof. Based on the definition of $G_u(\delta)$ in (3.6)

$$\Big(G_u(\delta) = G_u \quad \forall \delta \in \Delta_l\Big) \iff \Big(\text{Lemma 3.5.2}\Big). \tag{3.39}$$

□

Example 1 (System with RSG property)**.** *Consider the uncertain LTI system* (3.1) *with*

$$A(\delta) = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ -\delta_1 & -4\delta_1 & 1-\delta_1\delta_3 & 0 \\ -6\delta_2 & -5\delta_2 & 0 & 1-\delta_2\delta_4 \end{bmatrix}, \; B(\delta) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}, \; E = \begin{bmatrix} 5 & 1 \\ 2 & 10 \\ 3 & 0 \\ 0 & 4 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}. \tag{3.40}$$

The parametric uncertainty $\delta = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \in \Delta_4 \subset \mathbb{R}^4$ *takes values from the interval* $\Delta_4 \coloneqq \begin{bmatrix} -5\mathbb{1}_4 & 20\mathbb{1}_4 \end{bmatrix}$.

Computing the steady-state gain $G_u(\delta)$ *for arbitrary values of* $\delta \in \Delta_4$ *confirms that the above system indeed has the RSG property with*

$$G_u(\delta) = \begin{bmatrix} -0.2632 & 0.2105 \\ 0.3158 & -0.0526 \end{bmatrix} \quad \forall \delta \in \Delta_4. \tag{3.41}$$

Theorem 3.5.1 enables the reduction of the FOMPC problem to a generalized tracking model predictive control problem (GTMPC). As a result, a tracking MPC controller can be designed to track the unknown optimum of the the steady-state optimization $\mathcal{RTO}$, thereby bringing the benefits of feedback (such as an inherent robustness to uncertainty) to the steady-state optimization while retaining the complexity of standard tracking MPC.

3.6 Feedback Optimizing Model Predictive Control with Quadratic Steady-state Cost

For certain applications (such as optimal load-frequency control [55] [211], implicit trajectory planning [232] and optimal internet congestion control [140]), the steady-state performance criterion is the quadratic function

$$\Phi(\bar{z}) = \frac{1}{2}\begin{bmatrix}\bar{u}\\ \bar{y}\end{bmatrix}^\top \begin{bmatrix}Q_{uu} & Q_{uy}\\ Q_{yu} & Q_{yy}\end{bmatrix}\begin{bmatrix}\bar{u}\\ \bar{y}\end{bmatrix} + \begin{bmatrix}R_u\\ R_y\end{bmatrix}^\top \begin{bmatrix}\bar{u}\\ \bar{y}\end{bmatrix} = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z} \tag{3.42}$$

and the resulting steady-state optimality error (3.34) is the piece-wise affine function,

$$e(k) := \tilde{G}(\delta)\nabla\tilde{\Phi}(z(k)) = \begin{cases}\tilde{G}(\delta)\nabla\Phi(z(k)) & z(k) \in int(\mathbb{Z})\\ \tilde{G}(\delta)\nabla\Phi(z(k)) + \tilde{G}(\delta)\nabla g(z(k))\Gamma g(z(k))) & z(k) \notin int(\mathbb{Z})\end{cases}. \tag{3.43}$$

For the polytopic constraints, $\mathbb{U} := \{u\colon P_u u \le q_u\}$ and $\mathbb{Y} := \{y\colon P_y y \le q_y\}$, $g(\bar{z}) = H\bar{z} - h$ and $\nabla g(\bar{z}) = H^\top$ where $H = \begin{bmatrix}P_u & \mathbf{0}_{m\times p}\\ \mathbf{0}_{p\times m} & P_y\end{bmatrix}$ and $h = \begin{bmatrix}q_u\\ q_y\end{bmatrix}$.
The tracking error in (3.43) can then be written as the set-based piecewise affine equation

$$e(k) = \Lambda_{j,y}(\delta)y(k) + \Lambda_{j,u}(\delta)u(k) + r_j(\delta) \quad \text{for} \quad z(k) \in \mathbb{S}_j,\ j \in \mathbb{J}. \tag{3.44}$$

The set $\mathbb{S}_j = \{\mathbb{S}_1, \mathbb{S}_2\}$ are the set of polyhedral partitions of the state-space into inactive and active regions respectively, where $\mathbb{J} = \{1,\ 2\}$ is the index set of such partitions that is,

$$\mathbb{S}_1 = \{\bar{z}\colon g(\bar{z}) < \mathbf{0}_{n_c}\}, \quad \mathbb{S}_2 = \{\bar{z}\colon g(\bar{z}) \ge \mathbf{0}_{n_c}\}. \tag{3.45}$$

$$\Lambda_{j,y}(\delta) = \begin{cases}\Lambda_y(\delta) & \text{if } z(k) \in \mathbb{S}_1\\ \Lambda_y(\delta) + \tilde{\Lambda}_y(\delta) & \text{if } z(k) \in \mathbb{S}_2\end{cases}, \quad \Lambda_{j,u}(\delta) = \begin{cases}\Lambda_u(\delta) & \text{if } z(k) \in \mathbb{S}_1\\ \Lambda_u(\delta) + \tilde{\Lambda}_u(\delta) & \text{if } z(k) \in \mathbb{S}_2\end{cases}, \tag{3.46}$$

and

$$r(\delta) = \begin{cases}r(\delta) & \text{if } z(k) \in \mathbb{S}_1\\ r(\delta) + \tilde{r}(\delta) & \text{if } z(k) \in \mathbb{S}_2\end{cases}, \tag{3.47}$$

where

$$\Lambda_y(\delta) = \frac{1}{2}[(Q_{yy} + Q_{yy}^\top) + (G_u(\delta)^\top)^\dagger(Q_{uy} + Q_{yu}^\top)], \tag{3.48}$$

$$\Lambda_u(\delta) = \frac{1}{2}[(Q_{yu} + Q_{uy}^\top) + (G_u(\delta)^\top)^\dagger(Q_{uu} + Q_{uu}^\top)], \tag{3.49}$$

$$r(\delta) = R_y + (G_u(\delta)^\top)^\dagger R_u, \tag{3.50}$$

$$\tilde{\Lambda}_u(\delta) = (G_u(\delta)^\top)^\dagger P_u^\top \Gamma_u P_u, \tag{3.51}$$

$$\tilde{\Lambda}_y = P_y^\top \Gamma_y P_y, \tag{3.52}$$

$$\tilde{r}(\delta) = -[(G_u(\delta)^\top)^\dagger P_u^\top \Gamma_u q_u + P_y^\top \Gamma_y q_y]. \tag{3.53}$$

This error may be computed directly from the input $u(k)$ and output $y(k)$, provided the objective $\tilde{\Phi}$ and the input–output DC gain matrix $G_u(\delta)$ are known. As a result, the FOMPC problem by virtue of Theorem 3.5.1 reduces to a piece-wise quadratic tracking control problem.

For the remainder of this thesis, we will focus on adapting conventional tracking MPC algorithms to solve the FOMPC problem for the case of quadratic steady-state optimization problems with inequality constraints that are inactive in steady-state. As will become evident later in the thesis, this adaptation is not trivial and often leads to a generalization of conventional tracking MPC beyond set-point tracking.

Relation to Tracking MPC

Because FOMPC tracks an (unknown) reference that is the minimum of a steady-state optimization problem, we can easily show that it generalizes the classical Model Predictive tracking control algorithms introduced in Section 2.2 for a known reference, y_{sp}. We define the steady-state optimization cost function as

$$\Phi(\bar{z}) = \frac{1}{2}(\bar{y} - y_{\text{sp}})^\top(\bar{y} - y_{\text{sp}}). \tag{3.54}$$

The minimum is, obviously, $\bar{y} = y_{\text{sp}}$. The resulting steady-state optimality error at sample time k is then easily obtained as

$$e(k) = \tilde{G}(\delta)\nabla\Phi(z(k)) = y(k) - y_{\text{sp}} \tag{3.55}$$

via the designs $\Lambda_y = I_p$, $\Lambda_u = \mathbf{0}$ and $r = -y_{\text{sp}}$. The FOMPC tracking error (3.34) reduces to a standard tracking error $y(k) - \bar{y}_{sp}$, which may be steered to zero using conventional methods.

3.7 Conclusion

In this chapter, the feedback optimizing model predictive control problem has been formulated for a defined steady-state optimization. Also, as a first step towards solving the defined problem, the steady-state tracking error was expressed as a function of the measurable inputs and outputs. From this tracking error, a generalized tracking MPC problem has been formulated from which a solution to the feedback optimizing MPC problem can be devised. In the following chapters, the focus will be on devising model predictive control algorithms that solve the generalized tracking MPC problem for a defined steady-state optimization.

Chapter 4

Deterministic Feedback Optimizing Linear Quadratic Control

In this chapter, we consider the FOMPC problem in the absence of dynamic inequality constraints and under deterministic conditions (i.e., state is measurable and no model mismatch is present). We call this the deterministic feedback optimizing linear quadratic control (FOLQC) problem. We propose several solutions to the deterministic FOLQC problem when the steady-state optimization problem is a quadratic program. The proposed algorithms in this chapter enable the design of novel controllers that autonomously track the optimal solution of a steady-state quadratic program with guaranteed optimal transient performance. Also, the FOLQC algorithms proposed reduce to the standard tracking linear quadratic regulator when the goal is to track a known steady-state set-point. Therefore, we can alternatively view FOLQC as a generalization of tracking linear quadratic control to cases where the reference set-points are implicitly generated from a known steady-state optimization problem rather than being explicitly available a-priori. A benefit of FOLQC over the standard tracking LQ regulator is its self-optimizing capabilities. Although the FOLQC formulation presented in this chapter is deterministic (i.e. assumes model is known and disturbance is piece-wise constant), the control algorithm still enjoys some inherent robustness to model uncertainty and unknown disturbances. We analyse this inherent robustness using standard results on the robustness of the linear quadratic control. Also, unlike most feedback optimization algorithms, the deterministic FOLQC regulator under simple assumptions can guarantee the closed-loop convergence of a dynamic system to its optimal equilibrium, while also optimizing the dynamic performance according to a defined cost. This chapter is organized as follows. In Section 4.1, we formulate the deterministic FOLQC problem associated with a given (non-linear) steady-state optimization problem. In Section 4.2, the FOLQC problem is specialized to the case of quadratic steady-state optimization with linear dynamics. Also in this section, the FOLQC problem is solved for a steady-state quadratic program using the principle of dynamic programming and a velocity model of the system dynamics. Alternative solutions to the deterministic FOLQC problem based on linear matrix inequalities are proposed in Section 4.3. These alternative will be useful in the extension of the FOLQC problem to uncertain LTI systems in Chapter 5. In Section 4.4, we present an output feedback variant to the FOLQC regulator when the state is not available a-priori. Section 4.5, presents a feed-forward approach to the solution of the FOLQC problem. In Section 4.6, we study the inherent robustness of FOLQC and finally present some numerical simulation results in Section 4.8.

4.1 Introduction

Linear quadratic control (LQC) is a well-known and widely applied approach to achieving optimal control of dynamic systems. Not only is LQC a powerful design method, but it also has some interesting properties. Firstly, the optimal control in a LQC problem can be expressed as a linear state feedback control law allowing for a simple feedback implementation, with little or no online computation. Also, the feedback nature of the LQC law gives it good robustness properties when the state of the system is measurable.

In the LQC problem, the goal is to find the control input that minimizes a quadratic function of the state/output and the input, subject to linear dynamic constraints. Since its introduction by Kalman in the 1960s, the LQC problem has played a key role in many control design methods [132], with several extensions such as reference tracking, disturbance rejection, integral control and stochastic control achieving functionalities other than regulation.

Closely related to FOLQC is the linear quadratic tracking (LQT) problem. The LQT problem is a well studied extension of LQC with the goal of finding the optimal control input that drives the output and/or input of a linear dynamic system towards known setpoint/reference values, while rejecting unknown disturbances. In the literature, the infinite-horizon LQT problem is an actively studied area of the LQC problem with several proposed solutions [16, 52, 133, 108, 80]. In the following, we extend the LQT problem beyond the tracking of known setpoints to the tracking of the

solution to a steady-state optimization problem under unknown but piece-wise constant disturbances.
The plant under consideration is the discrete-time linear time-invariant dynamic system represented as follows

$$x(k+1) = Ax(k) + Bu(k) + Ew(k), \tag{4.1a}$$
$$y(k) = Cx(k). \tag{4.1b}$$

Here: k denotes the sample time; $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$, $y(k) \in \mathbb{R}^{n_y}$, and $w(k) \in \mathbb{R}^{n_w}$ denote the system states, inputs, outputs and exogenous disturbances respectively; and, $A \in \mathbb{R}^{n_x \times n_x}$, $B \in \mathbb{R}^{n_x \times n_u}$, $E \in \mathbb{R}^{n_x \times n_w}$, and $C \in \mathbb{R}^{n_y \times n_x}$ are *known* system matrices. Assuming that the plant state $x(0)$ at a given time $k = 0$ is known, then the objective of feedback optimizing linear quadratic control (FOLQC) is to find a control law which asymptotically drives the outputs $y(k)$ and/or inputs $u(k)$ of the LTI system (4.1) to the solution, $\bar{z}^*(\bar{w})$ of the steady-state optimization

$$\bar{z}^*(\bar{w}) = \arg\min_{\bar{z}} \left\{ \Phi(\bar{z}) \mid \bar{z} \in \mathbb{F}(\bar{w}),\ \bar{z} \in \mathbb{Z} \right\} \tag{4.2}$$

while minimizing the transient (often quadratic) performance index or cost functional

$$J_\infty(x(0)) = \sum_{k=0}^{\infty} l(e(k), u(k)), \tag{4.3}$$

where

$$e(k) = z(k) - \bar{z}^*(\bar{w}) \coloneqq \tilde{G}\nabla\tilde{\Phi}(z(k)) \ \Big(\text{see Theorem (3.5.1) }\Big), \tag{4.4}$$

and $\Phi \colon \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is a convex steady-state (economic) cost, continuous in the decision variable $\bar{z} = \begin{bmatrix} \bar{u} & \bar{y} \end{bmatrix}^\top$ and the sets $\mathbb{F}$ and $\mathbb{Z}$ retain their previous definition from Chapter 3 except with $\delta = 0$.

Remark 12. *Because the forced equilibrium set* $\mathbb{F}(\bar{w})$ *is included in problem* (4.2) *as a constraint, every solution,* $\bar{z}^*(\bar{w})$ *is an equilibrium of the system,* (4.1).

The following is a formal statement of the deterministic FOLQC problem.

Problem 4 (Deterministic FOLQC Problem)**.** *Design for the LTI system* (4.1) *a state feedback control law of the form*

$$u(k) = f_c(x(k), y(k), u(k-1)) \tag{4.5}$$

such that the following are achieved for every constant disturbance, $\bar{w}$, without using explicit knowledge of the disturbance or the solution to (4.2)*:*

1. *for all initial $x(0)$, the closed-loop dynamics*

$$x(k+1) = Ax(k) + Bf_c(x(k), y(k), u(k-1)) + Ew(k) \tag{4.6}$$

are asymptotically stable with respect to the (unknown) steady-state optimizers of (4.2)*.*

2. *all trajectories of the closed loop system are such that*

$$\lim_{k\to\infty} e(k) = \mathbf{0}. \tag{4.7}$$

3. *the feedback policy $f_c(x(k), y_k, u(k-1))$ minimizes the transient performance criterion*

$$J_\infty(x(0)) = \sum_{k=0}^{\infty} l(e(k), u(k)), \tag{4.8}$$

where $l(e(k), u(k))$ is a non-negative instantaneous loss incurred at time k .

4.2 Quadratic Feedback-Optimizing Linear Quadratic Control

In this section, we solve the FOLQC problem when the steady-state performance objective, Φ, is quadratic, i.e.

$$\Phi(\bar{z}) = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z}, \tag{4.9}$$

where $Q_{zz} = \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix}$ and $R_z = \begin{bmatrix} R_u \\ R_y \end{bmatrix}$.
We show that with the developed control algorithms, the system (4.1) can be steered asymptotically to the optimal steady-state equilibrium (i.e., the solution of (4.2)),

without explicit knowledge of this equilibrium (i.e. $z^*(\delta, \bar{w})$) or knowledge of the disturbance, $\bar{w}$, and while simultaneously minimizing a transient performance criterion, $J(\cdot, \cdot)$.

Remark 13. *When the cost $\Phi(\bar{z})$ is convex, i.e., $Q_{zz} \succeq \mathbf{0}$, FOLQC tracks the global minimum of* (4.2)*. However, when $\Phi(\bar{z})$ is non-convex i.e., $Q_{zz} \nsucceq \mathbf{0}$, FOLQC tracks a local minimum of* (4.2) *if it exists. (See [33], Section* 4.2.2 *for proof).*

In order to keep the presentation simple and maintain intuition of the concepts, we present the development of the FOLQC framework considering inequality ($\mathbb{Z}$) constraints that are inactive in steady-states. The tracking error, (4.4), under a quadratic steady-state cost, and inactive $\mathbb{Z}$, is an affine function of the measured output and input, and is given by

$$e(k) := \tilde{G}\nabla\Phi(z(k)) = \Lambda_y y(k) + \Lambda_u u(k) + r \tag{4.10}$$

where

$$\begin{aligned}
\Lambda_y &= \frac{1}{2}[(Q_{yy} + Q_{yy}^\top) + (G_u^\top)^+(Q_{uy} + Q_{yu}^\top)], \\
\Lambda_u &= \frac{1}{2}[(Q_{yu} + Q_{uy}^\top) + (G_u^\top)^+(Q_{uu} + Q_{uu}^\top)], \\
r &= R_y + (G_u^\top)^+ R_u.
\end{aligned}$$

This error may be computed directly from the input $u(k)$ and output $y(k)$, provided the objective Φ and the input–output DC gain matrix G_u is known. This obviates the need for complete knowledge of the optimal steady-state set-points $\bar{z}^*$ and the disturbances $\bar{w}$ as would otherwise have been the case using a feed-forward multi-layered approach.

Remark 14. *Though the error $e(k)$ in* (4.10) *appears to be independent of the additive disturbance $w(k)$, any change in $w(k)$ while $u(k)$ is constant will induce via the system dynamics, a corresponding change in the output $y(k)$ and hence $e(k)$.*

For the purpose of control design, we make the following assumptions

Assumption 3. *The system* (4.1) *is reachable and observable.*

Assumption 4. *The matrix*

$$S = \begin{bmatrix} A - I_{n_x} & B \\ \Lambda_y C & \Lambda_u \end{bmatrix} \tag{4.11}$$

is such that $\operatorname{rank} S = n_y + n_u$.

Assumption 3 is standard in the optimal control literature and Assumption 4 is standard in the output regulation literature. We now present the following algorithms for solving the FOLQC problem.
To design the FOLQC, we employ a velocity-model form of linear quadratic regulation, wherein the tracking error (between the measured $z(k)$ and the desired $\bar{z}^*(\bar{w})$) is driven to zero. A novel and practically significant feature of the approach is that we regulate the tracking error without knowledge of the desired steady-state and the disturbances.

4.2.1 Velocity and Error Dynamics

To regulate the system to $e = 0$, we consider the velocity, or incremental, form of the system dynamics (4.1) augmented with the tracking error dynamics, whose output is the tracking error $e(k)$:

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k), \tag{4.12a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k), \tag{4.12b}$$

where

$$\epsilon(k) := \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x(k) &:= x(k) - x(k-1), \\ \delta u(k) &:= u(k) - u(k-1), \end{aligned} \tag{4.13}$$

and

$$\mathcal{A} = \begin{bmatrix} A & \mathbf{0}_{n_x \times n_y} \\ \Lambda_y C & I_{n_y} \end{bmatrix}, \qquad \mathcal{B} = \begin{bmatrix} B \\ \Lambda_u \end{bmatrix}, \tag{4.14a}$$

$$\mathcal{C} = \begin{bmatrix} \Lambda_y C & I_{n_y} \end{bmatrix}, \qquad \mathcal{D} = \Lambda_u. \tag{4.14b}$$

4.2.2 FOLQC Formulation

Given the tracking error and velocity dynamics, the feedback optimizing linear quadratic control problem is defined, for a state $\epsilon(k)$, as

$$\min_{\delta u(k)} J_\infty(\epsilon(k)) \tag{4.15}$$

subject to,

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k), \tag{4.16a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k). \tag{4.16b}$$

In this problem, the decision variable is the control law

$$\delta u(k) = -\mathcal{K}\epsilon(k). \tag{4.17}$$

This control law is chosen to minimize the performance index $J_\infty(\epsilon(k))$, which is defined as

$$J_\infty(\epsilon(k)) = \frac{1}{2}\sum_{k=0}^{\infty} l(e(k), \delta u(k)), \tag{4.18}$$

where

$$l(e(k), \delta u(k)) := e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k). \tag{4.19}$$

The matrices Q_e and R_δ are the respective penalty matrices on the squared tracking error $e(k)$ and input deviation $\delta u(k)$.

Remark 15 (Performance index)**.** *The performance index $J_\infty(\epsilon(k))$ represents the costs accrued during the operation of the system under the FOLQC law over the time interval $[0, \infty]$. This cost is a measure of how well the closed loop system performs under the FOLQC law and can be used to indirectly influence the transient performance of the closed loop system. The term $e(k)^\top Q_e e(k)$ penalizes the distance of the tracking error from steady-state optimality with the weight Q_e used to determine how quickly the closed loop system achieves steady-state optimality for the static optimization problem* (4.2)*. The term $\delta u(k)^\top R_\delta \delta u(k)$ penalizes the incremental control effort needed to drive the system to the optimal steady-state equilibrium. Therefore based on the choice of Q_e relative to R_δ, the transient performance objectives of the closed loop system under FOLQC can be indirectly embedded in the control design.*

Problem (4.15 s.t. 4.16) is a standard infinite-horizon LQ optimal control problem, thus its solution is easily characterized and obtained. Firstly, existence of a solution is ensured by proper assumptions on the stabilizability and observability of the augmented velocity dynamics.

Proposition 2 (Stabilizability of augmented velocity dynamics)**.** *The pair $(\mathcal{A}, \mathcal{B})$ is stabilizable if and only if Assumptions 3 and 4 are satisfied.*

Proof. From the PBH controllability condition, $(\mathcal{A}, \mathcal{B})$ is stabilizable if and only if,

$$\begin{bmatrix} \mathcal{A} - \lambda I_{n_\epsilon} & \mathcal{B} \end{bmatrix} = \begin{bmatrix} A - \lambda I_{n_x} & \mathbf{0}_{n_x \times n_y} & B \\ \Lambda_y C & I_{n_y} - \lambda I_{n_y} & \Lambda_u \end{bmatrix} \tag{4.20}$$

has full rank for all $\lambda \in \operatorname{eig} \mathcal{A}$. The first set of n_x rows in (4.20) are linearly independent if (A, B) is stabilizable. The second set of n_y rows are linearly independent from the first set of n_x rows except possibly at $\lambda = 1$. We therefore need to check (4.20) at $\lambda = 1$, whence it becomes (4.11). □

Observability of $(\mathcal{C}, \mathcal{A})$ can be similarly established, following from the observability of (C, A). Secondly, uniqueness of the solution is ensured by satisfaction of regularity conditions on the objective function.

Proposition 3 (Positive definiteness of the cost)**.** *If* $R_\delta \succ 0$ *and*

$$Q_e - Q_e \Lambda_u (R_\delta + \Lambda_u^\top Q_e \Lambda_u)^{-1} \Lambda_u^\top Q_e^\top \succeq 0, \tag{4.21}$$

then $l(e(k), \delta u(k))$ *is for all* $\epsilon(k)$ *a positive definite function of* $\delta u(k)$.

Proof. Substituting (4.12b) in (4.19),

$$\begin{aligned} l(e(k), \delta u(k)) = \epsilon(k)^\top \mathcal{Q} \epsilon(k) + 2\epsilon^\top(k) \mathcal{N} \delta u(k) + \delta u(k)^\top \mathcal{R} \delta u(k) = \\ \begin{bmatrix} \epsilon(k) \\ \delta u(k) \end{bmatrix}^\top \begin{bmatrix} \mathcal{Q} & \mathcal{N} \\ \mathcal{N}^\top & \mathcal{R} \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ \delta u(k) \end{bmatrix}. \end{aligned} \tag{4.22}$$

The stage cost in (4.22) is positive definite if

$$\begin{bmatrix} \mathcal{Q} & \mathcal{N} \\ \mathcal{N}^\top & \mathcal{R} \end{bmatrix} \succ 0 \tag{4.23}$$

which is satisfied if $\mathcal{R} \succ 0$ and $\mathcal{Q} - \mathcal{N}\mathcal{R}^{-1}\mathcal{N}^\top \succeq 0$ resulting in the conditions of Proposition 4.19 upon substitution of $\mathcal{Q} = \mathcal{C}^\top Q_e \mathcal{C}$, $\mathcal{N} = \mathcal{C}^\top Q_e \mathcal{D}$, $\mathcal{D} = \Lambda_u$ and $\mathcal{R} = R_\delta + \mathcal{D}^\top Q_e \mathcal{D}$. □

Remark 16. *From* (4.3), $l(e(k), \delta u(k)) \succ 0$ *implies* $J_\infty(\epsilon(k), \delta \mathbf{u}(k)) \succ 0$.

4.2.3 Dynamic Programming Solution to the FOLQC problem

The finite- and infinite-horizon control laws are easily derived by applying the standard arguments of dynamic programming and Bellman's Principle of Optimality to the optimal control problem [133]:

Proposition 4. *The solution to (4.15 s.t. 4.16) is*

$$\delta u^*(k) = -\mathcal{K}(k)\epsilon(k) \quad k = 0, \dots, N-1 \tag{4.24}$$

where, for $k = 0, \dots, N-1$,

$$\mathcal{K}(k) = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}(k)\mathcal{B}\Big)^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}(k)\mathcal{A}\Big), \tag{4.25a}$$

$$\mathcal{P}(k) = \Big(\mathcal{A} - \mathcal{B}\mathcal{K}(k)\Big)^\top \mathcal{P}(k+1)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(k)\Big) + \mathcal{K}(k)^\top \mathcal{R}\mathcal{K}(k) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(k). \tag{4.25b}$$

Moreover, as $N \to \infty$ *then* $\mathcal{K}(N) \to \mathcal{K}$*, and* $\mathcal{P}(k+1) \to \mathcal{P}(k)$ *with* $\mathcal{P}(N) \to \mathcal{P}$ *where*

$$\mathcal{K} = (\mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B})^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}\mathcal{A}\Big), \tag{4.26a}$$

$$\mathcal{P} = \mathcal{A}^\top \mathcal{P}\mathcal{A} + \mathcal{Q} - \Big(\mathcal{N} + \mathcal{A}^\top \mathcal{P}\mathcal{B}\Big)(\mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B})^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}\mathcal{A}\Big). \tag{4.26b}$$

Proof. In this development, we adopt a backward dynamic programming approach starting from the time interval $k \in [0, N]$. Let $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ denote the optimum cost (4.15) of transferring the system (4.12) from an initial state $\epsilon(k)$ to the terminal state $\epsilon(N)$. At the terminal state $\epsilon(N)$, let the optimal cost be given by the function

$$J_N^*(\epsilon(N)) = \frac{1}{2}\epsilon(N)^\top \mathcal{P}(N)\epsilon(N), \tag{4.27}$$

where $\mathcal{P}$ is a symmetric positive (semi) definite matrix. Let $J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1))$ be the optimal cost evaluated from time $k+1$ to N. Then at any stage k, using the principle of optimality [133],

$$J_k^*\Big(\epsilon(k), \delta u(k)\Big) = \min_{\delta u(k)} \Big\{ l(e(k), \delta u(k)) + J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1)) \Big\}. \tag{4.28}$$

By solving (4.28) recursively, we can compute the optimal control law $u^*(k)$ that solves the OCP. To begin, at $k = N-1$,

$$J_{N-1}(\epsilon(N-1), \delta\mathbf{u}(N-1)) = J_N^*(\epsilon(N)) + l(e(N-1), \delta\mathbf{u}(N-1)). \tag{4.29}$$

Using the state equation (4.12) at $k = N-1$,

$$\epsilon(N) = \mathcal{A}\epsilon(N-1) + \mathcal{B}\delta u(N-1), \tag{4.30}$$

we can eliminate $\epsilon(N)$ from J_{N-1} to obtain

$$\begin{aligned} J_{N-1}(\epsilon(N-1), \delta u(N-1)) = \frac{1}{2}\Big(\mathcal{A}\epsilon(N-1) + \mathcal{B}\delta u(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{A}\epsilon(N-1) + \\ \mathcal{B}\delta u(N-1)\Big) + \frac{1}{2}\epsilon(N-1)^\top \mathcal{Q}\epsilon(N-1) \\ +\epsilon(N-1)^\top \mathcal{N}\delta u(N-1) + \frac{1}{2}\delta u(N-1)^\top \mathcal{R}\delta u(N-1). \end{aligned} \tag{4.31}$$

The optimal control law at $k = N-1$ i.e., $\delta u^*(N-1)$ can be obtained by applying the first order necessary optimality condition,

$$\frac{\partial J_{N-1}}{\partial \delta u(N-1)} = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}(N)\mathcal{B}\Big)\delta u^*(N-1) + \Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}(N)\mathcal{A}\Big)\epsilon(N-1) = \mathbf{0}. \tag{4.32}$$

Solving for $\delta u^*(N-1)$, we obtain

$$\delta u^*(N-1) = -\mathcal{K}(N-1)\epsilon(N-1), \tag{4.33}$$

where

$$\mathcal{K}(N-1) = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}(N)\mathcal{B}\Big)^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}(N)\mathcal{A}\Big). \tag{4.34}$$

With (4.33), we can compute the corresponding optimal cost at $k = N-1$ as,

$$\begin{aligned} J_{N-1}^*(\epsilon(N-1)) = \frac{1}{2}\epsilon(N-1)^\top \Big[\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N-1)\Big) + \\ \mathcal{Q} - 2\mathcal{N}\mathcal{K}(N-1) + \mathcal{K}(N-1)^\top \mathcal{R}\mathcal{K}(N-1)\Big]\epsilon(N-1), \end{aligned} \tag{4.35}$$

which can also be expressed as

$$J^*_{N-1}(\epsilon(N-1)) = \frac{1}{2}\epsilon(N-1)^\top \mathcal{P}(N-1)\epsilon(N-1) \tag{4.36}$$

resulting in the recursion

$$\begin{aligned}\mathcal{P}(N-1) = \Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N-1)\Big) + \\ \mathcal{K}(N-1)^\top \mathcal{R}\mathcal{K}(N-1) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(N-1).\end{aligned} \tag{4.37}$$

We can repeat this procedure for $k = N-2,\ N-3, \ldots$ giving the results (4.24) and (4.25). □

The main result of this chapter—that the infinite-horizon control law characterized by proposition 4 solves Problem 1—immediately follows.

Theorem 4.2.1. *Suppose that Assumptions 1 and 2 hold, and also the hypotheses of Proposition 2 and 3. The infinite-horizon control law $\delta u(k) = -\mathcal{K}\epsilon(k)$ solves Problem 4, minimizing the infinite-horizon criterion*

$$J_\infty(\epsilon(0)) := \frac{1}{2}\sum_{k=0}^{\infty} l(e(k), \delta u(k)). \tag{4.38}$$

Remark 17 (Finite- and receding-horizon implementations)**.** *The finite-horizon control gains $\mathcal{K}(N), \mathcal{K}(N-1), \ldots, \mathcal{K}(k), \ldots \mathcal{K}(0)$ (usually obtained from the online solution to a finite-horizon formulation of the FOLQC problem in Section 4.2.2) can also solve Problem 4 if $\mathcal{P}(N)$ satisfies the Lyapunov equation*

$$\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big)^\top \mathcal{P}(N+1)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big) - \mathcal{P}(N) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(N) + \mathcal{K}(N)^\top \mathcal{R}\mathcal{K}(N) = \mathbf{0}, \tag{4.39}$$

with some known stabilizing terminal control $\mathcal{K}(N)$.
In a receding-horizon (MPC) implementation, the applied control law is

$$\delta u(k) = -\mathcal{K}(k)\epsilon(k). \tag{4.40}$$

The result may be suboptimal with optimality achieved if

$$\mathcal{K}(N) = \mathcal{K}, \tag{4.41}$$

where $\mathcal{K}$ is the infinite-horizon FOLQC gain given by (4.26a). *However, such implementations may be useful—indeed necessary—when constraints are present on the system. The FOLQC problem subject to such constraints is the subject of following chapter.*

Proof. Assume at time instant k the control sequence

$$\delta\mathbf{u}(k) = \Big\{\delta u(k), \delta u(k+1), \ldots, \delta u(k+N-1)\Big\} \tag{4.42}$$

transfers the system (4.12) from an initial state $\epsilon(k)$ to the terminal state $\epsilon(N)$ with the associated finite-horizon performance cost

$$J_k(\epsilon(k), \delta\mathbf{u}(k)) = J_N(\epsilon(N)) + \frac{1}{2}\sum_{i=0}^{N-1} l(e(k+i), \delta u(k+i)), \tag{4.43}$$

where

$$J_N(\epsilon(N)) = \frac{1}{2}\epsilon(N)^\top \mathcal{P}(N)\epsilon(N) \tag{4.44}$$

is a terminal cost $\epsilon(N)$ with $\mathcal{P}(N)$ a symmetric positive (semi) definite matrix. Let $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ and $\delta\mathbf{u}^*(k)$ be the optimal cost and optimal control sequence respectively, obtained by minimizing the cost (4.43) subject to the velocity dynamics (4.16). For a receding-horizon (MPC) implementation, the optimal control law applied to the system at time k is given by the first element of $\delta\mathbf{u}^*(k)$ i.e.,

$$\delta u^*(k) = -\mathcal{K}(k)\epsilon(k), \tag{4.45}$$

where $\mathcal{K}(k)$ may be given (4.25a).

To establish convergence of the receding-horizon control law (4.45), we will show that $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ is a Lyapunov function for the closed-loop system under this control law if conditions of Remark 17 are satisfied.

Firstly, $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ is positive definite if the conditions of Proposition 3 are satisfied. Also, $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ is zero at the origin and radially unbounded everywhere else, since $J_k^*(0, \mathbf{0}) = 0$ and $J_k^*(\epsilon(k), \delta\mathbf{u}(k)) \geq \frac{1}{2}\epsilon(k)^\top \mathcal{Q}\epsilon(k),\ \forall \epsilon(k) \in \mathbb{R}^{n_x+n_y} \setminus \{0\}$ implying $J_k^*(\epsilon(k), \delta\mathbf{u}(k)) \to \infty$ as $\|\epsilon(k)\| \to \infty$. After establishing that $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ is zero at the origin and radially unbounded, the final step to proving that $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ is a

Lyapunov function is to show that

$$\Delta J_k^* = J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1)) - J_k^*(\epsilon(k), \delta\mathbf{u}(k)) \tag{4.46}$$

is negative semi-definite under conditions of Remark 17, where

$$\delta\mathbf{u}(k+1) = \Big\{\delta u(k+1), \delta u(k+2), \ldots, \delta u(k+N)\Big\} \tag{4.47}$$

is the control sequence required to transfer the system from an initial state $\epsilon(k+1)$ to the final state $\epsilon(k+N+1)$. Let $\delta\mathbf{u}^*(k+1)$ be the optimal control sequence at time $k+1$ with the corresponding optimal cost being $J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1))$. However, at time k, $\delta\mathbf{u}^*(k+1)$ is not available since the receding-horizon control is generated online by minimizing (4.43) at the current time, k. But if we assume an accurate model and a well posed optimal control problem, then the tail of $\mathbf{u}^*(k)$ i.e.,

$$\delta\mathbf{u}_{tail}^*(k) = \Big\{\delta u^*(k+1), \delta u^*(k+2), \ldots, \delta u^*(k+N-1)\Big\} \tag{4.48}$$

will be included in $\delta\mathbf{u}^*(k+1)$. Using the following shifted optimal control input sequence at time k,

$$\delta\tilde{\mathbf{u}}^*(k) = \Big\{\delta u^*(k+1), \delta u^*(k+2) \ldots, \delta u(k+N)\Big\} \tag{4.49}$$

where

$$\delta u(k+N) = -\mathcal{K}(N)\epsilon(k+N) \tag{4.50}$$

is a stabilizing terminal control law for the closed-loop velocity dynamics, (i.e. $\mathcal{K}(N)$ is such that the spectral radius (minimum eigenvalue) of $\mathcal{A} - \mathcal{B}\mathcal{K}(N)$ is less than unity), the shifted optimal cost $\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k))$ can be obtained which will be a known but suboptimal approximation of the true optimal cost $J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1))$ at time $k+1$. Since $\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k))$ is suboptimal, the inequality

$$\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) \geq J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1)) \tag{4.51}$$

is always true and therefore ΔJ_k^* will be negative (semi) definite if we can show that

$$\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) - J_k^*(\epsilon(k), \delta\mathbf{u}(k)) \preceq 0. \tag{4.52}$$

Given the cost function (4.43), the optimal cost at time k can be expressed as

$$
\begin{aligned}
J_k^*(\epsilon(k), \delta\mathbf{u}(k)) &= J_N(\epsilon(N)) + \frac{1}{2}\sum_{i=0}^{N-1} l(e(k+i), \delta u^*(k+i)) \\
&= \frac{1}{2}\epsilon(k+N)^\top \mathcal{P}(N)\epsilon(k+N) + \frac{1}{2}\sum_{i=0}^{N-1}\Big\{\epsilon(k+i)^\top \mathcal{Q}\epsilon(k+i)+ \\
&\qquad 2\epsilon(k+i)^\top \mathcal{N}\delta u^*(k+i) + \delta u^*(k+i)^\top \mathcal{R}\delta u^*(k+i)\Big\} \\
= \frac{1}{2}\epsilon(k+N)^\top \mathcal{P}(N)\epsilon(k+N) &+ \frac{1}{2}\Big\{\epsilon(k)^\top \mathcal{Q}\epsilon(k) + 2\epsilon^\top(k)\mathcal{N}\delta u^*(k) + \delta u^*(k)^\top \mathcal{R}\delta u^*(k)\Big\}+ \\
\ldots + \frac{1}{2}\Big\{\epsilon(k+N-1)^\top \mathcal{Q}\epsilon(k+N-1) &+ 2\epsilon(k+N-1)^\top \mathcal{N}\delta u^*(k+N-1) \\
&+\delta u^*(k+N-1)^\top \mathcal{R}\delta u^*(k+N-1)\Big\}.
\end{aligned}
\tag{4.53}
$$

Also, the suboptimal shifted cost at time k, $\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k))$, used as an approximation of the cost $J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1))$ at time $k+1$ is

$$
\begin{aligned}
\tilde{J}_{k+1}^*(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) &= J_{N+1}(\epsilon(k+N+1)) + \frac{1}{2}\sum_{i=1}^{N} l(e(k+i), \delta u^*(k+i)) \\
&= \frac{1}{2}\epsilon(k+N+1)^\top \mathcal{P}(N+1)\epsilon(k+N+1) + \frac{1}{2}\sum_{i=1}^{N}\Big\{\epsilon(k+i)^\top \mathcal{Q}\epsilon(k+i)+ \\
&\qquad 2\epsilon(k+i)^\top \mathcal{N}\delta u^*(k+i) + \delta u^*(k+i)^\top \mathcal{R}\delta u^*(k+i)\Big\} \\
&= \frac{1}{2}\epsilon(k+N+1)^\top \mathcal{P}(N+1)\epsilon(k+N+1) + \frac{1}{2}\Big\{\epsilon(k+1)^\top \mathcal{Q}\epsilon(k+1)+ \\
&\qquad 2\epsilon^\top(k+1)\mathcal{N}\delta u^*(k+1) + \delta u^*(k+1)^\top \mathcal{R}\delta u^*(k+1)\Big\} + \ldots + \\
\frac{1}{2}\Big\{\epsilon(k+N)^\top \mathcal{Q}\epsilon(k+N) &+ 2\epsilon(k+N)^\top \mathcal{N}\delta u(k+N) + \delta u(k+N)^\top \mathcal{R}\delta u(k+N)\Big\}.
\end{aligned}
\tag{4.54}
$$

From the velocity dynamics (4.12a),

$$
\epsilon(k+N+1) = \mathcal{A}\epsilon(k+N) + \mathcal{B}\delta u(k+N), \tag{4.55}
$$

and substituting (4.50) in (4.55), we obtain

$$
\epsilon(k+N+1) = \Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big)\epsilon(k+N). \tag{4.56}
$$

Substituting (4.56) and (4.50) in (4.54) and using the expression in (4.53), it can be easily shown that

$$\tilde{J}^*_{k+1}(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) - J^*_k(\epsilon(k), \delta\mathbf{u}(k)) = -\frac{1}{2}l(e(k+i), \delta u(k+i)) + \frac{1}{2}\epsilon(k+N)^\top\Big\{\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big)^\top\mathcal{P}(N+1)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big) - \mathcal{P}(N) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(N) + \mathcal{K}(N)^\top\mathcal{R}\mathcal{K}(N)\Big\}\epsilon(k+N). \tag{4.57}$$

If the conditions of Proposition 3 are met, then $l(e(k+i), \delta u(k+i)) \succeq 0$. Also, if $\mathcal{P}(N)$ is a positive definite solution to the Lyapunov equation,

$$\mathcal{P}(N) = \Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big)^\top\mathcal{P}(N+1)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(N)\Big) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(N) + \mathcal{K}(N)^\top\mathcal{R}\mathcal{K}(N) \tag{4.58}$$

for any stabilizing $\mathcal{K}(N)$ as stipulated in Remark 17, then the inequality in (4.57) reduces to

$$\tilde{J}^*_{k+1}(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) - J^*_k(\epsilon(k), \delta\mathbf{u}(k)) \preceq -\frac{1}{2}l(e(k+i), \delta u(k+i)). \tag{4.59}$$

Because $l(e(k+i), \delta u(k+i)) \succeq 0$,

$$\tilde{J}^*_{k+1}(\epsilon(k+1), \delta\tilde{\mathbf{u}}^*(k)) - J^*_k(\epsilon(k), \delta\mathbf{u}(k)) \preceq 0, \tag{4.60}$$

hence $J^*_k(\epsilon(k), \delta\mathbf{u}(k))$ is a Lyapunov function for the closed-loop velocity dynamics under the receding horizon control law (4.40). Therefore, the velocity dynamics (4.12) under the receding horizon control law (4.40) asymptotically converges to the origin, i.e., $\epsilon(k) \to \mathbf{0}$ and $\delta u(k) \to \mathbf{0}$ as $k \to \infty$. But

$$\epsilon(k) \to \mathbf{0} \implies \Big(\delta x(k), e(k-1)\Big) \to \mathbf{0} \tag{4.61}$$

which implies that,

$$x(k) = x(k-1), \tag{4.62a}$$

$$e(k-1) = 0. \tag{4.62b}$$

Equation (4.62) satisfies the conditions of Proposition 1 which shows convergence of the closed-loop system to the optimum of the steady-state optimization problem (4.2) thereby solving Problem 4.

Although convergence of the velocity dynamics has been shown, optimality of the receding-horizon FOLQC law (4.40) with respect to the performance objective (4.3) is not guaranteed. The receding horizon control law (4.40) is in-fact suboptimal. However, if the terminal control law (4.50) is chosen to be the infinite-horizon FOLQC law in (4.26a), i.e.,

$$\mathcal{K}(N) = \mathcal{K}, \tag{4.63}$$

then the receding-horizon control laws (4.40) are optimal with respect to the dynamic performance objective (4.3). □

4.3 An LMI formulation of the deterministic feedback optimizing linear quadratic control

In this section, the deterministic FOLQC problem presented will be revisited in the framework of linear matrix inequalities (LMI). This alternative formulation of the deterministic FOLQC will form the basis for designing robust FOLQC laws under various uncertainty scenarios later on in Chapter 5.

4.3.1 LMI Formulation of the FOLQC Problem

We here recall the FOLQC problem formulation in Section 4.3.3. Consider the deterministic velocity dynamics (4.12), the problem of interest is to design a linear state feedback control law of the form:

$$\delta u(k) = -\mathcal{K}\epsilon(k) \tag{4.64}$$

which minimizes the following performance objective:

$$\begin{aligned} J_\infty(\epsilon(0)) &= \frac{1}{2}\sum_{k=0}^{\infty}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big) \\ &= \frac{1}{2}\sum_{k=0}^{\infty}\epsilon(k)^\top\Big((\mathcal{C}-\mathcal{D}\mathcal{K})^\top Q_e(\mathcal{C}-\mathcal{D}\mathcal{K}) + \mathcal{K}^\top R_\delta \mathcal{K}\Big)\epsilon(k), \end{aligned} \tag{4.65}$$

where $Q_e \succeq \mathbf{0}$ and $R_\delta \succ \mathbf{0}$, so that $J_\infty \succ 0$ for all $\epsilon(k)$ except for $\epsilon(k) = \mathbf{0}$ where $J_\infty = 0$. For finiteness of J, it is necessary that the system (4.12) is stable in closed loop with the control law (4.64). To ensure this, we assume reachability of $(\mathcal{A}, \mathcal{B})$. From the analysis done in section 4.3.3, we know that the optimal value of the cost (4.65) is given by

$$J_\infty^* = \frac{1}{2}\epsilon(0)^\top \mathcal{P}\epsilon(0) \tag{4.66}$$

where $\mathcal{P}$ is any symmetric positive definite matrix satisfying the algebraic Riccati equation (ARE)

$$\mathcal{P} = \mathcal{A}^\top \mathcal{P}\mathcal{A} + \mathcal{Q} - (\mathcal{N} + \mathcal{A}^\top \mathcal{P}\mathcal{B})\mathcal{S}^{-1}(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}\mathcal{A}), \tag{4.67}$$

with $\mathcal{S} = \mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B}$, and $\epsilon(0)$ an arbitrary initial state. The existence of a stabilizing solution $\mathcal{P}$ to the ARE (4.67) is guaranteed by the reachability of $(\mathcal{A}, \mathcal{B})$ and the regularity assumptions on the weights (Q_e, R_δ).
It can also be easily shown that

$$J(\epsilon(k)) = \frac{1}{2}\epsilon(k)^\top \mathcal{P}\epsilon(k) \tag{4.68}$$

is a Lyapunov function for the closed loop system with the control law (4.64).
To solve the FOLQC problem, rather than solving the ARE (4.67), we instead assume $\mathcal{P}$ unknown and utilize the Lyapunov function (4.68) to formulate an LMI which when solved yields $\mathcal{P}$ and the associated FOLQC control law. In order to formulate the LMI problem, we will heavily rely on the following lemma originally proposed in continuous-time by R. Bellman (see [111], Lemma 1).

Lemma 4.3.1 (R. Bellman)**.** *The value of the functional*

$$V(x(0)) = \frac{1}{2}\sum_{k=0}^{\infty} x(k)^\top W x(k), \quad W \succ 0 \tag{4.69}$$

along the solutions of the system

$$x(k+1) = Ax(k),\ x(0) = x_0 \tag{4.70}$$

with stable matrix A is equal to

$$V(x(0)) = \frac{1}{2}x_0^\top P_{eq} x_0, \tag{4.71}$$

where the matrix P_{eq} satisfies the Lyapunov equation

$$A^\top P_{eq} A - P_{eq} = -W. \tag{4.72}$$

Proof. The solution of the system

$$x(k+1) = Ax(k),\ x(0) = x_0 \tag{4.73}$$

with stable matrix A can be shown to be equal to

$$x(k) = A^k x_0. \tag{4.74}$$

Therefore, the functional $V(x(0))$ can be equivalently expressed as

$$V(x(0)) = \frac{1}{2} x_0^\top \left(\sum_{k=0}^{\infty} (A^k)^\top W (A^k) \right) x_0, \quad W \succ 0. \tag{4.75}$$

Letting

$$\begin{aligned} P_{eq} = \sum_{k=0}^{\infty} (A^k)^\top W (A^k) &= W + A^\top W A + (A^2)^\top W (A^2) + (A^3)^\top W (A^3) + \ldots \\ &= W + A^\top (W + A^\top W A + (A^2)^\top W (A^2) + \ldots) A \\ &= W + A^\top P_{eq} A. \end{aligned} \tag{4.76}$$

Then,

$$V(x(0)) = \frac{1}{2} x_0^\top P_{eq} x_0 \tag{4.77}$$

where the matrix P_{eq} satisfies the Lyapunov equation (4.72). □

Remark 18 (The Lyapunov inequality and equation [111, 175])**.** *Let the matrix A be Hurwitz and let $W \succ 0$; then the Lyapunov inequality*

$$A^\top P A - P \preceq -W \tag{4.78}$$

is feasible and any solution P satisfies

$$P \succeq P_{eq}, \tag{4.79}$$

where $P_{eq} \succ 0$ is the solution of the associated Lyapunov equation (4.72)*. This implies that the minimal solution (with respect to the partial order $\preceq$) of the Lyapunov inequality* *is the solution of the associated Lyapunov equation. Therefore, for every initial condition* x_0 *we have*

$$x_0^\top P_{eq} x_0 \preceq x_0^\top P x_0, \tag{4.80}$$

so that the value of the functional V *in the Bellman lemma can be found as a solution of the following semi-definite program (SDP):*

$$\min_{0 \prec P = P^\top} \quad x_0^\top P x_0 \tag{4.81}$$

subject to:

$$A^\top P A - P \preceq -W. \tag{4.82}$$

This is the property we will exploit to reformulate the velocity model based FOLQC problem as a LMI.
It is important to note that although the SDP represented by the equation (4.81) *and* (4.82) *contain the initial condition,* x_0*, in the objective function,* (4.81)*, the resulting optimal control policy is independent of* x_0*. However, when the system matrix* A *is uncertain, then the above formulation would be difficult to implement since every specific admissible uncertainty will result in a range of values for the objective function,* (4.81)*,making it challenging to solve the resulting SDP.*

We now present the following results on the LMI solution of the FOLQC problem.

Theorem 4.3.2 (LMI solution of FOLQC)**.** *Let* $\mathcal{Y}$ *and* $\mathcal{W}$ *be the solution of the following SDP:*

$$\min_{\gamma, \mathcal{W}, 0 \prec \mathcal{Y} = \mathcal{Y}^\top} \quad \gamma \tag{4.83}$$

subject to:

$$\begin{bmatrix} \gamma I & \epsilon(0)^\top \\ \epsilon(0) & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{4.84a}$$

$$\begin{bmatrix} -\mathcal{Y} & (\mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W})^\top & (\mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W})^\top & \mathcal{W}^\top \\ \mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W} & \mathbf{0} & -Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}. \tag{4.84b}$$

Then the controller (4.64) *with*

$$\mathcal{K} = \mathcal{W}\mathcal{Y}^{-1}, \tag{4.85}$$

solves problem 4 (the FOLQC problem) with $J(\epsilon(k)) = \frac{1}{2}\epsilon(k)\mathcal{Y}^{-1}\epsilon(k)$ *a corresponding Lyapunov function guaranteeing stability of the closed-loop system.*

Proof. With the FOLQC state feedback control law (4.64), the closed loop dynamics are

$$\epsilon(k+1) = (\mathcal{A} - \mathcal{B}\mathcal{K})\epsilon(k). \tag{4.86}$$

Using Remark 1, the minimum value of the functional J_∞ in (4.65) subject to the closed loop dynamics (4.86) is given by the solution of the problem:

$$\min_{\mathcal{K}, 0 \prec \mathcal{P} = \mathcal{P}^\top} \quad \epsilon(0)^\top \mathcal{P} \epsilon(0) \tag{4.87}$$

subject to:

$$(\mathcal{A} - \mathcal{B}\mathcal{K})^\top \mathcal{P} (\mathcal{A} - \mathcal{B}\mathcal{K}) - \mathcal{P} \preceq -\Big((\mathcal{C} - \mathcal{D}\mathcal{K})^\top Q_e (\mathcal{C} - \mathcal{D}\mathcal{K}) + \mathcal{K}^\top R_\delta \mathcal{K}\Big). \tag{4.88}$$

The above problem is non-convex in the decision variables $\mathcal{K}$ and $\mathcal{P}$. To convexify, we perform the following manipulations on the inequality (4.88). Pre- and post-multiplying (4.88) by the matrix

$$\mathcal{Y} = \mathcal{P}^{-1} \tag{4.89}$$

and introducing

$$\mathcal{W} = \mathcal{K}\mathcal{Y}, \tag{4.90}$$

we obtain the inequality

$$(\mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W})^\top \mathcal{Y}^{-1} (\mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W}) - \mathcal{Y} + (\mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W})^\top Q_e (\mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W}) + \mathcal{W}^\top R_\delta \mathcal{W} \preceq \mathbf{0}. \tag{4.91}$$

Then applying the Schur complement, we can rewrite (4.91) as the LMI,

$$\begin{bmatrix} -\mathcal{Y} & (\mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W})^\top & (\mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W})^\top & \mathcal{W}^\top \\ \mathcal{A}\mathcal{Y} - \mathcal{B}\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}\mathcal{Y} - \mathcal{D}\mathcal{W} & \mathbf{0} & -Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}. \tag{4.92}$$

Given (4.92), the minimum of J_∞ can then be obtained by solving the SDP:

$$\min_{\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \epsilon(0)^\top \mathcal{Y}^{-1}\epsilon(0) \tag{4.93}$$

subject to the LMI (4.92).
However, the cost (4.93) is non-linear in the variable $\mathcal{Y}$; therefore to linearise, we reformulate the problem above as the SDP:

$$\min_{\gamma,\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \gamma \tag{4.94}$$

subject to:

$$\epsilon(0)^\top \mathcal{Y}^{-1}\epsilon(0) \prec \gamma, \tag{4.95a}$$

$$\begin{bmatrix} -\mathcal{Y} & (\mathcal{A}\mathcal{Y}-\mathcal{B}\mathcal{W})^\top & (\mathcal{C}\mathcal{Y}-\mathcal{D}\mathcal{W})^\top & \mathcal{W}^\top \\ \mathcal{A}\mathcal{Y}-\mathcal{B}\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}\mathcal{Y}-\mathcal{D}\mathcal{W} & \mathbf{0} & -Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}. \tag{4.95b}$$

Although linear in the objective, the above problem now contains the non-linear constraint (4.95a) making it non-convex. To obtain a more tractable formulation, we again make use of the Schur complement to express (4.95a) as the LMI

$$\begin{bmatrix} \gamma & \epsilon(0)^\top \\ \epsilon(0) & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}. \tag{4.96}$$

□

LMI-based FOLQC independent of initial conditions

The FOLQC formulation in theorem 4.3.2 appears to depend on the initial state $\epsilon(0)$ which can make it challenging to implement for uncertain systems (see Remark 18 for details). In this section, we reformulate the LMIs above to be independent of the initial conditions. This will be useful in Chapter 5 where the FOLQC problem is solved under model uncertainty.

To obtain the FOLQC control gain independent of $\epsilon(0)$, we average the solution of the FOLQC problem over all initial conditions in the unit ball [131]. Let $\epsilon(0)$ be random, uniformly distributed on the surface of the unit ball in the Euclidean norm. Then the FOLQC problem can be re-stated as the SDP:

$$\min_{\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \mathbf{E}\Big(\epsilon(0)^\top\mathcal{Y}^{-1}\epsilon(0)\Big) \tag{4.97}$$

subject to the LMI (4.92), where $\mathbf{E}(\cdot)$ is the mathematical expectation of the random variable $\epsilon(0)^\top\mathcal{Y}^{-1}\epsilon(0)$ given by

$$\mathbf{E}\Big(\epsilon(0)^\top\mathcal{Y}^{-1}\epsilon(0)\Big) = \frac{1}{n_\epsilon}tr(\mathcal{Y}^{-1}) \tag{4.98}$$

with $n_\epsilon = n_x + n_y$.

Remark 19. *The SDP obtained by minimizing* (4.97) *subject to* (4.98) *has the following properties [131]:*

1. *it is optimal in an average sense,*
2. *it retains many of the properties of linear systems which are optimal with respect to the standard quadratic criterion.*

Please refer to [131] for detailed analysis of this approach.

We can then obtain the FOLQC gain by minimizing $tr(\mathcal{Y}^{-1})$ subject to the LMI constraint (4.92). This leads to the following result.

Lemma 4.3.3 (LMI solution of FOLQC II)**.** *Let $\mathcal{Y}$ and $\mathcal{W}$ be the solution of the following SDP:*

$$\min_{\gamma,\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \gamma \tag{4.99}$$

subject to:

$$\begin{bmatrix} \gamma I & I \\ I & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{4.100a}$$

$$\begin{bmatrix} -\mathcal{Y} & (\mathcal{A}\mathcal{Y}-\mathcal{B}\mathcal{W})^\top & (\mathcal{C}\mathcal{Y}-\mathcal{D}\mathcal{W})^\top & \mathcal{W}^\top \\ \mathcal{A}\mathcal{Y}-\mathcal{B}\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}\mathcal{Y}-\mathcal{D}\mathcal{W} & \mathbf{0} & -Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}. \tag{4.100b}$$

Then the controller (4.64) *with*

$$\mathcal{K} = \mathcal{W}\mathcal{Y}^{-1} \tag{4.101}$$

solves problem 4 (the FOLQC problem) with $J(\epsilon(k)) = \frac{1}{2}\epsilon(k)\mathcal{Y}^{-1}\epsilon(k)$ *a corresponding Lyapunov function guaranteeing stability of the closed-loop system.*

Proof. The result is obtained by minimizing $tr(\mathcal{Y}^{-1})$ subject to (4.92) which is equivalent to the SDP:

$$\min_{\gamma,\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \gamma \tag{4.102}$$

subject to the constraints $tr(\mathcal{Y}^{-1}) \preceq \gamma$ and (4.92).
Applying the Schur complement to the constraint $\gamma I - tr(\mathcal{Y}^{-1}) \succeq \mathbf{0}$ gives the result obtained. □

Alternatively, we can obtain an $\epsilon(0)$ independent control gain by solving the discrete algebraic Riccati equation (4.26b) via semi-definite programming [34]. We summarize this approach with the result below.

Lemma 4.3.4 (LMI solution of FOLQC III)**.** *Let* $\mathcal{P}$ *be the solution of the following SDP:*

$$\max_{\mathcal{P}=\mathcal{P}^\top} \quad tr(\mathcal{P}) \tag{4.103}$$

subject to:

$$\begin{bmatrix} \mathcal{A}^\top\mathcal{P}\mathcal{A} + \mathcal{Q} - \mathcal{P} & \mathcal{N} + \mathcal{A}^\top\mathcal{P}\mathcal{B} \\ \mathcal{N}^\top + \mathcal{B}^\top\mathcal{P}\mathcal{A} & \mathcal{R} + \mathcal{B}^\top\mathcal{P}\mathcal{B} \end{bmatrix} \succeq \mathbf{0}, \tag{4.104a}$$

$$\mathcal{P} \succ 0. \tag{4.104b}$$

Then the controller (4.64) *with*

$$\mathcal{K} = \left(\mathcal{R} + \mathcal{B}^\top\mathcal{P}\mathcal{B}\right)^{-1}\left(\mathcal{N}^\top + \mathcal{B}^\top\mathcal{P}\mathcal{A}\right), \tag{4.105}$$

solves problem 4 (the FOLQC problem) with $J(\epsilon(k)) = \frac{1}{2}\epsilon(k)\mathcal{P}\epsilon(k)$ *a corresponding Lyapunov function guaranteeing stability of the closed-loop system.*

Proof. From the dynamic programming solution, the FOLQC gain is given by

$$\mathcal{K} = \left(\mathcal{R} + \mathcal{B}^\top\tilde{\mathcal{P}}\mathcal{B}\right)^{-1}\left(\mathcal{N}^\top + \mathcal{B}^\top\tilde{\mathcal{P}}\mathcal{A}\right) \tag{4.106}$$

where $\tilde{\mathcal{P}}$ is any symmetric positive definite matrix satisfying the discrete algebraic Ricatti equation

$$\begin{aligned}&\mathcal{A}^\top \tilde{\mathcal{P}} \mathcal{A} + \mathcal{Q} - \tilde{\mathcal{P}} \\ &\qquad - \left(\mathcal{N} + \mathcal{A}^\top \tilde{\mathcal{P}} \mathcal{B}\right)\left(\mathcal{R} + \mathcal{B}^\top \tilde{\mathcal{P}} \mathcal{B}\right)^{-1}\left(\mathcal{N}^\top + \mathcal{B}^\top \tilde{\mathcal{P}} \mathcal{A}\right) = \mathbf{0}.\end{aligned} \tag{4.107}$$

But as shown in [34], any feasible $\mathcal{P}$ for the Ricatti inequality

$$\begin{aligned}&\mathcal{A}^\top \mathcal{P} \mathcal{A} + \mathcal{Q} - \mathcal{P} \\ &\qquad - \left(\mathcal{N} + \mathcal{A}^\top \mathcal{P} \mathcal{B}\right)\left(\mathcal{R} + \mathcal{B}^\top \mathcal{P} \mathcal{B}\right)^{-1}\left(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P} \mathcal{A}\right) \preceq \mathbf{0}\end{aligned} \tag{4.108}$$

is lower bounded by the solution $\tilde{\mathcal{P}}$ of the Ricatti equation (4.107) i.e.,

$$\mathcal{P} \succeq \tilde{\mathcal{P}}. \tag{4.109}$$

By the above inequality, $\tilde{\mathcal{P}}$ is the lower bound on all $\mathcal{P}$ that are feasible for the LMI (4.108) and finding this lower bound would yield the optimal value of $\tilde{\mathcal{P}}$ that solves the Riccati equation (4.107). Following similar reasoning as [14] and applying the Schur complement on the Riccati inequality, we can then formulate the SDP obtained in Lemma 4.3.4 which concludes the proof. □

4.4 Inherent robustness of the deterministic FOLQC

Under nominal conditions, the deterministic FOLQC is guaranteed closed-loop stable if there exists a solution $\mathcal{P}$ to the discrete algebraic Riccati equation (4.26b) (see Proposition 4). However, this nominal stability may not guarantee the stability of the actual closed-loop system when there is uncertainty in the model used for the control design. Real systems always have model uncertainty and may be plagued by unknown external disturbances. As a result, considering the stability of FOLQC under non-nominal conditions where there may be model uncertainties and unknown external disturbances is vital. The robust stability of standard linear quadratic control has been a widely studied topic with a plethora of results on the subject [133, 6]. It is common knowledge that the guaranteed stability margins of the standard continuous-time linear quadratic control are relatively large with a gain margin of $(0.5, \infty)$ and a phase margin of not less than 60 degrees [133].

Similar to other linear quadratic control algorithms, the deterministic FOLQC is expected to show some robustness to model perturbation. Although results on the robust stability of some feedback-based online steady-state optimization algorithms have been presented in the literature [169, 46], an investigation of the robust stability of feedback-optimizing algorithms that achieve optimal transient performance under a linear-quadratic performance cost have not been made. Also, most studies on the robust stability of feedback-optimizing control have been limited to nominal systems with additive disturbances.

In this section, we study the robustness of the velocity model-based FOLQC algorithm given in Proposition 4. We investigate conditions under which the nominal stability of FOLQC is preserved when the transfer function of the LTI system (4.1) is perturbed. Primarily, we study the sensitivity of the stable modes of the closed-loop system with the deterministic FOLQC control law of Proposition 4 under small variations in the model parameters. The goal is to obtain conditions under which the stable modes of the closed-loop system under deterministic FOLQC remain inside the unit circle for small variations in the model parameters. We suggest guaranteed stability margins (expressed in terms of the elementary cost and system matrices) for the deterministic FOLQC by studying the behaviour of the singular values of the controller return difference matrix. Finally, we investigate the connections between the obtained stability margins and the selection of the weighting matrices for the state, input and cross-product terms. We rely heavily on the results obtained in [6] for the robust stability of discrete-time linear quadratic regulator with cross-product terms.

Preliminaries

Consider the nominal LTI system (4.1) with the corresponding nominal velocity dynamics for the steady-state optimality error (4.10):

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k), \tag{4.110a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k). \tag{4.110b}$$

We can express (4.110a) as the transfer function

$$\frac{\epsilon(z)}{\delta u(z)} = G(z) = (zI - \mathcal{A})^{-1}\mathcal{B}. \tag{4.111}$$

The nominal loop transfer function $T(z)$ for the closed-loop velocity feedback system,

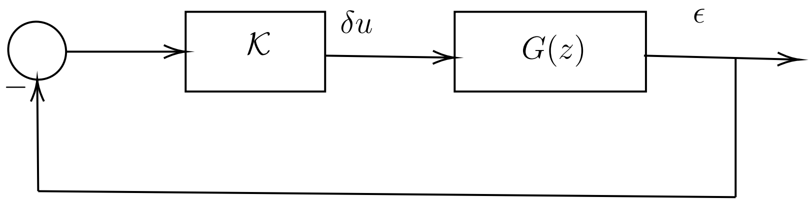

Fig. 4.1 Closed-loop velocity feedback system

depicted in Fig 4.1, is given by

$$T(s) = -\mathcal{K}G(z) = -\mathcal{K}(zI - \mathcal{A})^{-1}\mathcal{B}. \tag{4.112}$$

Due to model uncertainty, assume the model (4.10) is perturbed, and as a result the velocity dynamics (4.110) becomes similarly perturbed by $\Delta T(z)$ resulting in the following perturbed loop transfer function,

$$\tilde{T}(z) = T(z) + \Delta T(z). \tag{4.113}$$

We recall the following well-known result on the robust stability of discrete-time linear systems.

Lemma 4.4.1 (Robust stability of discrete-time LTI systems [6])**.** *Consider a stable discrete-time feedback system with loop transfer function $T(z)$ and return difference matrix $I + T(z)$, and suppose that $T(z)$ undergoes a stable norm-bounded additive change $\Delta T(z)$, i.e., $\tilde{T}(z) = T(z) + \Delta T(z)$ where $\Delta T(z)$ is bounded by some norm (knowledge of this bound is not necessary, only that it is bounded). Then, stability of the closed-loop system will be preserved if*

$$\overline{\sigma}[\Delta T(z)] \leq \underline{\sigma}[I + T(z)] \tag{4.114}$$

for z that traverses the unit circle, where $\overline{\sigma}[\cdot]$ and $\underline{\sigma}[\cdot]$ denote, respectively, the maximum and the minimum singular values.
Moreover, if $\exists \beta \in [0, 1]$, such that $\underline{\sigma}[I + T(z)] \geq \beta$, then multivariable gain and phase margins, GM and PM, respectively, are given by

$$GM = \Big(\frac{1}{1+\beta}, \frac{1}{1-\beta}\Big) \quad \text{and} \quad PM = \pm\ \arccos\Big(1 - \frac{\beta^2}{2}\Big). \tag{4.115}$$

Remark 20. *See section A.5.1 for definition of gain and phase margin.*

It is well known that the eigenvalues of the closed-loop matrix $\mathcal{A}-\mathcal{BK}$, for the velocity dynamics are the zeros of the return difference matrix,

$$\Omega(z) = I + T(z) = I - \mathcal{K}(zI - \mathcal{A})^{-1}\mathcal{B}. \tag{4.116}$$

The matrix $\Omega(z)$ has been shown in [6] to satisfy the following spectral factorization equality:

$$\Omega^\top(z^{-1})(\mathcal{R} + \mathcal{B}^\top\mathcal{P}\mathcal{B})\Omega(z) = \mathcal{R} + \mathcal{S}(z) + \mathcal{W}(z), \tag{4.117}$$

where

$$\begin{aligned} \mathcal{S}(z) &= \mathcal{B}^\top(z^{-1}I - \mathcal{A}^\top)^{-1}\mathcal{Q}(zI - \mathcal{A})^{-1}\mathcal{B}, \text{ and} \\ \mathcal{W}(z) &= \mathcal{B}^\top(z^{-1}I - \mathcal{A}^\top)^{-1}\mathcal{N} + \mathcal{N}^\top(zI - \mathcal{A})^{-1}\mathcal{B}. \end{aligned} \tag{4.118}$$

Guaranteed stability margins for the deterministic FOLQC

Using the return difference equality (4.117), we present the following results [6]

Theorem 4.4.2. *Suppose*

$$\mathcal{S} + \mathcal{W} \succeq 0 \tag{4.119}$$

and

$$\hat{\mathcal{R}} = \mathcal{B}\mathcal{R}^{-1}\mathcal{B}^\top \tag{4.120}$$

is nonsingular. Then the minimum singular value of the return difference matrix $I + T(z)$ is bounded from below for z that traverses the unit circle, by α_ϵ, where

$$\alpha_\epsilon = \left(\frac{\underline{\sigma}[\mathcal{R}]}{\overline{\sigma}[\mathcal{R}] + \overline{\sigma}^2[\mathcal{B}]\phi}\right)^{1/2}, \tag{4.121a}$$

$$\phi = \frac{\psi}{\underline{\sigma}^2[\mathcal{R}]\underline{\sigma}^2[\mathcal{B}]}, \tag{4.121b}$$

and

$$\psi = \left(\underline{\sigma}[\mathcal{R}]\overline{\sigma}[\mathcal{A}] + \overline{\sigma}[\mathcal{B}]\overline{\sigma}[\mathcal{N}]\right)^2 \overline{\sigma}[\mathcal{R}] + \underline{\sigma}[\mathcal{R}]\underline{\sigma}^2[\mathcal{B}]\left(\underline{\sigma}[\mathcal{R}]\overline{\sigma}[\mathcal{Q}] + \overline{\sigma}^2[\mathcal{N}]\right). \tag{4.122}$$

The guaranteed gain and phase margins of the deterministic FOLQC are then given by

$$GM_{\alpha_\epsilon} = (1 \pm \alpha_\epsilon)^{-1}, \quad PM_{\alpha_\epsilon} = \pm \arccos\left(1 - \frac{\alpha_\epsilon^2}{2}\right). \tag{4.123}$$

The direct consequence of Theorem 4.4.2 is the following.

Proposition 5. *Suppose* $\mathcal{Q} \succ 0$ *and* $\mathcal{W} \succeq 0$ *and that* $det(\hat{\mathcal{R}}) \neq 0$. *Then the lower bound of the form* (4.121a), *for* $\underline{\sigma}[I + T(z)]$, *as well as the stability margins of the form* (4.123) *are guaranteed for the deterministic FOLQC.*

Proof. See [6] for details □

Because the conditions presented above may be somewhat challenging to check or sometimes might not hold, we present in the following, an alternative result that submits easily to numerical verification.

Theorem 4.4.3. *Suppose*

$$\mathcal{Q} - I \succ 0, \quad \text{and} \quad \mathcal{R} \succeq \mathcal{N}^\top \mathcal{N} \succeq 0 \tag{4.124}$$

and that $det(\hat{\mathcal{R}}) \neq 0$. *Then the minimum singular value of the return difference matrix* $I + T(z)$ *of the deterministic FOLQC is bounded from below, for* z *that traverses the unit circle (i.e.* $\|z\| = 1$*), by* b*, where*

$$b = \left(\frac{\underline{\sigma}[\mathcal{R} - \mathcal{N}^\top \mathcal{N}]}{\overline{\sigma}[\mathcal{R}] + \overline{\sigma}^2[\mathcal{B}]\phi} \right)^{1/2} \tag{4.125}$$

with ϕ *given in* (4.121b). *The guaranteed gain and phase margins of the deterministic FOLQC are then given by*

$$GM_b = (1 \pm b)^{-1}, \quad \text{and } PM_b = \pm \arccos\left(1 - \frac{b^2}{2}\right). \tag{4.126}$$

Proof. See [6] for details. □

4.5 Relation with Other Approaches

4.5.1 Optimal Steady-State (OSS) Control

In [128], an OSS control algorithm was developed for regulating an LTI system to steady-states that solve the optimization problem (4.2) in feedback. The controller is designed in continuous-time and is based on an optimality model $\gamma(u, y)$ for problem (4.2) that serves as a proxy for the error in steady-state optimality. The designed OSS control law is given by the Proportional Integral (PI) controller,

$$\dot{\eta} = \gamma(u, y) = e, \tag{4.127a}$$
$$u = K_P e + K_I \eta \tag{4.127b}$$

where η is an integral of the error in optimality of (4.2).
We may discretize (4.127) to obtain a corresponding discrete-time PI controller

$$u(k) = u(k-1) - \mathcal{K}_{pi}\epsilon(k), \tag{4.128}$$

where $\mathcal{K}_{pi} = \begin{bmatrix} K_P C & K_I \delta t \end{bmatrix}$ and δt is the sampling interval. The control gains K_P and K_I in [128] were chosen via trial and error to achieve a stable closed loop system. By setting $\mathcal{K}_{pi} = \mathcal{K}$, optimal control gains with respect to the infinite-horizon criterion (4.38) are obtained. The proposed FOLQC approach therefore offers a more systematic approach to designing the (discrete-time version of the) OSS control law presented in [128].

4.6 Illustrative Example

We consider the continuous-time open-loop unstable system :

$$\dot{x}(t) = \begin{bmatrix} -1 & 3 \\ 5 & 2 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 1 \\ 1 & 4 \end{bmatrix} u(t) + w(t), \tag{4.129a}$$
$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t). \tag{4.129b}$$

The system is stabilizable and observable, meeting Assumption 3. The disturbance $w(t)$ is unknown but slowly varying as

$$w(t) = \begin{cases} \begin{bmatrix} -1 & 3 \end{bmatrix}^\top & 0 \le t < 20, \\ \begin{bmatrix} 2 & -3 \end{bmatrix}^\top & 20 \le t < 40, \\ \begin{bmatrix} 1 & 0 \end{bmatrix}^\top & t \ge 40. \end{cases} \tag{4.130}$$

The objective function (4.9) is used for the steady-state optimization problem with $Q_{uu} = 10 \times I_2,\ Q_{uy} = Q_{yu}^\top = 5 \times \mathbf{1}_{2\times 2},\ Q_{yy} = 5 \times I_2,\ R_u = \mathbf{1}_{2\times 1},$ and $R_y = \mathbf{1}_{2\times 1}$. For design and implementation of the discrete-time FOLQC, the system is discretized using zero-order hold with a sampling time of 0.1 seconds. The corresponding input–output DC gain matrix of the discrete-time system is,

$$G_u = \begin{bmatrix} -0.1765 & -0.5882 \\ -0.5882 & -0.5294 \end{bmatrix}. \tag{4.131}$$

The velocity state for the controller can be obtained from

$$\epsilon(k) = \begin{bmatrix} x(k) - x(k-1) \\ \Lambda_y y(k-1) + \Lambda_u u(k-1) + r \end{bmatrix} \tag{4.132}$$

where Λ_y, Λ_u and r follow from the choices of Φ parameters as

$$\Lambda_y = \begin{bmatrix} -35 & -40 \\ 35 & 40 \end{bmatrix}, \ \Lambda_u = \begin{bmatrix} -85 & 15 \\ 105 & -25 \end{bmatrix}, \text{ and } r = \begin{bmatrix} -7 \\ 8 \end{bmatrix}. \tag{4.133}$$

The matrix

$$S = \begin{bmatrix} 0.019 & -0.3245 & -0.0157 & -0.1606 \\ -0.5409 & -0.3055 & -0.1134 & -0.4799 \\ -35 & -40 & -85 & 15 \\ 35 & 40 & 105 & -25 \end{bmatrix} \tag{4.134}$$

has full rank satisfying Assumption 4. The transient performance criterion is chosen with $Q_e = 500 \times I_2$ and $R_\delta = 1 \times I_2$; these values satisfy the hypothesis of Proposition 3.

From the design parameters above, the FOLQC has the following parameters:

$$\mathcal{A} = \begin{bmatrix} 0.9810 & 0.3245 & 0 & 0 \\ 0.5409 & 1.3055 & 0 & 0 \\ -35.00 & -40.00 & 1.00 & 0 \\ 35.00 & 40.00 & 0 & 1.00 \end{bmatrix}, \ \mathcal{B} = \begin{bmatrix} 0.0157 & 0.1606 \\ 0.1134 & 0.4799 \\ -85.00 & 15.00 \\ 105.00 & -25.00 \end{bmatrix}$$
$$\mathcal{C} = \begin{bmatrix} -35.00 & -40.00 & 1.00 & 0 \\ 35.00 & 40.00 & 0 & 1.00 \end{bmatrix}, \ \mathcal{D} = \begin{bmatrix} -85.00 & 15.00 \\ 105.00 & -25 \end{bmatrix},$$
$$\mathcal{R} = \begin{bmatrix} 9.125 & -1.95 \\ -1.95 & 0.425 \end{bmatrix} \times 10^6, \ \mathcal{N} = \begin{bmatrix} 3.325 & -0.700 \\ 3.800 & -0.800 \\ -0.0425 & 0.0075 \\ 0.0525 & -0.0125 \end{bmatrix} \times 10^6, \tag{4.135}$$
$$\mathcal{Q} = \begin{bmatrix} 1.2250 & 1.40 & -0.0175 & 0.0175 \\ 1.400 & 1.600 & -0.02 & 0.02 \\ -0.0175 & -0.02 & 0.0005 & 0 \\ 0.0175 & 0.02 & 0 & 0.0005 \end{bmatrix} \times 10^6, \text{ and}$$
$$\mathcal{K} = \begin{bmatrix} 0.6364 & 0.7273 & -0.0454 & -0.0273 \\ 1.2727 & 1.4545 & -0.1908 & -0.1545 \end{bmatrix}.$$

Figure 4.2 shows the result of applying FOLQC to the system while the disturbances change in the way described.

Remark 21. *The FOLQC gain implemented in this example was computed by all the methods described in the Chapter and they all gave approximately the same solution.*

It can be seen that FOLQC tracks $z^*(\bar{w})$ asymptotically, and also maintains stability and good transient performance, overcoming the issues of (i) non-existence of stabilizing PI gains for the static OSS controller presented in [128], and (ii) poor dynamic performance (large overshoot and slow convergence) and potential high dimensionality of the dynamic OSS controller proposed in the same work. Moreover, it should be noted that designing the FOLQC controller for closed-loop stability and optimal steady-state tracking is significantly easier, and more systematic, than the manual tuning of the PI OSS controller of [128]; indeed, for the presented example it was not possible to find stabilizing PI gains for the system under OSS control. However, a limitation of FOLQC is the need for full state measurement.
A benefit of the FOLQC over standard feed-forward optimization/control is its inherent

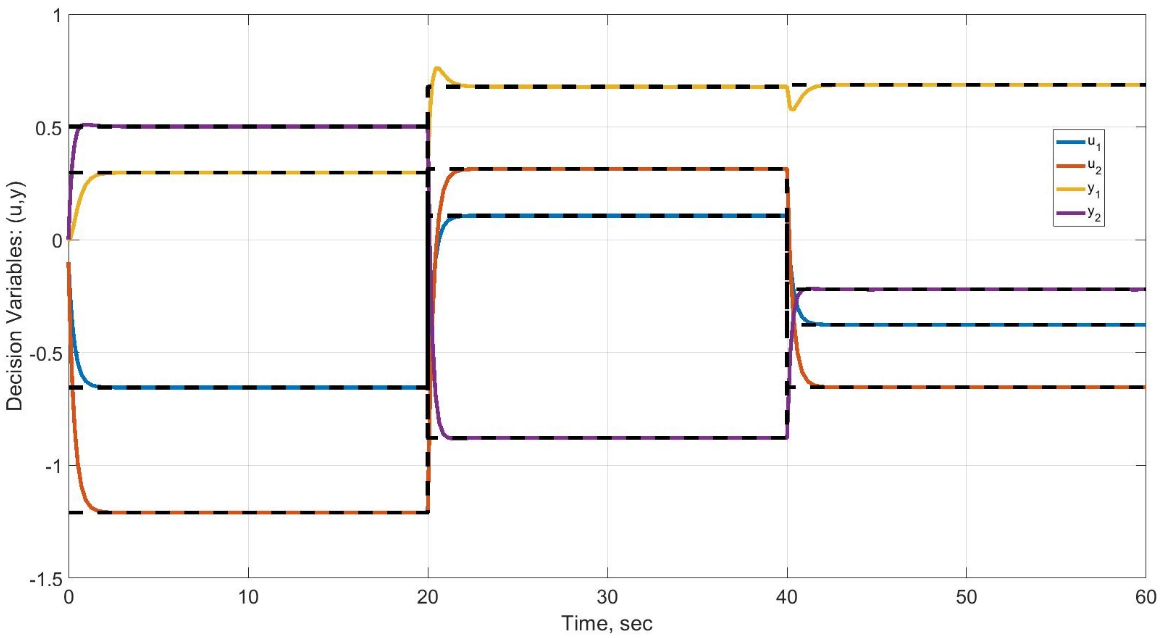

Fig. 4.2 FOLQC: outputs $(y_1, y_{2)}$ and input $(u_1, u_{2)}$ plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

robustness to model uncertainty. In the next chapter, this inherent robustness will be demosntrated for a system with model uncertainty.

4.7 Conclusion

An LQC approach to asymptotically track the unknown optimum to a steady-state optimization problem while minimizing a transient performance criterion has been presented. The approach avoids the need to measure or estimate system disturbances (if they are piecewise-constant) or set-point. Unlike related algorithms, our controller is of a low order and can systematically guarantee a priori, the stability and dynamic performance of the closed loop system. However, it is limited by the lack of constraint handling capabilities. In the presence of hard inequality constraints, the performance of the controller may be severally degraded with possible loss of stability. Future work will focus on developing the approach in several directions: for instance, systems subject to constraints, which pose challenges to how the optimal solution to the steady-state problem is characterized and how the tracking error is defined and regulated. Also dealing with model or parametric uncertainty can be challenging. These issues will be the subject of subsequent chapters.

Chapter 5

Robust Feedback Optimizing Linear Quadratic Control

For most feedback optimization algorithms, a knowledge of the input-output sensitivity function is a pre-requisite. This function is often obtained from precise knowledge of the system dynamics or via system identification. However, most real world systems are plagued with uncertainty which makes it difficult to have access to accurate system models. This issue of model uncertainty have not been given much attention in the feedback optimization literature as most algorithms rely on inherent robustness to deal with the effects of model uncertainty. In this chapter, we make pioneering efforts at developing novel feedback optimization algorithms for linear systems with parametric uncertainty. We develop our algorithm by extending the FOLQC algorithm presented in the previous chapter to the case of polytopic uncertain linear systems with quadratic steady state optimization. Firstly, we develop a model of the uncertain polytopic linear system. Then using linear matrix inequalities (LMI), we derive robust FOLQC laws from the solution to robust semi-definite programs.

5.1 Introduction

In the feedback optimization literature, very few results directly consider the model uncertainty in the feedback optimizing control design. Instead, the focus has been on analysing the inherent robustness of feedback optimization controllers. In [46, 173], theoretical analysis of the inherent robustness of feedback optimization was made. The obtained results indicate that feedback optimizing control is highly robust against model inaccuracy. Also in [169], the inherent robustness of feedback optimization was experimentally verified. Attempts made to synthesize robust feedback optimizing controllers have either imposed restrictive assumptions on the system (such as having a robust feasible subspace property [127]), or adopted a data-driven approach to either estimate the uncertain model online [169, 174, 30, 168] or directly learn the feedback optimizing control law from the input-output data [90]. However no rigorous design of robust feedback optimization has been made such that the controller is dynamically optimal and purposefully robust to a pre-specified class of uncertainty in the dynamic model. In this chapter, we propose a robust feedback optimization control algorithm that directly utilizes information about the uncertainty to compute optimal control laws that drive a system autonomously to the unknown optimum of an uncertain quadratic program.

5.2 Uncertain System Model

Due to parametric uncertainties and non-linearities, a discrepancy between the mathematical model and the real system always exists. In the previous chapters, this discrepancy was either ignored or captured by the additive disturbances. However, some uncertainties may be multiplicative and therefore cannot be accurately captured by additive disturbances. To characterise these uncertainties, we adopt the following linear parameter varying (LPV) model

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k) + Ew(k), \tag{5.1a}$$

$$y(k) = Cx(k), \tag{5.1b}$$

$$\delta \in \Delta_l,\ w(k) \in \mathbb{W}. \tag{5.1c}$$

Here $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$, $y(k) \in \mathbb{R}^{n_y}$, and $w(k) \in \mathbb{R}^{n_w}$ are the state, input, output and additive uncertainty (disturbance) vectors respectively. The system matrices $A(\delta) \in \mathbb{R}^{n_x \times n_x}$, $B(\delta) \in \mathbb{R}^{n_x \times n_u}$, $C \in \mathbb{R}^{n_y \times n_x}$ and $E \in \mathbb{R}^{n_x \times n_w}$ are the uncertain system coefficients and $\delta = \begin{bmatrix}\delta_1 & \dots & \delta_l\end{bmatrix}^\top \in \mathbb{R}^l$ is a vector which ensembles all uncertain parameters in the system coefficients and is here assumed to take values from the $l-$unit simplex:

$$\Delta_l \coloneqq \left\{\delta \in \mathbb{R}^l :\ \sum_{i=1}^{l} \delta_i = 1,\ \delta_i \geq 0, \forall i \in \mathbb{I}_{[1:l]}\right\}. \tag{5.2}$$

Due to the polytopic nature of the unit simplex Δ_l, the uncertain coefficient matrices take values from the convex set Ω i.e.

$$\Big(A(\delta), B(\delta)\Big) \in \Omega \tag{5.3}$$

where Ω is defined as the polytope

$$\Omega := \ Co\Big\{\Big(A_1, B_1\Big), \ldots, \Big(A_l, B_l\Big)\Big\} = \sum_{i=1}^{l} \alpha_i \Big(A_i, B_i\Big). \tag{5.4}$$

Here Co refers to the convex hull and α_i are convex coefficients i.e. $\alpha_i > 0$ and $\sum_i^l \alpha_i = 1$.

The additive uncertainty is assumed as in the previous chapters to take values from the bounded set $\mathbb{W}$ i.e., $w(k) \in \mathbb{W}$.

For the robust control design undertaken in this chapter, the uncertain system (5.1) is required to be quadratically stabilizable. We define quadratic stabilizability below.

Definition 4 (Quadratic stabilizability)**.** *The uncertain linear system* (5.1) *is said to be quadratically stabilizable for perturbations* $\delta \in \Delta_l$ *if there exists matrices* $P \succ 0$ *and* K *such that the undisturbed system satisfies the inequality*

$$\Big(A(\delta) - B(\delta)K\Big)^{\top} P\Big(A(\delta) - B(\delta)K\Big) - P \preceq 0 \text{ for all } \delta \in \Delta_l. \tag{5.5}$$

Remark 22. *The rationale for this definition becomes clearer when considering the quadratic Lyapunov function*

$$V(x(k)) = x(k)^{\top} P x(k) \tag{5.6}$$

needed to guarantee stability of the undisturbed uncertain system

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k), \ \ \delta \in \Delta_l, \tag{5.7}$$

under the control law

$$u(k) = -Kx(k). \tag{5.8}$$

Indeed if K *and* $P \succ 0$ *both satisfy* (5.5), *then* $V(x(k))$ *and*

$$\Delta V(x(k)) = V(x(k+1)) - V(x(k)) \tag{5.9}$$

can be easily shown to be positive definite and negative semi-definite respectively, which ensures that (5.6) *is a Lyapunov function that guarantees closed loop stability of the undisturbed uncertain system dynamics* (5.7) *under the control law* (5.8)*.*

In the developments that follow, quadratic stabilizability will be assumed.

Assumption 5 (Quadratic stabilizability)**.** *The uncertain system* (5.1) *belongs to the class of quadratically stabilizable systems.*

5.3 The Robust Feedback-Optimizing Linear Quadratic Control Problem

The main objective of the robust feedback optimizing linear quadratic control (FOLQC) is to regulate the inputs and/or outputs of a disturbed linear time-invariant system to an equilibrium point that is the solution to an uncertain steady-state optimization problem. This regulation should be achieved without knowledge of the optimal steady-state set-points or explicit solution of the steady-state optimization problem, while also guaranteeing optimal dynamic performance between steady-states. We define the FOLQC problem precisely as follows.

Problem 5 (The robust FOLQC Problem)**.** *Design for the linear discrete-time uncertain system* (5.1) *a state feedback control law*

$$u(k) = \kappa_N(x(k), u(k-1)) \tag{5.10}$$

obtained from the solution to an optimal control problem, such that for any admissible $\bar{w} \in \mathbb{W}$ *and* $\forall \delta \in \Delta$*:*

1. *The point* $\bar{z}^*(\delta, \bar{w})$ *is an asymptotically stable equilibrium for the closed-loop system, satisfying*
$$\lim_{k \to \infty} (u(k), y(k)) = \bar{z}^*(\delta, \bar{w}). \tag{5.11}$$
2. *The feedback policy* $\kappa_N(\cdot, \cdot)$ *minimizes a transient performance criterion.*

Here, $\bar{z}^*(\delta, \bar{w})$ *is the solution to the uncertain steady-state optimization problem*

$$\mathcal{RTO}(\delta, \bar{w}): \quad \bar{z}^*(\delta, \bar{w}) = \arg\min_{\bar{z}} \left\{ \Phi(\bar{z}) \colon \bar{z} \in \mathbb{F}(\delta, \bar{w}),\ \bar{z} \in \mathbb{Z} \right\} \tag{5.12}$$

where $\Phi\colon \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ *is a continuous steady-state (economic) performance objective, the set* $\mathbb{F}(\delta, \bar{w})$ *is the feasible set of the uncertain system as defined in Chapter 2 and* $\mathbb{Z}$ *is a steady-state inequality constraints on* z*.*

5.4 Uncertain velocity model

For the purpose of control design, we can express the LPV model above in velocity form for the steady-state optimization (5.12) and the associated optimality tracking error (3.34) as :

$$\epsilon(k+1) = \mathcal{A}(\delta)\epsilon(k) + \mathcal{B}(\delta)\delta u(k) + \mathcal{E}\delta w(k), \tag{5.13a}$$

$$e(k) = \mathcal{C}(\delta)\epsilon(k) + \mathcal{D}(\delta)\delta u(k), \tag{5.13b}$$

where

$$\epsilon(k) := \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x(k) &:= x(k) - x(k-1), \\ \delta u(k) &:= u(k) - u(k-1), \\ \delta w(k) &:= w(k) - w(k-1), \end{aligned} \tag{5.14}$$

and

$$\mathcal{A}(\delta) = \begin{bmatrix} A(\delta) & \mathbf{0}_{n_x \times n_y} \\ \Lambda_y(\delta)C & I_{n_y} \end{bmatrix}, \qquad \mathcal{B}(\delta) = \begin{bmatrix} B(\delta) \\ \Lambda_u(\delta) \end{bmatrix}, \tag{5.15a}$$

$$\mathcal{C}(\delta) = \begin{bmatrix} \Lambda_y(\delta)C & I_{n_y} \end{bmatrix}, \qquad \mathcal{D}(\delta) = \Lambda_u(\delta), \qquad \mathcal{E} = \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix}. \tag{5.15b}$$

We characterise the uncertainty in the velocity dynamics with the following sets:

$$\Big(\mathcal{A}(\delta), \mathcal{B}(\delta), \mathcal{C}(\delta), \mathcal{D}(\delta)\Big) \in \Omega_\delta, \quad \delta w(k) \in \mathbb{D} \tag{5.16}$$

where

$$\Omega_\delta := \ Co\Big\{\Big(\mathcal{A}_1, \mathcal{B}_1, \mathcal{C}_1, \mathcal{D}_1\Big), \ldots, \Big(\mathcal{A}_l, \mathcal{B}_l, \mathcal{C}_l, \mathcal{D}_l\Big)\Big\} = \sum_{i=1}^{l} \alpha_i \Big(\mathcal{A}_i, \mathcal{B}_i, \mathcal{C}_i, \mathcal{D}_i\Big) \tag{5.17}$$

and $\mathbb{D} := \mathbb{W} + (-\mathbb{W})$.

Remark 23 (Nominal coefficients). *We can define nominal coefficients for the velocity dynamics via the centroid of the set Ω_δ i.e*

$$\Big(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}\Big) := \frac{1}{l}\sum_{i=1}^{l}\Big(\mathcal{A}_i, \mathcal{B}_i, \mathcal{C}_i, \mathcal{D}_i\Big). \tag{5.18}$$

5.5 Robust FOLQC under polytopic uncertainty: the case of piece-wise constant disturbances

In this section, we present robust formulations of the FOLQC problem under polytopic model uncertainty. We propose a solution to the FOLQC problem where the worst-case performance objective is minimized for the uncertainties in the system. We adopt an LMI approach to formulating the robust FOLQC control laws due to the convenience and elegance of LMIs in dealing with uncertainty.

Problem formulation

Consider the following velocity dynamics (5.13) for the uncertain system (5.1), the steady-state optimization (5.12) and under piece-wise constant disturbances i.e $\delta w(k) = 0$, then the resulting velocity dynamics become:

$$\epsilon(k+1) = \mathcal{A}(\delta)\epsilon(k) + \mathcal{B}(\delta)\delta u(k), \tag{5.19a}$$

$$e(k) = \mathcal{C}(\delta)\epsilon(k) + \mathcal{D}(\delta)\delta u(k) \tag{5.19b}$$

$$\tag{5.19c}$$

where for every instant k, $\delta \in \Delta_l$.
Our goal is to find the control law,

$$\delta u(k) = -\check{\mathcal{K}}\epsilon(k) \tag{5.20}$$

that minimizes the worst-case performance index

$$\check{J}(\epsilon(0)) = \max_{\delta\in\Delta_l} J_\infty(\epsilon(0)) \tag{5.21}$$

subject to $\lim_{k\to\infty} e(k) = 0$, where $J_\infty(\epsilon(0))$ is defined as

$$J_\infty(\epsilon(0)) = \frac{1}{2}\sum_{k=0}^{\infty}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big), \tag{5.22}$$

with $Q_e \succeq 0$ and $R_\delta \succ 0$.
Because the set Δ_l contains an infinite number of points, maximizing $J_\infty(\epsilon(0))$ over Δ_l is in general computationally challenging which makes the above min-max problem computationally intractable. To address this challenge, we derive an upper bound on the robust performance index $\check{J}(\epsilon(0))$. We then instead minimize this upper bound to obtain the robust FOLQC law (5.20).

Derivation of the upper bound

To derive an upper bound on the performance index $\check{J}$, we first define the following quadratic function of the velocity state $\epsilon(k)$

$$V(\epsilon(k)) = \frac{1}{2}\epsilon(k)^\top \mathcal{P}\epsilon(k),\ \mathcal{P} \succ 0. \tag{5.23}$$

At sampling time k, suppose V is a Lyapunov function for the velocity dynamics (5.19) and therefore satisfies the following inequality for any $\delta \in \Delta_l$, $k > 0$:

$$V(\epsilon(k+1)) - V(\epsilon(k)) \leq -\frac{1}{2}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big). \tag{5.24}$$

For $\check{J}$ to be finite, we must have $e(k) \to \mathbf{0}$ as $k \to \infty$ and hence $V(\epsilon(k)) \to 0$. Summing (5.24) from $k = 0$ to $k = \infty$, we get

$$\begin{aligned} -V(\epsilon(0)) \leq -\frac{1}{2}\sum_{k=0}^{\infty}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big) \\ \implies -V(\epsilon(0)) \leq -J_\infty(\epsilon(0)). \end{aligned} \tag{5.25}$$

Therefore,

$$\max_{\delta\in\Delta_l} J_\infty \leq V(\epsilon(0)). \tag{5.26}$$

The equation (5.26) gives the upper bound on the robust performance objective $\check{J}(\epsilon(0))$ as the quadratic function, $V(\epsilon(0))$. This upper bound just as $\check{J}(\epsilon(0))$ also depends on the initial condition, $\epsilon(0)$. Given (5.26), the robust FOLQC problem can then be redefined to synthesize a constant velocity state feedback control law, (5.20), to minimize $V(\epsilon(0))$. In the next section, a theorem for computing this robust FOLQC law will be presented.

Robust FOLQC law

Theorem 5.5.1 (LMI solution of robust FOLQC)**.** *Let $\epsilon(k) = \epsilon(0)$ be the state of the uncertain velocity dynamics* (5.20) *measured at sampling time k. Suppose the uncertain parameter vector δ takes values from the unit simplex Δ_l, then the robust state feedback matrix $\check{\mathcal{K}}$ in the control law* (5.20) *which minimizes the upper bound $V(\epsilon(0))$ on the robust performance objective* (5.21) *at sampling time k is given by*

$$\check{\mathcal{K}} = \mathcal{W}\mathcal{Y}^{-1} \tag{5.27}$$

where $\mathcal{Y} \succ 0$ and $\mathcal{W}$ are obtained from the solution (if it exists) of the following SDP:

$$\min_{\gamma, \mathcal{W}, 0 \prec \mathcal{Y} = \mathcal{Y}^\top} \quad \gamma \tag{5.28}$$

subject to :

$$\begin{bmatrix} 1 & \epsilon(0)^\top \\ \epsilon(0) & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{5.29a}$$

$$\begin{bmatrix} -\mathcal{Y} & \left(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\right)^\top & \left(\mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W}\right)^\top & \mathcal{W}^\top \\ \mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W} & \mathbf{0} & -\gamma Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -\gamma R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}, \ \forall \delta \in \Delta_l. \tag{5.29b}$$

The controller (5.27) *solves problem 4 (the FOLQC problem, see chapter 4) for the uncertain system* (5.1) *under constant (or slowly varying) disturbances, $w(k)$ with $V(\epsilon(k)) = \frac{1}{2}\epsilon(k)\mathcal{P}\epsilon(k)$, $\mathcal{P} = \gamma\mathcal{Y}^{-1}$, a corresponding Lyapunov function guaranteeing the stability (i.e., convergence to the unknown steady-state optimum) of the resulting closed-loop system.*

Proof. Minimization of $V(\epsilon(0)) = \frac{1}{2}\epsilon(0)^\top \mathcal{P}\epsilon(0)$, $\mathcal{P} \succ 0$ is equivalent to the problem

$$\min_{\gamma,\mathcal{P}} \gamma \quad \text{subject to:} \quad \frac{1}{2}\epsilon(0)^\top \mathcal{P}\epsilon(0) \le \gamma. \tag{5.30}$$

Defining $\mathcal{Y} = \gamma\mathcal{P}^{-1} \succ 0$ and using the Schur complement, this minimization problem is equivalent to

$$\min_{\gamma,\mathcal{W},0\prec\mathcal{Y}=\mathcal{Y}^\top} \quad \gamma \tag{5.31}$$

subject to :

$$\begin{bmatrix} 1 & \epsilon(0)^\top \\ \epsilon(0) & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}. \tag{5.32}$$

The quadratic function $V(\epsilon(0))$ is also to satisfy (5.24). Substituting $\delta u(k) = -\check{\mathcal{K}}\epsilon(k)$, $e(k) = \big(\mathcal{C}(\delta) - \mathcal{D}(\delta)\check{\mathcal{K}}\big)\epsilon(k)$ and $\epsilon(k+1) = \big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\check{\mathcal{K}}\big)\epsilon(k)$, the inequality (5.24) becomes:

$$\begin{aligned} \big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\check{\mathcal{K}}\big)^\top \mathcal{P}\big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\check{\mathcal{K}}\big) - \mathcal{P} + \big(\mathcal{C}(\delta) - \mathcal{D}(\delta)\check{\mathcal{K}}\big)^\top Q_e\big(\mathcal{C}(\delta) - \mathcal{D}(\delta)\check{\mathcal{K}}\big) \\ + \check{\mathcal{K}}^\top R_\delta \check{\mathcal{K}} \le 0. \end{aligned} \tag{5.33}$$

Performing the following change of variables:

$$\mathcal{P} = \gamma\mathcal{Y}^{-1} \text{ and } \mathcal{W} = \check{\mathcal{K}}\mathcal{Y}, \tag{5.34}$$

the following inequality is obtained

$$\begin{aligned} \big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\big)^\top \mathcal{Y}^{-1}\big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\big) - \mathcal{Y} + \big(\mathcal{C}(\delta)\mathcal{Y} + \mathcal{W}^\top(\gamma R_\delta^{-1})^{-1}\mathcal{W} \\ -\mathcal{D}(\delta)\mathcal{W}\big)^\top (Q_e^{-1})^{-1}\big(\mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W}\big) \le 0. \end{aligned} \tag{5.35}$$

Applying the Schur complement, the inequality (5.35) can be expressed as the linear matrix inequality

$$\begin{bmatrix} -\mathcal{Y} & \big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\big)^\top & \big(\mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W}\big)^\top & \mathcal{W}^\top \\ \mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W} & \mathbf{0} & -\gamma Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -\gamma R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}. \tag{5.36}$$

The inequality (5.36) is affine in the coefficient matrices $\Big(\mathcal{A}(\delta), \mathcal{B}(\delta), \mathcal{C}(\delta), \mathcal{D}(\delta)\Big)$, hence it is satisfied for all $\Big(\mathcal{A}(\delta), \mathcal{B}(\delta), \mathcal{C}(\delta), \mathcal{D}(\delta)\Big) \in \Omega_\delta$ if and only if there exists $\mathcal{Y} \succ 0$, $\mathcal{W} = \check{\mathcal{K}}\mathcal{Y}$ and γ such that the LMI (5.36) is satisfied for all $\delta \in \Delta_l$ which concludes the proof. □

Remark 24. *Although the optimal control policy does not depend on the initial state, $\epsilon(0)$, solving the SDP (minimize (5.28) subject to (5.29a) and (5.29b)) does require a knowledge of $\epsilon(0)$. Because the system is uncertain, $\epsilon(0)$ will belong to a set rather than being single valued making it challenging to properly set up and solve the SDP above. To avoid this difficulty, the result of Lemma 4.3.3 can be utilized to reformulate the problem as the following $\epsilon(0)-$ independent SDP:*

$$\min_{\gamma, \mathcal{W}, 0 \prec \mathcal{Y} = \mathcal{Y}^\top} \quad \gamma \tag{5.37}$$

subject to :

$$\begin{bmatrix} \gamma I & I \\ I & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{5.38a}$$

$$\begin{bmatrix} -\mathcal{Y} & \Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)^\top & \Big(\mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W}\Big)^\top & \mathcal{W}^\top \\ \mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W} & \mathbf{0} & -\gamma Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -\gamma R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}, \ \forall \delta \in \Delta_l. \tag{5.38b}$$

Remark 25 (Dynamic performance and robustness). *The robust FOLQC controller proposed in Theorem 5.5.1 minimizes the upper bound on the dynamic performance cost $J_\infty(\epsilon(0))$ for all $\delta \in \Delta_l$. As a result, the robust FOLQC controller is not guaranteed to optimize the actual performance cost for the true value of the uncertain parameter δ. However, the robust FOLQC can guarantee the performance of the closed loop uncertain system is better than or equal to the worst-case (optimal) performance achievable within the uncertainty range Δ_l. This means that robust FOLQC trades performance for robustness. This is indeed not unique to the robust FOLQC algorithm above but a fundamental limitation of all control systems. In the simulation results* *shown in the next section, this trade-off between performance and robustness will be observed in multiple occasion. Although interesting, we will defer any investigation of this trade-off between robustness and performance to future research and will refer the reader to the excellent work done by [32, 58] on this topic.*

Remark 26 (The curious case of no solution)**.** *One may wonder what conclusions can be made if the semi-definite program stated in Theorem 5.5.1 has no solution. In this case, the robust FOLQC problem cannot be solved using the LMI approach described in the theorem for the given uncertainty interval* Δ_l*. This would normally be the case for uncertain systems that do not satisfy the quadratic stabilizabilty assumption made in Assumption 5. A way to fix this would be to redefine the uncertainty interval* Δ_l *or adopt an alternative approach to the robust control design.*

5.6 Numerical Simulation and Comparative Studies

In this section, numerical simulation and comparative studies of robust feedback optimizing control formulations will be presented. We study the performance of a conventional tracking linear quadratic controller, nominal FOLQC and the robust FOLQC developed in this chapter. The following uncertain linear system will be used for this study.

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k) + Ew(k), \tag{5.39a}$$

$$y(k) = Cx(k), \tag{5.39b}$$

where

$$A(\delta) = \begin{bmatrix} 2 & 0.1 \\ 0 & 1 - 0.1\delta \end{bmatrix}, \ B(\delta) = \begin{bmatrix} 0 \\ 0.1\delta \end{bmatrix}, \ E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and } C = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{5.40}$$

with $\delta \in \Delta_l$ and $\Delta_l \coloneqq \{\delta \mid \delta \in [4, 10]\}$.
We assume the system is quadratically stabilizable and the disturbance $w(k)$ is unknown and constant with the value

$$w(t) = \begin{bmatrix} -1 & 3 \end{bmatrix}^\top, \ 0 \le k < 500. \tag{5.41}$$

In a steady-state, the system (5.39) satisfies the following equation

$$0 = A(\delta)x + B(\delta)u + Ew, \tag{5.42a}$$

$$y = Cx. \tag{5.42b}$$

From (5.42) the steady-state input-output map for (5.39) is computed as

$$y = h(u, w) = G_u u + G_w w, \tag{5.43}$$

where

$$\begin{aligned} G_u(\delta) &= C(I_n - A(\delta))^{-1}B(\delta) = -0.1, \\ G_w(\delta) &= C(I_n - A(\delta))^{-1}E = \begin{bmatrix} -1 & -\delta^{-1} \end{bmatrix}. \end{aligned} \tag{5.44}$$

Note that the system has the robust steady-state gain property as $G_u(\delta)$ is independent of δ. We define the steady-state optimization problem of interest as

$$\begin{aligned} \bar{z}^*(\delta, w) &= \arg\min_{\bar{z}} \Phi(\bar{z}) \\ \text{subject to:} \quad G_z\bar{z} &= d(\delta, w), \end{aligned} \tag{5.45}$$

where

$$z = \begin{bmatrix} u \\ y \end{bmatrix}, \ \Phi(\bar{z}) = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z}, \ Q_{zz} = \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix}, \ R_z = \begin{bmatrix} R_u \\ R_y \end{bmatrix}, \ Q_{uu} = 5,$$
$$Q_{uy} = Q_{yu} = 2, \ Q_{yy} = 1, \ R_u = 0.1, \ R_y = 0.5, G_z = \begin{bmatrix} -G_u(\delta) & I_p \end{bmatrix}, \ d(\delta, w) = G_w(\delta)w. \tag{5.46}$$

To regulate the uncertain system (5.39) to steady-state equilibria that are the solution of problem (5.45), we adopt three approaches in this example:

1. A conventional approach where problem (5.45) is explicitly solved online for a nominal value of the uncertain parameter δ and an estimated value of the unknown disturbance. The set-points obtained are then used in a tracking linear quadratic control formulation to achieve the control objective.

2. A nominal FOLQC approach based on a nominal model of the uncertain system, and

3. A robust FOLQC approach based on knowledge of the set Δ_l and the piecewise-constant nature of the disturbance i.e $w(k) = w(k-1)$.

5.6.1 Conventional approach: *Disturbance estimation, explicit solution and tracking LQC design*

Disturbance estimation

To estimate the disturbance, a nominal value of the uncertain parameter $\delta = \bar{\delta}$ is chosen and the following nominal dynamics are written

$$\bar{x}(k+1) = A(\bar{\delta})\bar{x}(k) + B(\bar{\delta})u(k) + Ew(k), \tag{5.47a}$$

$$\bar{y}(k) = C\bar{x}(k). \tag{5.47b}$$

From (5.47) and the fact that $w(k) = w(k-1)$, we define the following disturbance augmented dynamics

$$\begin{bmatrix} \bar{x}(k+1) \\ \bar{w}(k+1) \end{bmatrix} = A_a(\bar{\delta})\bar{x}(k) + B_a(\bar{\delta})\bar{u}(k), \tag{5.48a}$$

$$\bar{y}(k) = C_a \begin{bmatrix} \bar{x}(k+1) \\ \bar{w}(k+1) \end{bmatrix} \tag{5.48b}$$

where

$$A_a(\bar{\delta}) = \begin{bmatrix} A(\bar{\delta}) & E \\ \mathbf{0} & I_{n_w} \end{bmatrix}, \; B_a(\bar{\delta}) = \begin{bmatrix} B(\bar{\delta}) \\ \mathbf{0} \end{bmatrix}, \text{ and } C_a = \begin{bmatrix} C & \mathbf{0} \end{bmatrix}. \tag{5.49}$$

The disturbance augmented observer is then designed as the dynamical system

$$\begin{bmatrix} \hat{x}(k+1) \\ \hat{w}(k+1) \end{bmatrix} = \Big(A_a(\bar{\delta}) - L_a C_a\Big) \begin{bmatrix} \hat{x}(k) \\ \hat{w}(k) \end{bmatrix} + B_a(\bar{\delta})u(k) + L_a y(k). \tag{5.50}$$

The observer gain L_a is designed such that the estimation error dynamics

$$e_a(k+1) = \Big(A_a(\bar{\delta}) - L_a C_a\Big) e_a(k) \tag{5.51}$$

is driven asymptotically to zero for any non-zero value of $e_a(0)$ where

$$e_a(k) = \begin{bmatrix} \bar{x}(k) \\ \bar{w}(k) \end{bmatrix} - \begin{bmatrix} \hat{x}(k) \\ \hat{w}(k) \end{bmatrix}. \tag{5.52}$$

Remark 27. *1. The nominal value of δ can be chosen to be its most probable value. This is usually the centroid of the uncertainty set Δ_l.*

2. The observer gain L_a should be chosen such that the poles of error dynamics are much faster than the poles of the closed-loop system. This is to guarantee that the estimated disturbance, $\hat{w}$, converges very fast to w (or a (hopefully) close approximation in this case) before the linear system settles at a steady-state.

The poles of the error dynamics are placed at $0.1 + 0.3i,\ 0.1 - 0.3i,\ 0.1,$ and 1 to give the observer gain matrix

$$L_a = \begin{bmatrix} 3.2 & 14.2 & 0 & 8.1 \end{bmatrix}. \tag{5.53}$$

Explicit solution

Given the estimated disturbance at time k, $\hat{w}(k)$, and a nominal value of the uncertain parameter, $\bar{\delta}$, the steady-state optimization problem (5.45) can be solved online as the quadratic program

$$\begin{aligned} (\hat{u}_{ref}, \hat{y}_{ref}) = \bar{z}^*(\bar{\delta}, \hat{w}) &= \arg\min_{\bar{z}} \Phi(\bar{z}) \\ \text{subject to:} \quad \begin{bmatrix} -0.1 & 1 \end{bmatrix} \bar{z} &= \begin{bmatrix} -1 & -\bar{\delta}^{-1} \end{bmatrix} \hat{w}. \end{aligned} \tag{5.54}$$

Remark 28. *Because $\bar{\delta}$ is only an approximation to the true value of the parameter δ, which is assumed uncertain, the estimated disturbance, $\hat{w}$, therefore only approximates the actual disturbance, w. As a result, the computed set-points, $(\hat{u}_{ref}, \hat{y}_{ref})$, are an approximation of the actual steady-state optimum, $\bar{z}^*(\delta, w)$, of* (5.45).

Tracking LQC design

Given the computed set-points, $(\hat{u}_{ref}, \hat{y}_{ref})$, to track the optimal solution (or in this case an approximation) of the steady-state optimization problem (5.45), we design a tracking linear quadratic control algorithm based on the following velocity model of the nominal dynamics

$$\bar{\epsilon}(k+1) = \bar{\mathcal{A}}\bar{\epsilon}(k) + \bar{\mathcal{B}}\delta\bar{u}(k), \tag{5.55a}$$

$$\bar{e}(k) = \bar{\mathcal{C}}\bar{\epsilon}(k) \tag{5.55b}$$

where

$$\bar{\epsilon}(k) := \begin{bmatrix} \delta\bar{x}(k) \\ \bar{e}(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta\bar{x}(k) &:= \bar{x}(k) - \bar{x}(k-1), \\ \delta\bar{u}(k) &:= \bar{u}(k) - \bar{u}(k-1), \end{aligned} \tag{5.56}$$

and

$$\bar{\mathcal{A}} = \begin{bmatrix} A(\bar{\delta}) & \mathbf{0} \\ C & I_{n_y} \end{bmatrix}, \qquad \bar{\mathcal{B}} = \begin{bmatrix} B(\bar{\delta}) \\ \mathbf{0} \end{bmatrix}, \qquad \bar{\mathcal{C}} = \begin{bmatrix} C & I_{n_y} \end{bmatrix}. \tag{5.57a}$$

The nominal tracking error, $\bar{e}(k)$ is defined as

$$\bar{e}(k) = \bar{y}(k) - \hat{y}_{ref} = C\bar{x}(k) - \hat{y}_{ref}. \tag{5.58}$$

Given the nominal tracking error and nominal velocity dynamics, the tracking linear quadratic control law applied to the uncertain system is given by

$$u(k) = u(k-1) - \bar{\mathcal{K}}\bar{\epsilon}(k), \tag{5.59}$$

where $\bar{\mathcal{K}}$ is the optimal control gain obtained by minimizing the performance objective

$$J_\infty(\bar{\epsilon}(k)) = \frac{1}{2}\sum_{k=0}^{\infty}\Big(\|\bar{e}(k)\|^2_{\bar{Q}_e} + \|\delta\bar{u}(k)\|^2_{\bar{\mathcal{R}}}\Big), \tag{5.60}$$

subject to the nominal velocity dynamics (5.55), with $\bar{Q}_e \succeq 0$ and $\bar{\mathcal{R}} \succ 0$. From [133], the linear quadratic control gain $\bar{\mathcal{K}}$ is given by

$$\bar{\mathcal{K}} = \Big(\bar{\mathcal{R}} + \bar{\mathcal{B}}^\top \bar{\mathcal{P}} \bar{\mathcal{B}}\Big)^{-1} \bar{\mathcal{B}}^\top \bar{\mathcal{P}} \bar{\mathcal{A}} \tag{5.61}$$

where $\bar{\mathcal{P}} \succ 0$ is the stabilizing positive definite solution to discrete-time algebraic Riccati equation

$$\bar{\mathcal{P}} = \bar{\mathcal{A}}^\top \bar{\mathcal{P}} \bar{\mathcal{A}} + \bar{\mathcal{C}}^\top \bar{Q}_e \bar{\mathcal{C}} - \bar{\mathcal{A}}^\top \bar{\mathcal{P}} \bar{\mathcal{B}}\Big(\bar{\mathcal{R}} + \bar{\mathcal{B}}^\top \bar{\mathcal{P}} \bar{\mathcal{B}}\Big)^{-1} \bar{\mathcal{B}}^\top \bar{\mathcal{P}} \bar{\mathcal{A}}. \tag{5.62}$$

Remark 29. *To implement this controller, the nominal velocity state $\bar{\epsilon}(k)$ at time k, is computed from the estimated state $\hat{x}(k)$ and the computed output reference $\hat{y}_{ref}$ as*

$$\bar{\epsilon}(k) := \begin{bmatrix} \hat{x}(k) - \hat{x}(k-1) \\ C\hat{x}(k) - \hat{y}_{ref} \end{bmatrix}. \tag{5.63}$$

Because the disturbance is assumed constant, the output reference, $\hat{y}_{ref}$ will also be constant within a sampling interval.

5.6.2 Robust FOLQC approach

For the robust FOLQC design, Λ_y, Λ_u and r follow from the choices of Φ and the input-output sensitivity, G_u as

$$\Lambda_y = -15, \ \Lambda_u = -48, \ \text{and } r = -5. \tag{5.64}$$

The transient performance criterion is chosen with $Q_e = 100$ and $R_\delta = 25$; these values satisfy the hypothesis of Proposition 3.
From the design parameters above, the FOLQC law is computed from the following parameters:

$$\mathcal{A}(\delta) = \begin{bmatrix} 2 & 0.1 & 0 \\ 0 & 1-0.1\delta & 0 \\ -19 & 0 & 1 \end{bmatrix}, \ \mathcal{B}(\delta) = \begin{bmatrix} 0 \\ 0.1\delta \\ -48 \end{bmatrix}, \ \mathcal{C}(\delta) = \begin{bmatrix} -19 & 0 & 1.00 \end{bmatrix}, \ \mathcal{D}(\delta) = -48. \tag{5.65}$$

The robust state feedback matrix $\check{\mathcal{K}}$ in the control law (5.20) which minimizes the upper bound $V(\epsilon(0))$ on the robust performance objective (5.21) at sampling time k is given by

$$\check{\mathcal{K}} = \mathcal{W}\mathcal{Y}^{-1} \tag{5.66}$$

where $\mathcal{Y} \succ 0$ and $\mathcal{W}$ are obtained from the solution of the SDP (5.28) as

$$\begin{aligned} \mathcal{W} &= 1.0 \times 10^{-3} \times \begin{bmatrix} 0.0003 & -0.0027 & -0.3766 \end{bmatrix}, \\ \mathcal{Y} &= \begin{bmatrix} 0.0000 & -3.3 \times 10^{-}7 & -3.49 \times 10^{-5} \\ -3.27 \times 10^{-7} & 3.93 \times 10^{-6} & 3.5 \times 10^{-4} \\ -3.5 \times 10^{-5} & 0.0003 & 0.0815 \end{bmatrix}. \end{aligned} \tag{5.67}$$

5.6.3 Simulation results

For the simulation, the true value of the uncertain parameter is fixed at $\delta = 8$. The robust FOLQC law was computed using the extremal values of the uncertainty range i.e. $\delta_1 = 4$ and $\delta_2 = 10$. A nominal FOLQC and tracking LQC controllers

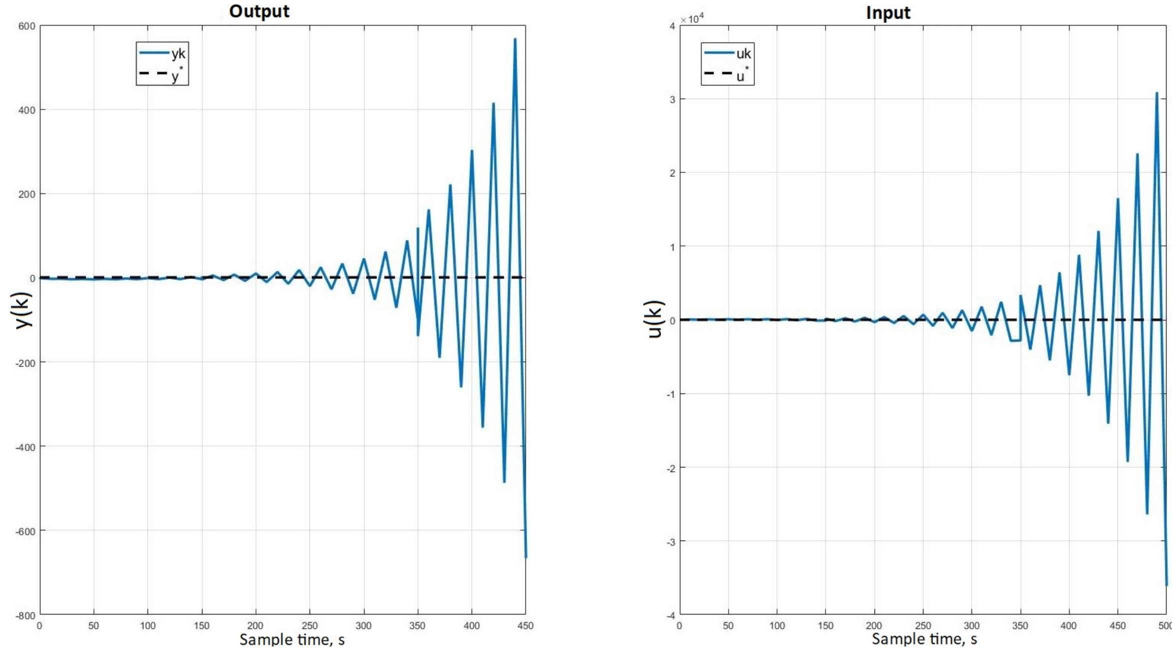

Fig. 5.1 Tracking LQC, $\bar{\delta} = 5$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

are also implemented for the following nominal values of the uncertain parameter, $\bar{\delta} = \{5, 7, 8, 9, 10\}$.

Performance of tracking LQC with disturbance estimation and explicit solution

Figure 5.1 shows the input-output performance of the tracking LQC combined with online optimization and disturbance estimation for $\bar{\delta} = 5$.
From the figure, we can see that the closed-loop system is unstable and does not track the optimum to the steady-state optimization. This shows that the conventional combination of a tracking controller with online steady-state optimization using disturbance estimates may not always guarantee the stability or convergence to the optimal steady-state when the nominal value of the uncertain parameter is far from its true value.

To improve the nominal accuracy, we set $\bar{\delta} = 7$ and implement the tracking LQC again. This time, the controller was stabilizing but convergence could only be guaranteed to a neighbourhood of the optimal steady-state. Figure 5.2 shows the obtained performance. Although a stable closed-loop system is obtained, the dynamic performance is good but not optimal for the defined linear quadratic cost.

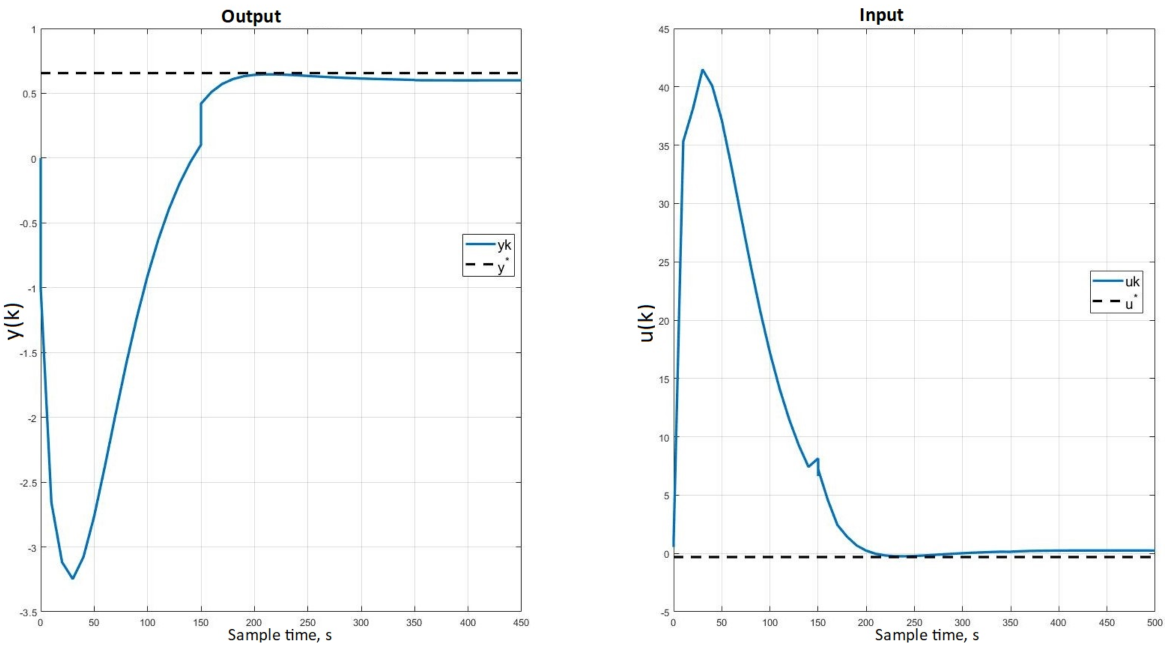

Fig. 5.2 Tracking LQC, $\bar{\delta} = 7$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

To improve the performance and accurately track the optimal steady-state, the nominal value of the uncertain parameter must closely approximate the true value. We see this from Figure 5.3 where the tracking LQC was implemented for a value of $\bar{\delta} = 8$, the true value. The figure shows that accurate tracking of the true optimal equilibrium is achieved with optimal dynamic performace. Similar results were obtained for overestimated nominal values of $\bar{\delta} = 12$ (Fig. 5.4) and $\bar{\delta} = 19$ (Fig. 5.5).

In conclusion, the simulation results here have shown that under model uncertainty, a tracking LQC formulation may not guarantee the stability of the closed-loop system. Even when stability is achieved, convergence can only be guaranteed to a neigbourhood of the optimal steady-state equilibrium. Also, the dynamic performance of a tracking LQC formulation deteriorates the further the nominal model is from the true model.

Performance of nominal FOLQC

Here, we investigate the performance of the nominal FOLQC from Chapter 4 under model uncertainty. We design nominal FOLQC for the following nominal model values: $\bar{\delta} = \{5, 7, 8, 9, 10\}$. At $\bar{\delta} = 5$, we see from Figure 5.6 that the nominal FOLQC does not achieve closed-loop stability.

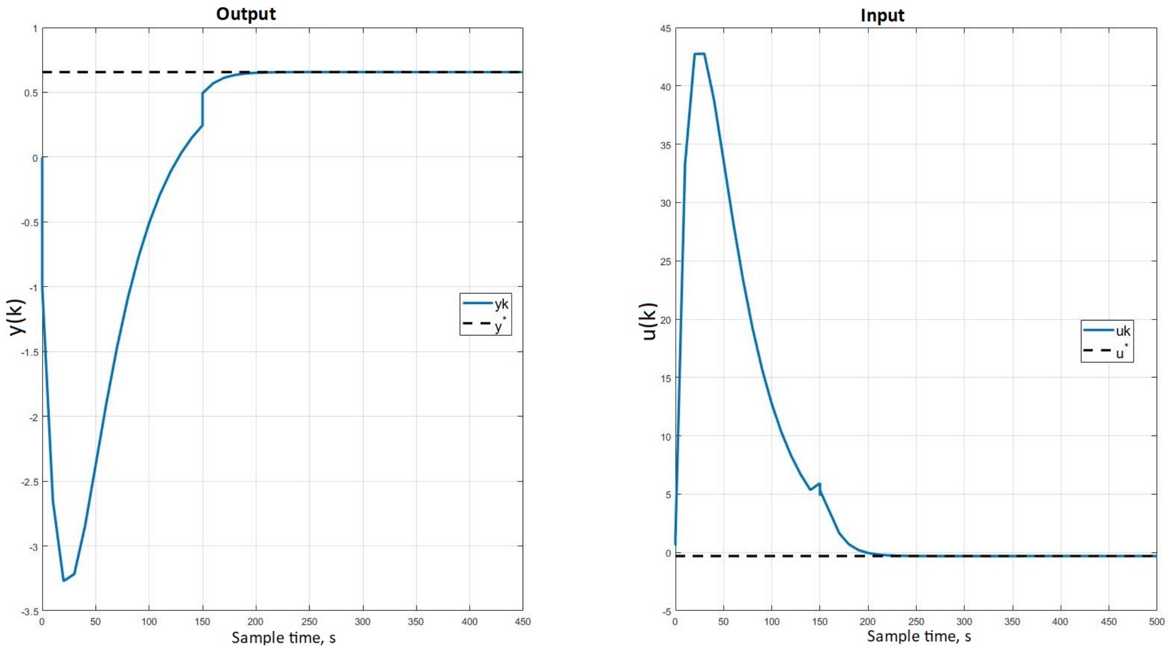

Fig. 5.3 Tracking LQC, $\bar{\delta} = 8$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

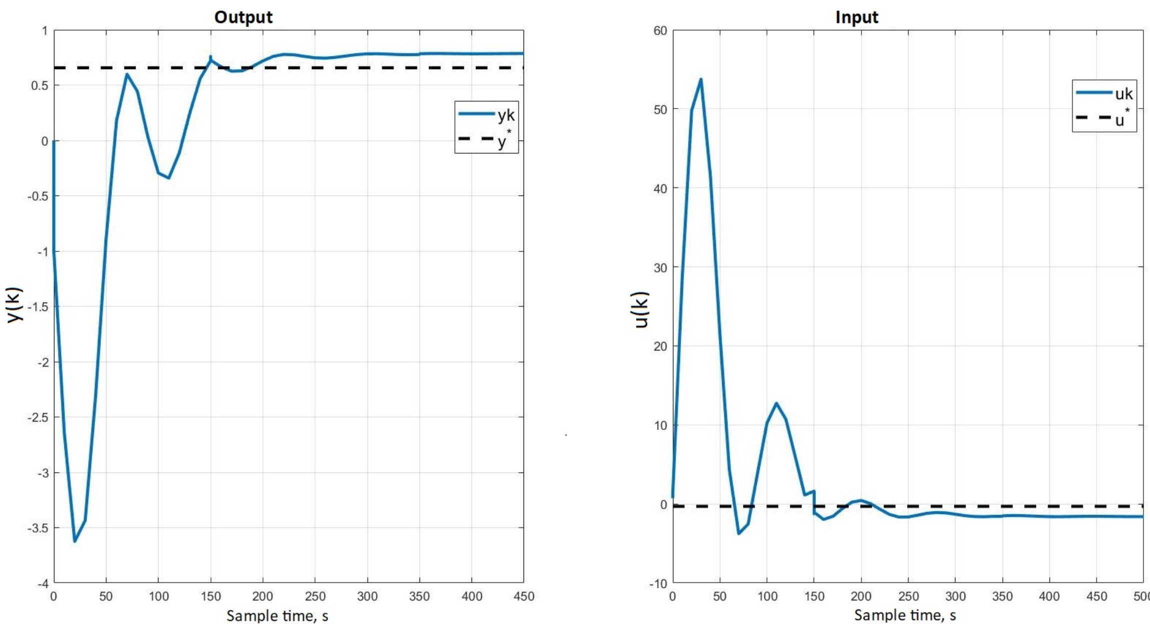

Fig. 5.4 Tracking LQC, $\bar{\delta} = 12$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

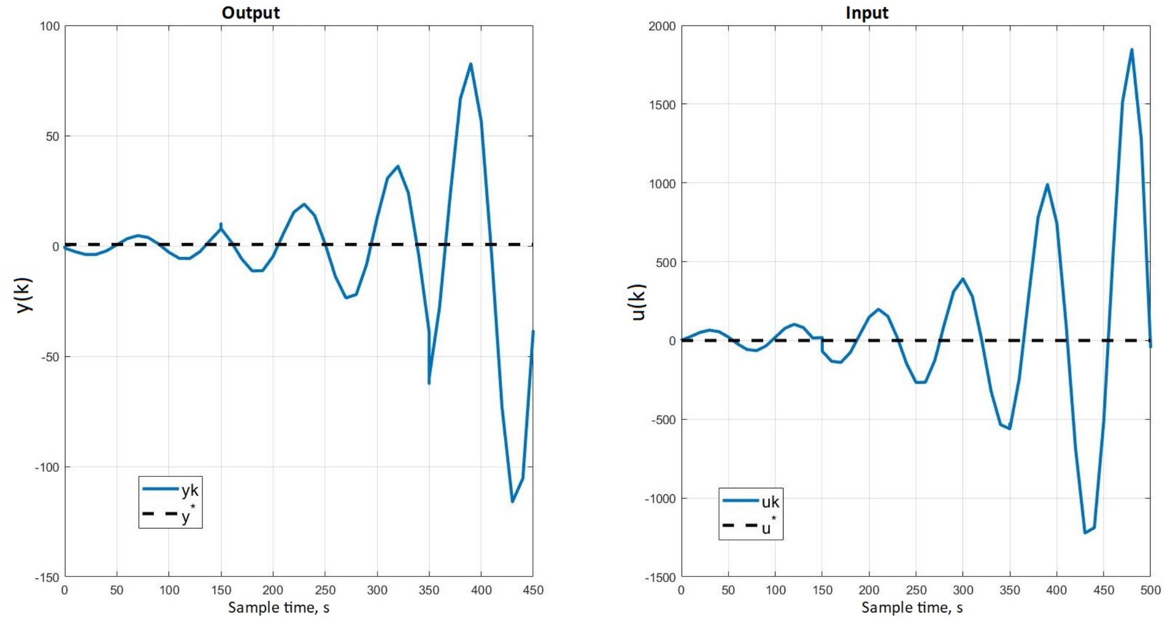

Fig. 5.5 Tracking LQC, $\bar{\delta} = 19$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

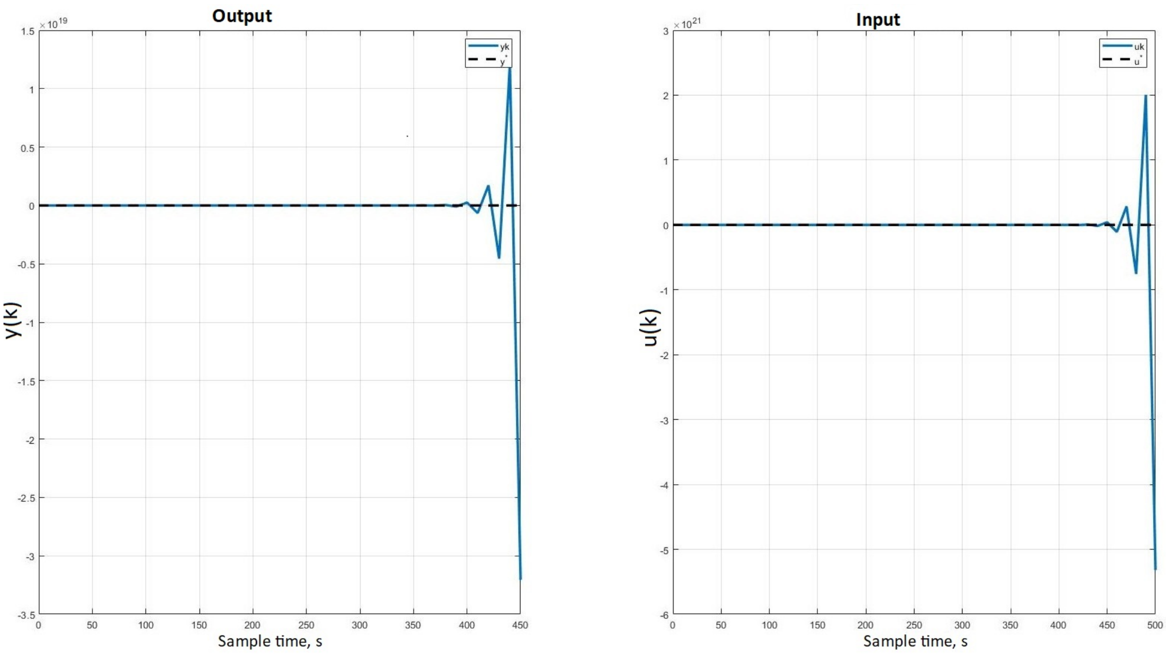

Fig. 5.6 Nominal FOLQC, $\bar{\delta} = 5$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

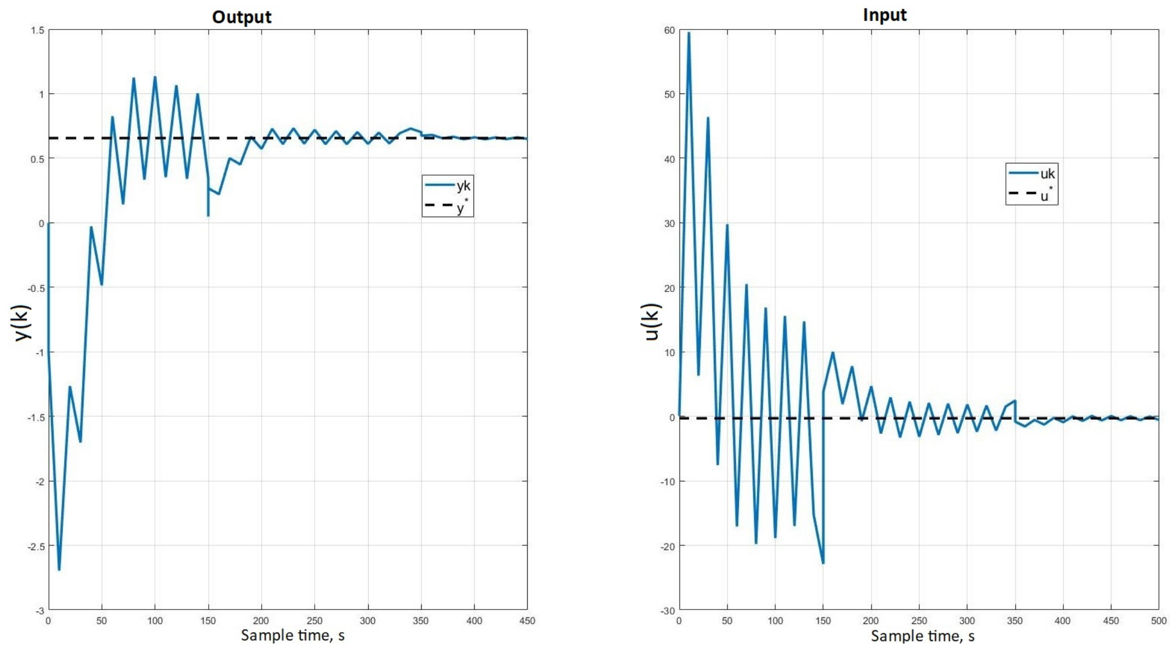

Fig. 5.7 Nominal FOLQC, $\bar{\delta} = 7$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

However at an improved but underestimated nominal value of $\bar{\delta} = 7$, the nominal FOLQC can be seen from Figure 5.7 to stabilize the uncertain system while also tracking the optimal steady-state with a poor dynamic performance. In contrast with the tracking LQC controller which only achieves robust stability but cannot robustly track the true steady-state optimum, the nominal FOLQC achieves both robust stability and robust convergence to the true optimal steady-state provided the nominal parameter estimate is close to its true value. This shows that unlike tracking LQC which only has an inherent robust stability property, the nominal FOLQC has both inherent robust stability and inherent robust convergence to the true steady-state optimum. This also adds to the advantage that nominal FOLQC does not rely on online disturbance estimation.

To improve the dynamic performance of the nominal FOLQC, accurate knowledge of the uncertain model parameter δ will be required. Figure 5.8, shows the performance of nominal FOLQC with accurate knowledge of δ. It can be observed from the figure that with the nominal parameter set to the true value i.e., $\bar{\delta} = 8$, stability and convergence to the true optimal steady-state is achieved with much improved dynamic performance.

At an overestimated nominal value of $\bar{\delta} = 9$, the nominal FOLQC can be seen from Figure 5.9 to stabilize the uncertain system and also track the (unknown) optimal

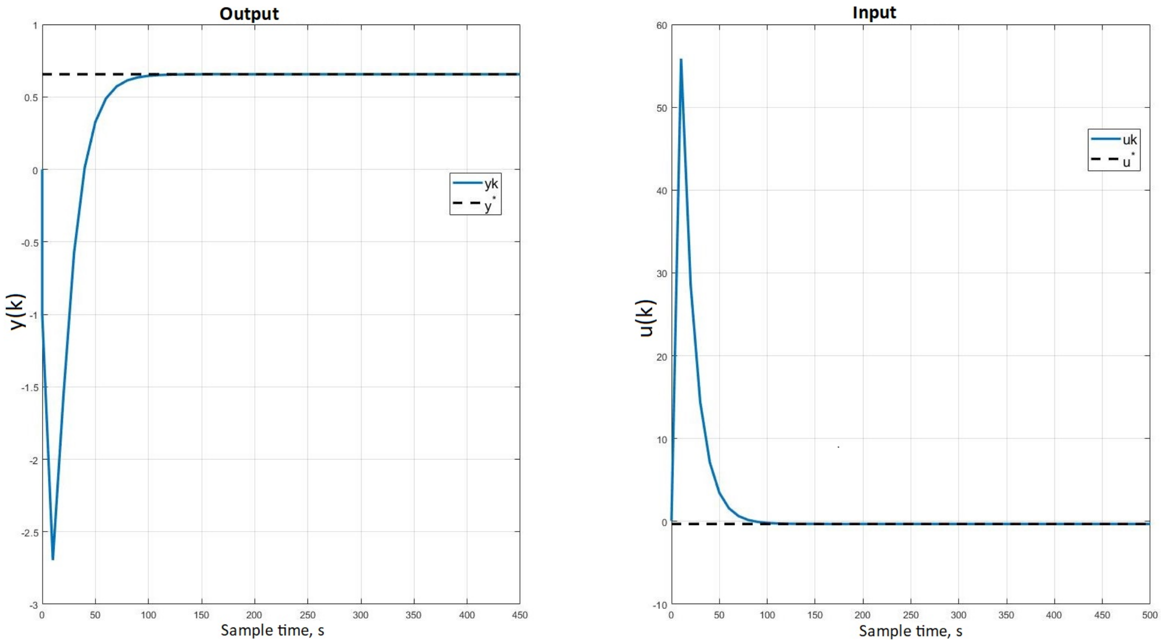

Fig. 5.8 Nominal FOLQC, $\bar{\delta} = 8$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

steady-state setpoints with a very good dynamic performance. It interesting to note that the nominal FOMPC performs better when the nominal value is overestimated rather than underestimated. It is not clear why this is the case. We leave any further analysis of this observation to future investigation.

Finally at the overestimated value of $\bar{\delta} = 10$, the nominal FOLQC can be observed from Figure 5.10 to achieve closed-loop stability for the true plant but with a severely degraded dynamic performance. The controller takes very long to settle at the optimal steady-state setpoints and experiences very large oscillations about these setpoints. This is a very poor performance but at least closed-loop stability is retained. At $\bar{\delta} > 10$, the nominal FOLQC becomes unstable (see Figure 5.11).

Performance of robust FOLQC

Although the tracking LQC is inherently robustly stable and the nominal FOLQC is both inherently robustly stable and inherently robustly convergent to the true optimum, both controllers do not guarantee these properties under model uncertainty. Exceptions are when the uncertainty satisfies some inherent robustness bound that is always very challenging to compute in practice. Therefore, to guarantee closed-loop stability and

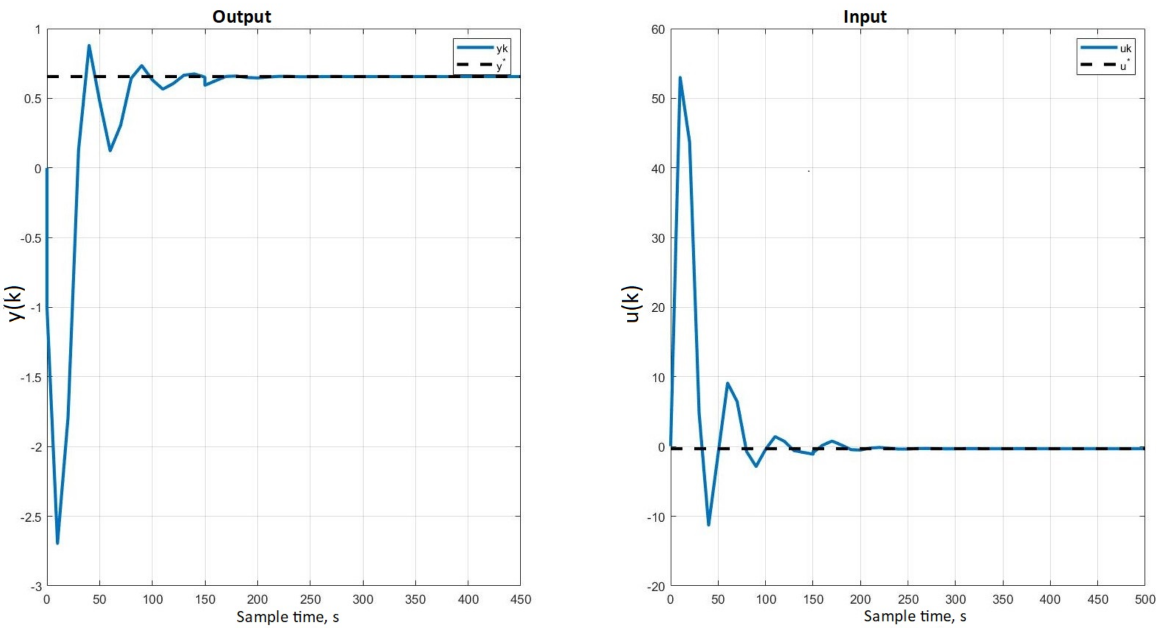

Fig. 5.9 Nominal FOLQC, $\bar{\delta} = 9$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

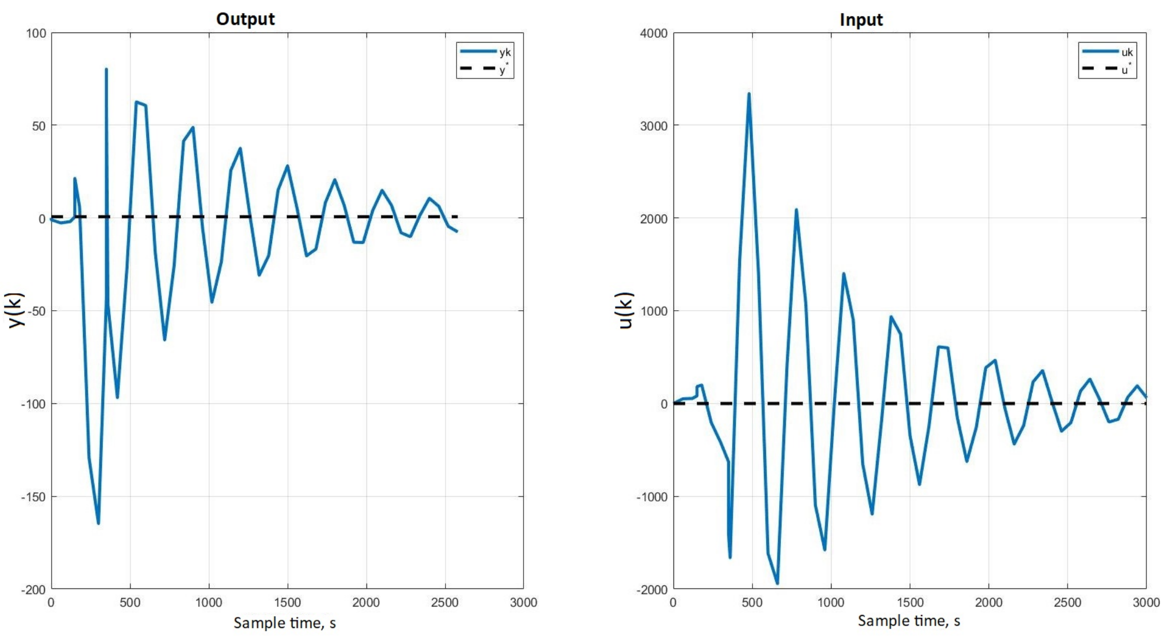

Fig. 5.10 Nominal FOLQC, $\bar{\delta} = 10$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

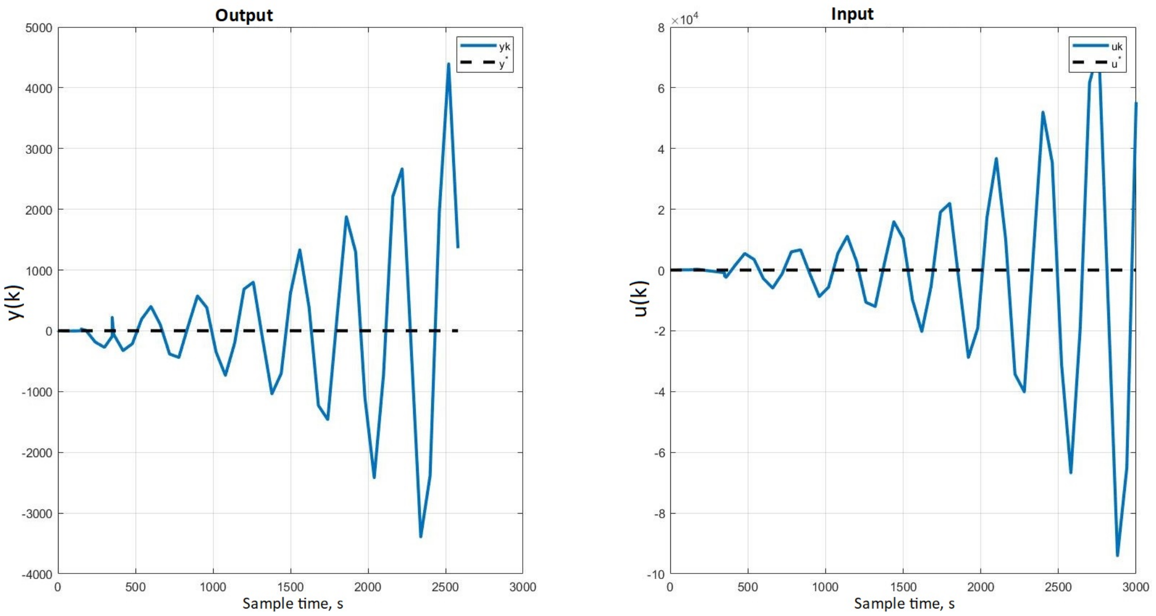

Fig. 5.11 Nominal FOLQC, $\bar{\delta} = 11$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

convergence to the true steady-state optimum under model uncertainty, the robust FOLQC formulation is needed.

Unlike the other two control formulations, robust FOLQC can guarantee the closed-loop stability and convergence to the true optimal steady-state as long as the uncertainty belongs to a defined set. We can see this from Figure 5.12. Here, the robust FOLQC is designed for all $\delta \in [4, 10]$, i.e., δ is assumed to take values between 4 and 10. It is clear from Figure 5.12 that the robust FOLQC is both robustly stable and robustly convergent to the true steady-state optimum. Also, while the other two control formulation were unstable when the uncertain parameter was set at a nominal value less than or equal to 5 i.e. $\bar{\delta} \leq 5$, the robust FOLQC achieves a stable and convergent closed-loop performance, with a decent dynamic performance for all values of δ in the specified range i.e., $\delta \in [4, 10]$. In fact, this range can be relaxed but at the expense of poorer dynamic performance. Also, tightening this range will improve the dynamic performance. This is seen in Figure 5.13 where the uncertainty range is tightened to $\delta \in [6, 10]$. The improved performance of the robust FOLQC is clearly seen in the figure.

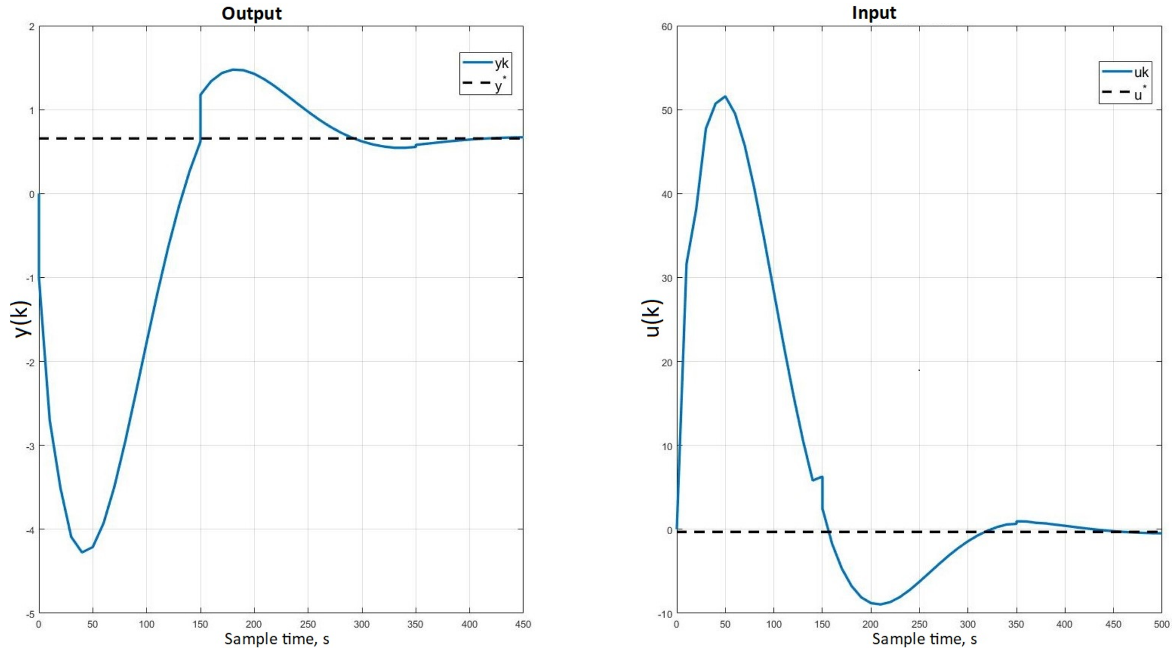

Fig. 5.12 Robust FOLQC, $\bar{\delta} \in [4, 10]$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

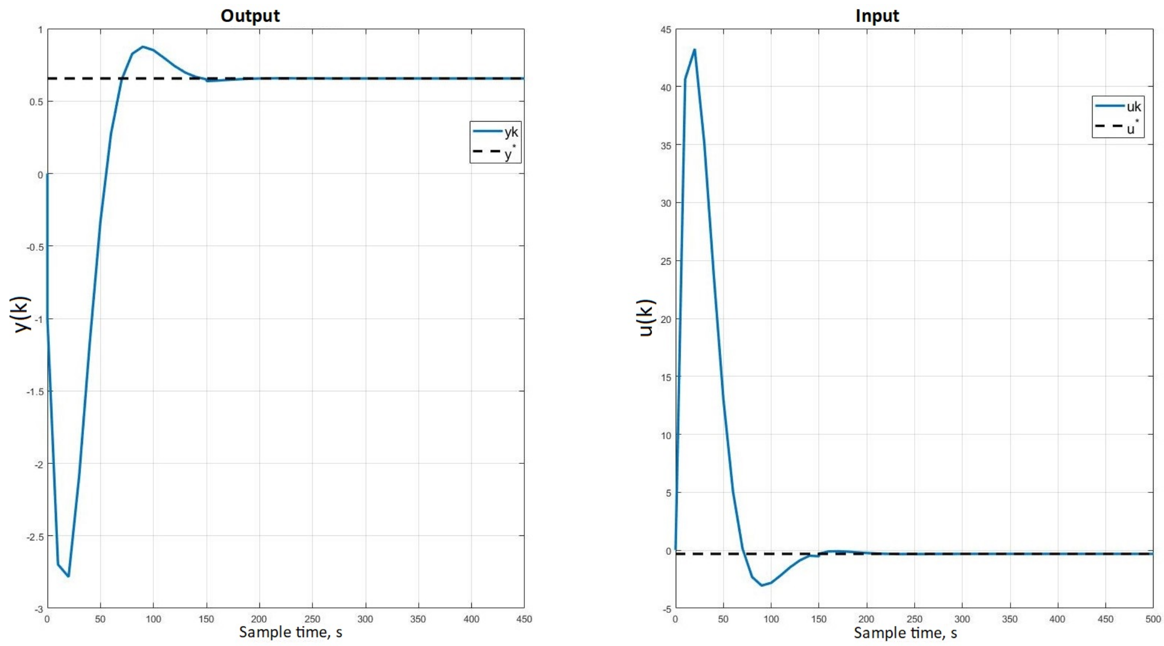

Fig. 5.13 Robust FOLQC, $\bar{\delta} \in [6, 10]$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

5.7 Conclusion

In this chapter, we have presented a robust FOLQC algorithm for the feedback optimization of uncertain linear systems with polytopic model uncertainty and a quadratic steady-state optimization problem. A linear matrix inequality approach was used to formulate semi-definite programs which can be solved offline to generate robust FOLQC laws. A limitation of the developed solution is that it requires the polytopic system to be quadratically stabilizing. This can limit its application to only a small class of uncertain linear systems. Future work will focus on relaxing this assumption and developing less conservative robust FOLQC laws.

Chapter 6

Nominal Feedback Optimizing Model Predictive Control

In this chapter, we propose feedback optimizing control laws based on a model predictive control framework to regulate a constrained linear system to the optimum of a steady-state optimization problem while optimizing the transient performance and satisfying dynamic inequality constraints in the inputs and outputs, without numerically solving the steady-state optimization problem. This is the FOMPC problem introduced in Chapter 3 when the model is assumed free of model uncertainty i.e., a nominal FOMPC problem. We begin with an introduction and a statement of the control problem for a steady-state quadratic program. We then propose a solution to the FOMPC problem based on a velocity model formulation of a standard tracking MPC algorithm. The theoretical properties of the proposed solution is analysed and a detailed analysis of the inherent robustness of the proposed nominal FOMPC algorithms is given. Finally, we present illustrative simulation results and a conclusion in the final section.

6.1 Introduction

Standard MPC schemes have primarily been designed to minimize the deviation of the states and inputs of a dynamic system from known reference values. These references are often generated in a feed-forward manner through the solution of steady-state optimization problems at timescales slower than the MPC regulator. These MPC schemes are generally referred to as tracking MPC. Approaches that combine feedback optimizing control and *model predictive control* (MPC) have the potential to augment the advantages of feedback optimizing control with the desirable features of MPC:

guaranteed constraint satisfaction and (near-)optimal transient performance. Two recent approaches to feedback optimizing MPC achieve steady-state optimality via the explicit online (numerical) solution to the steady-state optimization problem [75, 170]. Not only is it potentially computationally expensive to optimize steady-states online alongside solving the conventional MPC problem, but such an approach may also lack robustness to the effects of unknown disturbances on the steady-state solution. [74] addressed the latter shortcoming by proposing an economic MPC (EMPC) algorithm integrating modifier adaptation techniques with offset-free MPC. The resulting MPC algorithm is able to track the optimal steady-state equilibrium without explicitly computing the optimal steady-state set-points. The algorithm although simple requires online estimation of the true plant gradients and disturbances; adopts a tailored formulation of the optimal control problem (OCP); requires the open loop OCP exhibiting turnpike properties; and gives no formal convergence guarantees.

In this chapter, we present results addressing some of the above limitations by developing a new feedback optimizing model predictive control (FOMPC) algorithm. The proposed approach integrates steady-state optimization and feedback regulation for a linear time-invariant (LTI) system subject to linear inequality constraints and a constant additive state disturbance. By replacing the standard tracking error with the residual of the Karush–Kuhn–Tucker (KKT) optimality conditions of the steady-state optimization problem and using this residual within a velocity form of MPC for tracking, we develop an approach *guaranteed* to converge to the steady-state optimum while minimizing quadratic transient performance criterion and guaranteeing constraint satisfaction.

Compared to a two-step approach, the presented algorithm is a novel and systematic way of integrating RTO and MPC to reduce/eliminate the dependence on timescale separation, while also guaranteeing closed-loop convergence to the actual (unknown) RTO set-points.

6.2 Problem statement and preliminaries

System description and steady-state optimization

Consider a constrained discrete-time linear time-invariant system, $\mathscr{P}$, described by

$$x(k+1) = Ax(k) + Bu(k) + Ew(k), \tag{6.1a}$$

$$y(k) = Cx(k), \tag{6.1b}$$

$$(u(k), y(k)) \in \mathbb{Z} \coloneqq \mathbb{U} \times \mathbb{Y}, \tag{6.1c}$$

and the steady-state optimization problem

$$\mathcal{RTO}(\bar{w})\colon \quad \bar{z}^*(\bar{w}) = \arg\min_{\bar{z}} \Big\{\Phi(\bar{z}) \mid \bar{z} \in \mathbb{F}(\bar{w}),\ \bar{z} \in \mathbb{Z}\Big\}, \tag{6.2}$$

where

- $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{U} \subseteq \mathbb{R}^{n_u}$, $y(k) \in \mathbb{Y} \subseteq \mathbb{R}^{n_y}$, and $w(k) \in \mathbb{W} \subseteq \mathbb{R}^{n_w}$ are the state, input, output and additive uncertainty (disturbance) vectors respectively and $z(k) = \begin{bmatrix} u(k) & y(k) \end{bmatrix}^\top$.
- $A \in \mathbb{R}^{n_x \times n_x},\ B \in \mathbb{R}^{n_x \times n_u},\ E \in \mathbb{R}^{n_x \times n_w},\ C \in \mathbb{R}^{n_y \times n_x}$ are the system coefficient matrices.
- $\Phi\colon \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}$ is a continuous scalar economic performance index to be minimized in steady-state.
- $\mathbb{F}(\bar{w})$ is the set of controlled equilibria for the system (6.1) defined similarly to (3.4) with $\delta = 0$.

Problem (6.2) is a static optimization problem whose solution varies according to the disturbance $\bar{w}$, which is unknown a priori. Therefore, the explicit online computation of the solution of (6.2), using estimates of the disturbances, may result in steady-state errors and sub-optimality [65]. In the remainder of this chapter, we develop an approach based on model predictive control that solves problem (6.2) implicitly via feedback control, without knowledge or estimates of the disturbances, while regulating the system in an optimal and admissible way with respect to a transient performance criterion and constraints.

Problem statement

For the system (6.1) we consider the problem of designing a state feedback control law, based on model predictive control, for tracking the optimal solution $z^*(\bar{w})$ of the steady-state optimization problem (6.2), i.e., asymptotically steer the system output $y(k)$, and/or input $u(k)$ to their respective steady-state optimal values $z^*(\bar{w})$ while respecting the inequality constraint $(u(k), y(k)) \in \mathbb{Z}$ at all times. Formally, the control problem we consider is defined as follows.

Problem 6 (The FOMPC Problem)**.** *Design for the discrete linear time-invariant system* (6.1) *a state feedback control law*

$$u(k) = \kappa_N(x(k), u(k-1)) \tag{6.3}$$

obtained from the online solution to finite-horizon horizon optimal control problems, such that for any constant admissible $\bar{w} \in \mathbb{W}$:

1. *for all feasible $x(0)$, the point $\bar{z}^*(\bar{w})$ is an asymptotically stable equilibrium for the closed-loop system, satisfying*

$$\lim_{k\to\infty} (u(k), y(k)) = \bar{z}^*(\bar{w}). \tag{6.4}$$

2. *The feedback policy $\kappa_N(\cdot, \cdot)$ minimizes a transient performance criterion.*

3. *The constraints $(u(k), y(k)) \in \mathbb{Z}$ are satisfied at all times.*

In the next section, we show an MPC controller can be constructed to have a suitable tracking error $e(k)$ that facilitates regulation of $(u(k), y(k))$ to the optimum of (6.2) without knowledge of the optimal set-points $z^*(\bar{w})$ or the disturbance $\bar{w}$. We consider the case where the steady-state performance objective is a quadratic function of the outputs and inputs, and show that this results in the steady-state optimality error $e(k)$ as defined by (3.34) being an affine function of the measured output and input. Using the velocity model form of the LQ optimal control problem [172], we develop an MPC formulation that steers the system asymptotically and admissibly to the optimal steady-state equilibrium, without knowledge of this equilibrium and while minimizing a standard LQ transient performance criterion. For simplicity and to maintain intuition of the concepts, we consider the case where the inequality constraints $(u(k), y(k)) \in \mathbb{Z}$ are inactive in steady-states.

6.3 Proposed Solution

In this section, we propose solutions to the FOMPC problem when the steady-state performance objective is a quadratic function of the input and output variables.

Quadratic steady-state performance index

We consider that

$$\Phi(\bar{z}) = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z} \tag{6.5}$$

where $Q_{zz} = \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix}$ and $R_z = \begin{bmatrix} R_u \\ R_y \end{bmatrix}$.

Tracking error

Instead of defining and regulating the tracking error as the difference between $z(k)$ and the unknown optimum $\bar{z}^*(\bar{w})$, we define the tracking error as the residual of the KKT optimality condition for problem (6.2) (see details in chapter 3, equation (3.34)) which under inactive steady-state inequality constraints yields

$$e(k) := \tilde{G}\nabla\tilde{\Phi}(z(k)) = \Lambda_y Cx(k) + \Lambda_u u(k) + r \tag{6.6}$$

where Λ_y, Λ_u and r retain their meaning from previous chapters. This error may be computed directly from the input $u(k)$ and output $y(k)$ measurements, provided the objective Φ and the input–output DC gain matrix G_u are known. This choice therefore obviates the need for knowledge of the optimal equilibrium $\bar{z}^*$ and the disturbance $\bar{w}$.

Basic assumptions

Concerning the system (6.1) and the steady-state cost (6.5), we make the following assumptions

Assumption 6 (Basic system assumptions)**.**

(i) The system (6.1) *is reachable and observable.*

(ii) The state $x(k)$ is measurable.

(iii) the number of outputs, n_y is less than or equal to the number of inputs, n_u.

(iv) $\mathbb{Y}$ and $\mathbb{U}$ contain the origin in their interiors; $\mathbb{W}$ contains the origin.

(v) $\mathbb{U}$ and $\mathbb{W}$ are compact.

(vi) $\text{rank}(S) = n_u + n_y$*, where*

$$S := \begin{bmatrix} A - I_{n_x} & B \\ \Lambda_y C & \Lambda_u \end{bmatrix}. \tag{6.7}$$

Assumption 7 (Basic assumptions on the steady-state problem)**.**

(i) The cost $\Phi(\cdot)$ is differentiable and convex.

(ii) The set of admissible disturbances, $\mathbb{W}$, is such that, for all $\bar{w} \in \mathbb{W}$,

(a) the set $\mathbb{F}(\bar{w})$ has a non-empty relative interior.

(b) the minimizer $\bar{z}^*(\bar{w})$exists and is unique.

Remark 30. *Assumption 6(iii) refers to the number of controlled/optimized states rather than measured outputs; the latter are not considered in this chapter since we assume state measurements. In practice, if $y(k) = Cx(k)$ defines a vector of measured outputs, possibly with $n_y \geq n_u$, it is possible to follow the common practice of defining a controlled output $\tilde{y}(k) = Hy(k) = HCx(k)$ a linear combination of these measured outputs, such that dim$(\tilde{y}(k)^) \leq n_u$ [181].*

Remark 31. *Assumption* 6(*vi*) *is, as shown in Proposition 2 (chapter 4), necessary for reachability of the velocity dynamics* (4.12) *required for the control design.*

Velocity and tracking error dynamics

To regulate the system to $e = 0$, we consider the velocity form of the system dynamics (6.1) augmented with the tracking error dynamics (see equation (4.12)). Conditions for reachability and observability of the velocity dynamics have been established in Chapter 4.

6.3.1 Nominal FOMPC

In this section, we present an MPC algorithm based on a velocity form of the system dynamics (6.1) that solves the FOMPC problem for systems with unknown constant (or slowly varying) disturbances i.e., $w(k) = w(k-1)$, and a quadratic steady-state performance criterion. Because the additive disturbance in the velocity model is zero i.e., $\delta w(k) = 0$, the resulting velocity dynamics is a nominal model. To guarantee convergence of the closed-loop nominal velocity dynamics to steady-state set-points $\bar{z}^*$ that are admissible and optimal for the actual system, we make use of suitably derived auxiliary control laws and terminal invariant sets. Since $\delta w(k) = 0$, the velocity dynamics (4.12) reduces to:

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k), \tag{6.8a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k). \tag{6.8b}$$

In view of the reachability of $(\mathcal{A}, \mathcal{B})$, there exists a gain $\mathcal{K}$ such that $\mathcal{F} = \mathcal{A} - \mathcal{B}\mathcal{K}$ is Schur (see Definition 36). Note that such gain is the same as the unconstrained, feedback-optimizing linear quadratic control gain obtained in chapter 4, equation (4.26a).

The dynamics of (6.8a), under the feedback control law

$$\delta u(k) = -\mathcal{K}\epsilon(k) \tag{6.9}$$

is given by

$$\epsilon(k+1) = \mathcal{F}\epsilon(k). \tag{6.10}$$

The maximal output admissible set (MOAS)

In the following, the velocity dynamics (6.8a) will be used to design an MPC algorithm to track the unknown optimum of the steady-state cost (7) with guarantees of stability and recursive feasibility for the inequality constraints $\mathbb{U}$ and $\mathbb{Y}$. Towards this goal, we will adopt (6.9) as an auxiliary control law to guarantee stability of the closed-loop velocity dynamics in line with [181]. To achieve recursive feasibility of the inequality constraints, the constraints on u and y must be reformulated in terms of the MPC optimization variables δu and ϵ. To this end, similarly to [28], we write the relation between these variables as (see section 6.6 for details)

$$\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \tilde{\mathcal{C}} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} + \tilde{\mathcal{D}} w(k-1), \tag{6.11}$$

where

$$\tilde{\mathcal{C}} := \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix},$$
$$\tilde{\mathcal{D}} := \begin{bmatrix} E \\ \mathbf{0}_{n_u \times n_x} \end{bmatrix} - \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_x} \end{bmatrix}.$$

Let $\mathbb{G}$ be the set which enforces the constraints $(u(k), y(k)) \in \mathbb{Z}$ on the velocity state $\epsilon(k)$, i.e,

$$\mathbb{G} := \{\epsilon : (u, y) \in \mathbb{Z}\}.$$

From (6.11), the set $\mathbb{G}$ can be computed in terms of ϵ as,

$$\mathbb{G} := \left\{ \epsilon : \tilde{\mathcal{C}} \begin{bmatrix} \epsilon \\ r \end{bmatrix} \in (\mathbb{X} \times \mathbb{U}) \ominus \tilde{\mathcal{D}} \mathbb{W} \right\} \tag{6.12}$$

where $\mathbb{X} \subseteq \mathbb{R}^n := \{x : y \in \mathbb{Y}\}$.

To guarantee that the system inputs and outputs $u(k)$ and $y(k)$, respectively, lie in the feasibility sets $\mathbb{U}$ and $\mathbb{Y}$, we need to compute a suitable invariant set, $\mathbb{G}_f$, where $\epsilon(k)$ must lie in order to guarantee that the constraints $(u(k), y(k)) \in \mathbb{Z}$ are verified for all k. Assume the pair $(\mathcal{A}, \mathcal{B})$ is reachable and the set $\mathbb{G}$ is a closed polytope (both true if Assumption 6 are satisfied), then the set $\mathbb{G}_f$ can be computed as the (projection of the) maximal constraint admissible set [114] for the velocity dynamics (6.8a) under the auxiliary control law, $\delta u(k) = -\mathcal{K}\epsilon(k)$ and is constructed such that:

$$\epsilon \in \mathbb{G}_f \implies \epsilon \in \mathbb{G} \text{ and } \mathcal{F}\epsilon \in \mathbb{G}_f. \tag{6.13}$$

Remark 32 (Computation of $\mathbb{G}_f$)**.** *Algorithms for computing $\mathbb{G}_f$ and discussions on the computational complexity of the associated algorithms are presented in [81, 114]. Generally, the computation of $\mathbb{G}_f$ is more difficult for linear systems with higher dimensions i.e. very large number of state variables.*

Optimal control problem

Given the velocity dynamics (6.8) and the sets $\mathbb{G}$ and $\mathbb{G}_f$, the feedback optimizing model predictive control problem is defined, for a state $\epsilon(k)$, as

$$\min_{\delta\mathbf{u}(k)} V_N(\epsilon(k), \delta\mathbf{u}(k)) = V_f(\epsilon(k+N)) + \sum_{i=0}^{N-1} l(e(k+i), \delta u(k+i)) \tag{6.14}$$

subject to:
$\forall i \in \mathbb{I}_{[0,N-1]}$,

$$\epsilon(k+i+1) = \mathcal{A}\epsilon(k+i) + \mathcal{B}\delta u(k+i), \tag{6.15a}$$

$$e(k+i) = \mathcal{C}\epsilon(k+i) + \mathcal{D}\delta u(k+i), \tag{6.15b}$$

$$\epsilon(k+i) \in \mathbb{G}, \tag{6.15c}$$

and

$$\epsilon(k+N) \in \mathbb{G}_f. \tag{6.16a}$$

In this problem, the decision variable is the sequence of control increments over the N-step prediction horizon:

$$\delta\mathbf{u}(k) := \Big\{\delta u(k), \delta u(k+1) \dots, \delta u(k+N-1)\Big\}. \tag{6.17}$$

These sequences are chosen to minimize the objective $V_N(\epsilon(k), \delta\mathbf{u}(k))$, which is composed of a stage cost

$$l(e(k), \delta u(k)) := \frac{1}{2}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big) \tag{6.18}$$

and a terminal cost

$$V_f(\epsilon(k+N)) := \frac{1}{2}\epsilon(k+N)^\top \mathcal{P}\epsilon(k+N). \tag{6.19}$$

The terminal cost is employed, in the usual way, towards guaranteeing stability and is chosen to satisfy the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K})^\top \mathcal{P}(\mathcal{A} - \mathcal{B}\mathcal{K}) - \mathcal{P} = -[\mathcal{C}^\top Q_e \mathcal{C} - 2\mathcal{N}\mathcal{K} + \mathcal{K}^\top \mathcal{R}\mathcal{K}] \tag{6.20}$$

where $\mathcal{N} = \mathcal{C}^\top Q_e \mathcal{D}$ and $\mathcal{K}$ is such that $\mathcal{A} - \mathcal{B}\mathcal{K}$ is Schur.

This performance index (6.14) captures the transient performance objectives for the closed-loop system: the term $e(k)^\top Q_e e(k)$ penalizes the distance of the tracking error from steady-state optimality and therefore determines the duration of the transient phase, while the term $\delta u(k)^\top R_\delta \delta u(k)$ penalizes the incremental control effort. It is simple to verify that the following assumption ensures positive definiteness of this cost.

Assumption 8. *The matrices R_δ, Q_e and $\mathcal{P}$ satisfy $R_\delta \succ 0$, $\mathcal{P} \succeq 0$ and*

$$Q_e - Q_e \Lambda_u (R_\delta + \Lambda_u^\top Q_e \Lambda_u)^{-1} \Lambda_u Q_e^\top \succeq 0. \tag{6.21}$$

Solution of the optimal control problem presented above, followed by the application of the first control in the optimized sequence, yields the control law

$$\delta u(k) = \kappa_N \Big(\epsilon(k), u(k-1) \Big). \tag{6.22}$$

Main result

The following result summarizes the stability and recursive feasibility of the FOMPC algorithm, and follows directly from well established results on conventional linear MPC [181].

Theorem 6.3.1 (Stability and feasibility)**.** *For piecewise-constant disturbances, the FOMPC control law $u(k) = u(k-1) + \kappa_N \Big(\epsilon(k), u(k-1) \Big)$ solves Problem 6.*

Remark 33 (**Practical asymptotic stability**)**.** *Where the set $\mathbb{G}_f$ cannot be efficiently computed e.g. when the system has very high dimension, it may be possible to still achieve asymptotic stability and feasibility in practice without these terminal ingredients. In this case, the terminal constraint* (6.16) *and the terminal cost* (6.19) *can both be omitted from the FOMPC formulation above without losing asymptotic stability if the prediction horizon, N is made large enough. How big N can be to theoretically guarantee this stability and feasibility is still an important theoretical question that may need to be answered. In most practical settings however, a stabilizing value of N can be determined in simulation using a trial-and-error approach. A major difficulty when formulating a FOMPC problem using very large values of N is that the resulting online optimization problem can get very large and therefore difficult to solve in real time.*

6.3.2 Inherent robustness of nominal FOMPC: An ISS stability analysis

Due to the implicit feedback implementation, nominal FOMPC is expected to show an inherent robustness to uncertainty in the system dynamics. The goal of this section is to analyse this nominal robustness, i.e., we want to inquire if nominal FOMPC (i.e., FOMPC based on known system coefficients and piece-wise constant disturbances) is sufficiently robust to uncertainty and if possible establish obtainable bounds. For simplicity of presentation and to avoid the complications caused by the cross-term, we express the OCP (minimize (6.14) subject to (6.15), and (6.16)) for the FOMPC problem above in the following equivalent form:

$$\mathbb{P}(\epsilon(k))\colon \min_{\mathbf{v}(k)\in\mathscr{U}_N} V_N(\epsilon(k),\mathbf{v}(k)) \tag{6.23}$$

where the feasible region $\mathscr{U}_N(\epsilon(k))$ is defined as

$$\mathscr{U}_N(\epsilon(k)) \triangleq \left\{\mathbf{v}(k) \middle| \begin{array}{r} \epsilon(k+i+1) = \hat{\mathcal{A}}\epsilon(k+i) + \mathcal{B}v(k+i),\ \forall i \in \mathbb{I}_{[0,N-1]} \\ \epsilon(k+i) \in \mathbb{G},\ \forall i \in \mathbb{I}_{[0,N-1]} \\ \epsilon(k+N) \in \mathbb{G}_f \end{array}\right\}, \tag{6.24}$$

and the cost function $V_N(\epsilon(k),\mathbf{v}(k))$ is given by

$$V_N(\epsilon(k),\mathbf{v}(k)) = V_f(\epsilon(k+N)) + \sum_{i=0}^{N-1} l(e(k+i),v(k+i)). \tag{6.25}$$

Here,

$$v(k) = \delta u(k) + \mathcal{R}^{-1}\mathcal{N}^\top\epsilon(k), \quad l(e(k),v(k)) := \frac{1}{2}\Big(\epsilon(k)^\top\hat{\mathcal{Q}}\epsilon(k) + v(k)^\top\mathcal{R}v(k)\Big),$$
$$\hat{\mathcal{A}} = \mathcal{A} - \mathcal{B}\mathcal{R}^{-1}\mathcal{N}^\top,\ \hat{\mathcal{Q}} = \mathcal{Q} - \mathcal{N}\mathcal{R}^{-1}\mathcal{N}^\top \text{ and } V_f(\epsilon(k+N)) = \frac{1}{2}\epsilon(k+N)^\top\hat{\mathcal{P}}\epsilon(k+N), \tag{6.26}$$

and $\hat{\mathcal{P}}$ is a terminal penalty required to guarantee the closed loop stability of the FOMPC. The computation of $\hat{\mathcal{P}}$ will be detailed later on in this analysis. The vector $\mathbf{v}(k)$ is the sequence of translated control increments over the N-step prediction horizon:

$$\mathbf{v}(k) := \Big\{v(k), v(k+1), \ldots, v(k+N-1)\Big\}. \tag{6.27}$$

The solution of $\mathbb{P}(\epsilon(k))$ yields the optimal control sequence $\mathbf{v}^*(k)$ and the corresponding optimal cost $V_N^0(\epsilon(k))$ which is also referred to as the value function. The nominal FOMPC law is then given by the first element of $\mathbf{v}^*(k)$ i.e.

$$\kappa_N(\epsilon(k)) = v(k) = \mathbf{v}^*(0). \tag{6.28}$$

The domain of the value function is the set of velocity states $\epsilon(k)$ for which $\mathbb{P}(\epsilon(k))$ has a solution:

$$\mathscr{E}_N \triangleq \{\epsilon(k) \in \mathbb{R}^{n_x+n_y} \colon \mathscr{U}_N(\epsilon(k)) \neq \emptyset\}. \tag{6.29}$$

To analyse the robust stability of the nominal FOMPC algorithm above, the following assumptions are in order.

Assumption 9 (Continuity of velocity dynamics, stage cost and terminal cost)**.** *The velocity dynamics $\epsilon(k+1) = \hat{\mathcal{A}}\epsilon(k) + \mathcal{B}v(k)$, the stage cost $l(\cdot,\cdot)$ and the terminal cost $V_f(\cdot)$ are continuous. Furthermore, we assume $l(0,0) = 0$ and $V_f(0) = 0$.*

Assumption 10 (Properties of the constraint set)**.** *The set $\mathbb{G}$ is compact and contains the origin*

Assumption 11 (Stage cost bounds)**.** *There exists a function $\alpha_1 \in \mathcal{K}_\infty$ such that*

$$l(e(k), v(k)) \geq \alpha_1(\|\epsilon(k)\|), \ \forall \epsilon \in \mathbb{G}. \tag{6.30}$$

Assumption 12 (Terminal Constraint)**.** *The set $\mathbb{G}_f$ is a constraint admissible positive invariant set for $\epsilon(k+1) = \hat{\mathcal{A}}\epsilon(k) + \mathcal{B}v(k)$ with $v(k) = \mathcal{K}_v\epsilon(k)$, where $\mathcal{K}_v = -\mathcal{K} + \mathcal{R}^{-1}\mathcal{N}$ i.e.,*

$$(\hat{\mathcal{A}} + \mathcal{B}\mathcal{K}_v)\mathbb{G}_f \subseteq \mathbb{G}_f \text{ and } \mathbb{G}_f \subseteq \mathbb{G}. \tag{6.31}$$

Remark 34. *Assumption 12 is met if $\mathbb{G}_f$ is chosen to satisfy equation* (6.13)*. See Remark 32 for details on the computation of this set.*

Assumption 13 (Terminal Cost)**.** *The function $V_f(\epsilon(k))$ is a control Lyapunov function i.e.,*

$$V_f(\epsilon(k+1)) - V_f(\epsilon(k)) \leq -l(\epsilon(k), \mathcal{K}_v\epsilon(k)), \ \forall \epsilon(k) \in \mathbb{G}_f. \tag{6.32}$$

Remark 35. *Assumption 13 is met if the terminal penalty $\hat{\mathcal{P}}$ is chosen to satisfy the standard discrete algebraic Riccati equation (DARE),*

$$\hat{\mathcal{P}} = \hat{\mathcal{A}}^\top \hat{\mathcal{P}} \hat{\mathcal{A}} + \hat{\mathcal{Q}} - \hat{\mathcal{A}}^\top \hat{\mathcal{P}} \mathcal{B}(\mathcal{R} + \mathcal{B}^\top \hat{\mathcal{P}} \mathcal{B})^{-1} \mathcal{B}^\top \hat{\mathcal{P}} \hat{\mathcal{A}}. \tag{6.33}$$

Given the above assumptions, it can be easily shown that the value function has the following properties.

Lemma 6.3.2 (Properties of the value function)**.** *Suppose Assumptions 9, 10, 11, 12, and 13 hold, then there exists constants* $c_2 > c_1 > 0$ *such that the value function satisfies:*

$$c_1\|\epsilon\|^2 \leq \|V_N^0(\epsilon)\| \leq c_2\|\epsilon\|^2, \tag{6.34a}$$
$$V_N^0(\epsilon^+) \leq \gamma V_N^0(\epsilon) \tag{6.34b}$$

for all $\epsilon \in \mathscr{E}_N$*, where* $\gamma = 1 - \frac{c_1}{c_2}$ *and* $\epsilon^+ = \hat{\mathcal{A}}\epsilon + \mathcal{B}\kappa_N(\epsilon)$*.*

Proof. See [181] for proof of this result. □

The following remarks show how to estimate the constants c_1 and c_2.

Remark 36 (Estimation of c_1)**.** *Since the value function is, by definition, the optimal cost*

$$V_N^0(\epsilon) = \frac{1}{2}\epsilon(N)^\top\hat{\mathcal{P}}\epsilon(N) + \frac{1}{2}\sum_{k=0}^{N-1}\Big(\epsilon(k)^\top\hat{\mathcal{Q}}\epsilon(k) + v(k)^\top\mathcal{R}v(k)\Big) \tag{6.35}$$

It follows that $V_N^0(\epsilon) \geq \epsilon(0)^\top\hat{\mathcal{Q}}\epsilon(0) \geq \lambda_{min}(\hat{\mathcal{Q}})|\epsilon|^2$*, hence*

$$c_1 = \lambda_{min}(\hat{\mathcal{Q}}) = \lambda_{min}(\mathcal{Q} - \mathcal{N}\mathcal{R}^{-1}\mathcal{N}^\top). \tag{6.36}$$

Remark 37 (Estimation of c_2)**.** *From [182], the terminal cost upper bounds the value function such that*

$$V_N^0(\epsilon) \leq \epsilon(N)^\top\hat{\mathcal{P}}\epsilon(N), \quad \forall\epsilon \in \mathbb{G}_f \tag{6.37}$$

and therefore, $V_N^0(\epsilon) \geq \lambda_{max}(\hat{\mathcal{P}})|\epsilon|^2 = \alpha(\|\epsilon\|)$ *is an upper bounding* $\mathcal{K}_\infty$ *function within* $\mathbb{G}_f$*. To extend this definition to the entire region of attraction of the FOMPC, that is the set* $\mathscr{E}_N$*, we utilize Proposition 2.18 of [182] which guarantees the existence of* $c > \eta > 0$ *such that the* $\mathcal{K}_\infty$ *function* $\beta(\|\epsilon\|) = \frac{c}{\eta}\alpha(\|\epsilon\|) = \frac{c}{\eta}\lambda_{max}(\hat{\mathcal{P}})\|\epsilon\|^2$ *bounds the value function within* $\mathscr{E}_N$*. Therefore,*

$$c_2 = \frac{c}{\eta}\lambda_{max}(\hat{\mathcal{P}}). \tag{6.38}$$

It remains to find the least conservative pair (c, η) *such that* c_2 *is the tightest upper bounding constant. From [182], this can be computed as*

$$\eta = \max_{\epsilon \in \mathscr{B}_R} \alpha(\|\epsilon\|) \tag{6.39}$$

where $\mathscr{B}_R := \{\epsilon \in \mathbb{R}^{n_x+n_y} \mid \|\epsilon| \leq R\}$ *and* c *is defined in [182] as the constant which implicitly defines the largest level set such that*

$$V_N^0(\epsilon) \leq c, \quad \forall \epsilon \in \mathscr{E}_N. \tag{6.40}$$

Refer to [182] for further details.

Lemma 6.3.3 (Lipschitz continuity of value function). *The value function* $V_N^0(\cdot)$ *satisfies*

$$\|V_N^0(\epsilon_1) - V_N^0(\epsilon_2)\| \leq L\|\epsilon_1 - \epsilon_2\| \text{ over } \mathscr{E}_N \tag{6.41}$$

with Lipschitz constant $L \geq 0$*.*

Proof. The proof follows the same procedure as Theorem C.29 of [182]. □

Lemma 6.3.2 establishes asymptotic stability of the nominal dynamics but may fail to guarantee the stability of the disturbed dynamics. In the next section, we analyze the stability of the nominal FOMPC law in disturbed conditions.

Stability analysis of nominal FOMPC in disturbed conditions

Having reformulated the nominal FOMPC problem as the OCP $\mathbb{P}(\epsilon(k))$ above and given the stated assumptions, we now analyse the stability properties of the nominal FOMPC for the disturbed system:

$$\epsilon^d(k+1) = \hat{\mathcal{A}}\epsilon(k) + \mathcal{B}v(k) + d(k) \tag{6.42}$$

where $d(k)$ is the aggregate (unknown) perturbation to the nominal velocity dynamics i.e. parametric uncertainty and additive disturbances (See remark below).

Remark 38. *Consider the uncertain linear system :*

$$x(k+1) = (A + \Delta A)x(k) + (B + \Delta B)u(k) + Ew(k), \tag{6.43a}$$

$$y(k) = Cx(k) \tag{6.43b}$$

where $(\Delta A, \Delta B)$ *are unknown, norm bounded perturbations to the nominal system coefficients* (A, B) *and* $w(k)$ *is the unknown additive disturbance at sample time* k*. The uncertainty in* $(\Delta A, \Delta B)$ *is propagated to the coefficient matrices of the steady-state optimality error* (6.6) *resulting in the uncertain error equation*

$$e(k) = (\Lambda_y + \Delta\Lambda_y)Cx(k) + (\Lambda_u + \Delta\Lambda_u)u(k) + r. \tag{6.44}$$

From the equations above, the additive disturbance $d(k)$ *can be expressed as*

$$d(k) = (E_w\Delta A + E_p\Delta\Lambda_y C)C_x\epsilon(k) + (E_w\Delta B + E_p\Delta\Lambda_u)\delta u(k) + E_w E\delta w(k) \tag{6.45}$$

where

$$\delta w(k) = w(k) - w(k-1),\ E_w = \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix},\ E_p = \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y \times n_y} \end{bmatrix} \text{ and } C_x = \begin{bmatrix} I_{n_x} & \mathbf{0} \end{bmatrix}. \tag{6.46}$$

Due to the disturbance $d(k)$, the true successor state is no longer $\epsilon(k+1)$ but $\epsilon^d(k+1)$ and therefore Lemma 6.3.2 may no longer guarantee closed loop stability as the inequality $V_N^0(\epsilon^{d+}) \leq \gamma V_N^0(\epsilon^d)$ may not be satisfied. However, if the origin is asymptotically stable for the nominal velocity dynamics (i.e., the system without d), then it may be possible to find a set $\Omega(\bar{R})$ for which the conditions of Lemma 6.3.2 are satisfied and therefore guarantee asymptotic stability of the uncertain system [182] with respect to the set $\Omega(\bar{R})$. We now present the main result which establishes the stability properties of the uncertain velocity dynamics assuming the origin is an asymptotically stable equilibrium for the nominal velocity dynamics. Using the obtained results, we also establish an upper bound on the uncertainty $d(k)$ which if met guarantees closed-loop stability of the uncertain velocity dynamics (6.42) under the nominal FOMPC control law:

$$\kappa_N(\epsilon(k)) = v(k) = \mathbf{v}^*(0). \tag{6.47}$$

Theorem 6.3.4 (Robust stability of nominal FOMPC)**.** *Suppose Assumptions 9, 10, 11, 12, and 13 hold, and let*

$$\epsilon(0) \in \Omega(\bar{R}) \triangleq \{\ \epsilon \mid\ V_N^0(\epsilon) \leq \bar{R}\} \subset \mathscr{E}_N \tag{6.48}$$

where $\Omega(\bar{R})$ denotes the largest sublevel set contained in $\mathscr{E}_N$.
If there exists scalars $R \in (0, \bar{R}]$ and L satisfying Lemma 6.3.3 such that for all admissible ϵ and δw

$$L\|(E_w\Delta AC_x + E_p\Delta\Lambda_y CC_x)\epsilon\| + L\|(E_w\Delta B + E_p\Delta\Lambda_u)\kappa_N(\epsilon)\| + L\|E_w E\delta w\| \leq (1-\gamma)R \tag{6.49}$$

for all $\epsilon \in \Omega(\bar{R})$, then

(i) the set $\Omega(R) \triangleq \{ \epsilon \mid V_N^0(\epsilon) \leq R\} \subset \Omega(\bar{R})$ is positively invariant for the uncertain velocity dynamics (6.42)*.*

(ii) the set $\Omega(\bar{R})$ is also positively invariant for (6.42)*. Therefore, the states ϵ remains in $\Omega(\bar{R})$ for all time, and enter and remain within the set $\Omega(R)$ after some finite time.*

(iii) the actual inputs and outputs (u, y) of the uncertain system (6.43) *converge to a neighbourhood of the optimal equilibrium $\bar{z}^*$ for the steady-state optimization problem* (6.2)*.*

Proof. (i) Consider some $\epsilon \in \Omega(R)$, then by Lemma 6.3.2, $V_N^0(\epsilon^+) \leq \gamma V_N^0(\epsilon) \leq \gamma R$ and by Lipschitz continuity,

$$V_N^0(\epsilon^{d+}) - V_N^0(\epsilon^+) \leq L\|\epsilon^{d+} - \epsilon^+\|. \tag{6.50}$$

But the optimal one step ahead prediction is

$$\epsilon^+ = \hat{\mathcal{A}}\epsilon + \mathcal{B}v, \tag{6.51}$$

and the actual successor state is

$$\begin{aligned} \epsilon^{d+} &= \hat{\mathcal{A}}\epsilon^d + \mathcal{B}v \\ &= \hat{\mathcal{A}}\epsilon + \mathcal{B}v + d. \end{aligned} \tag{6.52}$$

Therefore, $\|\epsilon^{d+} - \epsilon^+\| = \|d\|$ and the bound (6.50) becomes

$$V_N^0(\epsilon^{d+}) - V_N^0(\epsilon^+) \leq L\|d\| \tag{6.53}$$

which simplifies to

$$V_N^0(\epsilon^{d+}) \leq \gamma R + L\|d\|. \tag{6.54}$$

If there exist some $R \in (0, \bar{R}]$ such that

$$L\|d\| \leq (1-\gamma)R \tag{6.55}$$

then

$$V_N^0(\epsilon^{d+}) \leq \gamma R + (1-\gamma)R \leq R, \tag{6.56}$$

which implies that $\epsilon^{d+} \in \Omega(R)$ and therefore the value function $V_N^0(\cdot)$ is an ISS Lyapunov function for the uncertain velocity dynamics (6.42); We obtain the bound (6.49) by substituting (6.45) in (6.55) and applying the triangle inequality.
(ii) Consider some $\epsilon(0) \in \Omega(\bar{R}) \setminus \Omega(R)$. Using the ISS property of the value function and assuming the existence of $\rho \in (\gamma, 1]$, then

$$V_N^0(\epsilon^+) \leq \gamma V_N^0(\epsilon) + (\rho - \gamma)V_N^0(\epsilon) \leq \rho V_N^0(\epsilon). \tag{6.57}$$

Consequently,

$$V_N^0(\epsilon(k)) \leq \rho^k \bar{R}. \tag{6.58}$$

Hence $V_N^0(\epsilon(k)) \leq R$ after some finite k', implying $\epsilon(k') \in \Omega(R)$.
(iii) Finally, if each $\epsilon(k) \to \Omega(R)$, then $(\delta x(k), e(k-1))$ converge to, and remain within a neighbourhood of the origin, and as a result $(u(k), y(k))$ converge to and remain within a neighbourhood of the optimal steady-state set-point $\bar{z}^*$ as defined by the steady-state optimization problem (6.2). □

Discussions

The above result establishes the convergence of the uncertain system (6.43) to the optimal solution of the steady-state optimization problem (6.2), when in closed-loop with the nominal FOMPC law $u(k) = u(k-1) + \delta u(k)$, and when the model uncertainty and disturbances satisfy the stated conditions. Although theoretically insightful, the condition (6.49) is somewhat abstract and may be difficult to compute in practice due to its dependence on uncertain parameters such as ΔA, ΔB, $\Delta\Lambda_y$ and $\Delta\Lambda_u$. Also even if it was possible to compute the bound in (6.49), it may only be satsified for a very limited value of the uncertainty hence a very conservative condition. However, the condition (6.49) can in most practical circumstances be used to derive intuition on how best to tune the FOMPC to maximize its inherent robustness. From (6.49), making the left hand side (LHS) smaller than the right hand side (RHS) by

- ensuring the model uncertainty and additive disturbances are sufficiently small, or
- the decay constant γ are sufficiently small (i.e., stronger regulatory action is required from the FOMPC controller)

can improve the robust stability of the FOMPC algorithm. This can be achieved via a higher penalty on the steady-state optimality error and smaller penalty on the input deviation.

If the model uncertainty is exclusive to the coefficient matrix A (i.e $\Delta B = 0$), the additive disturbance is constant (i.e. $\delta w = 0$) and the system has the robust steady-state gain property defined in Definition 3 (i.e $\Delta\Lambda_y = 0,\ \Delta\Lambda_u = 0$), then the condition in Theorem 6.3.4 can be simplified as given by the corollary below.

Corollary 6.3.4.1. *If the system* (6.43) *is uncertain only in A, the additive disturbance is piecewise constant and the robust steady-state gain property (see Definition 3) is satisfied, then the robust stability guarantee of Theorem 6.3.4 holds provided the uncertainty in A i.e ΔA satisfies the following bound:*

$$L\|E_w \Delta A C_x \epsilon\| \leq \frac{(1-\gamma)R}{\sqrt{R/c_1}} \tag{6.59}$$

which simplifies to

$$\|\Delta A\| \leq \frac{1}{\sqrt{R/c_1}}\left(\frac{(1-\gamma)R}{L\|E_w\|\|C_x\|}\right) \tag{6.60}$$

if E_w and C_x are square matrices of equal dimension as ΔA.

Proof. With no uncertainty in B, a robust steady-state gain property and piecewise constant disturbances,

$$\Delta B = 0,\ \Delta\Lambda_y = 0,\ \Delta\Lambda_u = 0,\ \text{and } \delta w = 0 \tag{6.61}$$

and the condition (6.49) simplifies to

$$L\|E_w \Delta A C_x \epsilon\| \leq (1-\gamma)R. \tag{6.62}$$

But

$$\|E_w \Delta A C_x \epsilon\| \leq \|E_w \Delta A C_x\|\|\epsilon\| \tag{6.63}$$

and therefore (6.62) reduces to

$$L\|E_w \Delta A C_x\|\|\epsilon\| \leq (1-\gamma)R. \tag{6.64}$$

According to Theorem 6.3.4, $\epsilon \in \Omega(R)$ and the lower bound of $V_N^0(\epsilon)$ implies

$$c_1\|\epsilon\|^2 \leq R \implies \|\epsilon\|^2 \leq \frac{R}{c_1}, \ \therefore \|\epsilon\| \leq \sqrt{\frac{R}{c_1}} \tag{6.65}$$

and therefore (6.62) reduces to

$$L\|E_w \Delta A C_x\| \leq \frac{(1-\gamma)R}{\sqrt{R/c_1}}. \tag{6.66}$$

However if E_w and C_x are square matrices, then by the sub-multiplicative property of matrix norms for square matrices [83] i.e.,

$$\|XY\| \leq \|X\|\|Y\| \tag{6.67}$$

we can rewrite (6.66) as (6.60) which concludes the proof. □

Also, in the absence of model uncertainty i.e. $(\Delta B = 0,\ \Delta A = 0,\ \Delta\Lambda_y = 0,\ \Delta\Lambda_u = 0)$, the condition in Theorem 6.3.4 can be further simplified as given by the following corollary.

Corollary 6.3.4.2. *If the system* (6.43) *has no model uncertainty but the additive disturbance is not piecewise constant, then the robust stability guarantee of Theorem 6.3.4 holds provided the change in the additive disturbance between consecutive sample times (i.e., δw) satisfies the following bound:*

$$\|\delta w\| \leq \frac{(1-\gamma)R}{L\|E_w E\|}. \tag{6.68}$$

Proof. The result is obtained by substituting $\Delta B = 0,\ \Delta A = 0,\ \Delta\Lambda_y = 0,\ \Delta\Lambda_u = 0$ in the inequality (6.49) and simplifying the resulting expression. □

6.4 Numerical examples

In this section, we demonstrate the regulation and feedback-optimizing capabilities of the proposed velocity model-based FOMPC regulator. We present two examples of feedback-optimizing model predictive control using numerical simulation studies of the proposed algorithm. In the first example, we utilize a simple system to illustrate key design elements of the FOMPC algorithm. We then present in the second example, an application of FOMPC to a more complicated multivariable system.

6.4.1 Example 1: 2nd order system

We consider the continuous-time open-loop unstable system :

$$\dot{x}(t) = \begin{bmatrix} -1 & 3 \\ 1 & 4 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 4 \end{bmatrix} u(t) + w(t), \tag{6.69a}$$

$$y(t) = \begin{bmatrix} 1 & 5 \end{bmatrix} x(t). \tag{6.69b}$$

The system is stabilizable and observable, meeting Assumption 6. The disturbance $w(t)$ is unknown but slowly varying as

$$w(t) = \begin{cases} \begin{bmatrix} -0.1 & 0.3 \end{bmatrix}^\top & 0 \leq t < 20, \\ \begin{bmatrix} 0.2 & -0.3 \end{bmatrix}^\top & 20 \leq t < 40, \\ \begin{bmatrix} 0.1 & 0 \end{bmatrix}^\top & t \geq 40. \end{cases} \tag{6.70}$$

The following inequality constraints are present on the input, output and disturbances;

$$\begin{aligned} \mathbb{U} := \{u : -20 \leq u \leq 20\}, \ \mathbb{Y} := \{y : -10 \leq y \leq 10\}, \\ \mathbb{W} := \{w : -2 \times I_2 \leq w \leq 2 \times I_2\}. \end{aligned} \tag{6.71}$$

The objective function for the steady-state optimization problem is

$$\Phi(z) = \frac{1}{2} \begin{bmatrix} u \\ y \end{bmatrix}^T \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} + \begin{bmatrix} R_u \\ R_y \end{bmatrix}^\top \begin{bmatrix} u \\ y \end{bmatrix} \tag{6.72}$$

where

$$Q_{uu} = 10 \ , Q_{yy} = 5 \ , Q_{uy} = Q_{yu}^\top = 5, \ R_u = 1, \ R_y = 1.$$

For design and implementation of the discrete-time FOMPC, the system is discretized using zero-order hold with a sampling interval of 0.1 seconds. The input–output DC gain matrix of the discrete-time system is,

$$G_u = -4.7143. \tag{6.73}$$

The velocity state for the controller is

$$\epsilon(k) = \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} = \begin{bmatrix} x(k) - x(k-1) \\ \Lambda_y y(k-1) + \Lambda_u u(k-1) + r \end{bmatrix} \tag{6.74}$$

where Λ_y, Λ_u and r follow from the choices of Φ parameters as

$$\Lambda_y = 3.9394 \quad \Lambda_u = 2.8788, \quad r = 0.7879.$$

The matrix

$$S = \begin{bmatrix} -0.0790 & 0.3539 & 0.1626 \\ 0.1180 & 0.5109 & 0.4998 \\ 3.9394 & 19.6970 & 2.8788 \end{bmatrix}$$

has full rank satisfying Assumption 6.

The transient performance criterion is chosen with $Q_e = 500 \times I_2$ and $R_\delta = 1 \times I_2$; these values satisfy the hypothesis of Proposition 3. A prediction horizon of $N = 5$ is used. From the design parameters above, the FOMPC has the following parameters:

$$\begin{gathered} \mathcal{A} = \begin{bmatrix} 0.9210 & 0.3539 & 0 \\ 0.1180 & 1.5109 & 0 \\ 3.9394 & 19.6970 & 1.00 \end{bmatrix}, \ \mathcal{B} = \begin{bmatrix} 0.1626 \\ 0.4998 \\ 2.8788 \end{bmatrix}, \\ \mathcal{C} = \begin{bmatrix} 3.9394 & 19.6970 & 1.00 \end{bmatrix}, \ \mathcal{D} = 2.8788, \\ \mathcal{R} = 419.3710, \ \mathcal{N} = \begin{bmatrix} 0.5670 \\ 2.8352 \\ 0.1439 \end{bmatrix} \times 10^3, \\ \mathcal{Q} = \begin{bmatrix} 0.0776 & 0.3880 & 0.0197 \\ 0.3880 & 1.9399 & 0.0985 \\ 0.0197 & 0.0985 & 0.005 \end{bmatrix} \times 10^4, \text{ and} \\ \mathcal{K} = \begin{bmatrix} 0.9467 & 4.7837 & 0.1634 \end{bmatrix}. \end{gathered} \tag{6.75}$$

From the inequality $\mathbb{X} \times \mathbb{U}$ given by,

$$\mathbb{X} \times \mathbb{U} \coloneqq \left\{ \begin{bmatrix} x \\ u \end{bmatrix} : \begin{bmatrix} 1 & 5 & 0 \\ -1 & -5 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq \begin{bmatrix} 10 \\ 10 \\ 20 \\ 20 \end{bmatrix} \right\} \tag{6.76}$$

we compute the constraint $\mathbb{G}$ according to (6.12) as:

$$\mathbb{G} \coloneqq \left\{ \epsilon : \begin{bmatrix} 0.8236 & 3.3270 & 0.3004 \\ -0.8236 & -3.3270 & -0.3004 \\ 0.2414 & 2.2893 & -0.0637 \\ -0.2414 & -2.2893 & 0.0637 \end{bmatrix} \epsilon \leq \begin{bmatrix} 9.7650 \\ 9.2916 \\ 19.3043 \\ 19.4047 \end{bmatrix} \right\}. \tag{6.77}$$

From the set $\mathbb{G}$, we compute the maximum output admissible invariant set for the velocity dynamics as:

$$\mathbb{G}_f \coloneqq \left\{ \epsilon : \begin{bmatrix} 0.8236 & 3.3270 & 0.3004 \\ -0.8236 & -3.3270 & -0.3004 \\ 0.2414 & 2.2893 & -0.0637 \\ -0.2414 & -2.2893 & 0.0637 \\ -0.1853 & -1.4963 & -0.1345 \\ 0.1853 & 1.4963 & 0.1345 \end{bmatrix} \epsilon \leq \begin{bmatrix} 9.7650 \\ 9.2916 \\ 19.3043 \\ 19.4047 \\ 9.7650 \\ 9.2916 \end{bmatrix} \right\}. \tag{6.78}$$

Figure 6.1 shows the result of applying the FOMPC law designed above to the continuous-time system (6.69) while the disturbances change according to (6.70). We can see that the FOMPC tracks the optimal solution $z^*(\bar{w})$ to the steady-state optimization with the cost function (6.72) with a very good transient performance (i.e. small overshoot and fast convergence). Stability is also maintained and the inequality constraints are satisfied at all times. A benefit of the FOMPC over standard feed-forward optimization/control is its inherent robustness to model uncertainty which was theoretically shown in Theorem 6.3.4. In simulation, we obtained better robust stability performance under mild model perturbations whenever Q_e was significantly larger than R_δ confirming the intuition derived from Theorem 6.3.4.

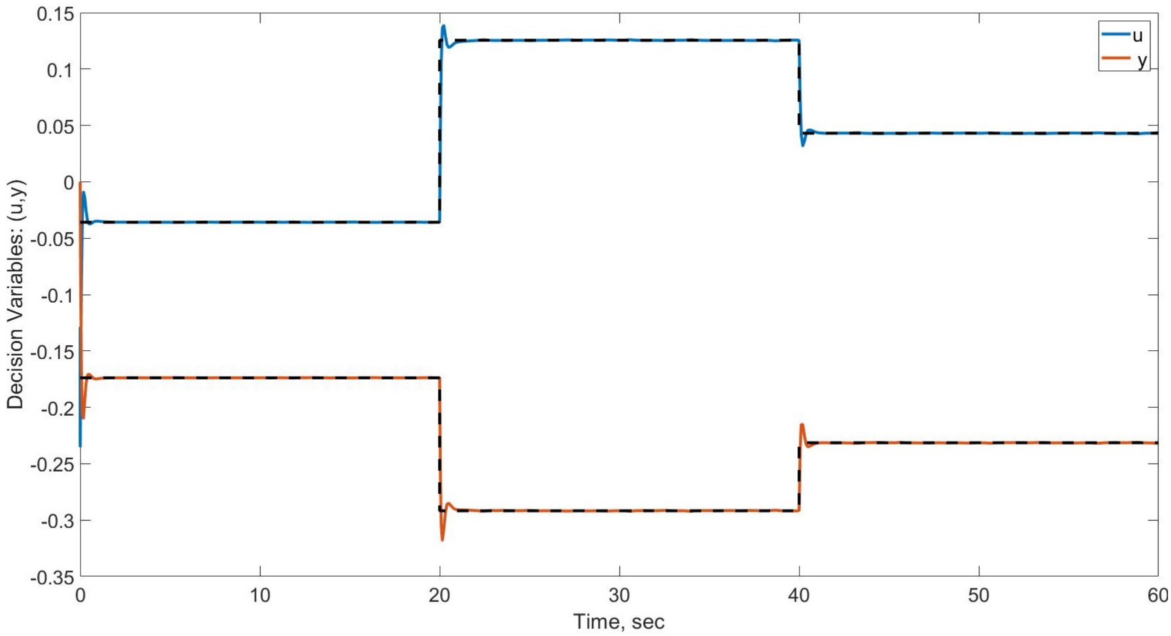

Fig. 6.1 FOMPC: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

6.4.2 Example 2: 4th order system

We consider here a modified (two-input) version of the continuous-time open-loop unstable system in [128]:

$$\dot{x}(t) = \begin{bmatrix} -1 & -4 & -1 & 3 \\ 1 & -4 & -1 & -3 \\ -1 & 4 & -1 & -9 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 1 \\ 1 & 4 \\ 4 & 2 \\ 1 & 0 \end{bmatrix} u(t) + w(t), \tag{6.79a}$$

$$y(t) = \begin{bmatrix} 1 & -1 & 0 & -4 \\ 1 & 0 & 2 & 0 \end{bmatrix} x(t). \tag{6.79b}$$

The system is reachable and observable. The disturbance $w(t)$ is unknown but slowly varying as

$$w(t) = \begin{cases} \begin{bmatrix} -1 & 3 & 1 & 2 \end{bmatrix}^\top & 0 \le t < 5, \\ \begin{bmatrix} 2 & 3 & 0 & 1.5 \end{bmatrix}^\top & 5 \le t < 10, \\ \begin{bmatrix} 1 & 0 & 0 & 5 \end{bmatrix}^\top & t \ge 10. \end{cases} \tag{6.80}$$

For design and implementation of the discrete-time controller, the system is discretized using zero-order hold. The following inequality constraints are present on the input, output and disturbances;

$$\begin{aligned} \mathbb{U} &:= \left\{ u : \begin{bmatrix} -12 \\ -15 \end{bmatrix} \leq u \leq \begin{bmatrix} 12 \\ 15 \end{bmatrix} \right\}, \ \mathbb{Y} := \left\{ y : \begin{bmatrix} -3 \\ -15 \end{bmatrix} \leq y \leq \begin{bmatrix} 3 \\ 15 \end{bmatrix} \right\}, \\ \mathbb{W} &:= \left\{ w : \begin{bmatrix} -1.5 & -3.5 & -1.5 & -2.5 \end{bmatrix}^\top \leq w \leq \begin{bmatrix} 1.5 & 3.5 & 1.5 & 2.5 \end{bmatrix}^\top \right\}. \end{aligned} \tag{6.81}$$

The objective function for the steady-state optimization problem is

$$\Phi(z) = \frac{1}{2} \begin{bmatrix} u \\ y \end{bmatrix}^T \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} + \begin{bmatrix} R_u \\ R_y \end{bmatrix}^\top \begin{bmatrix} u \\ y \end{bmatrix} \tag{6.82}$$

where

$$Q_{uu} = 2 \times I_2, Q_{yy} = \begin{bmatrix} 1/2 & 0 \\ 0 & 1 \end{bmatrix}, Q_{uy} = Q_{yu}^\top = \mathbf{1}_{2\times 2}$$

$$R_u = \mathbf{1}_{2\times 1}, \ R_y = 5 \times \mathbf{1}_{2\times 1}$$

The input–output DC gain matrix of the discrete-time system is,

$$G_u = \begin{bmatrix} 2.5 & -1.375 \\ 13.5 & 4.5 \end{bmatrix}. \tag{6.83}$$

The velocity state for the controller is

$$\epsilon_k = \begin{bmatrix} \delta x_k \\ e_{k-1} \end{bmatrix} = \begin{bmatrix} x_k - x_{k-1} \\ \Lambda_y y_{k-1} + \Lambda_u u_{k-1} + r \end{bmatrix} \tag{6.84}$$

where Λ_y, Λ_u and r follow from the choices of Φ parameters as

$$\Lambda_y = \begin{bmatrix} 0.1981 & -0.3019 \\ 0.1300 & 1.1300 \end{bmatrix} \quad \Lambda_u = \begin{bmatrix} 1.3019 & 0.0943 \\ 1.0922 & 1.1677 \end{bmatrix}$$

$$r = \begin{bmatrix} 4.6981 \\ 5.1300 \end{bmatrix}.$$

The transient performance criterion is chosen with $Q_e = 25 \times I_2$ and $R = 150 \times I_2$, satisfying Assumption 8. Figure 4.2 shows the result of applying FOMPC to the system

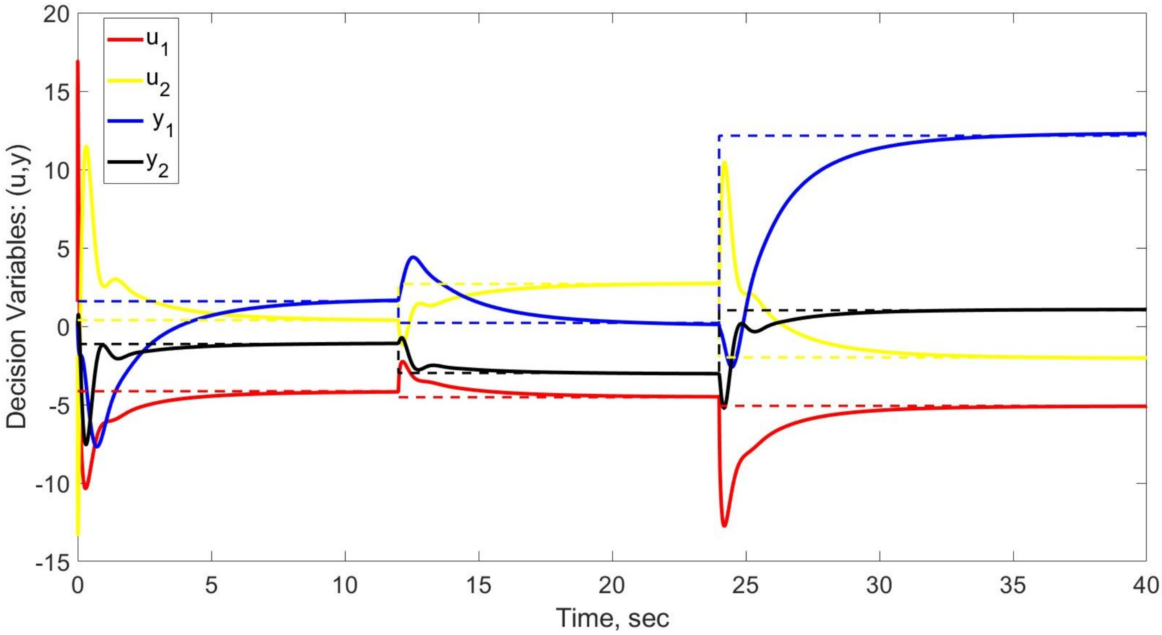

Fig. 6.2 FOMPC: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

while the disturbances change in the way described. It can be seen that FOMPC successfully tracks the optimal equilibrium points while maintaining stability and good transient performance, overcoming the issues of (i) non-existence of stabilizing PI gains for the static OSS controller presented in [128], and (ii) poor dynamic performance (large overshoot and slow convergence) and potential high dimensionality of the dynamic OSS controller proposed in the same work. Moreover, it should be noted that designing the FOMPC controller for closed-loop stability and optimal steady-state tracking is significantly easier, and more systematic, than the manual tuning of the PI OSS controller of [128]; indeed, for the presented example it was not possible to find stabilizing PI gains for the system under OSS control.

6.4.3 Discussion

The closed-loop performance of the FOMPC algorithm above is determined by the following tuning parameters: the weighting matrices Q_e and R_δ, and the prediction horizon N. It was observed that larger values of N yielded much improved responses but resulted in an MPC problem that took longer to solve. This is because larger values of N ensures the model prediction represents a significant portion of the system dynamics but ultimately lead to bigger online optimization problems. A prediction horizon of $N = 5$ was found to be a good compromise between performance and computational complexity for the simulation examples presented above.

The weights Q_e and R_δ are penalties placed on the magnitude of the steady-state tracking error and the size of the control input deviation respectively. From the simulations, we observed that larger values of Q_e relative to R_δ produced input/output responses that were under-damped with faster settling times and aggressive control moves. This often resulted in the possibility of overshoots and oscillations in the closed-loop response.

In the same vein, larger values of R_δ relative to Q_e resulted in very cautious and slow control moves producing input/output responses that were highly damped with longer settling times and fewer oscillations. A good compromise between an aggressive under-damped response and a slower highly damped response was achieved for the simulation results above by careful selection of Q_e and R_δ.

In order to obtain a cost function that is positive definite and as a result guarantee closed-loop stability of the FOMPC algorithm, the weights Q_e and R_δ had to satisfy the condition in Proposition 3. It may be useful to note that for some steady-state economic cost functions, finding values of Q_e that meet the conditions of Proposition 3 can be non-trivial. It would be interesting to investigate more systematic ways of tuning the FOMPC controller such that the conditions of Proposition 3 are much easier to satisfy. For the simulation results in this thesis, a laborious trial and error approach has been adopted.

Finally, uncertainties in the dynamic model of the linear system was observed to degrade the stability and performance of the FOMPC controller. However, for small values of model uncertainty, the inherent robustness of the FOMPC was adequate to handle this uncertainty and still guarantee closed-loop stability and decent dynamic performance. A detailed analysis of the impact of model uncertainty on the nominal FOMPC controller will be discussed in the next chapter.

6.5 Conclusion

A model predictive control algorithm that achieves asymptotic convergence to an unknown, optimal steady-state equilibrium while minimizing a transient performance criterion and respecting the system inequality constraints in the transients between optimal steady-states has been presented. The approach uses the residual of the KKT optimality condition for the steady-state optimization as a tracking error, and employs this as a state in a velocity-model form of MPC for tracking. The proposed controller is only moderately more complex than a conventional linear tracking MPC controller, but avoids the the need to explicitly solve the steady-state optimization problem. The results presented in this Chapter are only valid for systems with known dynamics and unknown additive disturbances. In the next Chapter, we will relax this assumption by considering model uncertainty in the FOMPC design. Also, future work will focus on generalizations such as relaxing the assumptions of convexity in the cost, linearity in the dynamics, and considering the important problem of tracking unreachable set-points.

6.6 Appendix A: Proof of Equation (6.11)

From the system dynamics,

$$x(k) = Ax(k-1) + Bu(k-1) + Ew(k-1) \tag{6.85}$$

we obtain

$$\delta x(k) = (A - I_{n_x})x(k-1) + Bu(k-1) + Ew(k-1). \tag{6.86}$$

But ,

$$e(k-1) = \Lambda_y Cx(k-1) + \Lambda_u u(k-1) + r, \tag{6.87}$$

$$\therefore \quad \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} = S \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix} w(k-1) + \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix} r. \tag{6.88}$$

Making $\begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix}$ the subject of the above equation yields,

$$\begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} = S^{-1}\left(\begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} - \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix} w(k-1) \right). \tag{6.89}$$

Also from the system dynamics, we can derive,

$$\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} E \\ \mathbf{0}_{n_u \times n_w} \end{bmatrix} w(k-1). \tag{6.90}$$

$$
\begin{aligned}
\therefore \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} &= \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} \\
&\quad + \left(\begin{bmatrix} E \\ \mathbf{0}_{n_u \times n_x} \end{bmatrix} - \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_x} \end{bmatrix} \right) w(k-1).
\end{aligned} \tag{6.91}
$$

We can therefore compactly express (6.91) as the equation (6.11).

Chapter 7

Robust Feedback Optimizing Model Predictive Control

Being a model based solution, the nominal FOMPC design presented previously is adversely affected by model uncertainty. The inherent robustness of nominal FOMPC can only handle small perturbation to the nominal model. To deal with polytopic model uncertainty, we present in this chapter two robust FOMPC designs. The first approach is based on the computationally efficient but conservative tube-based MPC while the second approach adopts the more computationally complex min-max MPC formulation.

This chapter is organized as follows. Section 7.1 is the introduction. Section 7.2 describes the tube-based robust FOMPC design. Section 7.3 describes the min-max robust FOMPC design. Section 7.4 presents illustrative examples. Section 7.5 is the conclusion and Section 7.6 contains the appendix.

7.1 Introduction

Although nominal FOMPC is inherently robust against uncertainty (as shown in Chapter 6), it cannot explicitly deal with model uncertainty. To explicitly handle model uncertainty, there exists two broad classes of robust MPC algorithms. One is based on a min-max formulation of the optimal control problem solved to generate the MPC laws [116]. The other adopts the so-called tube-based MPC design where the effect of the disturbance over the prediction horizon is handled a priori using a robust state feedback control law and disturbance invariant sets [28, 154]. Although the tube-based approach is simpler with a computational complexity comparable to nominal FOMPC, it may yield conservative performance and also the computation of a minimal RPI set for the uncertainty can be non-trivial and sometimes impossible. On the other hand, min-max robust MPC although computationally less efficient can guarantee convergence to the actual steady-state optimum for quadratically stabilizable systems without relying on the explicit computation of disturbance invariant sets.

7.2 Robust FOMPC Design: A tube-based approach

To obtain FOMPC controllers that are guaranteed stable and feasible for arbitrary but known uncertainty bounds, a tube-based approach to FOMPC will be developed in this section. Although tube-based FOMPC can result in more conservative performance compared to a nominal one, it allows feasibility and convergence guarantees to be obtained for arbitrary bounds on the uncertainty. We develop the tube-based robust FOMPC algorithm based on a velocity model of the linear system with polytopic model uncertainty and constant but bounded additive disturbances—i.e., $w(k) \in \mathbb{W}$, and $w(k) = w(k-1)$, where $\mathbb{W}$ is a known bounded set. Convergence of the closed-loop system to an invariant set centred around the optimal steady-state set-points $\bar{z}^*$ is guaranteed by use of a suitably defined auxiliary control law, terminal and disturbance invariant sets.

Consider the uncertain linear system (5.1) here recalled:

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k) + Ew(k), \tag{7.1a}$$

$$y(k) = Cx(k), \tag{7.1b}$$

$$\delta \in \Delta_l, \; w(k) \in \mathbb{W}. \tag{7.1c}$$

where all parameters and quantities retain their previous interpretation. Also, for all $\delta \in \Delta_l$, the following inequality constraints must be enforced,

$$x \in \mathbb{X}, \; y \in \mathbb{Y}, \text{ and } u \in \mathbb{U} \tag{7.2}$$

where $\mathbb{X} := \{x : y \in \mathbb{Y}\} \subseteq \mathbb{R}^n$.

Due to the properties of the unit simplex set Δ_l, the uncertain coefficients are contained in the polytope Ω i.e.,

$$\Big(A(\delta), B(\delta)\Big) \in \Omega \tag{7.3}$$

where Ω is defined as

$$\Omega := \; Co\Big(\big(A_1, B_1\big), \ldots, \big(A_l, B_l\big)\Big) = \sum_{i=1}^{l} \alpha_i \big(A_i, B_i\big). \tag{7.4}$$

Here Co refers to the convex hull.

For the uncertain velocity dynamics (7.1), let (A, B) be nominal values of the uncertain coefficients $\big(A(\delta), B(\delta)\big)$ such that:

$$\Delta A = A(\delta) - A, \text{ and } \Delta B = B(\delta) - B. \tag{7.5}$$

We can therefore rewrite (7.1) as the LTI system with the lumped additive disturbance $\tilde{w}$:

$$x(k+1) = Ax(k) + Bu(k) + \tilde{w}(k) \tag{7.6}$$

where

$$\tilde{w}(k) = \Delta A x(k) + \Delta B u(k) + Ew(k) \text{ and } \tilde{w}(k) \in \tilde{\mathbb{W}}. \tag{7.7}$$

The lumped disturbance set, $\tilde{\mathbb{W}}$ is defined as

$$\tilde{\mathbb{W}} := \Big\{ \tilde{w} \,:\, (A(\delta), B(\delta)) \in \Omega, \; (x, u) \in \mathbb{X} \times \mathbb{U}, \; w \in \mathbb{W} \Big\} \tag{7.8}$$

and explicitly computed as

$$\tilde{\mathbb{W}} = Co\Big(\bigcup_{i=1,\ldots,l} \big\{ (A_i - A)\mathbb{X} \oplus (B_i - B)\mathbb{U} \oplus E\mathbb{W} \big\} \Big). \tag{7.9}$$

The uncertainty in $(A(\delta), B(\delta))$ is propagated to the coefficient matrices of the steady-state optimality error (6.6) resulting in the uncertain error equation

$$e(k) = \Lambda_y(\delta) C x(k) + \Lambda_u(\delta) u(k) + r \tag{7.10}$$

where

$$\big(\Lambda_y(\delta) C, \Lambda_u(\delta)\big) \in \Omega_\Lambda, \tag{7.11}$$

and

$$\Omega_\Lambda := \; Co\Big(\big(\Lambda_{y1} C, \Lambda_{u1}\big), \ldots, \big(\Lambda_{yl} C, \Lambda_{ul}\big)\Big) = \sum_{i=1}^{l} \alpha_i \big(\Lambda_{yi} C, \Lambda_{ui}\big). \tag{7.12}$$

Let (Λ_y, Λ_u) be nominal values of the uncertain coefficients $(\Lambda_y(\delta), \Lambda_u(\delta))$, then similar to the state equation, we can express (7.10) as the disturbed error equation:

$$e(k) = \Lambda_y Cx(k) + \Lambda_u u(k) + r + \tilde{p}(k), \tag{7.13}$$

where

$$\begin{aligned} \tilde{p}(k) &= \Delta\Lambda_y Cx(k) + \Delta\Lambda_u u(k), \\ \Delta\Lambda_y = \Lambda_y(\delta) - \Lambda_y,\ \Delta\Lambda_u &= \Lambda_u(\delta) - \Lambda_u,\ \tilde{p}(k) \in \tilde{\mathbb{P}}. \end{aligned} \tag{7.14}$$

The set $\tilde{\mathbb{P}}$ is defined as

$$\tilde{\mathbb{P}} := \left\{ \tilde{p} \,:\, (\Lambda_y(\delta)C, \Lambda_u(\delta)) \in \Omega_\Lambda,\ (x,u) \in \mathbb{X} \times \mathbb{U} \right\}, \tag{7.15}$$

and explicitly computed as

$$\tilde{\mathbb{P}} = Co\left(\bigcup_{i=1,\ldots,l} \left\{ (\Lambda_{yi} - \Lambda_y)C\mathbb{X} \oplus (\Lambda_{ui} - \Lambda_u)\mathbb{U} \right\} \right). \tag{7.16}$$

Given (7.6) and (7.13), we can rewrite the uncertain velocity dynamics (5.13) as

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k) + d(k), \tag{7.17a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k) + f(k) \tag{7.17b}$$

where

$$d(k) = \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix} \delta\tilde{w}(k) + \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y \times n_y} \end{bmatrix} \delta\tilde{p}(k),\ f(k) = \delta\tilde{p}(k) \tag{7.18}$$

and

$$\epsilon(k) := \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x(k) &:= x(k) - x(k-1), \\ \delta u(k) &:= u(k) - u(k-1), \\ \delta\tilde{w}(k) &:= \tilde{w}(k) - \tilde{w}(k-1), \\ \delta\tilde{p}(k) &:= \tilde{p}(k) - \tilde{p}(k-1), \end{aligned} \tag{7.19}$$

with the coefficients defined thus:

$$\mathcal{A} = \begin{bmatrix} A & \mathbf{0}_{n\times p} \\ \Lambda_y C & I_p \end{bmatrix}, \qquad \mathcal{B} = \begin{bmatrix} B \\ \Lambda_u \end{bmatrix}, \tag{7.20a}$$

$$\mathcal{C} = \begin{bmatrix} \Lambda_y C & I_p \end{bmatrix}, \qquad \mathcal{D} = \Lambda_u. \tag{7.20b}$$

The disturbances $d(k)$ and $f(k)$ belong to the following sets:

$$d(k) \in \mathbb{D} := \begin{bmatrix} I_{n_x\times n_w} \\ \mathbf{0}_{n_y\times n_w} \end{bmatrix} \delta\tilde{\mathbb{W}} + \begin{bmatrix} \mathbf{0}_{n_x\times n_y} \\ I_{n_y\times n_y} \end{bmatrix} \delta\tilde{\mathbb{P}}, \text{ and } f(k) \in \mathbb{H} := \delta\tilde{\mathbb{P}}. \tag{7.21}$$

Remark 39. *The sets $\delta\tilde{\mathbb{W}}$ and $\delta\tilde{\mathbb{P}}$ can be computed from the sets $\tilde{\mathbb{W}}$ and $\tilde{\mathbb{P}}$ as [28]:*

$$\delta\tilde{\mathbb{W}} = \tilde{\mathbb{W}} \oplus (-\tilde{\mathbb{W}}), \text{ and } \delta\tilde{\mathbb{P}} = \tilde{\mathbb{P}} \oplus (-\tilde{\mathbb{P}}). \tag{7.22}$$

To facilitate the control design, we make the following assumptions on the sets $\mathbb{D}$ and $\mathbb{H}$.

Assumption 14 (Properties of uncertainty sets)**.** *The sets $\mathbb{D}$ and $\mathbb{H}$ are both bounded and contain the origin in their interior.*

Tube MPC design

To deal with the non-zero but bounded disturbances $d(k)$ and $f(k)$ in the velocity dynamics (7.17), we adopt an MPC approach based on tubes. The main feature of the tube based MPC approach as originally proposed in [154] is that the actual control action ($\delta u(k)$) is composed of a nominal control ($\delta\bar{u}(k)$) derived from an MPC problem for the following nominal (i.e., disturbance free) velocity dynamics,

$$\bar{\epsilon}(k+1) = \mathcal{A}\bar{\epsilon}(k) + \mathcal{B}\delta\bar{u}(k), \tag{7.23a}$$

$$\bar{e}(k) = \mathcal{C}\bar{\epsilon}(k) + \mathcal{D}\delta\bar{u}(k) \tag{7.23b}$$

and a linear feedback control ($\mathcal{K}$), designed to reject the error between the actual and nominal velocity state predictions i.e., ($\epsilon(k) - \bar{\epsilon}(k)$), thereby maintaining the trajectory of the actual velocity dynamics (7.17) within a tube centred around the nominal predictions of (7.23), despite the disturbances.
To simplify the presentation in the following paragraphs, the actual and nominal

velocity dynamics will be expressed in terms of the extended state variable $\begin{bmatrix} \epsilon(k) \\ r \end{bmatrix}$, respectively as

$$\begin{bmatrix} \epsilon(k+1) \\ r \end{bmatrix} = \tilde{\mathcal{A}} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} + \tilde{\mathcal{B}} \delta u(k) + \tilde{\mathcal{E}} d(k), \tag{7.24}$$

and

$$\begin{bmatrix} \bar{\epsilon}(k+1) \\ r \end{bmatrix} = \tilde{\mathcal{A}} \begin{bmatrix} \bar{\epsilon}(k) \\ r \end{bmatrix} + \tilde{\mathcal{B}} \delta \bar{u}(k) \tag{7.25}$$

where

$$\tilde{\mathcal{A}} := \begin{bmatrix} \mathcal{A} & \mathbf{0}_{(n_x+n_y)\times n_y} \\ \mathbf{0}_{n_y\times(n_x+n_y)} & I_{n_y} \end{bmatrix}, \ \tilde{\mathcal{B}} := \begin{bmatrix} \mathcal{B} \\ \mathbf{0}_{n_y\times n_u} \end{bmatrix}, \ \tilde{\mathcal{E}} := \begin{bmatrix} I_{(n_x+n_y)\times n_w} \\ \mathbf{0}_{n_y\times n_w} \end{bmatrix}. \tag{7.26}$$

Let

$$\varepsilon(k) = \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} - \begin{bmatrix} \bar{\epsilon}(k) \\ r \end{bmatrix}, \tag{7.27}$$

and according to [154] assume that, for all k, for the actual extended velocity dynamics (7.24) the following control law is considered

$$\delta u(k) = \delta \bar{u}(k) - \tilde{\mathcal{K}} \varepsilon(k) \tag{7.28}$$

where the gain $\tilde{\mathcal{K}}$ is defined a priori such that $\tilde{\mathcal{F}} = \tilde{\mathcal{A}} - \tilde{\mathcal{B}}\tilde{\mathcal{K}}$ is Schur.

Remark 40. *To obtain $\tilde{\mathcal{K}}$ such that $\tilde{\mathcal{F}}$ is Schur, we can define $\tilde{\mathcal{K}}$ as*

$$\tilde{\mathcal{K}} = \begin{bmatrix} \mathcal{K} & \mathbf{0}_{n_u\times n_y} \end{bmatrix} \tag{7.29}$$

where $\mathcal{K}$ is the FOLQC gain that stabilizes the entire family of velocity dynamics obtained from the parameter sets Ω and Ω_Λ (refer to the robust FOLQC laws in Chapter 5).

From (7.24), (7.25) and (7.28), it directly follows that

$$\varepsilon(k+1) = \tilde{\mathcal{F}} \varepsilon(k) + \tilde{\mathcal{E}} d(k). \tag{7.30}$$

If $\tilde{\mathcal{F}}$ is Schur and the set $\mathbb{D}$ is bounded, then there exists some set $\tilde{\mathbb{E}}$ such that $\varepsilon(k) \in \tilde{\mathbb{E}}$ for all $k \geq 0$. The set $\tilde{\mathbb{E}}$ is therefore a robust positive invariant (RPI) set for the error dynamics (7.30) i.e., for any $\varepsilon(k) \in \tilde{\mathbb{E}}$, $\varepsilon(k+1) \in \tilde{\mathbb{E}}$ for all $d(k) \in \mathbb{D}$.
Because (7.30) is independent of the nominal control input, $\delta\bar{u}(k)$, the set $\tilde{\mathbb{E}}$ can be computed offline and used to tighten the constraints in the nominal (constant disturbance) MPC problem to implicitly account for the disturbances omitted from the predictions of the nominal dynamics (7.23). In view of [179], one may be able to compute an invariant outer-approximation to the minimal RPI set $\tilde{\mathbb{E}}$ for the error dynamics (7.30) defined in (7.31).

$$\tilde{\mathbb{E}} = \bigoplus_{i=0}^{\infty} \tilde{\mathcal{F}}^i \tilde{\mathcal{E}} \mathbb{D}. \tag{7.31}$$

An algorithm for computing this invariant outer approximation to (7.31) can be devised in line with [179, 212].

The maximal output admissible set (MOAS) with tightened constraints

In the following, the nominal velocity dynamics (7.23) will be used to design an MPC algorithm to track the unknown optimum of the steady-state cost (7) with guarantees of stability and recursive feasibility for the inequality constraints $\mathbb{U}$ and $\mathbb{Y}$. Towards this goal, we will adopt

$$\delta\bar{u}(k) = -\mathcal{K}\bar{\epsilon}(k) \tag{7.32}$$

as a terminal control law to guarantee closed-loop stability of the nominal velocity dynamics (7.23) in line with [181]. To achieve recursive feasibility of the inequality constraints for any $d(k) \in \mathbb{D}$, the constraints on u and y must be reformulated in terms of tightened versions of the constraints on the MPC optimization variables $\delta\bar{u}$ and $\bar{\epsilon}$. To this end, similarly to [28], we write the relation between these variables as (see section 7.6 for details)

$$\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \tilde{\mathcal{C}} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} + \tilde{\mathcal{D}} q(k-1) \tag{7.33}$$

where $q(k) = \begin{bmatrix} \tilde{w}(k) \\ \tilde{p}(k) \end{bmatrix} \in (\tilde{\mathbb{W}} \times \tilde{\mathbb{P}})$,

$$\begin{aligned}
\tilde{\mathcal{C}} &= \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix}, \\
\tilde{\mathcal{D}}_{\tilde{w}} &= \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_u \times n_x} \end{bmatrix} - \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_x} \end{bmatrix}, \\
\tilde{\mathcal{D}}_{\tilde{p}} &= - \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix}, \text{and } \tilde{\mathcal{D}} = \begin{bmatrix} \tilde{\mathcal{D}}_{\tilde{w}} & \tilde{\mathcal{D}}_{\tilde{p}} \end{bmatrix}.
\end{aligned} \tag{7.34}$$

Let $\bar{\mathbb{G}}$ be the set which enforces the constraints $(u(k), y(k)) \in (\mathbb{U} \times \mathbb{Y})$ on the nominal velocity state $\bar{\epsilon}(k)$ for all admissible disturbances, $d(k) \in \mathbb{D}$, i.e.,

$$\bar{\mathbb{G}} := \left\{ \bar{\epsilon} : (u, y) \in (\mathbb{U} \times \mathbb{Y}), \quad \forall d(k) \in \mathbb{D} \right\}. \tag{7.35}$$

Taking the difference between (7.33) and its nominal version i.e.,

$$\begin{bmatrix} \bar{x}(k) \\ \bar{u}(k-1) \end{bmatrix} = \tilde{\mathcal{C}} \begin{bmatrix} \bar{\epsilon}(k) \\ r \end{bmatrix} \tag{7.36}$$

we obtain the following relation,

$$\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \tilde{\mathcal{C}} \begin{bmatrix} \bar{\epsilon}(k) \\ \bar{r} \end{bmatrix} + \tilde{\mathcal{C}} \varepsilon(k) + \tilde{\mathcal{D}} q(k-1). \tag{7.37}$$

From the invariance of $\tilde{\mathbb{E}}$, $\varepsilon(k) \in \tilde{\mathbb{E}}$ for all $k \geq 0$. Also, $q(k-1) \in (\tilde{\mathbb{W}} \times \tilde{\mathbb{P}})$ and $\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} \in (\mathbb{X} \times \mathbb{U})$, therefore the following set relationship is obtainable from (7.37):

$$\tilde{\mathcal{C}} \begin{bmatrix} \bar{\epsilon} \\ r \end{bmatrix} \in (\mathbb{X} \times \mathbb{U}) \ominus \tilde{\mathcal{D}}(\tilde{\mathbb{W}} \times \tilde{\mathbb{P}}) \ominus \tilde{\mathcal{C}} \tilde{\mathbb{E}}. \tag{7.38}$$

The set $\bar{\mathbb{G}}$ can therefore be defined in terms of $\bar{\epsilon}$ as,

$$\bar{\mathbb{G}} := \left\{ \bar{\epsilon} : \tilde{\mathcal{C}} \begin{bmatrix} \bar{\epsilon} \\ r \end{bmatrix} \in (\mathbb{X} \times \mathbb{U}) \ominus \tilde{\mathcal{D}}(\tilde{\mathbb{W}} \times \tilde{\mathbb{P}}) \ominus \tilde{\mathcal{C}} \tilde{\mathbb{E}} \right\}. \tag{7.39}$$

To ensure that the system inputs and outputs $u(k)$ and $y(k)$, respectively, lie in the sets $\mathbb{U}$ and $\mathbb{Y}$, beyond the end of the prediction horizon, N, we need to compute a terminal invariant set, $\bar{\mathbb{G}}_f$, where $\bar{\epsilon}(k+N)$ must lie in order to guarantee that the constraints $(u(k+N), y(k+N)) \in (\mathbb{U} \times \mathbb{Y})$ are verified for all k.
That is, for any $\bar{\epsilon}(k+N) \in \bar{\mathbb{G}}_f$, it should be that $\mathcal{F}\bar{\epsilon}(k+N) \in \bar{\mathbb{G}}_f \subseteq \bar{\mathbb{G}}$ where $\mathcal{F} = \mathcal{A} - \mathcal{B}\mathcal{K}$. Assume the pair $(\mathcal{A}, \mathcal{B})$ is reachable and the set $\bar{\mathbb{G}}$ is a closed polytope (both true if Assumption 6 are satisfied), then the set $\bar{\mathbb{G}}_f$ can be computed as the (projection of the) maximal constraint admissible set [114] for the nominal velocity dynamics (7.23a) under the auxiliary control law, $\delta\bar{u}(k) = -\mathcal{K}\bar{\epsilon}(k)$ and is constructed such that:

$$\bar{\epsilon} \in \bar{\mathbb{G}}_f \implies \bar{\epsilon} \in \bar{\mathbb{G}} \text{ and } \mathcal{F}\bar{\epsilon} \in \bar{\mathbb{G}}_f. \tag{7.40}$$

Optimal control problem

Given the velocity dynamics (7.23), the sets $\bar{\mathbb{G}}$ and $\bar{\mathbb{G}}_f$, and the current velocity state $\epsilon(k)$ obtained from the most recent input-output and state measurements,

$$\epsilon(k) = \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} = \begin{bmatrix} x(k) - x(k-1) \\ \Lambda_y y(k-1) + \Lambda_u u(k-1) + r \end{bmatrix}, \tag{7.41}$$

the feedback optimizing model predictive control problem under bounded disturbance is defined at each time instant k as:

$$\min_{\bar{\epsilon}(k), \delta\bar{\mathbf{u}}(k)} V_N(\bar{\epsilon}(k), \delta\bar{\mathbf{u}}(k)) = V_f(\bar{\epsilon}(k+N)) + \sum_{i=0}^{N-1} l(\bar{e}(k+i), \delta\bar{u}(k+i)) \tag{7.42}$$

subject to:
$\forall i \in \mathbb{I}_{[0,N-1]}$,

$$\bar{\epsilon}(k+i+1) = \mathcal{A}\bar{\epsilon}(k+i) + \mathcal{B}\delta\bar{u}(k+i), \tag{7.43a}$$

$$\bar{e}(k+i) = \mathcal{C}\bar{\epsilon}(k+i) + \mathcal{D}\delta\bar{u}(k+i), \tag{7.43b}$$

$$\epsilon(k) - \bar{\epsilon}(k) \in \mathbb{E}, \tag{7.43c}$$

$$\bar{\epsilon}(k+i) \in \bar{\mathbb{G}}, \tag{7.43d}$$

and,

$$\bar{\epsilon}(k+N) \in \bar{\mathbb{G}}_f. \tag{7.44a}$$

In this problem, the decision variable is the sequence of nominal control increments over the N-step prediction horizon:

$$\delta\bar{\mathbf{u}}(k) := \Big\{\delta\bar{u}(k), \delta\bar{u}(k+1)\ldots, \delta\bar{u}(k+N-1)\Big\} \tag{7.45}$$

and the nominal velocity state at each time instant $\bar{\epsilon}(k)$. These sequences are chosen to minimize the objective $V_N(\bar{\epsilon}(k), \delta\bar{\mathbf{u}}(k))$, which is composed of a stage cost

$$l(\bar{e}(k), \delta\bar{u}(k)) := \frac{1}{2}\Big(\bar{e}(k)^\top Q_e \bar{e}(k) + \delta\bar{u}(k)^\top R_\delta \delta\bar{u}(k)\Big) \tag{7.46}$$

and a terminal cost

$$V_f(\bar{\epsilon}(k+N)) := \frac{1}{2}\bar{\epsilon}(k+N)^\top \mathcal{P}\bar{\epsilon}(k+N). \tag{7.47}$$

This cost aims to capture the transient performance objective for the system: the term $\bar{e}(k)^\top Q_e \bar{e}(k)$ penalizes the distance of the tracking error from steady-state optimality and therefore determines the duration of the transient phase, while the term $\delta\bar{u}(k)^\top R_\delta \delta\bar{u}(k)$ penalizes the incremental control effort. It is simple to verify that Assumption 8 ensures positive definiteness of this cost. The terminal cost is employed, in the usual way, towards guaranteeing stability where the matrix $\mathcal{P}$ satisfies (6.20). Finally, the set $\mathbb{E}$ is a RPI set on $\epsilon - \bar{\epsilon}$ derived from the set $\tilde{\mathbb{E}}$ as:

$$\mathbb{E} := \Big\{\epsilon - \bar{\epsilon} \colon \varepsilon \in \tilde{\mathbb{E}}\Big\}. \tag{7.48}$$

The solution $\Big(\delta\bar{\mathbf{u}}^*(k), \bar{\epsilon}^*(k)\Big)$ of the optimal control problem presented above, followed by the application of the first control, $\delta\bar{u}^*(k)$ in the optimized sequence, yields the control law

$$\delta u(k) = \delta\bar{u}^*(k) - \mathcal{K}\Big(\epsilon(k) - \bar{\epsilon}^*(k)\Big). \tag{7.49}$$

Main result

The following result summarizes the stability and recursive feasibility of the FOMPC algorithm, and follows directly from well established results on conventional linear MPC [181].

Theorem 7.2.1 (Robust Stability and feasibility)**.** *Let the assumptions 6, 14 and 7 be verified, the design parameters* $Q_e, R_\delta, \mathcal{P}, \mathcal{K}, \mathbb{E}, \bar{\mathbb{G}}$ *and* $\bar{\mathbb{G}}_f$ *chosen as stipulated above, and the uncertainties of the type specified in* (7.1c)*. Then if at time* $k = 0$*, a feasible solution to the optimization problem* (7.42)-(7.44a) *exists, the resulting MPC control law asymptotically steers the nominal error,* $\bar{e}(k)$ *in* (7.23b) *(not the actual error,* (7.13)*) to zero, and the actual inputs and outputs* $(u(k), y(k))$ *are driven to an invariant neighbourhood of the optimal steady-state* $z^*(\bar{w})$*.*
Moreover, $\delta\bar{u}(k) \to 0$ *as* $k \to \infty$*, and the constraints* $(u(k), y(k)) \in \mathbb{U} \times \mathbb{Y}$ *are fulfilled for all* $k > 0$*.*

Proof. To prove the theorem, we follow the standard robust tube-based MPC arguments in [154].
Feasibility: Consider that, at time k, a solution to the OCP (minimize (7.42) subject to (7.43) and (7.44)) is $(\bar{\epsilon}^*(k), \delta\bar{\mathbf{u}}^*(k))$. Then a feasible but not necessarily optimal solution to the OCP at time $k+1$ would be $(\bar{\epsilon}(k+1), \delta\bar{\mathbf{u}}(k+1))$ where

$$\delta\bar{\mathbf{u}}(k+1) := \Big\{\delta\bar{u}^*(k+1), \dots, \delta\bar{u}^*(k+N-1), \delta\bar{u}(k+N)\Big\} \tag{7.50}$$

with $\delta\bar{u}(k+N) = -\mathcal{K}\bar{\epsilon}(k+N)$ and $\Big\{\delta\bar{u}^*(k+1), \dots, \delta\bar{u}^*(k+N-1)\Big\}$ derived from the tail of $\delta\bar{\mathbf{u}}^*(k)$.
Convergence: Consider the above mentioned feasible solution at time $k+1$. Following standard arguments for the proof of the convergence of robust tube-based MPC [154] we obtain that

$$V_N(\bar{\epsilon}(k+1), \delta\bar{\mathbf{u}}^*(k+1)) - V_N(\bar{\epsilon}(k), \delta\bar{\mathbf{u}}^*(k)) \leq -l(\bar{e}(k), \delta\bar{u}^*(k)). \tag{7.51}$$

In view of the positive definiteness of $V_N(\cdot,\cdot)$ and of Q_e and R_δ, it holds that $\bar{\epsilon}(k) \to 0$ and $\delta\bar{u}(k) \to 0$ as $k \to \infty$. □

This result guarantee that for any feasible initial input/output pair i.e $(u(k), y(k)) \in \mathbb{U} \times \mathbb{Y}$, the FOMPC control law $u(k) = u(k-1) + \delta\bar{u}^*(k) - \mathcal{K}\Big(\epsilon(k) - \bar{\epsilon}^*(k)\Big)$ steers the input and output of the real system (7.1) to a neighbourhood of the optimal solution

of Problem 6, for all $w(k) \in \mathbb{W}$ and $\delta \in \Delta_l$.
A major challenge with the tube-based robust FOMPC algorithm above is that the computation of the sets $\bar{\mathbb{G}}$ and $\tilde{\mathbb{E}}$ are intertwined as they both require the set $\mathbb{X} \times \mathbb{U}$ i.e to compute $\tilde{\mathbb{E}}$ requires the set $\mathbb{D}$ which is defined based on the set $\tilde{\mathbb{W}}$ which in turn depends on the sets $\mathbb{X}$ and $\mathbb{U}$. This creates an interdependency problem between the nominal FOMPC problem solved online and the offline computation of the RPI set $\tilde{\mathbb{E}}$, thereby complicating the design of the above FOMPC algorithm. We will not discuss these complications here (see [95] for details on possible workarounds to this problem). Furthermore, if the set $\mathbb{D}$ is large, the tightened constraint set $\bar{\mathbb{G}}$ may be too small thereby limiting the region of attraction of the FOMPC controller resulting in very conservative performance. This conservatism is reduced in the absence of model uncertainty as the set $\mathbb{D}$ only depends on the additive disturbance bound, $\mathbb{W}$. See the remark below.

Remark 41. *In the absence of model uncertainty, i.e.* $\Delta_l = \emptyset$*, then*

$$\Omega := \ Co\Big((A_1, B_1), \ldots, (A_l, B_l)\Big) = (A, B) \tag{7.52}$$

and

$$\Omega_\Lambda := \ Co\Big((\Lambda_{y1}C, \Lambda_{u1}), \ldots, (\Lambda_{yl}C, \Lambda_{ul})\Big) = (\Lambda_y C, \Lambda_u) \tag{7.53}$$

resulting in the following disturbance sets

$$\tilde{\mathbb{W}} = E\mathbb{W} \text{ and } \tilde{\mathbb{P}} = \emptyset. \tag{7.54}$$

7.3 Robust FOMPC Design : A Min-max Approach

In the the previous section, a robust FOMPC control algorithm based on tube-based MPC was presented. Although very systematic, this approach has several limitations:

1. It relies on the heavy offline computation of a robust positive invariant (RPI) set, making it more complicated to implement and limiting its application to systems of lower dimensions.

2. It can result in very conservative performance if the RPI set is not small. Indeed lumping parametric uncertainty into an additive disturbance is a conservative way of dealing with the uncertainty.

3. It guarantees robust asymptotic convergence to an RPI set with the optimal steady-state equilibrium as centre. Although for non-persistent uncertainty, asymptotic convergence to the steady-state optimum is achievable.

We address these issues by formulating a robust FOMPC controller using an LMI approach. The proposed FOMPC controller minimizes the worst-case performance cost against the polytopic model uncertainties and under piecewise constant disturbances. The proposed algorithm also guarantees the recursive feasibility of the input and output constraints.

7.3.1 Problem formulation

Consider the following discrete-time velocity dynamics for the polytopic uncertain system (7.1), the steady-state optimization (5.12) and under piecewise constant disturbances:

$$\epsilon(k+1) = \mathcal{A}(\delta)\epsilon(k) + \mathcal{B}(\delta)\delta u(k), \tag{7.55a}$$

$$e(k) = \mathcal{C}(\delta)\epsilon(k) + \mathcal{D}(\delta)\delta u(k), \tag{7.55b}$$

$$\delta \in \Delta_l. \tag{7.55c}$$

All variables and parameters retain their previous meaning from Section 5.4.

Assumption 15. *The system in* (7.55) *above is assumed to be quadratically stabilizable for all* $\delta \in \Delta_l$ *according to Definition 4.*

Let the system in (7.1) be subject to the following ellipsoidal constraints on the inputs and outputs for all $k \geq 0$:

$$(u(k), y(k)) \in \mathscr{E} := \Big\{(u, y) \ : \|u(k)\|_2 \leq \bar{u}, \ \|y(k+1)\|_2 \leq \bar{y}\Big\}. \tag{7.56}$$

Using the triangle inequality, we can translate the constraints (7.56) above into following constraints in velocity form:

$$\|\delta u(k)\|_2 \leq \bar{u} - \|u(k-1)\|_2, \ \forall k \geq 0, \tag{7.57a}$$

$$\|CC_x\epsilon(k+1)\|_2 \leq \bar{y} - \|y(k)\|_2, \ \forall k \geq 0 \tag{7.57b}$$

where $C_x = \begin{bmatrix} I_{n_x} & \mathbf{0} \end{bmatrix}$. Given the velocity dynamics (7.55) and the constraints (7.57), the robust FOMPC problem is defined for a state $\epsilon(k)$ as

$$\min_{\delta u(k+i),\ i>0} \left(\check{J}(\epsilon(k)) = \max_{\delta \in \Delta_l} J_\infty(\epsilon(k)) \right) \tag{7.58}$$

subject to:
$\forall i \geq 0$,

$$\epsilon(k+i+1) = \mathcal{A}\epsilon(k+i) + \mathcal{B}\delta u(k+i), \tag{7.59a}$$

$$e(k+i) = \mathcal{C}\epsilon(k+i) + \mathcal{D}\delta u(k+i), \tag{7.59b}$$

$$\|\delta u(k+i)\|_2 \leq \bar{u} - \|u(k+i-1)\|_2, \tag{7.59c}$$

$$\|CC_x\epsilon(k+i+1)\|_2 \leq \bar{y} - \|y(k+i)\|_2. \tag{7.59d}$$

In this problem, $\epsilon(k+i)$ is the predicted velocity state at sample time $k+i$. The decision variable is the control action at sample time $k+i$, $\delta u(k+i)$. The control action is chosen to minimize the worst case performance index $\check{J}(\epsilon(k))$ which is given by (5.21), subject to the relevant constraints.
Similarly to the robust FOLQC, in order to obtain a more computationally tractable problem, we instead minimize the upper bound on the worst-case cost $\check{J}(\epsilon(k))$. This upper bound has been derived in Chapter 5 as

$$\check{J}(\epsilon(k)) \leq V(\epsilon(k)) \tag{7.60}$$

where

$$V(\epsilon(k)) = \frac{1}{2}\epsilon(k)^\top \mathcal{P}\epsilon(k),\ \mathcal{P} \succ 0 \tag{7.61}$$

and satisfies the inequality

$$V(\epsilon(k+1)) - V(\epsilon(k)) \leq -\frac{1}{2}\Big(e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big). \tag{7.62}$$

The robust FOMPC problem ((7.58)−(7.59)) can therefore be recast as the optimization problem

$$\min_{\delta u(k+i),\ i>0} V(\epsilon(k)) \tag{7.63}$$

subject to:
$\forall i \geq 0,$

$$V(\epsilon(k+i+1)) - V(\epsilon(k+i)) \leq -\frac{1}{2}\Big(e(k+i)^\top Q_e e(k+i) + \delta u(k+i)^\top R_\delta \delta u(k+i)\Big), \tag{7.64a}$$

$$\|\delta u(k+i)\|_2 \leq \bar{u} - \|u(k+i-1)\|_2, \tag{7.64b}$$

$$\|CC_x \epsilon(k+i+1)\|_2 \leq \bar{y} - \|y(k+i)\|_2. \tag{7.64c}$$

Although the robust FOMPC problem has been recast in a simpler and more tractable form, the optimization problem ((7.63)-(7.64)) is still not computationally tractable as the decision variable $\delta u(k+i),\ i > 0$ is infinite dimensional. To address this challenge, similar to [116], we parametrize the input $\delta u(k+i)$ in terms of the velocity state $\epsilon(k+i)$ via the feedback control law:

$$\delta u(k+i) = -\mathcal{K}\epsilon(k+i). \tag{7.65}$$

Although this control parametrization makes the FOMPC control problem tractable, it introduces some degree of conservatism. Efforts to reduce this conservatism will not be made in this work but deferred to future research. We now present our result for solving the robust FOMPC problem above with the following theorem.

Theorem 7.3.1 (Robust FOMPC using LMI)**.** *Let the current velocity state,* $\epsilon(k+i)$*, the previous control input* $u(k+i-1)$ *and the current output* $y(k+i)$ *of the system* (7.1) *be available at sample time* $k+i$*. Then, the receding horizon state feedback matrix* $\mathcal{K}$ *which at sample time* $k+i$ *is the solution to the robust FOMPC problem (*(7.63)-(7.64)*) with the control law* (7.65) *is given by*

$$\mathcal{K} = \mathcal{W}\mathcal{Y}^{-1}$$

where $\mathcal{Y} \succ 0$ *and* $\mathcal{W}$ *are the solutions (if they exists) to the following semi-definite program:*

$$\min_{\gamma, \mathcal{W}, 0 \prec \mathcal{Y} = \mathcal{Y}^\top} \quad \gamma \tag{7.66}$$

subject to :

$$\begin{bmatrix} 1 & \epsilon(k+i)^\top \\ \epsilon(k+i) & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{7.67a}$$

$$\begin{bmatrix} -\mathcal{Y} & \Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)^\top & \Big(\mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W}\Big)^\top & \mathcal{W}^\top \\ \mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W} & -\mathcal{Y} & \mathbf{0} & \mathbf{0} \\ \mathcal{C}(\delta)\mathcal{Y} - \mathcal{D}(\delta)\mathcal{W} & \mathbf{0} & -\gamma Q_e^{-1} & \mathbf{0} \\ \mathcal{W} & \mathbf{0} & \mathbf{0} & -\gamma R_\delta^{-1} \end{bmatrix} \preceq \mathbf{0}, \ \forall \delta \in \Delta_l, \tag{7.67b}$$

$$\begin{bmatrix} \Big(\bar{u} - \|u(k+i-1)\|_2\Big)^2 I_{n_u} & \mathcal{W} \\ \mathcal{W}^\top & \mathcal{Y} \end{bmatrix} \succ \mathbf{0}, \tag{7.67c}$$

$$\begin{bmatrix} \mathcal{Y} & \Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)^\top C_x^\top C^\top \\ CC_x\Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big) & \Big(\bar{y} - \|y(k+i)\|_2\Big)^2 I_{n_y} \end{bmatrix} \succ \mathbf{0}, \ \forall \delta \in \Delta_l. \tag{7.67d}$$

Also the resulting FOMPC control law $u(k+i) = u(k+i-1) - \mathcal{K}\epsilon(k+i)$ *under constant (or slowly varying) disturbances steers the input and output of the real system* (7.1) *to the optimal solution of Problem 6, for all* $w(k) \in \mathbb{W}$ *and* $\delta \in \Delta_l$ *with* (7.61) *a corresponding Lyapunov function guaranteeing the closed loop stability (i.e. convergence to the steady-state optimum) of the uncertain system.*

Proof. To minimize the upper bound for the robust performance index $\check{J}(\epsilon(k+i))$, we solve the following problem at sample time $k+i$:

$$\min_{\gamma} \gamma \quad \text{subject to:} \quad V(\epsilon(k+i)) < \gamma, \tag{7.68}$$

which is equivalent to the problem,

$$\min_{\gamma} \gamma \quad \text{subject to:} \quad \epsilon(k+i)^\top \mathcal{Y}^{-1} \epsilon(k+i) < 1 \tag{7.69}$$

where $\mathcal{Y} = \gamma\mathcal{P}^{-1}$. Using the Schur complement, $\epsilon(k+i)^\top\mathcal{Y}^{-1}\epsilon(k+i) < 1$ can be written as (7.67a).
To obtain (7.67b), we substitute $\delta u(k+i) = -\mathcal{K}\epsilon(k+i)$, $e(k+i) = \Big(\mathcal{C}(\delta)-\mathcal{D}(\delta)\mathcal{K}\Big)\epsilon(k+i)$ and $\epsilon(k+i+1) = \Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{K}\Big)\epsilon(k+i)$ in (7.64a) to obtain:

$$\Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{K}\Big)^\top\mathcal{P}\Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{K}\Big) - \mathcal{P} + \Big(\mathcal{C}(\delta) - \mathcal{D}(\delta)\mathcal{K}\Big)^\top Q_e\Big(\mathcal{C}(\delta) - \mathcal{D}(\delta)\mathcal{K}\Big) + \mathcal{K}^\top R_\delta\mathcal{K} \preceq 0. \tag{7.70}$$

Therefore if the inequality (7.70) is satisfied, the closed loop system is stable. By Schur's lemma, substitution of

$$\mathcal{P} = \gamma\mathcal{Y}^{-1} \text{ and } \mathcal{W} = \mathcal{K}\mathcal{Y}, \tag{7.71}$$

and considering the uncertainty δ is affine in the coefficient matrices, the inequality (7.70) can be rewritten as (7.67b).
Regarding the input and output constraints. Observe that if conditions (7.64a) and (7.67a) are satisfied then:

$$\epsilon(k+i)^\top\mathcal{Y}^{-1}\epsilon(k+i) < 1,\ \forall i \geq 1, \tag{7.72}$$

that is, the ellipsoid $\mathscr{E}_{inv} := \Big\{z \mid z^\top\mathcal{Y}^{-1}z < 1\Big\}$ is an invariant ellipsoid for the predicted values of the state $\epsilon(k+i)$. Then we may, at any instant $k+i$, impose the constraints (7.57a) on all future controls. It therefore follows that:

$$\begin{aligned}\max_{i\geq 0}\|\delta u(k+i)\|_2^2 = \max_{i\geq 0}\|-\mathcal{W}\mathcal{Y}^{-1}\epsilon(k+i)\|_2^2 &\leq \max_{z\in\mathscr{E}}\|-\mathcal{W}\mathcal{Y}^{-1}z\|_2^2 \\ &\leq \lambda_{max}\Big(\mathcal{Y}^{-\frac{1}{2}}\mathcal{W}^\top\mathcal{W}\mathcal{Y}^{-\frac{1}{2}}\Big).\end{aligned} \tag{7.73}$$

Therefore the input constraints (7.57a) can be written as :

$$\lambda_{max}\Big(\mathcal{Y}^{-\frac{1}{2}}\mathcal{W}^\top\mathcal{W}\mathcal{Y}^{-\frac{1}{2}}\Big) \leq \Big(\bar{u} - \|u(k+i-1)\|_2\Big)^2 \tag{7.74}$$

Using the Schur complement, we can express (7.74) as the LMI (7.67c).
Regarding the output, at any instant $k+i$, we impose the constraints (7.57b) on all

future outputs. From the output constraints we have

$$\max_{i>0} \Big(\|CC_x\epsilon(k+i+1)\|_2\Big)^2 \leq \Big(\bar{y} - \|y(k+i)\|_2\Big)^2. \tag{7.75}$$

Substituting $\epsilon(k+i+1) = \Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{K}\Big)\epsilon(k+i)$ and $\mathcal{K} = \mathcal{W}\mathcal{Y}^{-1}$, we obtain

$$\begin{aligned} \max_{i>0}\|CC_x\epsilon(k+i+1)\|_2^2 &= \max_{i>0}\|CC_x\Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{W}\mathcal{Y}^{-1}\Big)\epsilon(k+i)\|_2^2 \\ &\leq \max_{z>\mathscr{E}}\|CC_x\Big(\mathcal{A}(\delta) - \mathcal{B}(\delta)\mathcal{W}\mathcal{Y}^{-1}\Big)z\|_2^2 \\ \leq \lambda_{max}\Big(\mathcal{Y}^{-\frac{1}{2}}\Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)^\top & C_x^\top C^\top CC_x\Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)\mathcal{Y}^{-\frac{1}{2}}\Big). \end{aligned} \tag{7.76}$$

We can then express the output constraint (7.57b) as the inequality

$$\begin{aligned} \lambda_{max}\Big(\mathcal{Y}^{-\frac{1}{2}}\Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)^\top C_x^\top C^\top CC_x\Big(\mathcal{A}(\delta)\mathcal{Y} - \mathcal{B}(\delta)\mathcal{W}\Big)\mathcal{Y}^{-\frac{1}{2}}\Big) \\ \leq \Big(\bar{y} - \|y(k+i)\|_2\Big)^2. \end{aligned} \tag{7.77}$$

By the Schur complement and because the uncertainty is affine in the coefficient matrices, the inequality (7.77) can be expressed as the LMI (7.67d). □

7.4 Illustrative Example

In this section, numerical simulation and comparative studies of robust feedback optimizing model predictive control formulations will be presented. We study the performance of nominal FOMPC, tube-based robust FOMPC and a min-max robust FOMPC formulation. The following uncertain linear system (same as Chapter 5) will be used in this study.

$$x(k+1) = A(\delta)x(k) + B(\delta)u(k) + Ew(k), \tag{7.78a}$$

$$y(k) = Cx(k), \tag{7.78b}$$

where

$$A(\delta) = \begin{bmatrix} 2 & 0.1 \\ 0 & 1 - 0.1\delta \end{bmatrix}, \; B(\delta) = \begin{bmatrix} 0 \\ 0.1\delta \end{bmatrix}, \; E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and } C = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{7.79}$$

with $\delta \in \Delta_l \coloneqq \{\delta \mid \delta \in [5, 10]\}$.

We assume the system is quadratically stabilizable and the disturbance $w(k)$ is unknown and constant with the value

$$w(t) = \left\{ \begin{bmatrix} -1 & 3 \end{bmatrix}^\top, \quad 0 \leq k < 500. \right. \tag{7.80}$$

The following inequality constraints are present on the input, output and disturbances;

$$\begin{aligned} \mathbb{U} &:= \{u\colon -50 \leq u \leq 50\}, \\ \mathbb{Y} &:= \{y\colon -50 \leq y \leq 50\}, \\ \mathbb{W} &:= \{w\colon -20 \times I_2 \leq w \leq 20 \times I_2\}. \end{aligned} \tag{7.81}$$

In a steady-state, the system (7.78) satisfies the following equation

$$0 = A(\delta)x + B(\delta)u + Ew, \tag{7.82a}$$

$$y = Cx. \tag{7.82b}$$

From (7.82) the steady-state input-output map for (7.78) is computed as

$$y = h(u, w) = G_u u + G_w w \tag{7.83}$$

where

$$\begin{aligned} G_u(\delta) &= C(I_n - A(\delta))^{-1}B(\delta) = -0.1, \\ G_w(\delta) &= C(I_n - A(\delta))^{-1}E = \begin{bmatrix} -1 & -\delta^{-1} \end{bmatrix}. \end{aligned} \tag{7.84}$$

Note that the system has the robust steady-state gain property as $G_u(\delta)$ is independent of δ. We define the steady-state optimization problem of interest as

$$\begin{aligned} \bar{z}^*(\delta, w) &= \arg\min_{\bar{z}} \Phi(\bar{z}) \\ \text{subject to:} \quad & G_z \bar{z} = d(\delta, w), \end{aligned} \tag{7.85}$$

where

$$z = \begin{bmatrix} u \\ y \end{bmatrix}, \; \Phi(\bar{z}) = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z}, \; Q_{zz} = \begin{bmatrix} Q_{uu} & Q_{uy} \\ Q_{yu} & Q_{yy} \end{bmatrix}, \; R_z = \begin{bmatrix} R_u \\ R_y \end{bmatrix}, \; Q_{uu} = 5,$$
$$Q_{uy} = Q_{yu} = 2, \; Q_{yy} = 1, \; R_u = 0.1, \; R_y = 0.5, G_z = \begin{bmatrix} -G_u(\delta) & I_p \end{bmatrix}, \; d(\delta, w) = G_w(\delta)w. \tag{7.86}$$

To regulate the uncertain system (7.78) to steady-state equilibria that are the solution of problem (7.85), we adopt three approaches in this example:

1. A nominal FOMPC designed based on a nominal value, $\bar{\delta} = 7$ of the uncertain parameter δ and relying on the inherent robustness of FOMPC to deal with the uncertainty.

2. A robust tube-based FOMPC designed based on a nominal value, $\bar{\delta} = 7$, of the uncertain parameter δ and relying on an RPI set to deal with the uncertainty.

3. A robust min-max FOMPC designed using a knowledge of the uncertainty set $\delta \in [5, 10]$, and the piecewise-constant nature of the disturbance i.e., $w(k) = w(k-1)$.

7.4.1 Nominal FOMPC

The design of the nominal FOMPC is based on procedure outlined in Chapter 6. The uncertain parameter δ is chosen as $\delta = \bar{\delta} = 7$. The transient performance criterion has the following weights: $Q_e = 1$ and $R_\delta = 5$; these values satisfy the hypothesis of Proposition 3.

From the design parameters above, the nominal FOMPC has the following parameters:

$$\begin{aligned}
\mathcal{A} &= \begin{bmatrix} 2 & 0.1 & 0 \\ 0 & 0.3 & 0 \\ -15 & 0 & 1.00 \end{bmatrix}, \ \mathcal{B} = \begin{bmatrix} 0 \\ 0.7 \\ -48 \end{bmatrix}, \\
\mathcal{C} &= \begin{bmatrix} -15 & 0 & 1.00 \end{bmatrix}, \ \mathcal{D} = -48, \\
\mathcal{R} &= 2309, \ \mathcal{N} = \begin{bmatrix} 720 \\ 0 \\ -48 \end{bmatrix} \times 10^3, \\
\mathcal{Q} &= 10^4 \times \begin{bmatrix} 225 & 0 & -15 \\ 0 & 0 & 0 \\ -15 & 0 & 1 \end{bmatrix}, \text{ and} \\
\mathcal{K} &= \begin{bmatrix} 72.2239 & 4.2626 & 0.0105 \end{bmatrix}.
\end{aligned} \tag{7.87}$$

From the inequality $\mathbb{X} \times \mathbb{U}$ given by,

$$\mathbb{X} \times \mathbb{U} := \left\{ \begin{bmatrix} x \\ u \end{bmatrix} : \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq \begin{bmatrix} 50 \\ 50 \\ 50 \\ 50 \end{bmatrix} \right\} \tag{7.88}$$

we compute the constraint $\mathbb{G}$ according to (6.12) as:

$$\mathbb{G} := \left\{ \epsilon \colon \begin{bmatrix} 2.0323 & 0.129 & 0.0022 \\ -2.0323 & -0.129 & -0.0022 \\ -0.3226 & -0.0403 & -0.0215 \\ 0.3226 & 0.0403 & 0.0215 \end{bmatrix} \epsilon \leq \begin{bmatrix} 49.9892 \\ 50.0108 \\ 50.1075 \\ 49.8925 \end{bmatrix} \right\}. \tag{7.89}$$

From the set $\mathbb{G}$, we compute the maximum output admissible invariant set for the velocity dynamics as:

$$\mathbb{G}_f := \left\{ \epsilon \colon \begin{bmatrix} -72.8377 & -4.3241 & -0.032 \\ 72.8377 & 4.3241 & 0.032 \\ 2.0323 & 0.1290 & 0.0022 \\ -2.0323 & -0.1290 & -0.0022 \\ -0.3226 & -0.0403 & -0.0215 \\ 0.3226 & 0.0403 & 0.0215 \end{bmatrix} \epsilon \leq \begin{bmatrix} 50.1075 \\ 49.8925 \\ 49.9892 \\ 50.0108 \\ 50.1075 \\ 49.8925 \end{bmatrix} \right\}. \tag{7.90}$$

7.4.2 Tube-based robust FOMPC

For the tube based FOMPC controller, the nominal value of $\delta = \bar{\delta} = 7$ is used to define the nominal velocity dynamics, which happens to be the same nominal dynamics used in the preceding subsection. The lumped disturbance set $\tilde{\mathbb{W}}$ is computed as

$$\tilde{\mathbb{W}} := \left\{ \tilde{w} \colon \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \tilde{w} \leq \begin{bmatrix} 20 \\ 20 \\ 0 \end{bmatrix} \right\}. \tag{7.91}$$

Also the set $\delta\mathbb{W}$ is computed as

$$\delta\tilde{\mathbb{W}} := \left\{ \delta\tilde{w} \colon \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix} \delta\tilde{w} \leq \begin{bmatrix} 40 \\ 40 \end{bmatrix} \right\}. \tag{7.92}$$

Because the system has a robust steady-state gain, the uncertainty in A and B do not affect the steady-state error and therefore

$$\tilde{\mathbb{P}} = \emptyset, \ \ \delta\tilde{\mathbb{P}} = \mathbb{H} = \emptyset. \tag{7.93}$$

The bound on the disturbance affecting the velocity dynamics is therefore

$$\mathbb{D} := \left\{ d\colon \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} d \leq \begin{bmatrix} 40 \\ 40 \end{bmatrix} \right\}. \tag{7.94}$$

For this problem, the inequality constraints were not sufficient to compute the minimal RPI set $\tilde{\mathbb{E}}$. Therefore, the tube-based robust FOMPC could not be applied to this example. Redefining the inequality constraints could be a possible way to resolve this issue. However, this is left to future investigation.

7.4.3 Min-max robust FOMPC

To compute the robust FOMPC law based on the min-max approach described in Section 7.3, the following ellipsoidal constraints are imposed on the system,

$$\begin{aligned} \mathbb{U} &:= \{u\colon \|u\| \leq 50\} \\ \mathbb{Y} &:= \{y\colon \|y\| \leq 50\} \end{aligned} \tag{7.95}$$

Remark 42. *The ellipsoids* (7.95) *are the maximum volume ellipsoids contained in the respective polytopic input and output inequalities* (7.81) *computed using the algorithm described in Appendix C of [207].*

For the robust FOMPC design, Λ_y, Λ_u and r follow from the choices of Φ and the input-output sensitivity, G_u as

$$\Lambda_y = -15, \ \Lambda_u = -48, \ \text{and } r = -5. \tag{7.96}$$

The transient performance criterion is chosen with $Q_e = 1$ and $R_\delta = 5$; these values satisfy the hypothesis of Proposition 3.
From the design parameters above, the FOLQC law is computed from the following

parameters:

$$\mathcal{A}(\delta) = \begin{bmatrix} 2 & 0.1 & 0 \\ 0 & 1-0.1\delta & 0 \\ -19 & 0 & 1 \end{bmatrix}, \ \mathcal{B}(\delta) = \begin{bmatrix} 0 \\ 0.1\delta \\ -48 \end{bmatrix}, \ \mathcal{C}(\delta) = \begin{bmatrix} -19 & 0 & 1.00 \end{bmatrix}, \\ \mathcal{D}(\delta) = -48, \ \bar{y} = 50, \ \bar{u} = 50. \tag{7.97}$$

At sample time, k, the robust FOMPC law which minimizes the upper bound $V(\epsilon(0))$ on the robust performance objective (7.60) subject to the constraints (7.95) is given for all $\delta \in [5, 10]$ by

$$u(k) = u(k-1) - \check{\mathcal{K}}\epsilon(k) \tag{7.98}$$

where $\check{\mathcal{K}}$ is computed online at sample time k by solving the SDP (7.66).

7.4.4 Simulation results

For the simulation, the true value of the uncertain parameter is fixed at $\delta = 8$. The robust FOMPC law was computed using the extremal values of the uncertainty range i.e $\delta_1 = 5$ and $\delta_2 = 10$. The nominal FOMPC is implemented for the following nominal values of the uncertain parameter, $\bar{\delta} = \{7, 8, 9\}$.

Performance of nominal FOMPC:

Here, the performance of nominal FOMPC is discussed for the uncertain example system. The following nominal model values: $\bar{\delta} = \{5, 6, 7, 8\}$ are investigated. At $\bar{\delta} = 5, 6$, the FOMPC problem was infeasible for the uncertain system and therefore no control law could be computed. However at an improved nominal value of $\bar{\delta} = 7$, the nominal FOMPC can be seen from Figure 7.1 to stabilize the uncertain system while also tracking the optimal steady-state albeit with poor transient performance. This shows that the nominal FOMPC has both inherent robust stability and inherent robust convergence to the true steady-state optimum. Also, it can be observed from Figure 7.1 that the nominal FOMPC satisfies the inequality constraints on the input and output, keeping both bounded in the interval $[-50, 50]$. This is in contrast with the nominal FOLQC from Chapter 5 which violated the constraints on the input (see Figure 5.7).

To improve the dynamic performance of the nominal FOMPC, accurate knowledge of the uncertain model parameter δ is required. Figure 7.2 shows the performance

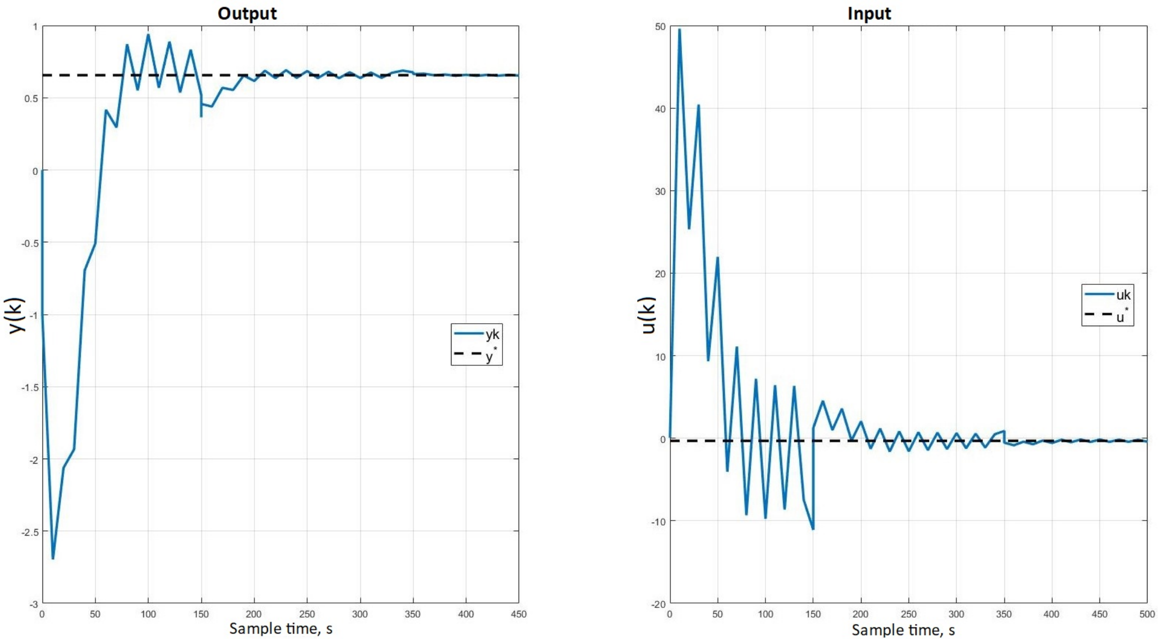

Fig. 7.1 Nominal FOMPC, $\bar{\delta} = 7$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

of nominal FOMPC designed using the true value of δ. It can be observed from the figure that with the nominal parameter set to the true value i.e., $\bar{\delta} = 8$, stability and convergence to the true optimal steady-state is achieved with much improved dynamic performance. Also, the constraints on the input and output are enforced as opposed to the case of nominal FOLQC in Figure 5.8 where the input can be seen to violate the constraint by exceeding $\bar{u} = 50$.

Performance of robust min-max FOMPC:

Although the nominal FOMPC has some inherent robustness to uncertainty, it does not guarantee the stability, optimal steady-state convergence and feasibility of the closed-loop system under model uncertainty. Unlike nominal FOMPC however, the robust FOMPC can guarantee the stability (and as a result, convergence to the true optimal steady-state) and feasibility of the closed-loop system as long as the uncertain parameter δ belongs to a defined set. We can see this from Figure 7.3. Here, the robust FOMPC is designed for all $\delta \in [5, 10]$, i.e., δ is assumed to take values between 5 and 10. It is clear from Figure 7.3 that the robust FOMPC is robustly stable, robustly convergent to the true steady-state optimum and enforces the inequality constraints on the input and output. The robust FOMPC guarantees the aforementioned properties

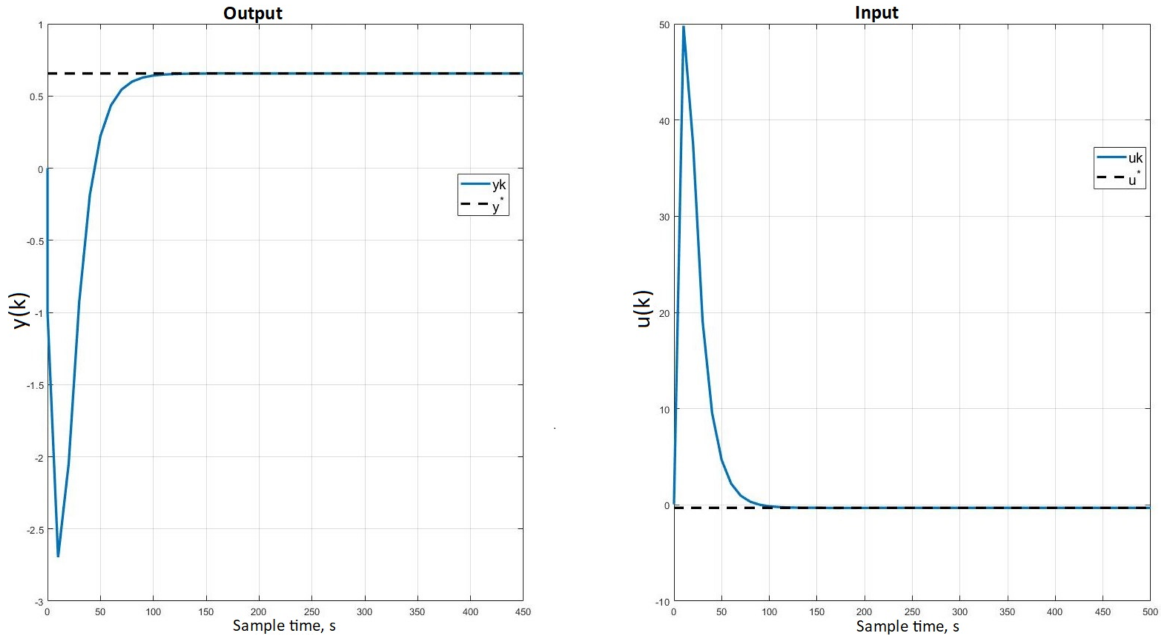

Fig. 7.2 Nominal FOMPC, $\bar{\delta} = 8$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

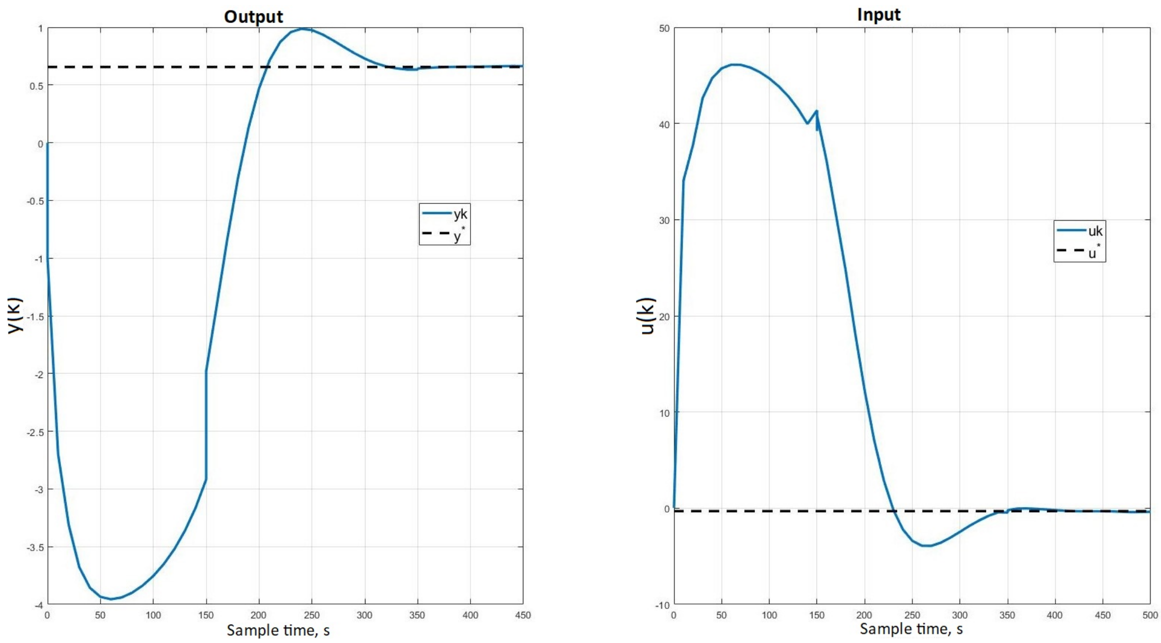

Fig. 7.3 Robust min-max FOMPC, $\delta \in [5, 10]$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

for all δ in the specified range i.e., $\delta \in [5, 10]$ while also ensuring a decent dynamic performance. Tightening this range will improve the dynamic performance albeit at the expense of robustness. Figure 7.4 shows the performance of the robust FOMPC under a much tighter uncertainty range of $\delta \in [6, 10]$. The performance improvements is apparent from the figure.

The constraint enforcing capability of robust FOMPC becomes apparent when the robust FOMPC performance in Figure 7.3 is compared with the robust FOLQC performance in Figure 5.12. It can be noticed here that while robust FOMPC satisfies the inequality constraint on the input, the robust FOLQC violates it.

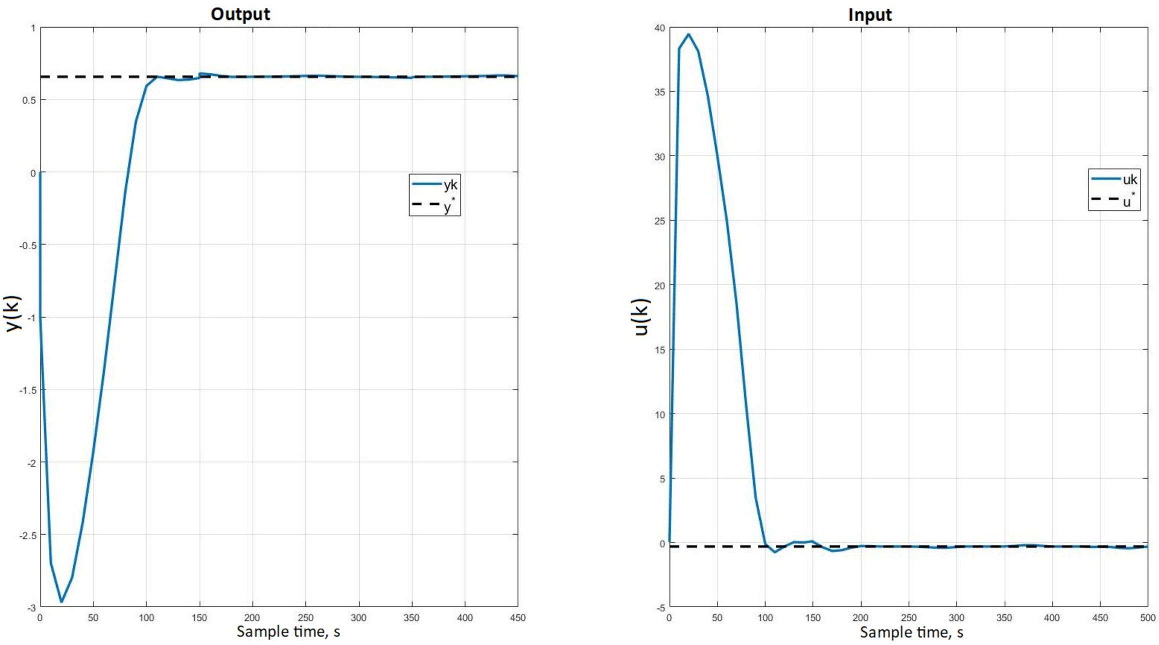

Fig. 7.4 Robust FOMPC, $\bar{\delta} \in [7, 10]$: outputs y and input u plotted as a function of time. The actual optimum (u^*, y^*) is shown using dashed lines.

7.5 Conclusion

In this chapter, we have presented robust FOMPC algorithms for the feedback optimization of uncertain linear systems with polytopic model uncertainty and a quadratic steady-state optimization problem. A linear matrix inequality (LMI) approach was used to formulate semi-definite programs which can be solved online to generate robust FOMPC laws. This approach requires the polytopic system to be quadratically stabilizable which limits its application to a small class of uncertain linear systems. Also a robust FOMPC algorithm based on the popular tube-based approach was presented. A major limitation of the tube MPC approach is that computing the minimal robust positive invariant set (mRPI) can be challenging for high dimensional systems. Also, where the disturbance is large, the corresponding mRPI sets can get bigger resulting in very conservative FOMPC algorithms which satisfy the inequality constraints for very limited range of the uncertainty. Future work will focus on developing less conservative and computationally more efficient robust FOMPC laws.

7.6 Appendix A: Proof of Equation (7.33)

From the lumped system dynamics (7.6),

$$x(k) = Ax(k-1) + Bu(k-1) + \tilde{w}(k-1), \tag{7.99}$$

we obtain

$$\delta x(k) = (A - I_{n_x})x(k-1) + Bu(k-1) + \tilde{w}(k-1). \tag{7.100}$$

But,

$$e(k-1) = \Lambda_y Cx(k-1) + \Lambda_u u(k-1) + r + \tilde{p}(k-1) \tag{7.101}$$

$$\therefore \quad \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} = S \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix} \tilde{w}(k-1) + \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix} r + \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix} \tilde{p}(k-1) \tag{7.102}$$

where $S = \begin{bmatrix} A - I_{n_x} & B \\ \Lambda_y C & \Lambda_u \end{bmatrix}$.

Making $\begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix}$ the subject of the above equation yields,

$$\begin{aligned} \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} = S^{-1} \Bigg(\begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix} - \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_x} \end{bmatrix} \tilde{w}(k-1) \\ - \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix} \tilde{p}(k-1) \Bigg). \end{aligned} \tag{7.103}$$

Also from the system dynamics, we can derive,

$$\begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} \begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_u \times n_x} \end{bmatrix} w(k-1). \tag{7.104}$$

$$\therefore \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} = \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x} & \mathbf{0}_{n_x \times n_y} & \mathbf{0}_{n_x \times n_y} \\ \mathbf{0}_{n_y \times n_x} & I_{n_y} & -I_{n_y} \end{bmatrix} \begin{bmatrix} \epsilon(k) \\ r \end{bmatrix}$$
$$+ \left(\begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_u \times n_x} \end{bmatrix} - \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} I_{n_x \times n_w} \\ \mathbf{0}_{n_y \times n_x} \end{bmatrix} \right) \tilde{w}(k-1) \tag{7.105}$$
$$- \begin{bmatrix} A & B \\ \mathbf{0}_{n_u \times n_x} & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} \mathbf{0}_{n_x \times n_y} \\ I_{n_y} \end{bmatrix} \tilde{p}(k-1).$$

We can therefore compactly express (7.105) as the equation (7.33)

Chapter 8

Distributed Feedback Optimizing Model Predictive Control

In this chapter, a distributed feedback optimization problem is formulated for large-scale constrained linear systems subject to external disturbances. The goal is to develop distributed feedback optimizing control laws that steer a collection of dynamically coupled linear systems to steady-states that are optimal for a defined optimization problem while guaranteeing feasibility and an optimal transient performance. We leverage on the results obtained for FOMPC in the previous chapters to formulate and solve this problem. The outline of this chapter is as follows: Section 8.1 gives an introduction to the control problem and a brief review of existing results. Section 8.2 presents the distributed FOMPC (DFOMPC) problem. In section 8.3 we propose solutions to the DFOMPC problem with theoretical analysis of convergence. Finally, Section 8.4 draws conclusion on the chapter.

8.1 Introduction

Although there exists a literature on distributed model predictive control [145], achieving feedback optimization in distributed MPC has been given no attention. Most distributed MPC schemes are designed to track known set-points and do not have any feedback optimization capabilities [25, 68, 67, 27, 70]. On the contrary, several distributed feedback optimization controllers have been proposed [206, 228, 208, 117, 160] mostly based on the feedback implementation of primal-dual algorithms. However, these primal-dual approaches cannot guarantee optimal transient performance and feasibility is only achieved for steady-state constraints. There is currently no result on distributed feedback optimization with guaranteed transient performance and feasibility. The result presented in this chapter addresses this gap by proposing feedback optimization control algorithms using a distributed model predictive control framework. For large-scale systems, the centralized FOMPC previously developed may not be suitable for feedback optimization due to the higher computational demand and the large geographical spread of such systems. Therefore, we develop in this chapter distributed MPC algorithms to solve the FOMPC problem for large-scale systems. We assume the large-scale system is composed of non-overlapping linear time-invariant subsystems with piecewise constant additive disturbances and coupling in the inputs and states. First, we remodel the large-scale system as a decentralized system with an additive disturbance that is composed of the dynamic interactions between neighbouring subsystems. Then using a velocity form of the plant model and a distributed form of the steady-state optimality error, we reformulate the feedback-optimizing model predictive control problem as a distributed tracking control problem for non-overlapping LTI systems. The developed algorithm is non cooperative, relies on a non-iterative exchange of information between neighbouring subsystems, and adopts a tube-based approach similar to the one developed for robust FOMPC to reject unplanned dynamic interactions between subsystems. Convergence of the proposed controllers to a neighbourhood of the true optimal steady-state setpoints are also shown.

8.2 The DFOMPC Problem

System dynamics, assumptions and steady-state optimization

Consider a linear discrete time invariant large scale system $\mathscr{P}$ described by the state-space model

$$\mathscr{P}: \begin{cases} x(k+1) = Ax(k) + Bu(k) + Ew(k), \\ y(k) = Cx(k), \\ (u(k), y(k)) \in \mathbb{Z} \coloneqq \mathbb{U} \times \mathbb{Y}, \quad w(k) \in \mathbb{W}, \end{cases} \tag{8.1}$$

where $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{U} \subseteq \mathbb{R}^{n_u}$, $y(k) \in \mathbb{Y} \subseteq \mathbb{R}^{n_y}$, and $w(k) \in \mathbb{W} \subseteq \mathbb{R}^{n_w}$ are the state, input, output and additive uncertainty (disturbance) vectors respectively. Let the system $\mathscr{P}$ consist of M interconnected and interacting subsystems. Letting

$$x(k) = \begin{bmatrix} x_1(k) \\ \vdots \\ x_M(k) \end{bmatrix}, \; u(k) = \begin{bmatrix} u_1(k) \\ \vdots \\ u_M(k) \end{bmatrix}, \; y(k) = \begin{bmatrix} y_1(k) \\ \vdots \\ y_M(k) \end{bmatrix}, \text{ and } w(k) = \begin{bmatrix} w_1(k) \\ \vdots \\ w_M(k) \end{bmatrix} \tag{8.2}$$

then the LTI system (8.1) can be decomposed into M dynamically coupled non-overlapping subsystems, each one described by the following state-space model:

$$\mathscr{P}_i\colon \begin{cases} x_i(k+1) = A_{ii}x_i(k) + B_{ii}u_i(k) + E_{ii}w_i(k) + \sum_{j\in\mathcal{N}_i} \Big\{A_{ij}x_j(k) + B_{ij}u_j(k)\Big\}, \\ \hfill y_i(k) = C_{ii}x_i(k), \\ \hfill (u_i(k), y_i(k)) \in \mathbb{Z}_i \coloneqq \mathbb{U}_i \times \mathbb{Y}_i, \end{cases} \tag{8.3}$$

where $x_i(k) \in \mathbb{R}^{n_{xi}}$, $y_i(k) \in \mathbb{Y}_i \subseteq \mathbb{R}^{n_{yi}}$, $u_i(k) \in \mathbb{U}_i \subseteq \mathbb{R}^{n_{ui}}$, and $w_i(k) \in \mathbb{W}_i \subseteq \mathbb{R}^{n_{wi}}$ are respectively the states, outputs, inputs and disturbances for the i^{th} subsystem. $\mathcal{N}_i$ is the index set of all subsystems neighbouring subsystem i i.e.,

$$\mathcal{N}_i \coloneqq \Big\{j \in \mathcal{N}\colon [A_{ij}\ B_{ij}] \neq \mathbf{0}, j \neq i\Big\}, \tag{8.4}$$

where $\mathcal{N} = \{1, \ldots, i, \ldots, M\}$ is the index set of all subsystems $\mathscr{P}_i$ and lastly,

$$n_y = \sum_{i=1}^{M} n_{yi},\; n_u = \sum_{i=1}^{M} n_{ui},\; n_w = \sum_{i=1}^{M} n_{wi}. \tag{8.5}$$

The subsystem $\mathscr{P}_j$ is a neighbour of the subsystem $\mathscr{P}_i$ if and only if the state, input and/or output of $\mathscr{P}_j$ affects the dynamics of $\mathscr{P}_i$ i.e. iff A_{ij}, B_{ij} are non zero. Here it is assumed the system $\mathscr{P}$ is decoupled in the output and the disturbance inputs (that is, C and E are block diagonal matrices). The topology of the large-scale system, $\mathscr{P}$ can be described elegantly using graph theoretical ideas by considering all family of sets $\mathcal{N}_i$ for all $i \in \mathcal{N}$. Refer to Appendix A for concepts on graph theory.

The main objective of DFOMPC is the design of distributed feedback control laws $u_i(k)$ that regulate respectively, the inputs and/or outputs of each subsystems $\mathscr{P}_i$ to steady-states that are collectively the solution of the following static optimization problem

$$\mathcal{RTO}(\bar{w})\colon \quad \bar{z}^*(\bar{w}) = \arg\min_{\bar{z}} \;\{\Phi(\bar{z}) \mid \bar{z} \in \mathbb{F}(\bar{w}),\; \bar{z} \in \mathbb{U} \times \mathbb{Y}\} \tag{8.6}$$

where $\mathbb{F}(\bar{w})$ is the feasible equilibrium set (i.e., $\mathbb{F}(\bar{\delta}, \bar{w})$ from chapter 3 with the parametric uncertainty vector δ set to zero). To properly formulate the control problem, the following assumptions are in order.

Assumption 16 (Local Reachability and Detectability)**.** *For each $i \in \mathcal{N}$, the pair (A_{ii}, B_{ii}) is reachable and the pair (C_{ii}, A_{ii}) is detectable.*

Assumption 17 (separable cost)**.** *The steady-state performance objective $\Phi(\bar{z})$ is separable i.e.,*

$$\Phi(\bar{z}) = \sum_{i \in \mathcal{N}} \Phi_i(\bar{z}_i). \tag{8.7}$$

Assumption 18 (Decoupled inequality constraints)**.** *The constraint sets $\mathbb{U}$, $\mathbb{Y}$ and $\mathbb{W}$ are separable i.e*

$$\mathbb{U} \coloneqq \prod_{i \in \mathcal{N}} \mathbb{U}_i, \;\; \mathbb{Y} \coloneqq \prod_{i \in \mathcal{N}} \mathbb{Y}_i \text{ and } \mathbb{W} \coloneqq \prod_{i \in \mathcal{N}} \mathbb{W}_i. \tag{8.8}$$

Assumption 19 (Sparsity of steady-state input/output gain)**.** *The steady-state input/output gain G_u is sparse or can be approximated by a sparse matrix.*

Assumption 19 allows the centralized steady-state optimality tracking error,

$$e(k) \coloneqq \tilde{G} \nabla \Phi(z(k)), \quad \tilde{G} = \begin{bmatrix} (G_u^\top)^\dagger & I_p \end{bmatrix} \tag{8.9}$$

to be written in the following distributed form

$$e_i(k) \coloneqq \nabla_{y_i} \Phi_i(z_i(k)) + \sum_{j \in \mathcal{N}_i} \gamma_{ij} \Big\{ \nabla_{u_i} \Phi_i(z_i(k)) - \nabla_{u_j} \Phi_j(z_j(k)) \Big\}, \tag{8.10}$$

where $z_i = (u_i, y_i)$ and γ_{ij} are the weights of the edge connecting the node $i \in \mathcal{N}$ to the node $j \in \mathcal{N}_i$. These weights can be obtained from the Laplacian matrix describing the topology of the large-scale system.

Remark 43 (Distributed power system dynamics)**.** *Assumption 19 is not restrictive as many practical distributed systems have steady-state input/output gain that are sparse or can be approximated by a sparse matrix. For example, consider the following multi-area power system dynamics [156, 222]*

$$\begin{aligned} \dot{\theta} &= f, \\ \dot{f} &= H^{-1} P^c - H^{-1} D f - H^{-1} P^{tie} - H^{-1} P^L \end{aligned} \tag{8.11}$$

where θ is the voltage angle deviation, f the frequency deviation, P^L the demand fluctuation, P^{tie} the inter-area power flow and P^c the load-reference set-points for all control areas. The variables H and D are the power system parameters.
For a multi-area power network with a connection topology defined by the graph $\mathscr{G}$, then P^{tie} can be written as [222],

$$P^{tie} = \mathscr{L}_{\mathscr{G}}\theta \tag{8.12}$$

where $\mathscr{L}_{\mathscr{G}}$ is the graph Laplacian matrix of the power network. Let the input, output and disturbance be defined respectively as: $u = P^c$, $y = \theta$ and $w = P^L$, then the power system dynamics (8.11) *can be written as the continuous-time dynamical system*

$$\begin{aligned} \dot{x} &= A_c x + B_c u + E_c w, \\ y &= C_c x, \end{aligned} \tag{8.13}$$

with

$$A_c = \begin{bmatrix} \mathbf{0} & I \\ -H^{-1}\mathscr{L}_{\mathscr{G}} & -H^{-1}D \end{bmatrix},\ B_c = \begin{bmatrix} \mathbf{0} \\ H^{-1} \end{bmatrix}, C_c = \begin{bmatrix} I & \mathbf{0} \end{bmatrix},\ E_c = \begin{bmatrix} \mathbf{0} \\ -H^{-1} \end{bmatrix} \text{ and } x = \begin{bmatrix} \theta \\ f \end{bmatrix}. \tag{8.14}$$

The steady-state gain for the power system above can be computed as

$$G_u = -C_c A_c^{-1} B_c = \mathscr{L}_{\mathscr{G}}^{-1}. \tag{8.15}$$

From the steady-state gain, the matrix $\tilde{G}$ is defined as the following (sparse) matrix

$$\tilde{G} = \begin{bmatrix} \mathscr{L}_{\mathscr{G}} & I_p \end{bmatrix}, \tag{8.16}$$

and therefore,

$$e(k) = \tilde{G}\nabla\Phi = \begin{bmatrix} \mathscr{L}_{\mathscr{G}} & I_p \end{bmatrix} \begin{bmatrix} \nabla_u \Phi \\ \nabla_y \Phi \end{bmatrix} = \mathscr{L}_{\mathscr{G}} \nabla_u \Phi + \nabla_y \Phi. \tag{8.17}$$

From (8.7)*, we can express* (8.17) *as*

$$e(k) = \mathscr{L}_{\mathscr{G}} \begin{bmatrix} \nabla_{u_1}\Phi_1(z_1) \\ \vdots \\ \nabla_{u_M}\Phi_M(z_M) \end{bmatrix} + \begin{bmatrix} \nabla_{y_1}\Phi_1(z_1) \\ \vdots \\ \nabla_{y_M}\Phi_M(z_M) \end{bmatrix}. \tag{8.18}$$

Using the properties of Laplacian matrices stated in Lemma A.5.1, we can express (8.18) *as*

$$\begin{bmatrix} e_1(k) \\ \vdots \\ e_i(k) \\ \vdots \\ e_M(k) \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_1} \gamma_{1j}\Big\{\nabla_{u_1}\Phi_1 - \nabla_{u_j}\Phi_j\Big\} \\ \vdots \\ \sum_{j\in\mathcal{N}_i} \gamma_{ij}\Big\{\nabla_{u_i}\Phi_i - \nabla_{u_j}\Phi_j\Big\} \\ \vdots \\ \sum_{j\in\mathcal{N}_M} \gamma_{Mj}\Big\{\nabla_{u_M}\Phi_M - \nabla_{u_j}\Phi_j\Big\} \end{bmatrix} + \begin{bmatrix} \nabla_{y_1}\Phi_1 \\ \vdots \\ \nabla_{y_i}\Phi_i \\ \vdots \\ \nabla_{y_M}\Phi_M \end{bmatrix}. \qquad (8.19)$$

Equation (8.10) *then follows easily from* (8.19)*.*

Problem statement

Given the above data, we now define the distributed feedback optimizing model predictive control (DFOMPC) problem.

Problem 7 (The DFOMPC Problem)**.** *Design for each linear discrete-time subsystem* (8.3)*, a distributed feedback control law, $u_i(k)$, obtained from the solution to an N-horizon optimal control problem, such that for any admissible $w_i \in \mathbb{W}_i$:*

1. *The point $\bar{z}^*(w)$ is for all feasible $x_i(0)$ and all $i \in \mathcal{N}$, an asymptotically stable equilibrium for the closed-loop systems, satisfying*

$$\lim_{k\to\infty} \left(\Big[u_i(k)\Big]_{\forall i\in\mathcal{N}}, \Big[y_i(k)\Big]_{\forall i\in\mathcal{N}} \right)(w) \;= \bar{z}^*(w), \qquad (8.20)$$

2. *The feedback policy $u_i(k)$ minimizes a transient performance criterion for each $i \in \mathcal{N}$.*

3. *The constraints $(u_i(k), y_i(k)) \in \mathbb{Z}_i$ are satisfied $\forall i \in \mathcal{N}$ at all times.*

By regulating the steady-state optimality error (8.10) to zero, the DFOMPC problem above can be translated into a distributed tracking model predictive control problem that achieves optimal (or near- optimal) steady-state operation. In the following section we present a robust model predictive control algorithm that solves the DFOMPC for a quadratic steady-state optimization problem.

8.3 Proposed Solution

In this section, we propose solutions to the distributed FOMPC control problem when the steady-state performance objective, Φ, is a sum of quadratic costs, i.e.

$$\Phi(\bar{z}) = \sum_{i\in\mathcal{N}} \Phi_i(\bar{z}_i) \tag{8.21}$$

where

- $\Phi_i(\bar{z}_i) = \frac{1}{2}\bar{z}_i^\top Q^i_{zz}\bar{z}_i + (R^i_z)^\top \bar{z}_i$,
- $Q^i_{zz} = \begin{bmatrix} Q^i_{uu} & Q^i_{uy} \\ Q^i_{yu} & Q^i_{yy} \end{bmatrix}$, and $R^i_z = \begin{bmatrix} R^i_u \\ R^i_y \end{bmatrix}$.

The steady-state tracking error (8.10) in this case can be expressed as the distributed affine equation

$$e_i(k) \coloneqq \Lambda_{yi} y_i(k) + \Lambda_{ui} u_i(k) + r_i + \sum_{j\in\mathcal{N}_i} (-\gamma_{ij})\Big[\Lambda_{yj} y_j(k) + \Lambda_{uj} u_j(k) + r_j\Big] \tag{8.22}$$

where

- $\Lambda_{yi} = \frac{1}{2}\Big[\Big(Q^i_{yy} + (Q^i_{yy})^\top\Big) + \sum_{j\in\mathcal{N}_i} \gamma_{ij}\Big(Q^i_{uy} + (Q^i_{yu})^\top\Big)\Big]$,
- $\Lambda_{ui} = \frac{1}{2}\Big[\Big(Q^i_{yu} + (Q^i_{uy})^\top\Big) + \sum_{j\in\mathcal{N}_i} \gamma_{ij}\Big(Q^i_{uu} + (Q^i_{uu})^\top\Big)\Big]$,
- $r_i = R^i_y + \sum_{j\in\mathcal{N}_i} \gamma_{ij} R^i_u$,
- $\Lambda_{yj} = \frac{1}{2}\Big[\Big(Q^j_{uy} + (Q^j_{yu})^\top\Big)\Big]$, $\Lambda_{uj} = \frac{1}{2}\Big[\Big(Q^j_{uu} + (Q^j_{uu})^\top\Big)\Big]$, and
- $r_j = R^j_u$.

This error may be computed directly from the input $u_i(k)$ and output $y_i(k)$ measurements of each subsystems, and those of neighbouring subsystems (i.e $u_j(k)$ and $y_j(k)$), provided the steady-state objectives Φ_i and interaction topology (γ_{ij}) of the distributed dynamical systems are known. It is clear from (8.22) that explicit knowledge of the steady-state set points ($z^*(\bar{w})$) or unknown disturbances ($\bar{w}$) are not needed to compute this error, and this provides opportunities to achieve robust tracking of the optimal steady-state under unknown additive disturbances. For the purpose of control design, we make the following assumption on the system and cost matrices

Assumption 20. *For each* $i \in \mathcal{N}$*, the matrix* $S_i = \begin{bmatrix} A_{ii} - I_{n_{xi}} & B_{ii} \\ \Lambda_{yi} C_{ii} & \Lambda_{ui} \end{bmatrix}$ *has full row rank i.e,* $\operatorname{rank} S_i = n_{yi} + n_{ui}$*.*

Remark 44. *Assumption 20 is standard in the output regulation literature and is required to guarantee the FOMPC problem is solvable for each decentralized subsystem dynamics. This assumption is independent of the connection topology of the large-scale system when the disturbance is separable i.e.,* $\mathbb{W} := \prod_{i \in \mathcal{N}} \mathbb{W}_i$*. Please refer to [7, 109] for further details on this assumption.*

We now present the following algorithms for solving the DFOLQC problem.

Velocity and Error Dynamics

To regulate each subsystem to $e_i = 0$, we consider the velocity, or incremental, form of the sub-systems dynamics (8.3) augmented with the steady-state optimality tracking error (8.22), whose output is the tracking error $e_i(k)$:

$$\epsilon_i(k+1) = \mathcal{A}_{ii}\epsilon_i(k) + \mathcal{B}_{ii}\delta u_i(k) + \mathcal{E}_{ii}\delta w_i(k) + \sum_{j \in \mathcal{N}_i} \left\{ \mathcal{A}_{ij}\epsilon_j(k) + \mathcal{B}_{ij}\delta u_j(k) \right\}, \tag{8.23a}$$

$$e_i(k) = \mathcal{C}_{ii}\epsilon_i(k) + \mathcal{D}_{ii}\delta u_i(k) + \sum_{j \in \mathcal{N}_i} \left\{ \mathcal{C}_{ij}\epsilon_j(k) + \mathcal{D}_{ij}\delta u_j(k) \right\} \tag{8.23b}$$

where

$$\epsilon_i(k) := \begin{bmatrix} \delta x_i(k) \\ e_i(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x_i(k) &:= x_i(k) - x_i(k-1), \\ \delta u_i(k) &:= u_i(k) - u_i(k-1), \\ \delta w_i(k) &:= w_i(k) - w_i(k-1), \end{aligned} \tag{8.24}$$

and

$$\mathcal{A}_{ii} = \begin{bmatrix} A_{ii} & \mathbf{0}_{n_{xi} \times n_{yi}} \\ \Lambda_{yi} C_{ii} & I_{n_{yi}} \end{bmatrix}, \qquad \mathcal{B}_{ii} = \begin{bmatrix} B_{ii} \\ \Lambda_{ui} \end{bmatrix}, \quad \mathcal{A}_{ij} = \begin{bmatrix} A_{ii} & \mathbf{0}_{n_{xi} \times n_{yi}} \\ -\gamma_{ij} \Lambda_{yi} C_{jj} & \mathbf{0}_{n_{yi} \times n_{yi}} \end{bmatrix}, \tag{8.25a}$$

$$\mathcal{C}_{ii} = \begin{bmatrix} \Lambda_{yi} C_{ii} & I_{n_{yi}} \end{bmatrix}, \qquad \mathcal{D}_{ii} = \Lambda_{ui}, \qquad \mathcal{E}_{ii} = \begin{bmatrix} E_{ii} \\ \mathbf{0}_{n_{yi} \times n_{xi}} \end{bmatrix}, \tag{8.25b}$$

$$\mathcal{C}_{ij} = \begin{bmatrix} -\gamma_{ij} \Lambda_{yj} C_{jj} & \mathbf{0}_{n_{yi} \times n_{yi}} \end{bmatrix}, \text{ and } \mathcal{D}_{ij} = -\gamma_{ij} \Lambda_{uj}. \tag{8.25c}$$

To enable the design of stablizing distributed control algorithms, the following additional assumption is made on the velocity dynamics

Assumption 21 (decentralized stabilizability)**.** *There exists a block diagonal matrix* $\mathcal{K} = \text{blkdiag}(\mathcal{K}_1, \ldots, \mathcal{K}_M)$ *with* $\mathcal{K}_i \in \mathbb{R}^{n_{xi}+n_{yi}}$, $i \in \mathbb{I}_{[1,M]}$ *such that*

(i) The matrix $\mathcal{F} = \mathcal{A} - \mathcal{B}\mathcal{K}$ *is Schur.*

(ii) The matrices $\mathcal{F}_{ii} = \mathcal{A}_{ii} - \mathcal{B}_{ii}\mathcal{K}_i$ *are Schur*

where $\mathcal{A} = \text{blkdiag}(\mathcal{A}_{11}, \ldots, \mathcal{A}_{MM})$ *and* $\mathcal{B} = \text{blkdiag}(\mathcal{B}_{11}, \ldots, \mathcal{B}_{MM})$*.*

Given the velocity dynamics, we can formulate the DFOMPC problem as a tracking MPC problem that regulates the velocity dynamics (8.23) to the origin while minimizing a dynamic performance objective for each subsystem. We achieve this by solving online, the following optimal control problem:

$$\mathbb{P}_i(\epsilon_i(k), v_{-i}(k))\colon \min_{\delta \mathbf{u}_i(k) \in \mathscr{U}_{N,i}} V_{N,i}(\epsilon_i(k), \delta \mathbf{u}_i(k)) \tag{8.26}$$

where the feasible region $\mathscr{U}_{Ni}(\epsilon_i(k), v_{-i}(k))$ is defined as

$$\mathscr{U}_{N,i}(\epsilon_i(k), v_{-i}(k)) \triangleq \left\{ \delta \mathbf{u}_i(k) \,\middle|\, \begin{array}{r} \epsilon_i(k+t+1) = \mathcal{A}_{ii}\epsilon_i(k+t) + \mathcal{B}_{ii}\delta u_i(k+t) + \mathcal{E}_{ii}\delta w_i(k+t) + \\ \sum\limits_{j \in \mathcal{N}_i} \Big\{ \mathcal{A}_{ij}\epsilon_j(k+t) + \mathcal{B}_{ij}\delta u_j(k+t) \Big\}, \ \forall t \in \mathbb{I}_{[0,N-1]} \\ \epsilon_i(k+t) \in \mathbb{G}_i, \ \forall t \in \mathbb{I}_{[0,N-1]} \\ \epsilon_i(k+N) \in \mathbb{G}_{f,i} \end{array} \right\} \tag{8.27}$$

and $v_{-i}(k+t) = (\epsilon_j(k+t), \delta u_j(k+t))$ is velocity state and input from the neighbouring subsystems.
In this problem, the decision variable is the sequence of control increments over the N-step prediction horizon for each subsystem:

$$\delta \mathbf{u}_i(k) \coloneqq \Big\{ \delta u_i(k), \delta u_i(k+1) \ldots, \delta u_i(k+N-1) \Big\}. \tag{8.28}$$

These sequences are chosen to minimize the local performance objective,

$$V_{N,i}(\epsilon_i(k), \delta \mathbf{u}_i(k)) = V_{f,i}(\epsilon_i(k+N)) + \sum_{t=0}^{N-1} l_i(e_i(k+t), \delta u_i(k+t)) \tag{8.29}$$

which is composed of a stage cost

$$l_i(e_i(k+t), \delta u_i(k+t)) := \frac{1}{2}\Big(e_i(k+t)^\top Q_{e,i} e_i(k+t) + \delta u_i(k+t)^\top R_{\delta,i} \delta u_i(k+t)\Big) \quad (8.30)$$

and a terminal cost

$$V_{f,i}(\epsilon_i(k+N)) := \frac{1}{2}\epsilon_i(k+N)^\top \mathcal{P}_i \epsilon_i(k+N). \quad (8.31)$$

This cost aims to capture the transient performance objective for each subsystem: the term $e_i(k+t)^\top Q_{e,i} e_i(k+t)$ penalizes the distance of the $i^{th}-$subsystem's tracking error from steady-state optimality and therefore determines the duration of the transient phase, while the term $\delta u_i(k+t)^\top R_{\delta,i} \delta u_i(k+t)$ penalizes the incremental control effort. The solution of $\mathbb{P}_i(\epsilon_i(k), v_{-i}(k))$ yields the optimal control sequence $\delta \mathbf{u}_i^*(k)$ and the corresponding optimal cost $V_{N,i}^0(\epsilon_i(k))$ which is also referred to as the value function. The DFOMPC law is then given by the first element of $\delta \mathbf{u}_i^*(k)$ i.e.

$$\delta u_i(k) = \kappa_{N,i}(\epsilon_i(k)) = \delta \mathbf{u}_i^*(k). \quad (8.32)$$

To solve $\mathbb{P}_i(\epsilon_i(k), v_{-i}(k))$ will require complete knowledge of the system dynamics (8.23) at subsystem i. Due to the interaction terms $v_{-i}(k+t)$, this information is not necessarily known by all subsystems, and the feasible set $\mathscr{U}_{N,i}(\epsilon_i(k), v_{-i}(k))$ will therefore be uncertain. Consequently, a solution to $\mathbb{P}_i(\epsilon_i(k), v_{-i}(k))$ cannot be precisely computed. To workaround this limitation, inspired by [27, 25] we propose two robust non-iterative approaches in this chapter. The first approach ignores the dynamic interactions from neighbouring subsystems and utilizes the resulting decentralized velocity dynamics to formulate a feedback optimizing model predictive control problem which when solved online generates a decentralized FOMPC law that approximates the actual distributed FOMPC law (8.32). This solution although simple, cannot guarantee convergence to the true global steady-state optimum, $z^*(\bar{w})$. The robust convergence of this control law to some approximation of $z^*(\bar{w})$ will be investigated, using the results from the inherent robustness of centralized FOMPC from Chapter 6 albeit applied in the context of a decentralized system. To improve upon the decentralized FOMPC algorithm, we will exploit neighbour-to-neighbour communication to exchange predictions of the dynamic interaction variable, $\hat{v}_{-i}(k)$ between subsystems. This will improve the accuracy of the feasible set $\mathscr{U}_{N,i}(\epsilon_i(k), \hat{v}_{-i}(k))$ computations. Also, to guarantee convergence and recursive feasibility, we will adopt a tube based approach to reject any uncertainty in the subsystem interactions that result from the discrepancies between the shared dynamic predictions and the actual interaction dynamics. We will assume this uncertainty lies within a known bounded set. We now present details of these developments in the sections that follow.

8.3.1 Inherently robust decentralized FOMPC (DeFOMPC)

In this section we present an inherently robust decentralized feedback-optimizing model predictive control (DeFOMPC) algorithm that regulates the distributed dynamical systems (8.3) to steady-state equilibrium points that approximate the optimal solution to the steady-state optimization problem (8.6). To design the control algorithm, the dynamic interaction between the subsystems are neglected and a nominal FOMPC regulator akin to that presented in Chapter 6 will be applied to each subsystem. As a result, we solve M optimal control problems in parallel using only local subsystem dynamics and local constraints.

To begin, consider the following nominal subsystem dynamics with the interactions neglected, $\mathscr{P}_{ii}$:

$$\bar{x}_i(k+1) = A_{ii}\bar{x}_i(k) + B_{ii}\bar{u}_i(k) + E_{ii}\bar{w}_i(k), \quad \bar{y}_i(k) = C_{ii}\bar{x}_i(k), \tag{8.33a}$$

$$\bar{e}_i(k) \coloneqq \Lambda_{yi}\bar{y}_i(k) + \Lambda_{ui}\bar{u}_i(k) + \bar{r}_i, \tag{8.33b}$$

$$(\bar{u}_i(k), \bar{y}_i(k)) \in \mathbb{Z}_i \coloneqq \mathbb{U}_i \times \mathbb{Y}_i. \tag{8.33c}$$

Assuming $\bar{w}_i(k)$ is constant or slowly varying, then (8.33) can be expressed as the following velocity dynamics

$$\bar{\epsilon}_i(k+1) = \mathcal{A}_{ii}\bar{\epsilon}_i(k) + \mathcal{B}_{ii}\delta\bar{u}_i(k), \tag{8.34a}$$

$$\bar{e}_i(k) = \mathcal{C}_{ii}\bar{\epsilon}_i(k) + \mathcal{D}_{ii}\delta\bar{u}_i(k). \tag{8.34b}$$

To compute the decentralized feedback optimizing model predictive control law, $\delta\bar{u}_i(k)$, we solve at each sampling time, k, following the approach in Chapter 6, a FOMPC problem for the decentralized velocity dynamics (8.34). We achieve this by solving online, the following optimal control problem:

$$\bar{\mathbb{P}}_{ii}(\bar{\epsilon}_i(k))\colon \min_{\delta\bar{\mathbf{u}}_i(k)\in\mathscr{U}_{N,i}} V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) \tag{8.35}$$

where the feasible region $\mathscr{U}_{N,i}(\epsilon_i(k))$ is defined as

$$\mathscr{U}_{N,i}(\bar{\epsilon}_i(k)) \triangleq \left\{ \delta\bar{\mathbf{u}}_i(k) \middle| \begin{array}{r} \bar{\epsilon}_i(k+t+1) = \mathcal{A}_{ii}\bar{\epsilon}_i(k+t) + \mathcal{B}_{ii}\delta\bar{u}_i(k+t)\ \forall t \in \mathbb{I}_{[0,N-1]}, \\ \bar{\epsilon}_i(k+t) \in \bar{\mathbb{G}}_i,\ \forall t \in \mathbb{I}_{[0,N-1]}, \\ \bar{\epsilon}_i(k+N) \in \bar{\mathbb{G}}_{f,i} \end{array} \right\} \tag{8.36}$$

and the cost function $V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k))$ is given by:

$$V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) = V_{f,i}(\bar{\epsilon}_i(k+N)) + \sum_{t=0}^{N-1} l_i(\bar{e}_i(k+t), \delta\bar{u}_i(k+t)), \tag{8.37}$$

where,

$$l_i(\bar{e}_i(k+t), \delta\bar{u}_i(k+t)) := \frac{1}{2}\Big(\bar{\epsilon}_i(k+t)^\top \mathcal{Q}_i\bar{\epsilon}_i(k+t) + 2\bar{\epsilon}_i^\top(k+t)\mathcal{N}_i\delta\bar{u}_i(k+t) + \delta\bar{u}_i(k+t)^\top\mathcal{R}_i\delta\bar{u}_i(k+t)\Big), \tag{8.38}$$

and

$$V_{f,i}(\bar{\epsilon}_i(k+N)) := \frac{1}{2}\bar{\epsilon}_i(k+N)^\top\mathcal{P}_i\bar{\epsilon}_i(k+N). \tag{8.39}$$

Here, $\mathcal{Q}_i = \mathcal{C}_{ii}^\top Q_{e,i}\mathcal{C}_{ii}$, $\mathcal{N}_i = \mathcal{C}_{ii}^\top Q_{e,i}\mathcal{D}_{ii}$ and $\mathcal{R}_i = R_{\delta,i} + \mathcal{D}_{ii}^\top Q_{e,i}\mathcal{D}_{ii}$ and $\bar{\epsilon}_i(k)$ is obtained at sample time k, from the local input/output, and state measurements at subsystem i:

$$\bar{\epsilon}_i(k) = \begin{bmatrix} \delta x_i(k) \\ \bar{e}_i(k-1) \end{bmatrix} = \begin{bmatrix} x_i(k) - x_i(k-1) \\ \Lambda_{y,i}y_i(k-1) + \Lambda_{ui}u_i(k-1) + r_i \end{bmatrix}. \tag{8.40}$$

Similar to the nominal FOMPC algorithm of Chapter 6, to guarantee positive definiteness of the performance objective $V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k))$, we choose the matrices $R_{\delta,i}$, $Q_{e,i}$ and $\mathcal{P}_i$ to satisfy $R_{\delta,i} \succ 0$, $\mathcal{P}_i \succeq 0$ and

$$Q_{e,i} - Q_{e,i}\Lambda_{ui}(R_{\delta,i} + \Lambda_{ui}^\top Q_{e,i}\Lambda_{ui})^{-1}\Lambda_{ui}Q_{e,i}^\top \succeq 0. \tag{8.41}$$

The constraints on $\bar{u}_i$ and $\bar{y}_i$ must be reformulated in terms of the constraints on the MPC optimization variables $\delta\bar{u}_i$ and $\bar{\epsilon}_i$. To this end, similarly to Chapter 6, we write the relation between these variables as (see section 7.6 for details)

$$\begin{bmatrix} \bar{x}_i(k) \\ \bar{u}_i(k-1) \end{bmatrix} = \tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i(k) \\ \bar{r}_i \end{bmatrix} + \tilde{\mathcal{D}}_i\bar{w}_i(k-1), \tag{8.42}$$

where

$$\tilde{\mathcal{C}}_i = \begin{bmatrix} A_{ii} & B_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} & I_{n_{ui}} \end{bmatrix} S_i^{-1} \begin{bmatrix} I_{n_{xi}} & \mathbf{0}_{n_{xi}\times n_{yi}} & \mathbf{0}_{n_{xi}\times n_{yi}} \\ \mathbf{0}_{n_{yi}\times n_{xi}} & I_{n_{yi}} & -I_{n_{yi}} \end{bmatrix},$$
$$\tilde{\mathcal{D}}_i = \begin{bmatrix} E_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} \end{bmatrix} - \begin{bmatrix} A_{ii} & B_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} & I_{n_{ui}} \end{bmatrix} S_i^{-1} \begin{bmatrix} E_{ii} \\ \mathbf{0}_{n_{yi}\times n_{xi}} \end{bmatrix}. \tag{8.43}$$

The online solution of the OCP, $\bar{\mathbb{P}}_{ii}(\bar{\epsilon}_i(k))$ yields the optimal control sequence $\delta\bar{\mathbf{u}}_i^*(k)$ and the corresponding optimal cost $V_{N,i}^0(\bar{\epsilon}_i(k))$ which is also referred to as the value function. The standard decentralized FOMPC law is then given by the first element of $\bar{\mathbf{u}}_i^*(k)$ i.e.,

$$\kappa_{N,i}(\bar{\epsilon}_i(k)) = \bar{\mathbf{u}}_i^*(k), \tag{8.44}$$

and its feasible region is given by

$$\mathscr{E}_{N,i} \triangleq \{\bar{\epsilon}_i(k) \in \mathbb{R}^{n_{xi}+n_{yi}} \colon \mathscr{U}_{N,i}(\bar{\epsilon}_i(k)) \neq \emptyset\}. \tag{8.45}$$

To guarantee closed-loop stability and recursive feasibility, the design of the OCP, $\bar{\mathbb{P}}_{ii}(\bar{\epsilon}_i(k))$ must meet the requirements summarized by the following assumptions.

Assumption 22 (Terminal cost)**.** *The matrix $\mathcal{P}_i$ is such that given the control law, $\delta\bar{u}_i(k) = -\mathcal{K}_i\bar{\epsilon}_i(k)$, for all $k \geq N$, the terminal cost $V_{f,i}(\bar{\epsilon}_i(k))$ is a control Lyapunov function i.e.,*

$$V_{f,i}(\bar{\epsilon}_i(k+1)) - V_{f,i}(\bar{\epsilon}_i(k)) \leq -l_i(\bar{\epsilon}_i(k), -\mathcal{K}_i\bar{\epsilon}_i(k)),\ \forall \bar{\epsilon}_i(k) \in \bar{\mathbb{G}}_{f,i} \tag{8.46}$$

where the set $\bar{\mathbb{G}}_{f,i}$ is a terminal constraint set to be defined later.

Remark 45. *To meet this requirement, we select $\mathcal{P}_i$ to satisfy the following Lyapunov equation:*

$$\Big(\mathcal{A}_{ii} - \mathcal{B}_{ii}\mathcal{K}_i\Big)^\top \mathcal{P}_i\Big(\mathcal{A}_{ii} - \mathcal{B}_{ii}\mathcal{K}_i\Big) - \mathcal{P}_i = -\Big(\mathcal{Q}_i - \mathcal{N}_i\mathcal{K}_i + \mathcal{K}_i^\top\mathcal{R}_i\mathcal{K}_i\Big) \tag{8.47}$$

where the matrix $\mathcal{K}_i$ is chosen such that Assumption 21 is satisfied.

Assumption 23 (Constraint set)**.** *The set $\bar{\mathbb{G}}_i$ is compact, contains the origin and enforces the constraints $(\bar{u}_i(k), \bar{y}_i(k)) \in \mathbb{U}_i \times \mathbb{Y}_i$ on the state $\bar{\epsilon}_i(k)$, i.e.,*

$$\bar{\mathbb{G}}_i \coloneqq \{\bar{\epsilon}_i \colon (\bar{u}_i, \bar{y}_i) \in \mathbb{U}_i \times \mathbb{Y}_i\}. \tag{8.48}$$

Remark 46. *Similar to the centralized FOMPC algorithm in Chapter 6, the set* $\bar{\mathbb{G}}_i$ *can be computed from* (8.42) *as:*

$$\bar{\mathbb{G}}_i := \left\{ \bar{\epsilon}_i : \tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i \\ \bar{r}_i \end{bmatrix} \in (\mathbb{X}_i \times \mathbb{U}_i) \ominus \tilde{\mathcal{D}}_i \bar{\mathbb{W}}_i \right\}. \tag{8.49}$$

Assumption 24 (Terminal constraint set)**.** *The set* $\bar{\mathbb{G}}_{f,i}$ *is as a local constraint admissible, positive invariant set for* (8.34a) *with* $\delta \bar{u}_i(k) = -\mathcal{K}_i \bar{\epsilon}_i(k)\ \forall k \geq N$ *i.e.,*

$$(\mathcal{A}_{ii} - \mathcal{B}_{ii}\mathcal{K}_i)\bar{\mathbb{G}}_{f,i} \subseteq \bar{\mathbb{G}}_{f,i} \text{ and } \bar{\mathbb{G}}_{f,i} \subseteq \bar{\mathbb{G}}_i. \tag{8.50}$$

The i^{th} subsystem with the terminal constraint set $\bar{\mathbb{G}}_{f,i}$, the terminal cost $V_{f,i}(\bar{\epsilon}_i(k))$ and the terminal control law $\delta \bar{u}_i(k) = -\mathcal{K}_i \bar{\epsilon}_i(k),\ \forall k \geq N$, is guaranteed stable without the dynamic interactions. In DeFOMPC, the transient performance objective is local to each subsystem and therefore does not guarantee optimal transient performance for the actual large scale system. Infact, DeFOMPC yields very conservative transient performance compared to its centralized counterpart. Also, due to the neglected interaction dynamics, decentralized FOMPC may not guarantee the closed-loop stability of the actual large-scale system. Therefore, the question of how much interaction dynamics can be neglected without compromising the closed-loop stability of the actual large-scale system under the decentralized FOMPC law above arises. We address this question by applying Theorem (6.3.4) of Chapter 6 in a distributed setting.

Theorem 8.3.1 (Robust stability of DeFOMPC)**.** *Let the velocity dynamics,* (8.34)*, the stage cost,* $l_i(\bar{e}_i(k), \delta \bar{u}_i(k))$ *and the terminal cost,* $V_{f,i}(\bar{\epsilon}_i(k))$ *be continuous for all subsystems* i*. Also, let* $l_i(\bar{e}_i(k), \delta \bar{u}_i(k))$ *be bounded above by a* κ_∞ *function, and the Assumptions 22, 23 and 24 be true for all subsystems. Suppose,*

$$\epsilon_i(0) \in \Omega(\bar{R}_i) \triangleq \{\ \epsilon_i \mid\ V^0_{N,i}(\epsilon_i) \leq \bar{R}_i\} \subset \mathscr{E}_{N,i}, \tag{8.51}$$

where $\Omega(\bar{R}_i)$ *denotes the largest sublevel set contained in* $\mathscr{E}_{N,i}$*, if there exists scalars* $R_i \in (0, \bar{R}_i]$ *and* L_i *satisfying Lemma 6.3.3 such that for all admissible* δv_i *and* δp_i

$$\|\mathcal{E}_{vi}\delta v_i\| + \|\mathcal{E}_{pi}\delta p_i\| \leq \frac{(1-\gamma_i)R_i}{L_i} \tag{8.52}$$

for all $\epsilon_i \in \Omega(\bar{R}_i)$*, then* *(i) the set* $\Omega(R_i) \triangleq \{\ \epsilon_i \mid \ V^0_{N,i}(\epsilon_i) \leq R_i\} \subset \Omega(\bar{R}_i)$ *is positively invariant for the subsystem velocity dynamics* (8.34)*.*

(ii) the set $\Omega(\bar{R}_i)$ *is also positively invariant for* (8.34)*. Therefore, the states* ϵ_i *remains in* $\Omega(\bar{R}_i)$ *for all time, and enter and remain within the set* $\Omega(R_i)$ *after some finite time.*

(iii) the actual inputs and outputs (u_i, y_i) *of the all subsystems* (8.3) *converge to steady-state values that are a close approximation to the optimal equilibrium* $\bar{z}^*$ *for the steady-state optimization problem* (6.2)*.*
Here,

$$\begin{aligned} v_i(k) &\coloneqq \sum_{j\in\mathcal{N}_i} \Big\{A_{ij}x_j(k) + B_{ij}u_j(k)\Big\}, \\ p_i(k) &\coloneqq \sum_{j\in\mathcal{N}_i} (-\gamma_{ij})\Big[\Lambda_{yj}y_j(k) + \Lambda_{uj}u_j(k) + r_j\Big], \\ \mathcal{E}_{vi} &= \begin{bmatrix} E_{ii} \\ \mathbf{0} \end{bmatrix},\ \mathcal{E}_{pi} = \begin{bmatrix} \mathbf{0} \\ I_{nyi} \end{bmatrix}. \end{aligned} \tag{8.53}$$

Proof. The proof follows directly from Theorem 6.3.4 applied to a distributed setting. □

This result establishes the stability of the controlled subsystems under the assumption that the interactions between the dynamics and the error, i.e., v_i and p_i respectively, of each subsystem i, satisfy the condition in (8.52). Although the above condition is somewhat abstract, it confirms the intuition that stability under decentralized control is more likely to be guaranteed if the lumped neglected interaction terms δv_i and δp_i are suitably small and the regulatory action is strong (i.e $Q_{e,i} \gg R_{\delta i}$).

8.3.2 Robust Tube-based Distributed FOMPC (tDFOMPC)

The DeFOMPC algorithm presented in the previous section relied on the inherent robustness of nominal FOMPC to reject the interactions between the subsystems. As a result, the closed loop system was only guaranteed stable for small range of dynamic interactions. Also, the algorithm drives the closed-loop system to an approximation of the true global steady-state optimum, $\bar{z}^*$ with a margin of error that is often significant if the state and error coupling are not negligible, which is more often the case in practical systems. In an effort to improve the performance of the DeFOMPC, reduce the conservatism and obtain more accurate tracking of the global steady-state optimum, we present in this section, a distributed FOMPC algorithm which extends the DeFOMPC algorithm above by considering information about the output and dynamic coupling transmitted between subsystems. To accommodate the unavoidable discrepancy that exists between the shared state and error trajectories, and the actual trajectories, we adopt a tube-based FOMPC approach to reject this uncertainty.

To begin, assume that, at any time instant k, each velocity subsystem (8.23) transmits its state and input reference trajectories $\hat{\epsilon}_i(k)$ and $\delta\hat{u}_i(k)$, $\forall k \in \mathbb{I}_{[0,N-1]}$ to its dynamic neighbours, $j \in \mathcal{N}_i$. By adding suitable constraints to each local FOMPC formulation, each subsystem is made to guarantee that, for all $k > 0$, the actual states $\epsilon_i(k)$, and the actual inputs $\delta u_i(k)$ both lie in a specified time-invariant neighbourhoods of $\hat{\epsilon}_i(k)$ and $\delta\hat{u}_i(k)$ respectively i.e.,

$$\epsilon_i(k) - \hat{\epsilon}_i(k) \in \mathbb{M}_i, \quad \delta u_i(k) - \delta\hat{u}_i(k) \in \mathbb{N}_i, \ \forall k \geq 0 \tag{8.54}$$

where $0 \in \mathbb{M}_i$ and $0 \in \mathbb{N}_i$. With this, the velocity dynamics (8.23) can be written as

$$\epsilon_i(k+1) = \mathcal{A}_{ii}\epsilon_i(k) + \mathcal{B}_{ii}\delta u_i(k) + \hat{d}_i(k) + \tilde{d}_i(k), \tag{8.55a}$$

$$e_i(k) = \mathcal{C}_{ii}\epsilon_i(k) + \mathcal{D}_{ii}\delta u_i(k) + \hat{f}_i(k) + \tilde{f}_i(k) \tag{8.55b}$$

where $\hat{d}_i(k)$ is a known state disturbance (obtained from the shared reference trajectories) given as

$$\hat{d}_i(k) = \sum_{j \in \mathcal{N}_i} \Big\{ \mathcal{A}_{ij}\hat{\epsilon}_j(k) + \mathcal{B}_{ij}\delta\hat{u}_j(k) \Big\}, \tag{8.56}$$

and $\tilde{d}_i(k)$ is an unknown, but bounded state disturbance (i.e., the deviation from the planned/reference trajectories in the velocity state), given by

$$\tilde{d}_i(k) = \sum_{j \in \mathcal{N}_i} \Big\{ \mathcal{A}_{ij}(\epsilon_j(k) - \hat{\epsilon}_j(k)) + \mathcal{B}_{ij}(\delta u_j(k) - \delta\hat{u}_j(k)) \Big\}, \tag{8.57}$$

with $\tilde{d}_i(k)$ bounded by the set,

$$\tilde{d}_i(k) \in \mathbb{D}_i \coloneqq \bigoplus_{j \in \mathcal{N}_i} \Big\{ \mathcal{A}_{ij}\mathbb{M}_j \oplus \mathcal{B}_{ij}\mathbb{N}_j \Big\}. \tag{8.58}$$

Similarly, for the error (8.55b), $\hat{f}_i(k)$ is a known disturbance on the error which is computed from the shared trajectories and given by

$$\hat{f}_i(k) = \sum_{j\in\mathcal{N}_i} \Big\{\mathcal{C}_{ij}\hat{\epsilon}_j(k) + \mathcal{D}_{ij}\delta\hat{u}_j(k)\Big\} \tag{8.59}$$

and $\tilde{f}_i(k)$ is an unknown but bounded disturbance on the error resulting from the deviation between the planned and actual reference trajectories, and is given by

$$\tilde{f}_i(k) = \sum_{j\in\mathcal{N}_i} \Big\{\mathcal{C}_{ij}(\epsilon_j(k) - \hat{\epsilon}_j(k)) + \mathcal{D}_{ij}(\delta u_j(k) - \delta\hat{u}_j(k))\Big\}, \tag{8.60}$$

with $\tilde{f}_i(k)$ bounded by the set,

$$\tilde{f}_i(k) \in \mathbb{H}_i \coloneqq \bigoplus_{j\in\mathcal{N}_i} \Big\{\mathcal{C}_{ij}\mathbb{M}_j \oplus \mathcal{D}_{ij}\mathbb{N}_j\Big\}. \tag{8.61}$$

Nominal models and control law

To formulate the control problem, in line with the tube-based robust MPC design philosophy, we will neglect the unknown disturbances in (8.55) and define the following nominal decentralized velocity dynamics:

$$\bar{\epsilon}_i(k+1) = \mathcal{A}_{ii}\bar{\epsilon}_i(k) + \mathcal{B}_{ii}\delta\bar{u}_i(k) + \hat{d}_i(k), \tag{8.62a}$$

$$\bar{e}_i(k) = \mathcal{C}_{ii}\bar{\epsilon}_i(k) + \mathcal{D}_{ii}\delta\bar{u}_i(k) + \hat{f}_i(k). \tag{8.62b}$$

To facilitate the tube based DFOMPC design, we will rewrite the actual and nominal velocity dynamics, respectively, in terms of the extended state variable $\begin{bmatrix}\epsilon_i(k)\\ r_i\end{bmatrix}$ as

$$\begin{bmatrix}\epsilon_i(k+1)\\ r_i\end{bmatrix} = \tilde{\mathcal{A}}_{ii}\begin{bmatrix}\epsilon_i(k)\\ r_i\end{bmatrix} + \tilde{\mathcal{B}}_{ii}\delta u_i(k) + \hat{\mathcal{E}}_i\hat{d}_i(k) + \tilde{\mathcal{E}}_i\tilde{d}_i(k) \tag{8.63}$$

and

$$\begin{bmatrix}\bar{\epsilon}_i(k+1)\\ \bar{r}_i\end{bmatrix} = \tilde{\mathcal{A}}_{ii}\begin{bmatrix}\bar{\epsilon}_i(k)\\ \bar{r}_i\end{bmatrix} + \tilde{\mathcal{B}}_{ii}\delta\bar{u}_i(k) + \hat{\mathcal{E}}_i\hat{d}_i(k), \tag{8.64}$$

where

$$\tilde{\mathcal{A}}_{ii} := \begin{bmatrix} \mathcal{A}_{ii} & \mathbf{0}_{(n_{xi}+n_{yi})\times n_{yi}} \\ \mathbf{0}_{n_{yi}\times(n_{xi}+n_{yi})} & I_{n_{yi}} \end{bmatrix},\ \tilde{\mathcal{B}}_{ii} := \begin{bmatrix} \mathcal{B}_{ii} \\ \mathbf{0}_{n_{yi}\times n_{ui}} \end{bmatrix},\ \tilde{\mathcal{E}}_{i} := \begin{bmatrix} I_{(n_{xi}+n_{yi})\times n_{wi}} \\ \mathbf{0}_{n_{yi}\times n_{wi}} \end{bmatrix}. \tag{8.65}$$

Let

$$\varepsilon_i(k) = \begin{bmatrix} \epsilon_i(k) \\ r_i \end{bmatrix} - \begin{bmatrix} \bar{\epsilon}_i(k) \\ \bar{r}_i \end{bmatrix}, \tag{8.66}$$

and according to [154] assume that, for all k, for the actual extended velocity dynamics (8.63) the following control law is considered

$$\delta u_i(k) = \delta \bar{u}_i(k) - \tilde{\mathcal{K}}_i \varepsilon(k) \tag{8.67}$$

where the gain $\tilde{\mathcal{K}}_i$ is defined a priori such that $\tilde{\mathcal{F}}_{ii} = \tilde{\mathcal{A}}_{ii} - \tilde{\mathcal{B}}_{ii}\tilde{\mathcal{K}}_i$ is Schur.

Remark 47. *To obtain $\tilde{\mathcal{K}}_i$ such that $\tilde{\mathcal{F}}_{ii}$ is Schur, we can define $\tilde{\mathcal{K}}_i$ as*

$$\tilde{\mathcal{K}}_i = \begin{bmatrix} \mathcal{K}_i & \mathbf{0}_{n_{ui}\times n_{yi}} \end{bmatrix} \tag{8.68}$$

where $\mathcal{K}_i$ is the chosen such that Assumption 21 is satisfied.

From (8.63), (8.64) and (8.67), it directly follows that

$$\varepsilon_i(k+1) = \tilde{\mathcal{F}}_{ii}\varepsilon_i(k) + \tilde{\mathcal{E}}_i \tilde{d}_i(k) \tag{8.69}$$

where $\tilde{d}_i(k) \in \mathbb{D}_i$. Since $\mathbb{D}_i$ is bounded and $\tilde{\mathcal{F}}_{ii}$ is Schur, there exists a robust positively invariant (RPI) set $\tilde{\mathbb{E}}_i$ for (8.69) such that, for all $\varepsilon_i(k) \in \tilde{\mathbb{E}}_i$, then $\varepsilon_i(k+1) \in \tilde{\mathbb{E}}_i$. Also, $\delta u_i(k) - \delta\bar{u}_i(k) \in \tilde{\mathcal{K}}_i\tilde{\mathbb{E}}_i$, for all $k \geq 0$.

Remark 48. *In view of [179], it may be possible to compute an invariant outer-approximation to the minimal RPI set $\tilde{\mathbb{E}}_i$ for the error dynamics* (8.69) *as defined in* (8.70) *below.*

$$\tilde{\mathbb{E}}_i = \bigoplus_{i=0}^{\infty} \tilde{\mathcal{F}}_{ii}^{i}\tilde{\mathcal{E}}_i\mathbb{D}_i. \tag{8.70}$$

From the set $\tilde{\mathbb{E}}_i$, we can obtain the set $\mathbb{E}_i$ which is a RPI set on $\epsilon_i - \bar{\epsilon}_i$ defined as:

$$\mathbb{E}_i := \left\{ \epsilon_i - \bar{\epsilon}_i \colon \varepsilon_i \in \tilde{\mathbb{E}}_i \right\}. \tag{8.71}$$

An algorithm for computing a polytopic outer approximation to (8.70) *can be devised in line with [179].*

Constraint reformulation and terminal constraint

To enforce the inequality constraints, the constraints on u_i and y_i must be reformulated in terms of the constraints on the MPC optimization variables $\delta\bar{u}_i$ and $\bar{\epsilon}_i$. To this end, similarly to [28], we write the relation between these variables as (see section 7.6 for details)

$$\begin{bmatrix} x_i(k) \\ u_i(k-1) \end{bmatrix} = \tilde{\mathcal{C}}_i \begin{bmatrix} \epsilon_i(k) \\ r_i \end{bmatrix} + \tilde{\mathcal{D}}_i q_i(k-1) \tag{8.72}$$

where $q_i(k) = \begin{bmatrix} \tilde{w}_i(k) \\ \tilde{p}_i(k) \end{bmatrix} \in (\tilde{\mathbb{W}}_i \times \tilde{\mathbb{P}}_i)$, $\tilde{w}_i(k) = E_{ii}w_i(k) + \sum_{j\in\mathcal{N}_i} \Big\{ A_{ij}x_j(k) + B_{ij}u_j(k) \Big\}$, $\tilde{p}_i(k) = \sum_{j\in\mathcal{N}_i}(-\gamma_{ij})\Big[\Lambda_{yj}y_j(k) + \Lambda_{uj}u_j(k) + r_j\Big]$, and,

$$\begin{aligned} \tilde{\mathcal{C}}_i &= \begin{bmatrix} A_{ii} & B_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} & I_{n_{ui}} \end{bmatrix} S_i^{-1} \begin{bmatrix} I_{n_{xi}} & \mathbf{0}_{n_{xi}\times n_{yi}} & \mathbf{0}_{n_{xi}\times n_{yi}} \\ \mathbf{0}_{n_{yi}\times n_{xi}} & I_{n_{yi}} & -I_{n_{yi}} \end{bmatrix}, \\ \tilde{\mathcal{D}}_{\tilde{w}i} &= \begin{bmatrix} I_{n_{xi}\times n_{wi}} \\ \mathbf{0}_{n_{ui}\times n_{xi}} \end{bmatrix} - \begin{bmatrix} A_{ii} & B_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} & I_{n_{ui}} \end{bmatrix} S_i^{-1} \begin{bmatrix} I_{n_{xi}\times n_{wi}} \\ \mathbf{0}_{n_{yi}\times n_{xi}} \end{bmatrix}, \\ \tilde{\mathcal{D}}_{\tilde{p}i} &= -\begin{bmatrix} A_{ii} & B_{ii} \\ \mathbf{0}_{n_{ui}\times n_{xi}} & I_{n_{ui}} \end{bmatrix} S_i^{-1} \begin{bmatrix} \mathbf{0}_{n_{xi}\times n_{yi}} \\ I_{n_{yi}} \end{bmatrix}, \text{and } \tilde{\mathcal{D}}_i = \begin{bmatrix} \tilde{\mathcal{D}}_{\tilde{w}i} & \tilde{\mathcal{D}}_{\tilde{p}i} \end{bmatrix}, \end{aligned} \tag{8.73}$$

and set $\tilde{\mathbb{W}}_i$ and $\tilde{\mathbb{P}}_i$ are defined as follows:

$$\begin{aligned} \tilde{\mathbb{W}}_i &:= \bigoplus_{j\in\mathcal{N}_i} \Big\{ A_{ij}\mathbb{X}_j \oplus B_{ij}\mathbb{U}_j \Big\} \oplus E_{ii}\mathbb{W}_i, \quad \mathbb{X}_i \subseteq \mathbb{R}^n := \{x_i : y_i \in \mathbb{Y}_i\}, \text{ and} \\ \tilde{\mathbb{P}}_i &:= \bigoplus_{j\in\mathcal{N}_i} (-\gamma_{ij}) \Big\{ \Lambda_{yj}\mathbb{Y}_j \oplus \Lambda_{uj}\mathbb{U}_j \oplus \mathbb{Y}_j \Big\}. \end{aligned} \tag{8.74}$$

Let $\bar{\mathbb{G}}_i$ be the set which enforces the constraints $(u_i(k), y_i(k)) \in (\mathbb{U}_i \times \mathbb{Y}_i)$ on the nominal velocity state $\bar{\epsilon}_i(k)$ for all admissible disturbances, $\tilde{d}_i(k) \in \mathbb{D}_i$, i.e,

$$\bar{\mathbb{G}}_i := \Big\{ \bar{\epsilon}_i : (u_i, y_i) \in (\mathbb{U}_i \times \mathbb{Y}_i), \quad \forall \tilde{d}_i(k) \in \mathbb{D}_i \Big\}. \tag{8.75}$$

Taking the difference between (8.72) and its nominal version i.e.,

$$\begin{bmatrix} \bar{x}_i(k) \\ \bar{u}_i(k-1) \end{bmatrix} = \tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i(k) \\ \bar{r}_i \end{bmatrix} \tag{8.76}$$

we obtain the following relation,

$$\begin{bmatrix} x_i(k) \\ u_i(k-1) \end{bmatrix} = \tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i(k) \\ \bar{r}_i \end{bmatrix} + \tilde{\mathcal{C}}_i \varepsilon_i(k) + \tilde{\mathcal{D}}_i q_i(k-1). \tag{8.77}$$

From the invariance of $\tilde{\mathbb{E}}_i$, $\varepsilon_i(k) \in \tilde{\mathbb{E}}_i$ for all $k \geq 0$. Also, $q_i(k-1) \in (\tilde{\mathbb{W}}_i \times \tilde{\mathbb{P}}_i)$ and $\begin{bmatrix} x_i(k) \\ u_i(k-1) \end{bmatrix} \in (\mathbb{X}_i \times \mathbb{U}_i)$, therefore the following set relationship is obtainable from (8.72):

$$\tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i \\ \bar{r}_i \end{bmatrix} \in (\mathbb{X}_i \times \mathbb{U}_i) \ominus \tilde{\mathcal{D}}_i(\tilde{\mathbb{W}}_i \times \tilde{\mathbb{P}}_i) \ominus \tilde{\mathcal{C}}_i \tilde{\mathbb{E}}_i. \tag{8.78}$$

The set $\bar{\mathbb{G}}_i$ can therefore be defined in terms of $\bar{\epsilon}_i$ as,

$$\bar{\mathbb{G}}_i := \left\{ \bar{\epsilon}_i : \tilde{\mathcal{C}}_i \begin{bmatrix} \bar{\epsilon}_i \\ \bar{r}_i \end{bmatrix} \in (\mathbb{X}_i \times \mathbb{U}_i) \ominus \tilde{\mathcal{D}}_i(\tilde{\mathbb{W}}_i \times \tilde{\mathbb{P}}_i) \ominus \tilde{\mathcal{C}}_i \tilde{\mathbb{E}}_i \right\} \tag{8.79}$$

where $\mathbb{X}_i \subseteq \mathbb{R}^n := \{\bar{x}_i : \bar{y}_i \in \mathbb{Y}_i\}$.
To guarantee that the inputs and outputs $u_i(k)$ and $y_i(k)$, of each subsystem i, respectively lie in the sets $\mathbb{U}_i$ and $\mathbb{Y}_i$, beyond the end of the horizon, N, we need to compute a terminal invariant set, $\bar{\mathbb{G}}_{f,i}$, where $\bar{\epsilon}_i(k)$ must lie in order to guarantee that the constraints $(u_i(k), y_i(k)) \in (\mathbb{U}_i \times \mathbb{Y}_i)$ are verified for all $k \geq N$. Assume the pair $(\mathcal{A}_{ii}, \mathcal{B}_{ii})$ is reachable and the set $\bar{\mathbb{G}}_i$ is a closed polytope (which are guaranteed by virtue of the assumptions made about the distributed system), then the set $\bar{\mathbb{G}}_{f,i}$ can be computed as the (projection of the) maximal constraint admissible set [114] for the nominal velocity dynamics (8.62a) under the auxiliary control law, $\delta\bar{u}_i(k) = -\mathcal{K}_i\bar{\epsilon}_i(k)$ and is constructed such that:

$$\bar{\epsilon}_i \in \bar{\mathbb{G}}_{f,i} \implies \bar{\epsilon}_i \in \bar{\mathbb{G}}_i \text{ and } \mathcal{F}_{ii}\bar{\epsilon}_i \in \bar{\mathbb{G}}_{f,i}. \tag{8.80}$$

Optimal control problem

To compute the tDFOMPC law, $\delta u_i(k)$, we solve at each sampling time, k, following the approach in Chapter 6, M optimal control problems for the nominal velocity dynamics (8.62) using knowledge of the current state $\epsilon_i(0)$ and the future input and output state trajectories for subsystem i and its neighbours $\hat{\epsilon}_j(k)$ and $\delta\hat{u}_j(k)$, $k = 0, \dots, N-1$. Starting from a knowledge of the sets $\mathbb{M}_j$, $\mathbb{N}_j$, $\bar{\mathbb{G}}_i$, $\tilde{\mathbb{E}}_i$ and $\bar{\mathbb{G}}_{f,i}$, the tDFOMPC law is obtained by solving online, the following optimal control problem:

$$\mathbb{P}_{ii}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k))\colon \min_{(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) \in \mathscr{U}_{N,i}} V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) \tag{8.81}$$

where the feasible region $\mathscr{U}_{N,i}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k))$ is defined as

$$\mathscr{U}_{N,i}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k)) \triangleq$$
$$\left\{ (\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) \middle| \begin{aligned} \bar{\epsilon}_i(k+t+1) = \mathcal{A}_{ii}\bar{\epsilon}_i(k+t) + \mathcal{B}_{ii}\delta\bar{u}_i(k+t) + \hat{d}_i(k+t),\ \forall t \in \mathbb{I}_{[0,N-1]} \\ \bar{e}_i(k+t) = \mathcal{C}_{ii}\bar{\epsilon}_i(k+t) + \mathcal{D}_{ii}\delta\bar{u}_i(k+t) + \hat{f}_i(k+t),\ \forall t \in \mathbb{I}_{[0,N-1]} \\ \epsilon_i(k) - \bar{\epsilon}_i(k) \in \mathbb{E}_i \\ \bar{\epsilon}_i(k+t) - \hat{\epsilon}_i(k+t) \in \mathbb{M}_i \ominus \mathbb{E}_i,\ \forall t \in \mathbb{I}_{[0,N-1]} \\ \delta\bar{u}_i(k+t) - \delta\hat{u}_i(k+t) \in \mathbb{N}_i \ominus \mathcal{K}_i\mathbb{E}_i,\ \forall t \in \mathbb{I}_{[0,N-1]} \\ \bar{\epsilon}_i(k+t) \in \bar{\mathbb{G}}_i,\ \forall t \in \mathbb{I}_{[0,N-1]} \\ \bar{\epsilon}_i(k+N) \in \bar{\mathbb{G}}_{f,i} \end{aligned} \right\} \tag{8.82}$$

and the cost function $V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k))$ is given by

$$V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k)) = V_{f,i}(\bar{\epsilon}_i(k+N)) + \sum_{t=0}^{N-1} l_i(\bar{e}_i(k+t), \delta\bar{u}_i(k+t)) \tag{8.83}$$

where,

$$l_i(\bar{e}_i(k+t), \delta\bar{u}_i(k+t)) := \frac{1}{2}\Big(\bar{e}_i(k+t)^\top Q_{e,i}\bar{e}_i(k+t) + \delta\bar{u}_i(k+t)^\top R_{\delta,i}\delta\bar{u}_i(k+t)\Big) \tag{8.84}$$

and

$$V_{f,i}(\bar{\epsilon}_i(k+N)) := \frac{1}{2}\bar{\epsilon}_i(k+N)^\top \mathcal{P}_i\bar{\epsilon}_i(k+N). \tag{8.85}$$

Here we assume at the current sampling time,k, the velocity state $\epsilon_i(k)$ is measurable from the local input/output, and the shared signal, $\hat{v}_{-i}$:

$$\epsilon_i(k) = \begin{bmatrix} \delta x_i(k) \\ \bar{e}_i(k-1) \end{bmatrix} = \begin{bmatrix} x_i(k) - x_i(k-1) \\ \Lambda_{y,i} y_i(k-1) + \Lambda_{ui} u_i(k-1) + r_i + \hat{v}_{-i} \end{bmatrix} \tag{8.86}$$

where $\hat{v}_{-i} = \sum_{j\in\mathcal{N}_i}(-\gamma_{ij})\Lambda_{yj}\hat{y}_j(k) + \Lambda_{uj}\hat{u}_j(k) + r_j$ is obtained via communication from neighbouring subsystems.
With regard to the OCP, $\mathbb{P}_{ii}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k))$ above, the following remarks are in order:

Remark 49. *1. To guarantee positive definiteness of the performance objective $V_{N,i}(\bar{\epsilon}_i(k), \delta\bar{\mathbf{u}}_i(k))$, the matrices $Q_{e,i} \succeq 0$ and $R_{\delta,i} \succ 0$ are chosen such that $l_i(\bar{e}_i(k+t), \delta\bar{u}_i(k+t)) \geq 0$ with $\mathcal{P}_i \succ 0$.*

2. To guarantee closed-loop convergence, the matrix $\mathcal{P}_i \succ 0$ is selected via the terminal control law

$$\delta\bar{u}_i(k+N) = -\mathcal{K}_i\bar{\epsilon}_i(k+N) \tag{8.87}$$

such that $V_{f,i}(\bar{\epsilon}_i(k+N))$ is a control Lyapunov function i.e.,

$$V_{f,i}(\bar{\epsilon}_i(k+N+1)) - V_{f,i}(\bar{\epsilon}_i(k+N)) \leq -l_i(\bar{e}_i(k+N), -\mathcal{K}_i\bar{\epsilon}_i(k+N)),\ \forall\bar{\epsilon}_i(k+N) \in \bar{\mathbb{G}}_{f,i}. \tag{8.88}$$

3. The optimal control problem, $\mathbb{P}_{ii}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k))$ does not yield control laws that track the global optimum of the centralized steady-state optimization problem, (8.6)*. Instead, the resulting control law regulates the system,* (8.1) *to an invariant neighbourhood of such a solution. How conservative this approximation is depebds on the size of the RPI set $\mathbb{E}_i$. Making $\mathbb{E}_i$ as small as possible (e.g. using the minimal RPI set instead) can improve the accuracy of the steady-state optimal solutions.*

The solution, $(\bar{\epsilon}_i^*(k), \delta\mathbf{u}_i^*(k))$ of the OCP $\mathbb{P}_{ii}(\epsilon_i(k), \hat{d}_i(k), \hat{f}_i(k))$ above, followed by the application of the first control, $\delta\bar{u}_i^*(k)$ in the optimized sequence, yield the control law

$$\delta u_i(k) = \delta\bar{u}_i^*(k) - \mathcal{K}_i\Big(\epsilon_i(k) - \bar{\epsilon}_i^*(k)\Big), \tag{8.89}$$

and the corresponding optimal cost $V_{N,i}^0(\epsilon_i(k))$ which is also referred to as the value function.

Lastly, letting $\bar{\epsilon}_i(k+t)$ be the trajectory stemming from $(\bar{\epsilon}_i^*(k), \delta\mathbf{u}_i^*(k))$, and the prediction dynamics (8.62), the reference trajectories to be used in the next time instant $k+1$ are incrementally updated by appending the values

$$\hat{\epsilon}_i(k+N) = \bar{\epsilon}_i(k+N), \text{ and } \delta\hat{u}_i = -\mathcal{K}_i\bar{\epsilon}_i(k+N) \tag{8.90}$$

to the reference trajectories previously defined for $k+t \leq k+N-1$.

Convergence results

In the following presentation, the notation $x(1:N)$ is the column vector

$$x(1:N) = \begin{bmatrix} x(1) \\ \vdots \\ x(N) \end{bmatrix}. \tag{8.91}$$

To prove the convergence properties of the proposed algorithm, we make use of results from [70]. First the set of admissible initial conditions $\boldsymbol{\epsilon}(k) = (\epsilon_1(k), \ldots, \epsilon_M(k))$ and initial reference trajectories $\hat{\epsilon}_j(k:k+N-1)$, $\hat{\delta}u_j(k:k+N-1)$, for all $j = 1, \ldots, M$ are defined as follows.

Definition 5. *Letting* $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_M)$*, we define the feasibility region,* Ξ_N *for all the* $\mathbb{P}_{ii}$ *optimal control problems as the set*

$$\begin{aligned}
\Xi_N \triangleq \Bigg\{ \boldsymbol{\epsilon}\colon \text{ if } \epsilon_i(k) = \epsilon_i \text{ for all } i = 1, \ldots, M \text{ then} \\
\exists \begin{bmatrix} \hat{\epsilon}_1(k:k+N-1) \\ \vdots \\ \hat{\epsilon}_M(k:k+N-1) \end{bmatrix}, \begin{bmatrix} \hat{\delta} u_1(k:k+N-1) \\ \vdots \\ \hat{\delta} u_M(k:k+N-1) \end{bmatrix}, \\
\begin{bmatrix} \bar{\epsilon}_1(k) \\ \vdots \\ \bar{\epsilon}_M(k) \end{bmatrix}, \begin{bmatrix} \bar{\delta} u_1(k:k+N-1) \\ \vdots \\ \bar{\delta} u_M(k:k+N-1) \end{bmatrix} \\
\text{such that (8.23) and the following constraints :} \\
\epsilon_i(k) - \bar{\epsilon}_i(k) \in \mathbb{E}_i \\
\bar{\epsilon}_i(k) - \hat{\epsilon}_i(k) \in \mathbb{M}_i \ominus \mathbb{E}_i, \\
\delta\bar{u}_i(k) - \delta\hat{u}_i(k) \in \mathbb{N}_i \ominus \mathcal{K}_i \mathbb{E}_i, \\
\bar{\epsilon}_i(k) \in \bar{\mathbb{G}}_i, \\
\bar{\epsilon}_i(k+N) \in \bar{\mathbb{G}}_{f,i} \\
\text{are satisfied for all } i = 1, \ldots, M \Bigg\}.
\end{aligned} \tag{8.92}$$

We also denote, for each $\boldsymbol{\epsilon} \in \Xi_N$*, the region of feasible initial reference trajectories as*

$$\hat{\boldsymbol{\Xi}}_N \triangleq \left\{ \begin{bmatrix} \hat{\epsilon}_1(k:k+N-1) \\ \vdots \\ \hat{\epsilon}_M(k:k+N-1) \end{bmatrix}, \begin{bmatrix} \hat{\delta u}_1(k:k+N-1) \\ \vdots \\ \hat{\delta u}_M(k:k+N-1) \end{bmatrix} : \begin{array}{r} \text{if } \epsilon_i(k)=\epsilon_i \text{ for all } i=1,\dots,M \text{ then} \\ \exists \begin{bmatrix} \bar{\epsilon}_1(k) \\ \vdots \\ \bar{\epsilon}_M(k) \end{bmatrix}, \begin{bmatrix} \bar{\delta u}_1(k:k+N-1) \\ \vdots \\ \bar{\delta u}_M(k:k+N-1) \end{bmatrix} \\ \text{such that (8.23) and the following constraints :} \\ \epsilon_i(k)-\bar{\epsilon}_i(k) \in \mathbb{E}_i \\ \bar{\epsilon}_i(k)-\hat{\epsilon}_i(k) \in \mathbb{M}_i \ominus \mathbb{E}_i, \\ \delta\bar{u}_i(k)-\delta\hat{u}_i(k) \in \mathbb{N}_i \ominus \mathcal{K}_i\mathbb{E}_i, \\ \bar{\epsilon}_i(k) \in \bar{\mathbb{G}}_i, \\ \bar{\epsilon}_i(k+N) \in \bar{\mathbb{G}}_{f,i} \\ \text{are satisfied for all } i=1,\dots,M \end{array} \right\}. \tag{8.93}$$

Assumption 25. *Letting*

$$\mathbb{E} = \prod_{i=1}^{M} \mathbb{E}_i,\ \ \bar{\mathbb{G}} = \prod_{i=1}^{M} \bar{\mathbb{G}}_i,\ \text{ and } \bar{\mathbb{G}}_f = \prod_{i=1}^{M} \bar{\mathbb{G}}_{f,i} \tag{8.94}$$

it holds that

(i) Assumption 21 is satisfied.

(ii) $\bar{\mathbb{G}}_f \subseteq \bar{\mathbb{G}}$ *is an invariant set for* $\bar{\boldsymbol{\epsilon}}(k+1) = (\mathcal{A}-\mathcal{B}\mathcal{K})\bar{\boldsymbol{\epsilon}}$.

(iii) $\delta\bar{\mathbf{u}} = -\mathcal{K}\bar{\boldsymbol{\epsilon}}$ *for any* $\bar{\boldsymbol{\epsilon}} \in \bar{\mathbb{G}}_f$.

(iv) for all $\bar{\boldsymbol{\epsilon}} \in \bar{\mathbb{G}}_f$,

$$\mathbf{V}_f(\bar{\boldsymbol{\epsilon}}(k+1)) - \mathbf{V}_f(\bar{\boldsymbol{\epsilon}}(k)) \le -\mathbf{l}(\bar{\mathbf{e}}(k), \delta\bar{\mathbf{u}}), \tag{8.95}$$

where

$$\mathbf{V}_f(\bar{\boldsymbol{\epsilon}}(k)) = \sum_{i=1}^{M} V_{f,i}(\bar{\epsilon}_i(k)) \text{ and } \mathbf{l}(\bar{\mathbf{e}}(k), \delta\bar{\mathbf{u}}) = \sum_{i=1}^{M} l_i(\bar{e}_i(k), \delta\bar{u}_i(k)). \tag{8.96}$$

(v) The constraint $\bar{\boldsymbol{\epsilon}} \in \bar{\mathbb{G}}$ *is satisfied.*

Assumption 25 (ii) and 25 (iv) are standard in stabilizing MPC algorithms, while Assumption 25 (v) ensures the velocity dynamics satisfies the constraints on $\mathbf{x}$ and $\mathbf{u}$.

Assumption 26. *Given the sets* $\mathbb{M}_i$*,* $\mathbb{N}_i$ *and the RPI sets* $\mathbb{E}_i$*, there exists a real positive constant* $\bar{\rho}_E > 0$ *such that* $\mathbb{E}_i \oplus \mathscr{B}_{\bar{\rho}_E}(0) \subseteq \mathbb{M}_i$ *and* $\mathcal{K}_i \mathbb{E}_i \oplus \mathscr{B}_{\bar{\rho}_E}(0) \subseteq \mathbb{N}_i$ *for all* $i = 1, \ldots, M$ *where* $\mathscr{B}_{\bar{\rho}_E}(0)$ *is a ball of radius* $\bar{\rho}_E > 0$ *centred at the origin.*

We now state the main convergence result;

Theorem 8.3.2. *Let the Assumption 25 and 26 be satisfied and let* $\mathbb{M}_i$ *and* $\mathbb{N}_i$ *be neighbourhoods of the origin. Then, for any initial reference trajectories in* $\hat{\boldsymbol{\Xi}}_N$*, the states* $\boldsymbol{\epsilon}(k+t)$*, starting from any initial condition* $\boldsymbol{\epsilon}(k) \in \boldsymbol{\Xi}_N$*, asymptotically converges to the origin, and as a result the inputs and outputs* (u, y) *of the large scale system* (8.1) *converges to a steady-state equilibrium that approximates the optimal solution* $z^*(w)$ *of the steady-state optimization problem* (8.6) *by a margin of error that depends on the size of the RPI set* $\mathbb{E}$*.*

Proof. The proof can be easily adapted from [70] to the velocity formulation used in this chapter. □

8.4 Conclusion

We have presented distributed model predictive control algorithms for the feedback optimization of large scale linear systems with unknown disturbances, and a quadratic steady-state optimization problem. To avoid the excessive communication common to iterative approaches, a non-iterative robust MPC approach was adopted. Theoretical analysis have shown that the developed algorithms converge to a neighbourhood of the optimal steady-state solution. Future work will focus on improving the performance of the algorithms, relaxing the requirement for constant disturbances and illustrating the algorithms with applications.

Chapter 9

Feedback Optimizing Model Predictive Load-Frequency Control for Real-time Economic Dispatch

In this chapter, a feedback optimizing model predictive load-frequency control (FOMPLFC) algorithm will be developed for real-time economic dispatch in power systems. We begin the chapter by reviewing the problem of real-time economic dispatch in load-frequency control. This problem is also referred to as the optimal load-frequency control problem. We then go on to present a detailed frequency control dynamics for a power system network. Next we formulate the optimal load frequency control problem in the language of feedback optimizing MPC and then proceed to develop the associated solution to the problem by applying the FOMPC algorithms developed in earlier chapters. Lastly, the performance of the controller is evaluated using numerical examples and comparisons to state-of-the-art control algorithms are made.

This chapter is structured as follows: Section 9.2 presents the multi-area power system model and describes the economic dispatch problem. Section 9.3 formulates the problem of real-time economic dispatch as a feedback-optimizing model predictive LFC (FOMPLFC) problem. Section 9.4 presents a solution to the FOMPLFC problem with the associated theoretical guarantees discussed. In section 9.5, numerical simulation studies of the FOMPLFC controller and other state-of-the-art LFC control algorithms are discussed using a two-area power system example. Section 9.6 concludes the chapter.

9.1 Introduction

The foremost objective in a power system is the reliable and cost-efficient supply of power to meet projected demand. Conventional power systems utilize a combination of load-frequency control (LFC) and economic dispatch (ED) operating at different timescales to achieve this objective. While ED numerically computes economically optimal set-points from demand projections, LFC tracks these set-points using feedback control. ED operates at slower timescales compared to the real-time operation of LFC. Due to the uncertainty and variability of demand, the set-points computed by ED may be suboptimal. In traditional power systems, the demand is predictable with small fluctuations that only occur at the frequency control timescale. Therefore, rejecting this fluctuation using load-frequency control results in economically optimal power system operation using the traditional hierarchical frequency control structure [106, 22].
With increased penetration of renewable generation and demand-responsive loads, future power systems will have reduced inertia, with faster, larger, and less predictable power fluctuations (uncontrollable load minus intermittent generation). Also, these fluctuations may occur across both the frequency control and economic dispatch timescales [165]. Consequently, multi-time scale dynamics are introduced into the power system, making it difficult to separate economic dispatch from real-time frequency control. The operation of ED and LFC at different timescales may therefore no longer guarantee economically efficient power system operation. As a result, novel control schemes that simultaneously achieve economic dispatch and frequency regulation without timescale separation are required.

In practice, this problem has been addressed by performing the economic dispatch computations closer to the timescale of frequency control. However, this approach can be computationally intensive, lacking in robustness, and may not guarantee the stability of the closed-loop power system [88, 37].
Recently, the interest in developing control schemes that simultaneously satisfy the objectives of ED and LFC without timescale separation or explicit numerical solution to an ED problem has increased dramatically [55, 160]. This interest has been chiefly motivated by the evolution of power systems into intelligent grids with increased renewable generation, responsive demand, and a deregulated system operation. Indeed, the field of optimal load-frequency control has recently emerged to classify all control schemes addressing this issue. In recent studies, the goal has been to autonomously solve the economic dispatch problem using the feedback loop of the load frequency

controllers without explicit numerical computation [55, 160]. For most of the proposed solutions, autonomous economic dispatch is achieved in the frequency control loop by the direct implementation of the optimization algorithms for solving the economic dispatch problem as dynamic controllers [55, 160]. One limitation of these approaches is that the power system dynamics are not considered in the control design, assuming a pre-stabilized power system. Therefore, the only dynamics considered by these studies are those induced by the online optimization algorithms. With sufficient timescale separation, these controllers have worked perfectly [156, 88]. However, as the timescale between the optimization dynamics and the system dynamics decreases, it has been shown in [88] that the stability of the closed-loop system can become compromised. In particular, sudden changes might lead to drastic fluctuation in the frequency dynamics, causing poor dynamic performance, economic losses, or even loss of closed-loop stability.

There is currently very little literature on control algorithms that simultaneously achieve economic dispatch and frequency regulation without explicit computation of the ED set-points while guaranteeing feasibility and good transient performance. In the following, we review recent efforts at addressing this gap.

The authors in [167, 128, 129] developed a feedback control algorithm that steers an LTI system to the optimum of a steady-state optimization problem (e.g., economic dispatch). The control algorithm provided closed-loop stability guarantees without resorting to timescale separation assumptions. However, the authors did not consider the dynamic performance or constraints satisfaction in transient operation. To address this gap, [113] presented an LFC algorithm for a multi-area power system based on distributed economic model predictive control. Although convergence to the economic dispatch solution was guaranteed, the algorithm was computationally expensive. It also required explicit estimation of the unknown disturbances. Also, stability guarantees required passivity assumptions on the power system dynamics (which is hard to satisfy for power systems with second-order turbine-governor dynamics [211]). It is worth noting that the passivity assumptions made in [113] were used to guarantee closed-loop stability in the economic MPC scheme and did not make the algorithm plug and play. In [201], the author proposed an economic optimizing model predictive control algorithm based on a multi-objective function approach incorporating both the economic dispatch and the frequency control performance objectives. The algorithm, however, suffers from a trade-off between economic optimality and dynamic performance. Also, the algorithm provided no stability and recursive feasibility guarantees. It is clear from the literature that opportunities exist for the design of load-frequency control algorithms that:

1. are both frequency stabilizing and economically dispatching,

2. do not explicitly compute the economic dispatch set-points,

3. achieve optimal dynamic performance with recursive feasibility for the inequality constraints, and

4. easily establish closed-loop performance guarantees.

We propose a load-frequency control algorithm based on feedback-optimizing MPC to address the above gaps in this chapter. The proposed controller is optimal (in the transient performance), recursively feasible and achieves economic dispatch autonomously via the feedback loop of the load-frequency controller. No timescale separation or explicit online economic optimization is required. We formulate our MPC algorithm by replacing the tracking error in a standard tracking MPC algorithm with the residual of the Karush–Kuhn–Tucker (KKT) optimality conditions of the ED problem. Our MPC-based LFC algorithm is guaranteed to converge to the optimum of the ED problem without advanced knowledge of the load disturbances or numerically solving the ED problem. Also, the algorithm is guaranteed to satisfy the power system inequality constraints (thermal line and capacity limits). In the absence of hard inequality constraints, the controller reduces to a static optimal control law (e.g., linear quadratic control laws), altogether eliminating online optimization. Furthermore, a vital benefit of the proposed controller compared to online dynamic economic optimal control (e.g., Economic MPC) is that tracking the economic dispatch set-points is possible without measuring or estimating the disturbances.

9.2 Power System Model, Load Frequency Control and Economic Dispatch

9.2.1 Multi-Area Power System Model [121, 22]

Consider a transmission level network with arbitrary topology described by a weighted directed graph $\mathscr{G} = (\mathcal{N}, \mathscr{E}, \mathscr{W})$, where $\mathcal{N} = \{1, \ldots, i, \ldots N\}$ is the set of nodes or control areas and $\mathscr{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of edges or tie-line interconnection between the nodes. The set $\mathscr{W}$ contains the weights of the edges $\mathscr{E}$ i.e $\mathscr{W} = \{T_{ij},\ \forall(i,j) \in \mathscr{E}\}$ where T_{ij} $[pu\ MW/Hz]$ is the synchronizing coefficient of the edge/tie-line $(i,j) \in \mathscr{E}$. The neighbourhood set of the node i is denoted by $\mathscr{N}_i$. For the transmission graph $\mathscr{G}$, the graph Laplacian $\mathscr{L}_{\mathscr{G}}$ contains the topology information in $\mathscr{G}$. For a connected graph, the reduced graph Laplacian $\mathscr{L}_{\mathscr{G}}^r$ contains as much information as the Laplacian while being nonsingular. Each node is assumed to have a single generator, demand responsive load, and uncontrollable load.

Assume that the power network is working around a nominal equilibrium which is determined by an ED problem at a slower timescale. As common in power system control at the transmission level, we make the following assumptions [22].

Assumption 27. *The voltage magnitudes are fixed at all nodes, i.e.* $v_i\ \forall i \in \mathscr{N}$ *are constant. The transmission lines are of negligible resistance and reactive power injections and flows are omitted.*

With Assumption 27 in place, the power network is modelled using the swing dynamics [22]. For all $i \in \mathscr{N}$,

$$H_i \dot{f}_i = P_i^m - P_i^{dr} - D_i f_i - P_i^{tie} - w_i, \tag{9.1a}$$

where f_i $[Hz]$, P_i^m $[pu\ MW]$, P_i^{dr} $[pu\ MW]$ and w_i $[pu\ MW]$ are respectively the deviation in frequency, mechanical power output, controllable demand, and the net load (uncontrollable load minus intermittent generation) in node i. The net tie-line power flow dynamics for node i is given by

$$\dot{P}_i^{tie} = \sum_{j \in \mathscr{N}_i} T_{ij}(f_i - f_j) \tag{9.2}$$

where P_i^{tie} is the power flow on the tie-line $(i,j) \in \mathscr{E}$.

Remark 50 (Accuracy of swing dynamics)**.** *The swing equation in* (9.1) *is a popularly adopted model used to approximate the actual non-linear power system dynamics when the power system is operating around the vicinity of a previous economic dispatch setpoint. This setpoint is usually obtained by solving a more complex alternating current optimal power flow (ACOPF) problem at a much slower timescale. In [38], the validity of the swing equation as a good approximation of the power system frequency dynamics was investigated and it was observed that for small deviation from the ACOPF operating points, the swing equation accurately captures the dominant frequency dynamics of the power system under load changes.*

Turbine Dynamics

The turbine dynamics in the i^{th} node is given for a non-reheat turbine by,

$$\dot{P}_i^m = -\frac{1}{\tau_{t,i}} P_i^m + \frac{1}{\tau_{t,i}} P_i^v \tag{9.3}$$

where $\tau_{t,i}$ and P_i^v $[pu\ MW]$ are the turbine charging time constant and the change in governor valve position for the generator in node i respectively.

Governor Dynamics

The governor dynamics in the i^{th} node is,

$$\dot{P}_i^v = -\frac{1}{\tau_{v,i}} P_i^v + \frac{1}{\tau_{v,i}} u_i^m - \frac{1}{\tau_{v,i} r_i} f_i. \tag{9.4}$$

Here r_i $[Hz/pu\ MW]$ and τ_i^v are respectively the droop coefficient and speed-governor time constant of the generator in the i^{th} node/control area. The control input u_i^m is traditionally obtained from the load-reference set-point u_i for the i^{th} control area via the generation participation factor α_i^m i.e.,

$$u_i^m = \alpha_i^m u_i. \tag{9.5}$$

Demand Response Dynamics

Demand response (DR) has been identified as a cost efficient alternative to generator based spinning reserve. To simplify the integration of DR into the grid, load aggregation companies (LAGCOs) manage and combine several DR capacity into a single aggregate DR load. To achieve demand response in LFC, the grid operator utilizes the demand set-point command, u_i^{dr} to modify the demand of LAGCOs in node i. Load aggregation allows the demand response dynamics to be approximated for small time delays with a first order model [176, 192] as shown below.

$$P_i^{dr}(t + \tau_i^{dr}) = u_i^{dr} \tag{9.6}$$

where τ_i^{dr} is the LAGCO time constant and is usually due to delays in computation and communication. Using Taylor's series, we can express (9.6) as,

$$P_i^{dr}(t+\tau_i^{dr}) = P_i^{dr}(t) + \tau_i^{dr}\dot{P}_i^{dr}(t) + \sum_{q\in[2,\infty]} \frac{(\tau_i^{dr})^q}{q!}\frac{d^q\{P_i^{dr}(t)\}}{dt^q} \tag{9.7}$$

and for small time delays i.e $\tau_i^{dr} \ll 1$,

$$\lim_{q\to\infty} \sum_{q\in[2,\infty]} \frac{(\tau_i^{dr})^q}{q!}\frac{d^q\{P_i^{dr}(t)\}}{dt^q} = \mathbf{0}. \tag{9.8}$$

Therefore (9.6) is approximated by the first order dynamics

$$\dot{P}_i^{dr} = -\frac{1}{\tau_i^{dr}}P_i^{dr} + \frac{1}{\tau_i^{dr}}u_i^{dr}. \tag{9.9}$$

The control input u_i^{dr} is derived from the load-reference set-point u_i for the i^{th} control area via the demand participation factor α_i^{dr} i.e.,

$$u_i^{dr} = \alpha_i^{dr}u_i. \tag{9.10}$$

Area Control Error (ACE)

To restore both the frequency and inter-area power flows in a node to their respective nominal values, it is conventional to regulate the area control error (ACE) to zero. The ACE for node i is given by,

$$ACE_i = \beta_i f_i + P_i^{tie},\ \forall i \in \mathcal{N} \tag{9.11}$$

where β_i is the frequency bias coefficient of node i. The measurable output, $y \in \mathbb{R}^p$ of the power network consists of the area control error, ACE_i for all nodes in the power network.

System constraints

The power system network is subject to the following inequality constraints on the generation, demand response and line flows:

$$\underline{P}_i^m \leq P_i^m \leq \overline{P}_i^m, \tag{9.12a}$$

$$\underline{P}_i^{dr} \leq P_i^{dr} \leq \overline{P}_i^{dr}, \tag{9.12b}$$

$$\underline{P}_i^{tie} \leq P_i^{tie} \leq \overline{P}_i^{tie}. \tag{9.12c}$$

The control inputs u^m and u^{dr} are also upper- and lower-bounded.

State-space dynamics

The (centralized) state space model for the complete power system is

$$\dot{x} = A_c x + B_c u + E_c w, \tag{9.13a}$$

$$y = C_c x \tag{9.13b}$$

where

$$\begin{aligned} x &:= \begin{bmatrix} P^{tie} & f & P^m & P^v & P^{dr} \end{bmatrix}^\top \in \mathbb{R}^{Nn}, \\ u &:= \begin{bmatrix} u^m & u^{dr} \end{bmatrix}^\top \in \mathbb{R}^{Nm}, \\ y &:= ACE \in \mathbb{R}^{Np}, \end{aligned}$$

and where a variable without subscripts denotes the vector of variables corresponding to each node; for example, $f := [f_i]_{i \in \mathcal{N}}$.

Finally, we write the constraints as

$$\begin{aligned} u &\in \mathbb{U} \subseteq \mathbb{R}^m := \{u : P_u u \leq q_u\}, \\ x &\in \mathbb{X} \subseteq \mathbb{R}^n := \{x : P_x x \leq q_x\}. \end{aligned} \tag{9.14}$$

9.2.2 Real-time Multi-Area Economic Dispatch

Assume the power network is operating around a nominal equilibrium determined by an ED problem solved at a slower timescale. Let a constant disturbance w_i, $\forall i \in \mathcal{N}$ occur in real-time, say due to unknown variation in renewable generation or unpredicted load changes. Then by means of LFC, the generators and controllable loads are made to adjust their power generation P_i^m, $\forall i \in \mathcal{N}$ and consumption P_i^{dr}, $\forall i \in \mathcal{N}$ respectively in order to restore the grid frequency in the most economically efficient manner. Traditionally, changes in w_i are assumed very subtle and therefore the equilibrium from a previous ED operation to a large extent remains economically optimal. However, with larger and faster changes in w_i, there is need to achieve economic re-dispatch in LFC. To this end, we define the following multi-area ED problem for the LFC,

$$\min_{P^{tie}, P^m, P^{dr}} \quad -\Phi(P^m, P^{dr}) = \sum_{i \in \mathcal{N}} C_i(P_i^m) - \sum_{i \in \mathcal{N}} U_i(P_i^{dr}) \tag{9.15a}$$

$$\text{s.t.} \quad P_i^m - P_i^{dr} - P_i^{tie} - w_i = \mathbf{0}, \quad \forall i \in \mathcal{N} \tag{9.15b}$$

where $\Phi(P^m, P^{dr})$ is the social welfare of the power network, $C_i(P_i^m) = \frac{1}{2}q_i(P_i^m)^2 + r_iP_i^m + s_i$ the generator cost function and $U_i(P_i^{dr}) = \frac{1}{2}\tilde{q}_i(P_i^{dr})^2 + \tilde{r}_iP_i^{dr} + \tilde{s}_i$ the utility function for controllable loads in node i. We make the following assumption about problem (9.15).

Assumption 28. *Problem* (9.15) *is feasible for the equality constraint* (9.15b)*, each* $C_i(P_i^m)$ *is a strictly convex function in* P_i^m *and each* $U_i(P_i^m)$ *is a strictly concave function in* P_i^{dr}*.*

9.3 Problem Statement and Preliminaries

This section develops and analyzes predictive control algorithms for autonomously driving the outputs/inputs of representative power system models to steady-states that are both frequency regulating and equilibrium optimizing, according to a defined economic dispatch optimization problem. The developed algorithms should be robust to additive disturbances resulting from model and demand uncertainties while also guaranteeing dynamic optimality and feasibility. We achieve this by applying the feedback optimizing model predictive control algorithms developed previously to design the load frequency control algorithms for the power system models presented above. To proceed, we discretize the continuous-time power system model (9.13) using a zero-order hold method with a sampling interval of 0.1 seconds to obtain the discrete time model,

$$x(k+1) = Ax(k) + Bu(k) + Ew(k), \;\; y(k) = Cx(k) \tag{9.16}$$

where A, B, E and C are the discrete-time equivalents of the continuous-time state-space matrices $\tilde{A}, \tilde{B}, \tilde{E}$ and $\tilde{C}$ respectively.

Remark 51. *Although an exact discretization of the centralized dynamics is adopted here, this will remove the sparsity/structure of the power system. An inexact discretization such as the method proposed in [69] will preserve the system structure but at the expense of accuracy. Because the FOMPC controller developed here is centralized, the structure of the power system is not relevant to the control design and therefore the issue of inexact discretization is beyond the scope of the present discussion.*

Given the power system dynamics (9.16) and a constant disturbance $w(k) = \bar{w}$, a forced steady-state equilibrium is given by,

$$A\bar{x} + B\bar{u} + E\bar{w} = \bar{x}, \ \bar{y} = C\bar{x} \tag{9.17}$$

with the steady-state input–output map

$$\bar{y} = G_u\bar{u} + G_w\bar{w} \tag{9.18}$$

where $G_u \coloneqq C(I_n - A)^{-1}B$ and $G_w \coloneqq C(I_n - A)^{-1}E$ are the DC gains of the discrete-time system power system network (9.16) and the quantities with an over-bar represent steady-state values.

We make the following assumption about the discrete-time dynamics (9.16).

Assumption 29 (Basic assumptions)**.**

1. *the dynamics* (9.16) *are reachable and observable.*
2. *the state* $x(k)$ *is measurable at every sampling instant.*
3. $p \leq m$.

For the multi-area ED problem (9.15), the constraint (9.15b) is the power flow balance for all nodes and is satisfied at any steady-state equilibrium of the power system network. The cost function (9.15a) can also be expressed compactly as,

$$-\Phi(\bar{u}) = \frac{1}{2}\bar{u}^\top Q\bar{u} + R^\top\bar{u} + s \tag{9.19}$$

where

$$Q = \Gamma_m^\top Q_m \Gamma_m - \Gamma_{dr}^\top Q_{dr} \Gamma_{dr},\ R = \Gamma_m^\top R_m - \Gamma_{dr}^\top R_{dr},\ s = \mathbb{1}_N^\top s_m - \mathbb{1}_N^\top s_{dr},$$
$$\Gamma_m = \text{blkdiag}(\alpha_1^m, \dots, \alpha_N^m),\ \Gamma_{dr} = \text{blkdiag}(\alpha_1^{dr}, \dots, \alpha_N^{dr}),\ s_m = [s_i]_{\forall i \in \mathcal{N}}\ s_{dr} = [\tilde{s}_i]_{\forall i \in \mathcal{N}},$$
$$R_m = [r_i]_{\forall i \in \mathcal{N}},\ R_{dr} = [\tilde{r}_i]_{\forall i \in \mathcal{N}},\ Q_m = \text{blkdiag}(q_1, \dots, q_N),\ Q_{dr} = \text{blkdiag}(\tilde{q}_1, \dots, \tilde{q}_N). \tag{9.20}$$

9.3.1 The Feedback Optimizing Model Predictive Load Frequency Control (MPLFC) Problem

For a step change in $w = (w_i)_{i \in \mathcal{N}}$, the power imbalance causes frequency and tie-line deviation from the nominal values. The power imbalance due to the step changes in w will be eliminated through the adjustment of P^m and P^{dr} for all nodes using a centralized LFC regulator. Precisely, the feedback optimizing model predictive load frequency control (MPLFC) problem is defined next.

Problem 8 (The MPLFC Problem)**.** *Design for the linear time-invariant power system* (9.13)*, a state feedback load-frequency control law*

$$u(k) = \kappa_{LFC}(x(k), u(k-1)) \tag{9.21}$$

such that for any constant admissible $\bar{w} \in \mathbb{W}$:

1. *the ACE_i is driven to zero for all $i \in \mathcal{N}$,*
2. *the control inputs P^m and P^{dr} maximizes the social welfare $\Phi(P^m, P^{dr})$ in real time,*
3. *the feedback policy $\kappa_{LFC}(\cdot,\cdot)$ minimizes a transient performance criterion and,*
4. *the constraints* (9.14) *are satisfied at all times.*

Parts (1) and (2) of the MPLFC problem are solved via the static optimization formulation,

$$\min_{\bar{u},\bar{y}} \quad -\Phi(\bar{u}) \tag{9.22a}$$
$$\text{s.t.} \quad \bar{y} - G_u \bar{u} - G_w \bar{w} = \mathbf{0}, \tag{9.22b}$$
$$\bar{y} = \mathbf{0}. \tag{9.22c}$$

The equality constraint (9.22c) is added to achieve zero ACE in the optimal steady-state and therefore regulate the frequency and inter-area power flow to their respective nominal values. If Assumption 28 is satisfied, then problem (9.22) is convex and feasible, and a unique minimizer $(\bar{u}^*, \bar{y}^*)$ exists for every disturbance $\bar{w}$. The third control goal is achieved by designing a model predictive control law to track the optimal solution of problem (9.22) while rejecting the unknown step disturbance $\bar{w}$.

Remark 52 (Optimization problem encodes AGC). *Because the economic dispatch problem 9.22 is formulated to encapsulate the objectives of conventional automatic generation control (AGC), the optimal closed-loop performance of the power system frequency dynamics is expected to replicate the exact performance of AGC. This will be seen in the simulation results.*

However, if the goal is to obtain a different performance from conventional AGC, then this should be defined in the optimization problem. For instance, if an LFC scheme that encourages economics/price-based control is desired, the constraint $y = ACE = 0$ should be replaced by $y = f = 0$. In this case, the ACE will no longer be forced to zero but to values that result in economically optimal flow of power between the control areas. This will be demonstrated in the simulation studies.

9.3.2 Karush-Kuhn-Tucker (KKT) Optimality Conditions

To design the MPLFC law that solves the optimization problem (9.22) in feedback, we first examine the necessary conditions for optimality of the problem. To handle the equality constraint (9.22c), we reformulate (9.22) as,

$$\min_{\bar{u},\bar{y}} \quad -\Phi(\bar{u}) + \frac{1}{2}\bar{y}^\top \Pi \bar{y} \tag{9.23a}$$

$$\text{s.t.} \quad \bar{y} - G_u\bar{u} - G_w\bar{w} = \mathbf{0}. \tag{9.23b}$$

where $\frac{1}{2}\bar{y}^\top \Pi \bar{y}$ is a positive definite function of $\bar{y}$ and adds a penalty to the cost $\Phi(\bar{u})$ on violation of the equality constraint (9.22c). Defining $\bar{z} = \begin{bmatrix}\bar{u} & \bar{y}\end{bmatrix}^\top$, the static optimization problem (9.23) can be written compactly as,

$$\min_{\bar{z}} \quad \tilde{\Phi}(\bar{z}) = \frac{1}{2}\bar{z}^\top Q_{zz}\bar{z} + R_z^\top \bar{z} + s \tag{9.24a}$$

$$\text{s.t.} \quad G\bar{z} - G_w\bar{w} = \mathbf{0}_p. \tag{9.24b}$$

where $Q_{zz} = \begin{bmatrix} Q & \mathbf{0}_{m\times p} \\ \mathbf{0}_{p\times m} & \Pi \end{bmatrix}$, $R_z = \begin{bmatrix} R & \mathbf{0}_p \end{bmatrix}^\top$ and $G = \begin{bmatrix} -G_u & I_p \end{bmatrix}$. Problem (9.24) is a convex optimization problem and strong duality holds [33]. Therefore, the Karush–Kuhn–Tucker (KKT) conditions are necessary and sufficient for optimality. We obtain the KKT conditions for (9.24) by forming the corresponding Lagrangian,

$$\mathcal{L}(\bar{z}, \lambda) = \tilde{\Phi}(\bar{z}) + \lambda^\top (G\bar{z} - G_w\bar{w}) \tag{9.25}$$

where λ is a multiplier of appropriate dimension. The corresponding KKT optimality conditions are

$$\nabla\mathcal{L}(\bar{z}, \lambda) = \begin{bmatrix} \nabla\tilde{\Phi}(\bar{z}) + G^\top\lambda \\ G\bar{z} - G_w\bar{w} \end{bmatrix} = \mathbf{0}_{m+2p}. \tag{9.26}$$

The optimum $\bar{z}^*$ of problem (9.24) must satisfy the KKT system of equations in (9.26). Solving (9.26) however requires a knowledge of $\bar{w}$ which is assumed unknown a priori. To circumvent this, we express (9.26) in the following subspace form [24]:

$$\nabla\mathcal{L}(\bar{z}, \lambda) = \mathbf{0}_{m+2p} \iff \begin{bmatrix} \nabla\tilde{\Phi}(\bar{z}) \in \text{range}(G)^\top \\ G\bar{z} - G_w\bar{w} = \mathbf{0}_p \end{bmatrix} \tag{9.27}$$

By a fundamental theorem of linear algebra, $\text{range}(G)^\top = \text{null}(G)^\perp$, and therefore

$$\nabla\tilde{\Phi}(\bar{z}) \in \text{range}(G)^\top \iff \nabla\tilde{\Phi}(\bar{z}) \in \text{null}(G)^\perp. \tag{9.28}$$

Therefore, let $\tilde{G}$ be any full-rank matrix such that,

$$\tilde{G}G^\top = \mathbf{0} \quad \text{or} \quad \text{range}(\tilde{G})^\top = \text{null}(G). \tag{9.29}$$

Remark 53. *For the model* (9.16) *with the steady-state input-output map* (9.18) *(*i.e., *the case of non-singular* $(I_n - A)$*, the matrix*

$$\tilde{G} = \begin{bmatrix} (G_u^\top)^\dagger & I_p \end{bmatrix} \tag{9.30}$$

satisfies (9.29).

Remark 54. *For the power system* (9.16)*,* $(G_u^\top)^\dagger$ *depends on the connection topology of the nodes/control areas in the power network which is encapsulated in the Laplacian matrix* $(\mathscr{L}_{\mathscr{G}})$ *of the power network graph* $\mathcal{G}$*.*

With this, the KKT optimality condition (9.26) becomes

$$\nabla\mathcal{L}(\bar{z},\lambda) = \begin{bmatrix} \tilde{G}\nabla\tilde{\Phi}(\bar{z}) \\ G\bar{z} - G_w\bar{w} \end{bmatrix} = \mathbf{0}_{m+2p}. \tag{9.31}$$

It follows that $\bar{z}$ is optimal with respect to problem (9.24) if, and only if, it satisfies (9.31). This establishes the following result, which—similar to [128]—allows the the steady-state equilibrium optimization problem to be posed as a stabilization problem.

Proposition 6. *Part (1) of the LFC problem is solved if, from any initial state $x(0) = x_0$ and any disturbance $\bar{w}$, the control law*

$$u(k) = \kappa_{LFC}(x(k), u(k-1)) \tag{9.32}$$

is such that $z(k) = (u(k), y(k))$:

1. *is regulated to a steady-state equilibrium,* ***and,***
2. *satisfies* $\lim_{k\to\infty} \tilde{G}\nabla\tilde{\Phi}(z(k)) = \mathbf{0}_{m+p}$.

Proof. Condition (1) is satisfied if and only if $G\bar{z} - G_w\bar{w} = \mathbf{0}$, which is necessary and sufficient for equilibrium. Condition (2) implies, and is implied by, the KKT conditions (9.31) being met in the limit, which is necessary and sufficient for optimality. □

9.4 Feedback-Optimizing Model Predictive Load Frequency Control

In this section, based on the results of Proposition 6, we construct an MPC controller to regulate the tracking error, $\tilde{G}\nabla\tilde{\Phi}(z(k))$ to zero, and consequently solve problem 8 without knowledge of $\bar{z}^*$ or $\bar{w}$. Using the velocity model form of the linear quadratic optimal control problem [172], we develop an MPC formulation that steers the power system network asymptotically and admissibly to the economically optimal steady-state equilibrium, without knowledge of this equilibrium and while minimizing a LQ transient performance criterion.

9.4.1 Tracking error

Instead of defining and regulating the tracking error as the difference between $z(k)$ and the unknown optimum $z^*(k)$, we define the tracking error as the residual of the KKT optimality condition (9.31). For the cost function $\tilde{\Phi}(\bar{z})$, under inactive steady-state inequality constraints and non-singular $(I_n - A)$, the tracking error $\tilde{G}\nabla\tilde{\Phi}(z(k))$ is an affine function of the measured output and input and is given by,

$$e(k) := \tilde{G}\nabla\tilde{\Phi}(u(k), y(k)) = \Lambda_u u(k) + \Lambda_y y(k) + r \tag{9.33}$$

where $\Lambda_u = (G_u^\top)^\dagger Q$, $\Lambda_y = \Pi$ and $r = (G_u^\top)^\dagger R$.

Remark 55. *In the tracking error given by* (9.33)*,* $\Lambda_u u(k) + r = (G_u^\top)^\dagger(Qu + R)$ *is related to the marginal cost differences between neighbouring control areas while* $\Lambda_y y(k) = \Pi y(k)$ *is related to the area control error of the power network.*

This error may be computed directly from the input $u(k)$ and output $y(k)$, provided the objective $\Phi(u(k))$ and the input–output DC gain matrix G_u is known. This choice therefore eliminates the need for knowledge of the optimal equilibrium $\bar{z}^*$ and the disturbance $\bar{w}$.

9.4.2 Feedback optimizing model predictive load-frequency control (MPLFC) formulation

To regulate the power system (9.13) to $e = 0$ and therefore simultaneously achieve frequency regulation and economic dispatch, we express the power system model (9.16) in velocity form and formulate the model predictive load-frequency control problem following the same design approach as the nominal feedback optimizing model predictive control presented in Chapter 6. Solution of the optimal control problem ((6.14),(6.15) and (6.16)), followed by the application of the first control in the optimized sequence, yields the MPLFC control law,

$$u(k) = u(k-1) + \kappa_{N_p}(\epsilon(k)).$$

9.4.3 Stability and Performance Guarantees

For the following analysis, we assume that the disturbance $\bar{w}$ stays constant (otherwise steady-state operation is not well defined). The following result summarizes the stability and recursive feasibility of the MPLFC algorithm, and follows directly from well established results on conventional linear MPC [181] and the efforts of earlier chapters.

Theorem 9.4.1 (Stability and feasibility)**.** *The control law* $u(k) = \kappa_{LFC}(x(k), u(k-1)) = u(k-1) + \kappa_{N_p}(\epsilon(k))$ *solves the MPLFC problem,* (8)*.*

9.5 Numerical Simulation

In this section, we present the design and simulation of the MPLFC algorithm above for the 2-area power system model described in example 12.4 in [186] (see Figure 9.1). We compare the results obtained with a conventional MPC based LFC and a centralized version of the distributed averaging PI (DAPI) control algorithm from [229]. All simulations are done using the linear model presented in Section 9.2.1.

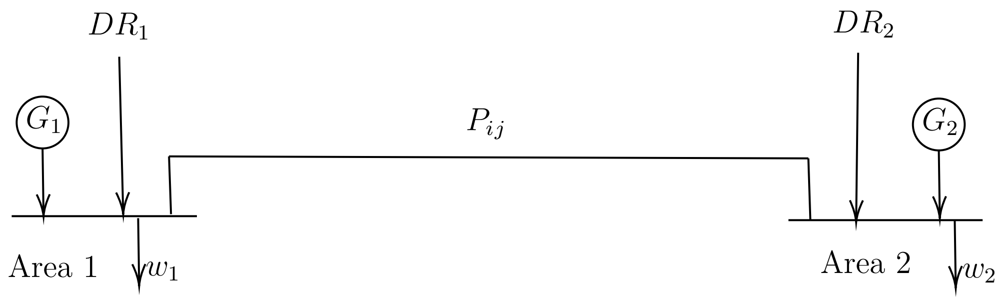

Fig. 9.1 2-area power system

Table 9.1 Power system model parameters

Area,i	H_i	$\tau_{t,i}$	$\tau_{v,i}$	τ_i^{dr}	α_i^m	α_i^{dr}	β_i	T_{ij}	D_i
1	10	0.5	0.2	1	0.5	0.5	20.6	2	0.6
2	8	0.6	0.3	1	0.5	0.5	16.9	2	0.9

Table 9.2 Power system cost parameters

Area,i	q_i	r_i	s_i	$\tilde{q}_i$	$\tilde{r}_i$	$\tilde{s}_i$
1	1	0.5	0	0.1	0.1	0
2	0.5	0.8	0	0.5	0.1	0

The parameters shown in Table 9.1 result in the continuous-time linear power system frequency dynamics in (9.13) with the following coefficient matrices,

$$A_c = \begin{bmatrix} 0 & 0 & 2.000 & -2 & 0 & 0 & 0 & 0 & 0 \\ -0.1 & -0.06 & 0 & 0.1 & 0 & 0 & 0 & 0.1 & 0 \\ 0.125 & 0 & -0.1125 & 0 & 0.1250 & 0 & 0 & 0 & 0.1250 \\ 0 & 0 & 0 & -2.00 & 0 & 2.00 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.667 & 0 & 1.667 & 0 & 0 \\ 0 & -100 & 0 & 0 & 0 & -5.00 & 0 & 0 & 0 \\ 0 & 0 & -53.333 & 0 & 0 & 0 & -3.333 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.00 \end{bmatrix}, \tag{9.34}$$

$$B_c = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 2.50 & 0 \\ 0 & 1.6667 \\ 0.500 & 0 \\ 0 & 0.500 \end{bmatrix}, \quad E_c = \begin{bmatrix} 0 & 0 \\ -1.0 & 0 \\ 0 & -0.125 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{9.35}$$

and

$$C_c = \begin{bmatrix} 1.00 & 20.60 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1.00 & 0 & 16.90 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{9.36}$$

The system is stabilizable and observable, meeting Assumption 6. The disturbance $w(t)$ is unknown but constant (or slowly varying). The following inequality constraints are present on the input, output and disturbances;

$$\begin{aligned} \mathbb{U} &:= \{u : \ -5I_2 \leq u \leq 5I_2\}, \\ \mathbb{Y} &:= \{y : \ -2I_2 \leq y \leq 2I_2\}, \\ \mathbb{W} &:= \{w : \ -2I_2 \leq w \leq 2I_2\}. \end{aligned} \tag{9.37}$$

The following disturbance scenarios will be utilized in subsequent discussions :
Scenario I (Step):

$$w(t) = \begin{bmatrix} 0.4 & 0 \end{bmatrix}^\top, 5 \leq k < 55 \tag{9.38}$$

Scenario II (Time-varying):

$$w(t) = \begin{cases} \begin{bmatrix} 0.4 & 0 \end{bmatrix}^\top & 5 \leq t < 55, \\ \begin{bmatrix} 0 & 0.9 \end{bmatrix}^\top & 55 \leq t < 105, \\ \begin{bmatrix} 1.5 & 0.1 \end{bmatrix}^\top & t \geq 105. \end{cases} \tag{9.39}$$

The objective function for the steady-state economic dispatch problem, derived from the social welfare using the cost parameters in Table 9.2 and a penalty of $\Pi = \begin{bmatrix} \pi_1 & 0 \\ 0 & \pi_2 \end{bmatrix}$ on each output of the power system is expressed in the form of equation (9.24) and is shown in (9.40) below:

$$\Phi(z) = \frac{1}{2} \begin{bmatrix} u \\ y \end{bmatrix}^T \begin{bmatrix} 0.2250 & 0 & 0 & 0 \\ 0 & 0.1125 & 0 & 0 \\ 0 & 0 & \pi_1 & 0 \\ 0 & 0 & 0 & \pi_2 \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} + \begin{bmatrix} 0.20 \\ 0.35 \\ 0 \\ 0 \end{bmatrix}^\top \begin{bmatrix} u \\ y \end{bmatrix} \tag{9.40}$$

9.5.1 Traditional MPC based LFC design

Traditionally, the control input u for the 2-area power system in (9.13) is designed using a tracking MPC formulation with the primary goal of regulating the output, y, which is the area control error (ACE) to zero.
Therefore, a tracking MPC based LFC algorithm will be designed in this subsection based on the following tracking velocity model of the power system dynamics in (9.13).

$$\bar{\epsilon}(k+1) = \bar{\mathcal{A}}\bar{\epsilon}(k) + \bar{\mathcal{B}}\delta\bar{u}(k), \tag{9.41a}$$

$$\bar{e}(k) = \bar{\mathcal{C}}\bar{\epsilon}(k) \tag{9.41b}$$

where

$$\bar{\epsilon}(k) := \begin{bmatrix} \delta x(k) \\ \bar{e}(k-1) \end{bmatrix} \quad \text{with} \quad \begin{matrix} \delta x(k) := x(k) - x(k-1), \\ \delta u(k) := u(k) - u(k-1), \end{matrix} \tag{9.42}$$

and

$$\bar{\mathcal{A}} = \begin{bmatrix} A & \mathbf{0} \\ C & I_{n_y} \end{bmatrix}, \qquad \bar{\mathcal{B}} = \begin{bmatrix} B \\ \mathbf{0} \end{bmatrix}, \qquad \bar{\mathcal{C}} = \begin{bmatrix} C & I_{n_y} \end{bmatrix}. \tag{9.43a}$$

Here $\bar{e}(k) = ACE = y(k)$ and the matrices (A, B, C) are the discrete-time equivalent of the continuous-time system matrices (A_c, B_c, C_c). A zero-order-hold method was used to discretize the power system dynamics with a sampling interval of $0.1s$.

Remark 56 (Note)**.** *To differentiate the velocity dynamics used for the tracking MPC based LFC formulation presented in this subsection from the velocity dynamics used in a subsequent subsection for the feedback optimizing model predictive LFC design, an overbar has been used.*

Given the tracking error and tracking velocity model, the MPC-based LFC law applied to the power system is given by

$$u(k) = u(k-1) - \bar{\kappa}_{N_p}(\bar{\epsilon}(k)) \tag{9.44}$$

where $\bar{\kappa}_{N_p}(\bar{\epsilon}(k))$ is the first control move to the optimal control sequence $\delta\bar{\mathbf{u}}^*(k)$ computed at time k by solving the following OCP,

$$\mathbb{P}(\bar{\epsilon}(k))\colon \min_{\delta\bar{\mathbf{u}}(k)\in\mathscr{U}_{N_p}} V_{N_p}(\bar{\epsilon}(k), \delta\bar{\mathbf{u}}(k)) \tag{9.45}$$

where the feasible region $\mathscr{U}_{N_p}(\bar{\epsilon}(k))$ is defined as

$$\mathscr{U}_{N_p}(\bar{\epsilon}(k)) \triangleq \left\{ \delta\bar{\mathbf{u}}(k) \left| \begin{aligned} \bar{\epsilon}(k+i+1) = \bar{\mathcal{A}}\bar{\epsilon}(k+i) + \bar{\mathcal{B}}\delta u(k+i),\ \forall i \in \mathbb{I}_{[0,N_p-1]} \\ \bar{\epsilon}(k+i) \in \bar{\mathbb{G}},\ \forall i \in \mathbb{I}_{[0,N_p-1]} \\ \bar{\epsilon}(k+N) \in \bar{\mathbb{G}}_f \end{aligned} \right. \right\} \tag{9.46}$$

and the performance objective is defined as

$$V_{N_p}(\bar{\epsilon}(k)) = \frac{1}{2}\|\bar{\epsilon}(k+N_p)\|^2_{\bar{\mathcal{P}}} + \frac{1}{2}\sum_{i=0}^{N_p-1}\Big(\|\bar{e}(k+i)\|^2_{\bar{Q}_e} + \|\delta\bar{u}(k+i)\|^2_{\bar{\mathcal{R}}}\Big) \tag{9.47}$$

where $\bar{Q}_e \succeq 0$, $\bar{\mathcal{R}} \succ 0$ and $\bar{\mathcal{P}} \succ 0$ is the stabilizing positive definite solution to discrete-time algebraic Riccati equation

$$\bar{\mathcal{P}} = \bar{\mathcal{A}}^\top\bar{\mathcal{P}}\bar{\mathcal{A}} + \bar{\mathcal{C}}^\top\bar{Q}_e\bar{\mathcal{C}} - \bar{\mathcal{A}}^\top\bar{\mathcal{P}}\bar{\mathcal{B}}\Big(\bar{\mathcal{R}} + \bar{\mathcal{B}}^\top\bar{\mathcal{P}}\bar{\mathcal{B}}\Big)^{-1}\bar{\mathcal{B}}^\top\bar{\mathcal{P}}\bar{\mathcal{A}}. \tag{9.48}$$

The matrices $\bar{\mathbb{G}}$ and $\bar{\mathbb{G}}_f$ are computed following the steps outlined in Chapter 6 except that the tracking velocity model parameters $(\bar{\mathcal{A}}, \bar{\mathcal{B}}, \bar{\mathcal{C}})$ are used in place of $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ and the following substitution, $\Lambda_y = I_p$, $\Lambda_u = \mathcal{D} = 0$ and $r = \mathbf{0}$ are made.

Remark 57. *One major limitation of this MPC based LFC design is that it does not utilize economic information to improve the efficiency of load frequency control in real-time. Therefore, it might suffer some efficiency loss under significant load changes and model uncertainty. This will be evident in the simulation studies.*

The LFC performance under the tacking MPC controller designed above is discussed in the following subsections.

Case I: LFC using tracking MPC under step disturbance

After 5 seconds, a step disturbance of $w_1 = 0.4p.u$ was applied to control area 1. The load-reference setpoints u for both control areas are generated at each sample time k by solving the MPC problem (9.45) and applying the centralized control law (9.44) to both control areas.
Figure 9.2 shows the inputs (load-reference setpoints) and outputs (area control error, ACE) for both control areas. As seen in the figure, the tracking MPC regulates the ACE to zero with good dynamic performance. Also, as evident from Figure 9.3, the frequency deviation is driven to zero, also with very good dynamic performance.

From the input-output performance of the tracking MPC shown in Figure 9.2, it is observed that although the tracking MPC was not designed to be feedback optimizing, it still tracks the optimal economic dispatch set-points (dashed lines). A reason for this is that the economic dispatch problem in (9.22) was formulated to achieve AGC or tie-line bias control at optimality. Therefore, any controller designed based on the tie-line bias control strategy (i.e., driving the ACE to zero), will result in the same steady-state equilibrium as those obtained by solving problem (9.22). This is why the tracking MPC controller could track the solution of the optimization problem without being designed to be feedback optimizing. However, as will become clear later, encoding tie-line bias control in an economic dispatch problem can lead to a loss of economic optimality in LFC. This limitation will be addressed in a later subsection.

Case II: LFC using tracking MPC under time-varying disturbance

The performance of the tracking MPC-based LFC is also studied under time-varying disturbances. The time-varying disturbance sequence in (9.39) was applied to the power system. From Figure 9.4 and Figure 9.5, we see that the performance of the controller is not significantly different from the case of step disturbances. Frequency regulation is also achieved here with good dynamic performance. Similar to the case

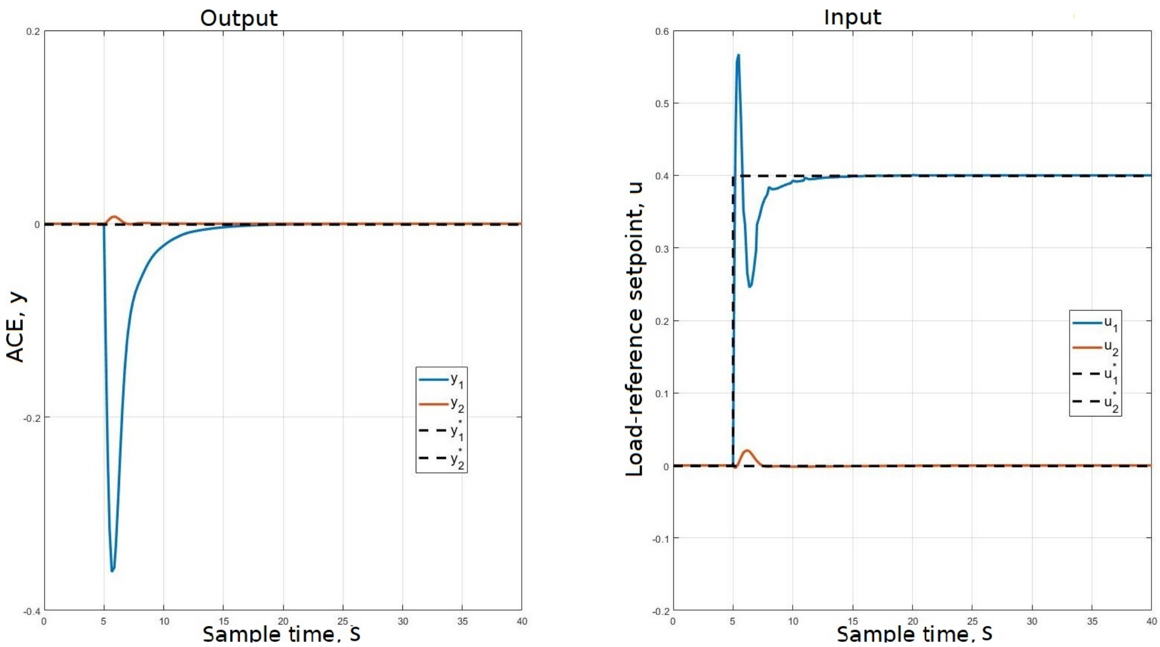

Fig. 9.2 Input and output LFC performance under tracking MPC ($\bar{Q}_e = 10I_2, \bar{R} = 50I_2$) with $w_1 = 0.4, w_2 = 0$. Economically dispatched reference values for the current disturbance are shown using dashed lines.

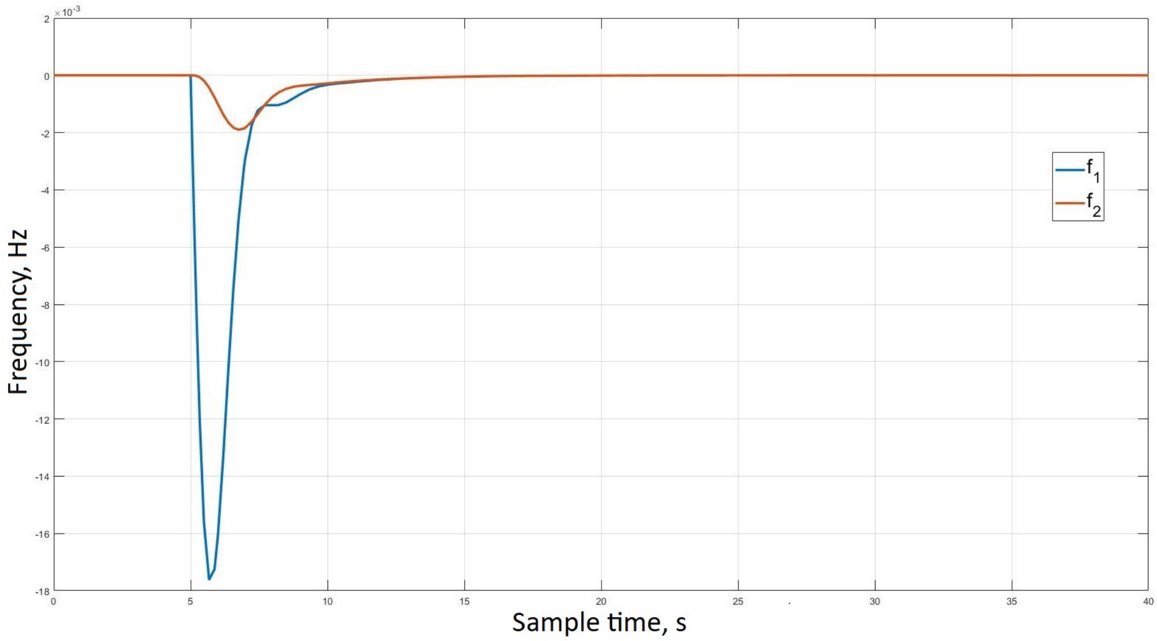

Fig. 9.3 Frequency regulation of LFC under tracking MPC ($\bar{Q}_e = 10I_2, \bar{R} = 50I_2$) with $w_1 = 0.4, w_2 = 0$.

of a step disturbance, the tracking MPC controller was able to track the inputs and outputs that happen to coincide with the optima of the economic dispatch problem in (9.22) because they were both (i.e the tracking MPC and the economic dispatch problem) formulated to implement the same tie-line bias control strategy of driving the ACE to zero.

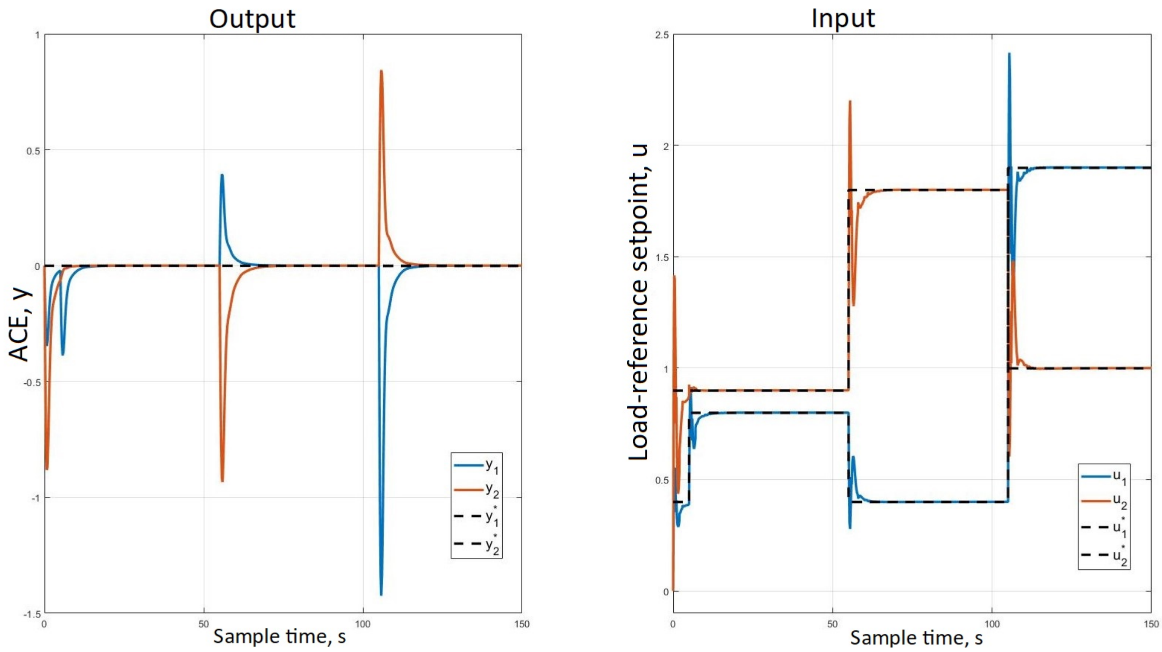

Fig. 9.4 Input and output LFC performance under tracking MPC ($\bar{Q}_e = 10I_2, \bar{R} = 50I_2$ $\Pi = 50I_2$) with time-varying disturbance. Economically dispatched reference values for the current disturbance are shown using dashed lines.

9.5.2 Centralized DAPI control design

To achieve real-time economic optimality in LFC, [229] proposed the distributed averaging PI (DAPI) control algorithm. This algorithm modifies a conventional PI control law by adding an averaging filter that encodes economic dispatch's equal marginal cost criterion into the PI control algorithm, thereby enabling the controller to achieve economic dispatch in steady-state. Also, the DAPI controller is a decentralized algorithm and gives plug and play capability for first order generation model. The design of a centralized version of the DAPI control algorithm as used in this section is shown in (9.49) below

$$
\begin{aligned}
\dot{e} &= y = ACE, \\
\dot{q} &= -\mathscr{L}_{\mathscr{G}} \nabla_u \Phi(u), \\
u &= -K_e e - K_q q
\end{aligned}
\tag{9.49}
$$

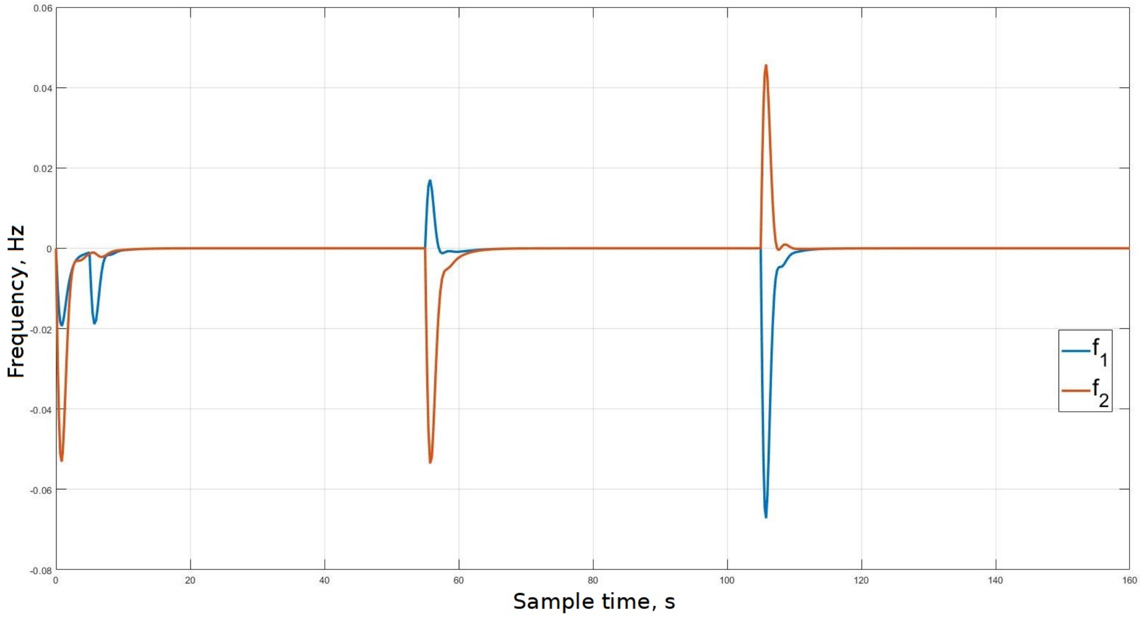

Fig. 9.5 Frequency regulation of LFC under tracking MPC ($\bar{Q}_e = 10I_2, \bar{R} = 50I_2$) with time-varying disturbance.

where K_e and K_q are tuning parameters that determine the rate of convergence of the algorithm. The actual control input to the generators and demand responsive loads are determined from u via the participation factors. For the simulation carried out in this section,

$$\mathscr{L}_{\mathscr{G}} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad \nabla_u \Phi(u) = \begin{bmatrix} 0.2250 & 0 \\ 0 & 0 \end{bmatrix} u + \begin{bmatrix} 0.2 \\ 0.35 \end{bmatrix}. \tag{9.50}$$

K_p and K_e are free parameters tuned in simulation to get a desired performance. The LFC performance under the DAPI controller designed above is discussed in the following subsections.

Case I: LFC using DAPI under step disturbance

A step disturbance of $w_1 = 0.4p.u$ was applied to control area 1 after 5 seconds. The control input, u for both control areas were generated in real-time by the DAPI controller. Figure 9.6 shows the inputs (load-reference setpoints) and outputs (area control error, ACE) for both control areas. As seen in the figure, the DAPI controller drives the ACE to zero while tracking the economic dispatch setpoints, which in this case happens to be the inputs and outputs that achieve tie-line bias control or AGC. Also, as evident from Figure 9.7, the DAPI controller drives the frequency deviation to zero, restoring the nominal frequency of the power network. However, unlike the tracking MPC controller, the DAPI controller has poorer dynamic performance with a transient response that is under-damped (i.e., oscillatory) and slowly converging. Although some performance improvements can be obtained by careful tuning, it was generally very challenging to tune the DAPI controller for a desired transient performance.

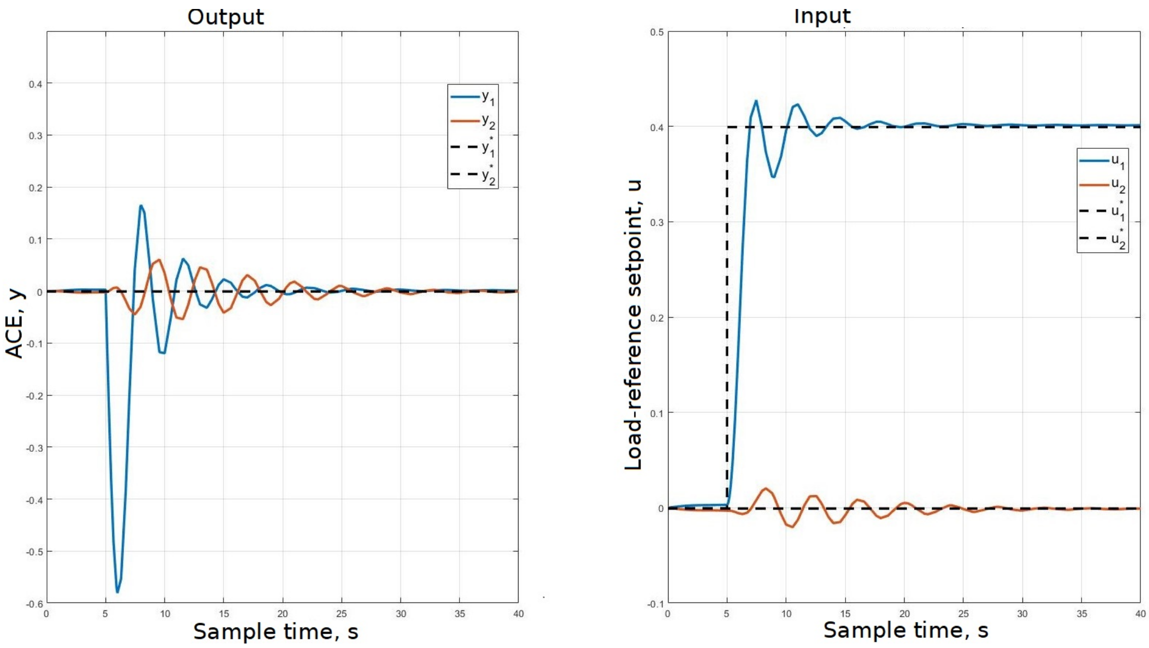

Fig. 9.6 Input and output LFC performance under DAPI ($K_e = 0.5I_2, K_q = 0.01I_2,\ \Pi = 400I_2$) with $w_1 = 0.4, w_2 = 0$. Economically dispatched reference values for the current disturbance are shown using dashed lines.

Case II: LFC using DAPI under time-varying disturbance

The performance of the DAPI controller was also investigated for time-varying disturbances using the disturbance sequence in (9.39). From Figure 9.8 and Figure 9.9, we see that the performance of the controller is not significantly different from the case of step disturbances. It is oscillatory and slowly converging as expected. Frequency regulation is also achieved here with a dynamic performance similar to the case of step disturbances. Again, the DAPI controller regulates the ACE and frequency to their nominal values using economically dispatched inputs which in this case are the same as the inputs required for tie-line bias control or AGC.

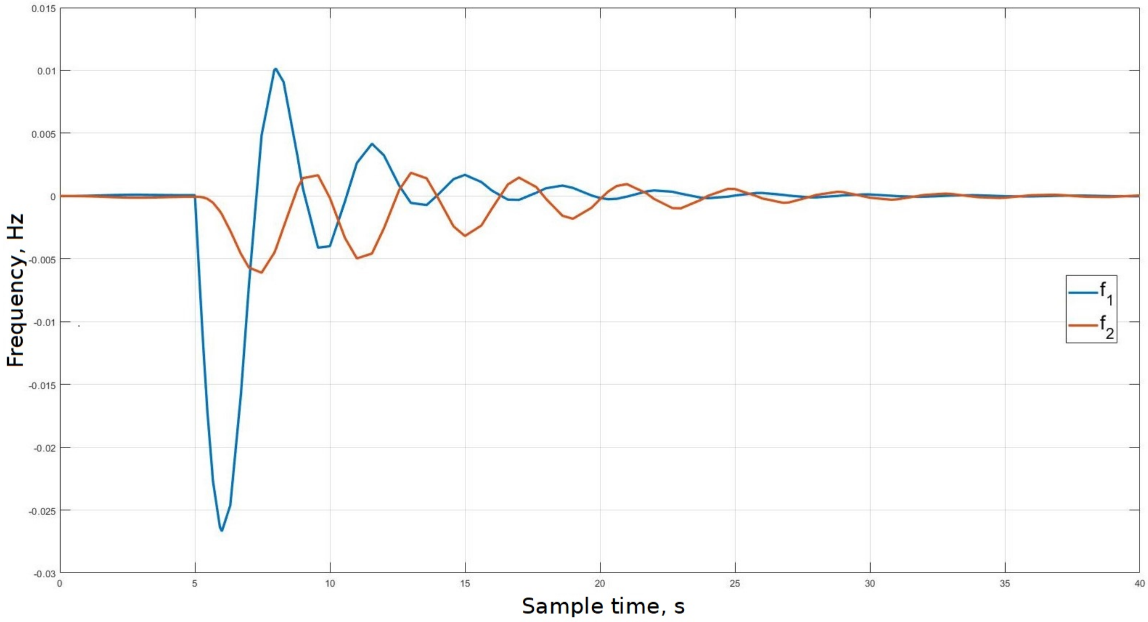

Fig. 9.7 Frequency regulation of LFC under DAPI ($K_e = 0.5I_2, K_q = 0.01I_2,\ \Pi = 400I_2$) with $w_1 = 0.4, w_2 = 0$.

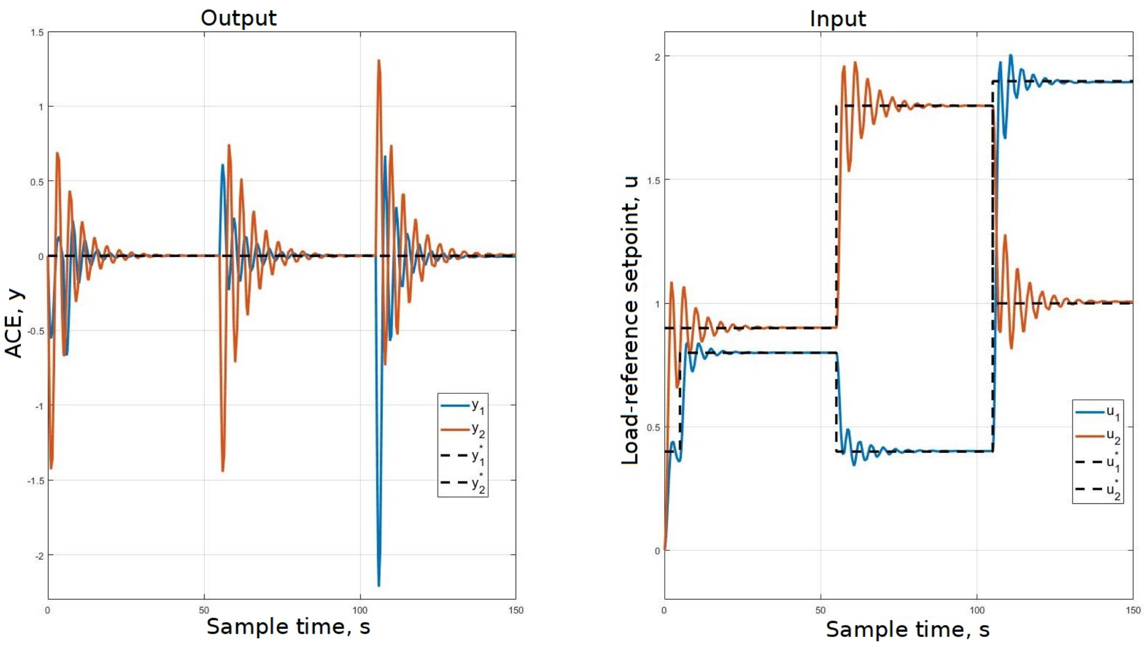

Fig. 9.8 Input and output LFC performance under DAPI ($K_e = 0.5I_2, K_q = 0.01I_2,\ \Pi = 400I_2$) with time-varying disturbance. Economically dispatched reference values for the current disturbance are shown using dashed lines.

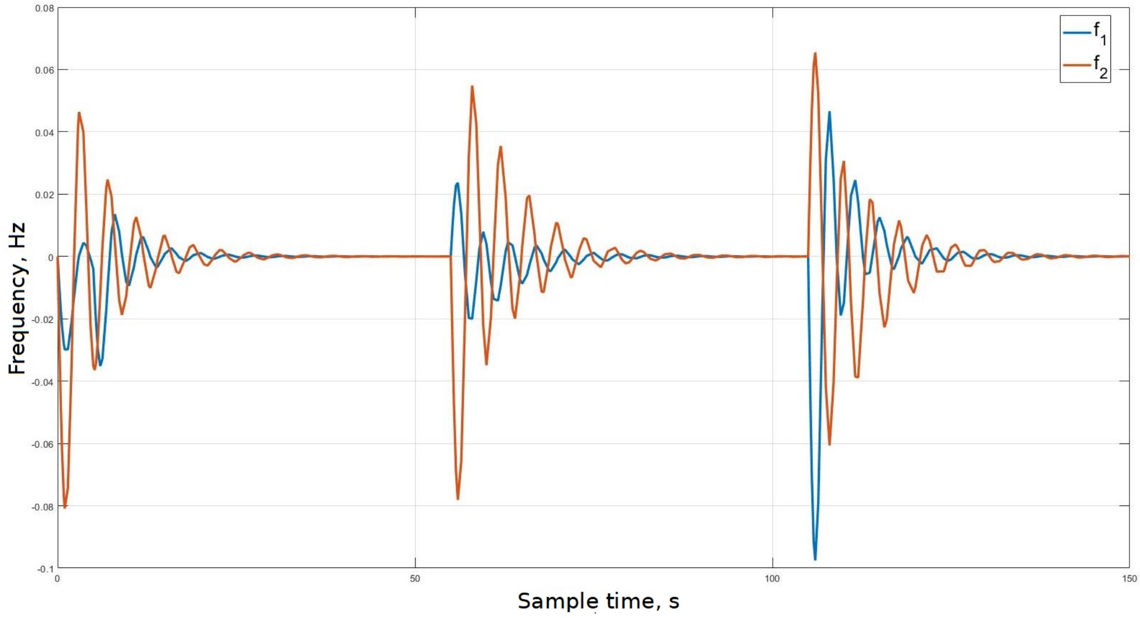

Fig. 9.9 Frequency regulation of LFC under DAPI ($K_e = 0.5I_2, K_q = 0.01I_2,\ \Pi = 400I_2$) with time-varying disturbances.

9.5.3 Feedback Optimizing MPC based LFC design

Despite the real-time economic optimizing capabilities of the DAPI controller, its design scope is severely limited. Firstly, the controller can only optimize the steady-state input or output but not both. Also, the controller has no means of guaranteeing optimal dynamic performance. Furthermore, the DAPI control design cannot take advantage of model information to improve performance, and finally, there is no way of systematically dealing with transient constraints in a DAPI controller. These limitations motivate a FOMPC approach to optimal load-frequency control.

To design the FOMPC, the system is first discretized using zero-order hold with a sampling time of 0.1 seconds. The input–output DC gain matrix of the resulting discrete-time system is,

$$G_u = \begin{bmatrix} 0.5493 & 0.5493 \\ 0.4507 & 0.4507 \end{bmatrix}. \tag{9.51}$$

The velocity state for the controller is

$$\epsilon(k) = \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} = \begin{bmatrix} x(k) - x(k-1) \\ \Lambda_y y(k-1) + \Lambda_u u(k-1) + r \end{bmatrix} \tag{9.52}$$

where Λ_y, Λ_u and r follow from the choices of $\Phi(z)$ parameters as

$$\Lambda_y = \Pi, \quad \Lambda_u = \begin{bmatrix} 0.2250 & 0 \\ 0 & 0 \end{bmatrix},$$
$$r = \begin{bmatrix} 0.20 \\ 0.35 \end{bmatrix}.$$

The matrix S satisfies Assumption 6 and together with the stabilizability of (9.1) guarantee the existence of a solution to the FOMPC problem.
The transient performance criterion is chosen with $Q_e = 0.1 \times I_2$ and $R_\delta = 500 I_2$; these values satisfy the hypothesis of Proposition 3. From the design parameters above, the FOMPLFC is designed following the approach outline in Chapter 6 with the prediction horizon, $N = 5$. In the following subsections, the performance of FOMPLFC will be discussed.

Case I: LFC using FOMPC under step disturbance

After 5 seconds, a step disturbance of $w_1 = 0.4 p.u$ is applied to control area 1. The control input, u for both control areas are generated at each sample time k by solving a FOMPC problem and applying the generated control law to both control areas.
Figure 9.10 shows the inputs (load-reference setpoints) and outputs (area control error, ACE) for both control areas. As seen in the figure, the FOMPC drives the ACE to zero and also tracks the economic dispatch setpoints (shown in dashed lines). Also, the FOMPC regulates the frequency deviation to zero as evident from Figure 9.11. The FOMPC has very good dynamic performance and unlike the DAPI controller, it produces a properly damped transient response with little to no oscillations. Convergence is also much faster than the DAPI controller.

Case II: LFC using FOMPC under time-varying disturbance

The performance of FOMPC is also studied under time-varying disturbances. The disturbance sequence is once again taken to be the time-varying disturbance in (9.39). From Figure 9.12 and Figure 9.13, we see that the performance of the controller is not significantly different from the case of step disturbances. Frequency regulation is also achieved here with very good dynamic performance. Similar to the case of a step disturbance, the economic dispatch setpoints are also tracked by the FOMPC controller under the time-varying disturbances.

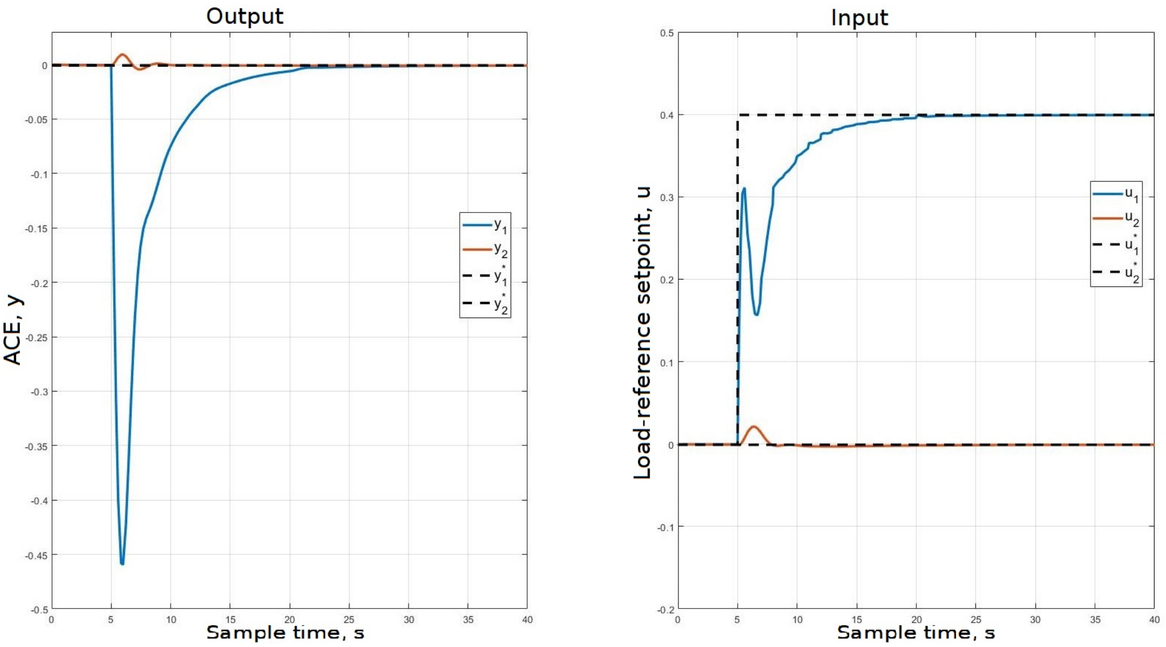

Fig. 9.10 Input and output LFC performance under FOMPC ($Q_e = 1 \times I_2, R_\delta = 50I_2, \Pi = 400 \times I_2$) with $w_1 = 0.4, w_2 = 0$. Economically dispatched reference values for the current disturbance are shown using dashed lines.

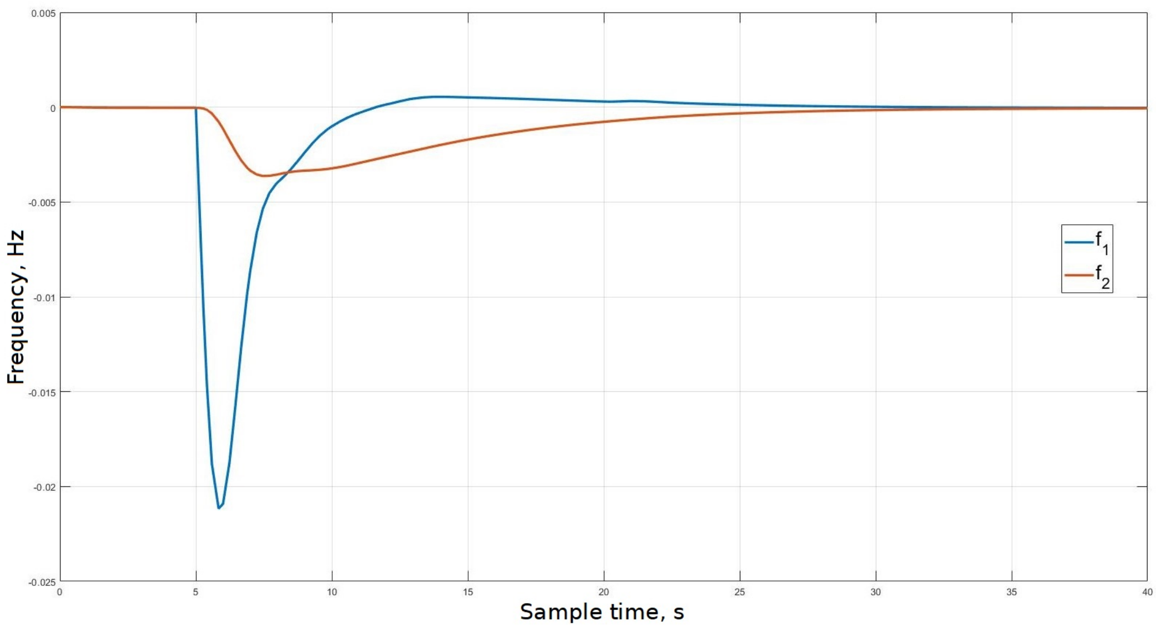

Fig. 9.11 Frequency regulation of LFC under FOMPC ($Q_e = 10^{-3} \times I_2, R_\delta = 5000I_2, \Pi = 400 \times I_2$) with $w_1 = 0.4, w_2 = 0$.

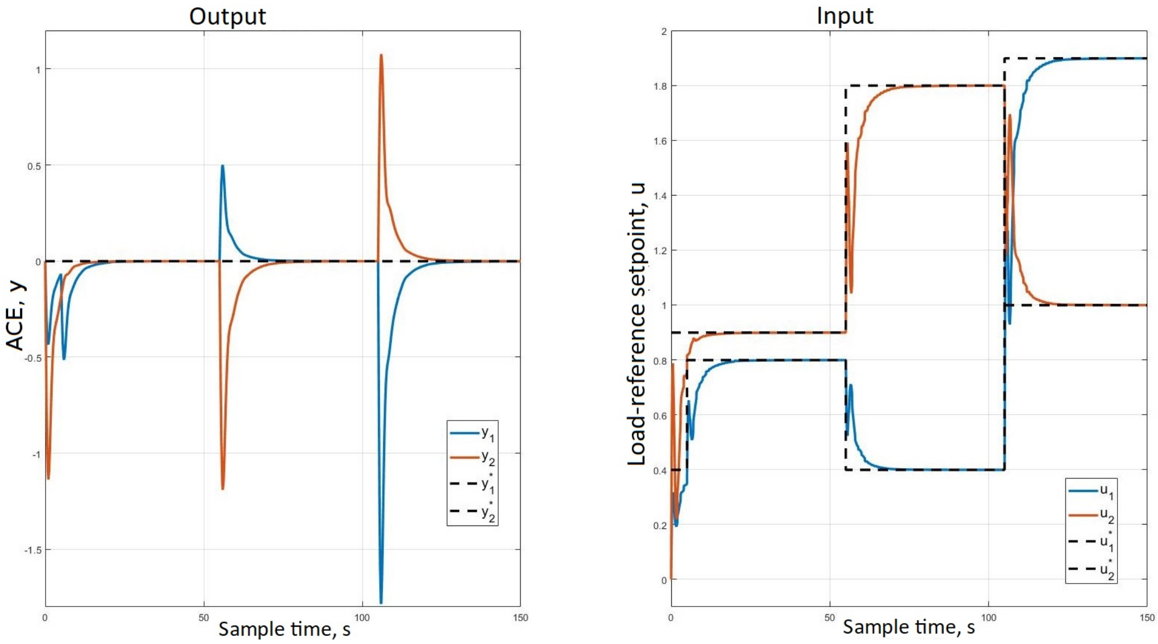

Fig. 9.12 Input and output LFC performance under FOMPC ($Q_e = 1 \times I_2, R_\delta = 50I_2, \Pi = 400 \times I_2$) with time-varying disturbance. Economically dispatched reference values for the current disturbance are shown using dashed lines.

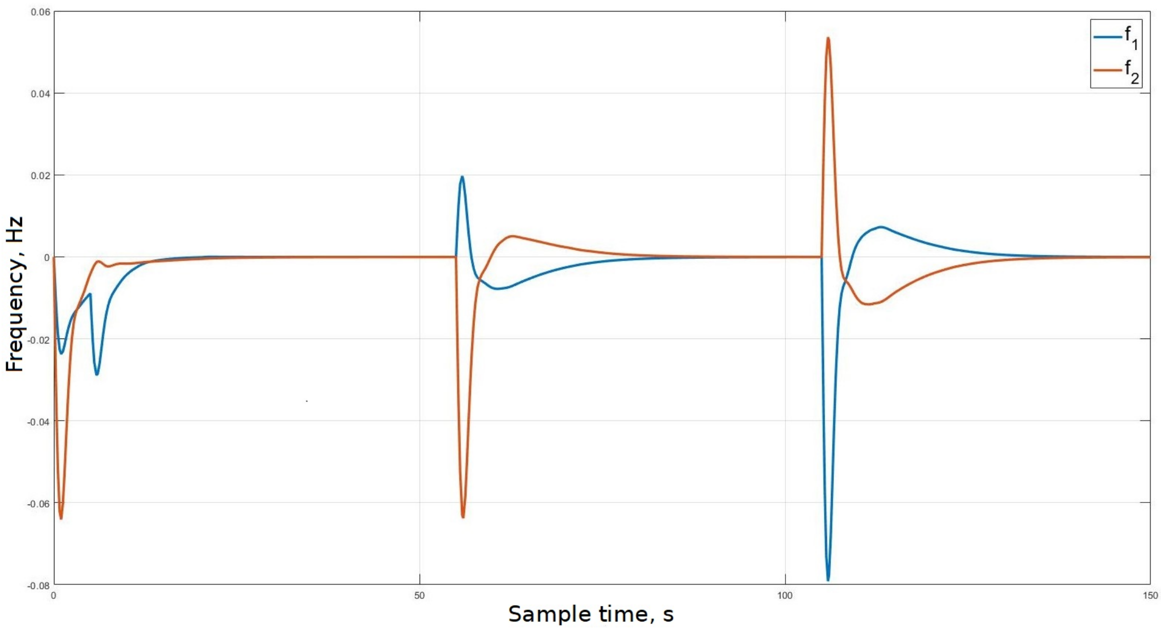

Fig. 9.13 Frequency regulation of LFC under FOMPC ($Q_e = I_2, R_\delta = 50I_2, \Pi = 400 \times I_2$) with time-varying disturbances.

9.5.4 Real-time multi-area economic dispatch in LFC

A common observation from the simulation studies presented previously is that in steady-state, all LFC control actions are local. That is, control resources (generators, demand response etc) in one control area do not 'asymptotically' respond to load changes in the other control area. The word asymptotically is important here as during the transient phase, inter-area coordination occurs but in steady-state, all LFC actions are local to the area where the disturbance originated. This is why in all the figures above, u_1 settles to the disturbance value of $w_1 = 0.4$ while u_2 settles to zero as no disturbance occurred in control area 2. This is the classical tie-line bias control approach or AGC and it is realized by driving the ACE to zero [197].

The problem with conventional tie-line bias control or AGC is that it does not encourage the asymptotic sharing of control resources between control areas and therefore limits the scope for achieving improved economic optimality in load-frequency control. Also, studies have shown that under significant variability and uncertainty in the demand, conventional AGC based on tie-line bias control can be severely inefficient [134]. The economic dispatch problem in (9.22) was defined to encode the conventional tie-line bias control approach of LFC. Therefore, solving problem (9.22) in feedback will merely results in controllers that implement the conventional tie-line bias control strategy albeit in a way that is supposedly optimal. In other words, problem (9.22) merely rediscovers AGC using an optimization approach. As a result, the controllers designed so far do not take advantage of the interconnectedness of the power system to achieve additional improvement in economic performance.
This limitation can be easily addressed by reformulating problem (9.22) to enforce the constraint $f = 0$ rather than $y = ACE = 0$ (i.e., drive the frequency to zero rather than the ACE). This simple reformulation removes the non-interaction constraint inherent in conventional AGC by allowing the control areas to coordinate resources to jointly regulate the frequency in steady-state, and as a result achieve economically optimal load-frequency control.

In this subsection, we demonstrate the economic improvement possible with this approach by solving a reformulated version of the economic dispatch problem (9.22) with the constraint $y = f = 0$ instead. We will discuss simulation results obtained from implementing the FOMPC and the DAPI control algorithms designed previously. To put things in perspective, a comparison will be made with the standard tracking MPC that is based on controlling the ACE. Figure 9.14 shows the performance of the standard tracking MPC-based LFC for the time-varying disturbance sequence in (9.39). From the figure, we can see that the standard tracking MPC achieves the objectives of LFC by consistently driving the frequency deviation and ACE to zero for both control areas. However, it fails to achieve this using inputs that are economically

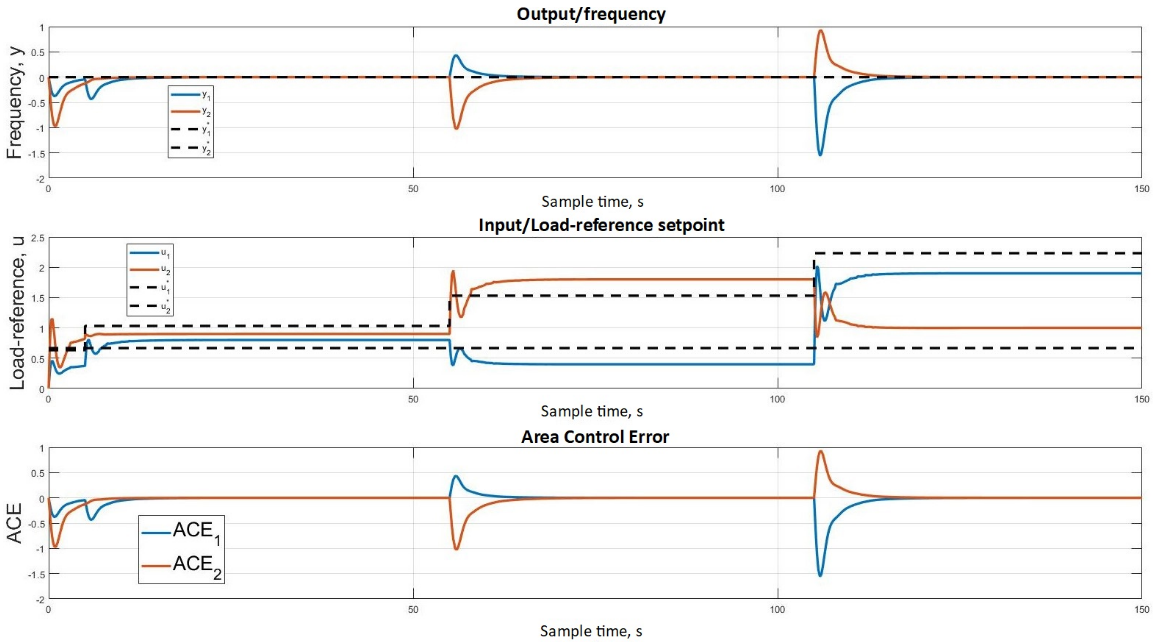

Fig. 9.14 Frequency regulation with economic dispatch objectives using tracking MPC ($Q_e = 10 \times I_2, R_\delta = 50I_2, \Pi = 400 \times I_2$) with time-varying disturbances. Economically dispatched reference values for the current disturbance are shown using dashed lines

dispatching. This is evident from Figure 9.14 as the inputs can be seen to not track the economic dispatch values shown using dashed lines. Therefore, tracking MPC does not achieve economically optimal frequency control. This is because the conventional AGC strategy implemented by the tracking MPC controller lack the mechanism to achieve an (economics/price-based) steady-state inter-area coordination within LFC.

To achieve economically optimal LFC (in a steady-state sense), the FOMPC algorithm was implemented to solve the reformulated economic dispatch problem. The results obtained are shown in Figure 9.15. From the figure, it is evident that FOMPC regulates the frequency to zero while also tracking the (unknown) economic dispatch setpoints, without measuring the load changes or explicitly solving the economic dispatch problem. The ACE is driven to non-zero values as the tie-line power flows are re-allocated online rather than driven to previously scheduled values (as commonly done in AGC). Also, it shows very good dynamic performance and like tracking MPC enforces the constraints on the power system (during the transient phase).

Finally, the DAPI controller is implemented for the power system. Figure 9.16 shows its performance for the problem being considered with the same disturbance sequence as previously. From the figure, it is evident that DAPI just like FOMPC achieves both frequency regulation (i.e., drives frequency to zero) and economic dispatch

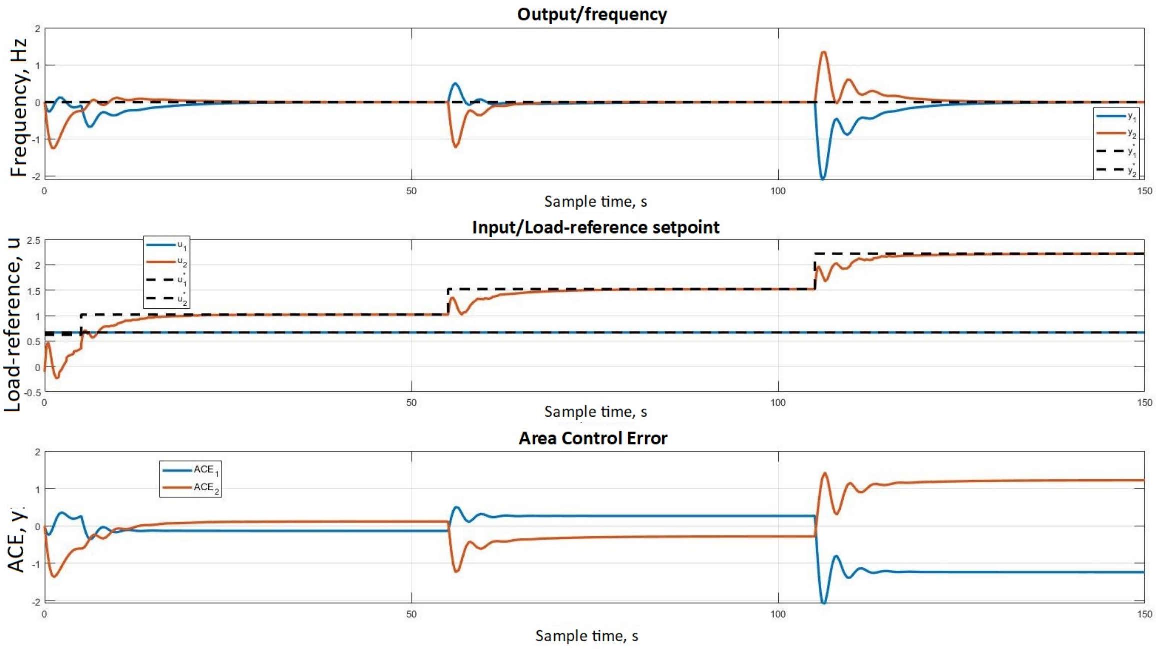

Fig. 9.15 Frequency regulation with economic dispatch objectives using FOMPC ($Q_e = 1 \times I_2, R_\delta = 50I_2, \Pi = 400 \times I_2$) with time-varying disturbances. Economically dispatched reference values for the current disturbance are shown using dashed lines

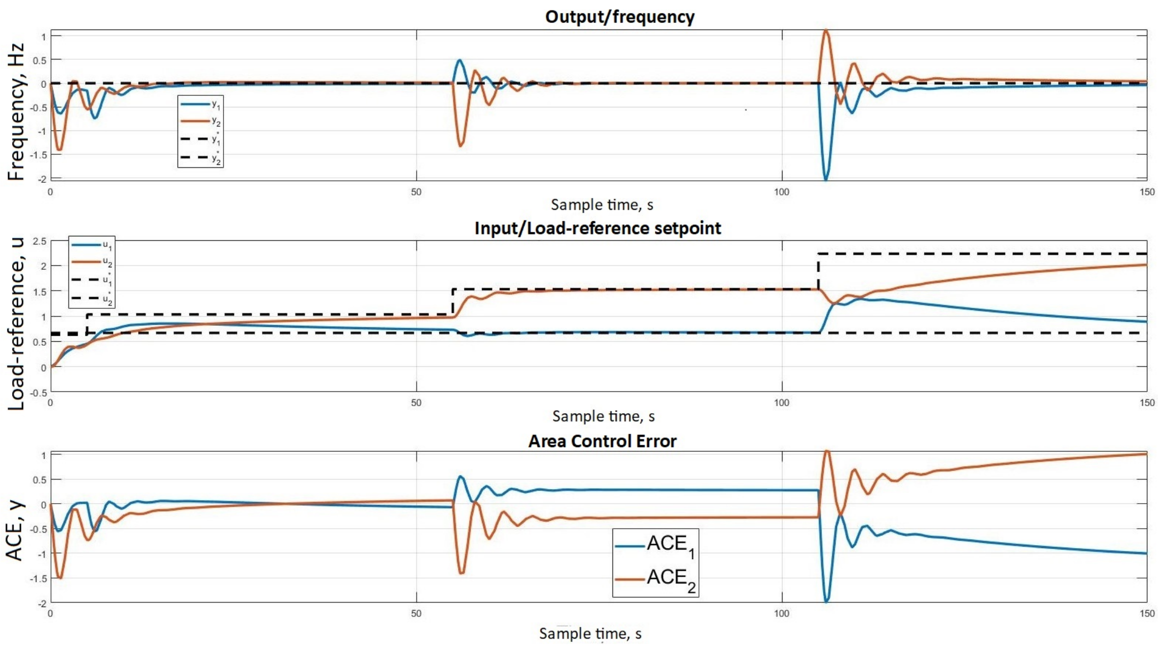

Fig. 9.16 Frequency regulation with economic dispatch objectives using DAPI ($K_q = 1 \times I_2, K_e = 0.5I_2, \Pi = 400 \times I_2$) with time-varying disturbances. Economically dispatched reference values for the current disturbance are shown using dashed lines

in closed-loop. The economic dispatch set-points can be seen in Figure 9.16 to be tracked as the disturbance changes. From the figure, it is clear that of the three control algorithms, the DAPI controller has the poorest transient performance. Its transient response is under-damped and it converges slowly to the economic dispatch setpoints. Also, tuning the DAPI controller for a desirable transient performance is a difficult task.

From the results obtained in the above simulations, it can be concluded that both FOMPC and DAPI can achieve LFC while simultaneously solving an economic dispatch problem using only the feedback from input-output measurements, without explicit knowledge of the load changes or explicitly computing the economic dispatch setpoints. This capability can be useful for real-time economic dispatch in future power system networks with highly variable and uncertain power fluctuations. The FOMPC has shown superior performance compared to DAPI, with better transient response, guaranteed constraint satisfaction and a much easier and intuitive tuning mechanism. However, unlike DAPI, FOMPC is computationally more expensive to implement and relies on a more detailed model of the power system dynamics.

9.6 Conclusion

This chapter has presented an MPC–based approach to the combined economic dispatch and load–frequency control in a multi-area power system. The approach uses a form of model predictive control that combines steady-state optimization and tracking to provide a feedback optimizing control law that drives the system states to the steady-state optimum, without explicitly computing this and using it as an explicit setpoint.

Recursive feasibility and stability of the closed loop were established under mild conditions on the system, cost, and constraints. To guarantee stability and transient constraint satisfaction, the proposed algorithm relied on the computation of invariant sets for the power system linear frequency dynamics. For larger power systems with a high number of state variables, it can be computationally challenging to compute these invariant sets. A practical approach to circumvent this difficulty will be to remove the terminal constraints from the FOMPC problem formulation and rely instead on very long prediction horizons. This however increases the size of the optimization problem solved online to generate the FOMPC actions at each sample time. With the current advancement in computing power, this may not be too much of a challenge in the future. Another important problem with the use of long prediction horizons to ensure stability and feasibility in FOMPC is that it becomes very challenging to obtain theoretical certifications of stability and feasibility over an infinite horizon. This is a very interesting theoretical question for future research.

Finally, simulation results on a two-area network have demonstrated the capability of the approach for frequency restoration while tracking a changing economic equilibrium. Comparisons with standard approaches in the literature have also been made. Results show that the FOMPC algorithm developed in this chapter shows superior performance to the other two controllers (DAPI and tracking MPC). The results presented in this chapter have only considered a centralized setting with states available for feedback and a linear swing dynamics assumed. Future work will consider decentralized and distributed implementations, and also test the algorithm using more realistic non-linear high dimensional power system models.

Chapter 10

Conclusion and Future Work

This thesis presented techniques for achieving feedback optimization (i.e., regulating a dynamic system in a closed-loop to the unknown solution of a steady-state optimization problem) using model predictive control. On the one hand, solving optimization problems implicitly using feedback rather than explicitly via numerical optimization can guarantee robust and computationally efficient solutions. On the other hand, model predictive control can ensure optimal dynamic performance, systematic constraint handling, and a prediction-based decision-making capability. In this thesis, we have combined the design philosophy of feedback optimization and model predictive control to obtain a feedback-optimizing MPC algorithm capable of optimizing both the steady-state and transient performance of a dynamic system while satisfying the system inequality constraints. We have achieved this without unnecessarily complicating the standard MPC algorithms. The capability of feedback optimizing MPC to achieve optimal steady-state control while optimizing the dynamic performance and satisfying system constraints makes it a promising approach for the optimal real-time operation of future power systems and indeed other plants that require economic operation alongside real-time control.

We present in this chapter a summary of the contributions of the thesis and recommendations for future work. Section 10.1 provides an overview of the significant contributions of the relevant contributing chapters, while section 10.2 states the possible directions for future research.

10.1 Thesis contributions

The contributions of each chapter of the thesis are stated in the following subsections. We split the contributions into the following key areas:

1. Contributions to Feedback Optimization

2. Contributions to Model Predictive Control

10.1.1 Contributions to Feedback Optimization

The field of feedback optimization deals with the design of feedback controllers that steer a physical system to a steady-state that solves a predefined static optimization problem. In feedback optimization, regulation to an optimal steady state is achieved without explicit numerical computation of the static optimization problem or the need for external set-points. Instead, the optimization algorithms for solving the static optimization problem are directly implemented as feedback controllers which regulate a dynamic system in closed-loop to the unknown optimum of the static optimization problem. This interconnection of off-the-shelf optimization algorithms with a dynamic system brings to bear the benefit of feedback control (i.e., robustness to unmeasured disturbances and un-modelled dynamics, simplicity and stability guarantees) in the solution of a static optimization problem.
However, the majority of controllers proposed for achieving feedback optimization ignore the dynamic performance of the closed-loop system assuming instead a pre-stabilized system with dynamic considerations made offline. Our first contribution in this thesis is to relax this assumption by directly considering dynamic performance objectives (such as transient optimality and recursive constraint satisfaction) within the feedback optimizing control design. Also, by using a model predictive control framework, we integrate model predictions into feedback optimizing control thereby bringing together the benefits of feedback and feed-forward control design. In the following, a chapter by chapter breakdown of our contributions to the field of feedback optimization is detailed.

1. In Chapter 2 a comprehensive review of the various techniques for achieving feedback optimization in dynamic systems is provided. The review in this chapter revealed that the design scope for most algorithms for feedback optimization is severely limited. This has resulted in the omission of important control objectives such as dynamic optimality, recursive constraint satisfaction and robust control

in most of the proposed algorithms. This in turn motivated the idea in this thesis to develop novel feedback optimizing control strategies based on a model predictive control framework.

2. Chapter 3 presented a general framework for designing feedback optimization controllers with optimal transient performance and recursive feasibility guarantees. This chapter poses the feedback optimization problem as a feedback optimizing model predictive control (FOMPC) problem. The absence of pre-computed set-points makes it challenging to solve the FOMPC problem. We addressed this difficulty by translating the FOMPC problem to a generalized tracking MPC problem with a tracking error derived from input-output measurements rather than a pre-computed set-point. We achieve this translation by defining the steady-state tracking error as the residual of the Karush-Kuhn-Tucker conditions of optimality for solving the static optimization problem. We also gave necessary and sufficient conditions for solvability of the FOMPC problem. A benefit of the FOMPC problem formulation presented in this chapter is its generality which makes it possible to develop an avalanche of MPC control laws that solve the FOMPC problem for a wide range of circumstances. This formed the central theme of the contributions in the subsequent chapters.

3. In Chapter 4, we proposed a novel feedback optimization algorithm that regulates a disturbed linear time-invariant system to the unknown optimum of a steady-state quadratic program while guaranteeing optimal transient performance. We call this the feedback optimizing linear quadratic control (FOLQC) algorithm. To develop this algorithm, we expressed the LTI system in velocity form. We then used dynamic and semi-definite programming tools to derive explicit linear quadratic control laws that are self-optimizing for a defined static quadratic programming problem. For most feedback optimization algorithms proposed in the literature, closed-loop stability is only established under nominal conditions (i.e., no model uncertainty). On the contrary, we present in this chapter robust stability results for the FOLQC under non-nominal conditions (i.e., with model perturbation). We present robust stability margins for FOLQC in terms of its tuning parameters i.e. the weights on the tracking error and input deviation. Finally, we illustrated the performance of FOLQC with numerical examples.

4. Most available feedback optimization algorithms assume precise knowledge of the system dynamics while relying on inherent robustness to reject any model

perturbation or disturbances. In chapter 5, we made pioneering efforts to develop a novel feedback optimization algorithm for linear systems with parametric model uncertainty by extending the FOLQC algorithm developed in Chapter 4 to the case of polytopic uncertain linear systems. Using linear matrix inequalities, we formulated robust semi-definite programs which can be solved offline to generate explicit robust feedback optimizing control laws capable of regulating uncertain linear systems to the unknown optimum of a defined quadratic optimization problem. This contribution is significant in the feedback optimization literature as insufficient attention has been paid to the design of feedback optimization algorithms that explicitly integrate uncertainty information to improve the control performance. Indeed this is the first robust feedback optimization algorithms with the capability to optimize the dynamic performance according to a defined transient cost criterion. Simulation results showing the success of the developed algorithm are also given.

5. In Chapter 6, we develop novel feedback optimization algorithms that regulate constrained linear systems to the unknown optimum of a steady-state quadratic program while satisfying system constraints and optimizing dynamic performance. As previously stated, most feedback optimization algorithms lack guarantees of transient performance and feasibility, providing only steady-state guarantees. In this chapter, we address this limitation for the case of a quadratic steady-state optimization by developing a novel feedback optimization algorithm based on MPC. The proposed algorithms systematically enforce the system's inequality constraints during the transient phase, as against relying on saturation functions to enforce constraints as is common with conventional feedback optimization algorithms. As a result, both hard and soft inequality constraints are handled without the undesirable effects of integrator windup common with saturation-based control laws. We also established robust stability of the proposed algorithm under model perturbation. We demonstrated the performance of the developed algorithm via numerical simulation.

6. In Chapter 7, robust feedback optimization algorithms are proposed for uncertain linear systems such that autonomous steady-state optimization is achieved while guaranteeing optimal dynamic performance and recursive feasibility. Here, the FOMPC problem is reformulated for the case of linear systems with polytopic

model uncertainty. Simulation results have been shown to demonstrate the performance of the proposed laws.

7. In Chapter 8, a novel distributed feedback optimizing control algorithm is presented for driving constrained large-scale systems closer to their (unknown) optimal steady-states with transient performance guarantees and recursive feasibility. The result obtained in this chapter are significant as they represent a pioneering attempt at synthesizing distributed feedback optimization algorithms that consider the dynamic performance objectives in the control design. We leverage on the results developed for robust FOMPC to develop the distributed feedback optimization algorithms presented in this chapter. Convergence analysis of the proposed algorithm has been proved showing that distributed FOMPC is guaranteed to converge to a neighbourhood of the optimal steady-state equilibrium while being recursively feasible.

8. In Chapter 9, we presented an optimal load-frequency control solution for autonomous real-time economic dispatch in power systems with guarantees on transient performance and constraint satisfaction. We posed the optimal load-frequency control problem as a FOMPC problem solved online to generate implicit feedback control laws that simultaneously achieve economic dispatch and frequency regulation without knowledge of the unknown load changes or the economic dispatch set-points. We also provided a theoretical analysis of the performance guarantees such as feasibility, transient optimality, and asymptotic convergence to the economic dispatch solutions. Finally, we presented simulation results showing the superior performance of the proposed solution compared to state-of-the-art solutions such as automatic generation control and distributed averaging PI control.

10.1.2 Contributions to Linear Quadratic and Model Predictive Control

It is conventional to adopt a tracking economic MPC formulation to achieve economically optimal steady-state control. Here, the optimal steady-state set-point is assumed to be available or directly computed within the MPC algorithm. A challenge with this approach is that the direct numerical solution to a steady-state economic optimization problem is non-robust. It requires accurate knowledge of the system model and unknown disturbances. Also, it is computationally expensive to solve the steady-state optimization problem within the MPC algorithm. Recently, there have been several techniques proposed to address these issues. One standard solution has been integrating modifier-adaptation techniques into the design of tracking economic MPC algorithms to improve their robustness. Compared to standard MPC, modifier-adaptation approaches incur an additional computational expense and result in more complex algorithms that are practically unappealing. Also, the need to know the disturbances in advance is still not addressed. In this thesis, we have presented an alternative approach to achieving robust asymptotically economic MPC that retains the computational and algorithmic complexity of standard tracking MPC and achieves optimal steady-state performance without disturbance estimation. The approach fuses the design philosophy of feedback optimization with a tracking MPC design. The result is a tracking MPC that achieves feedback optimization in closed-loop with the system. Here, input and output measurements are used to track the unknown optimum to a steady-state economic optimization problem rather than compute the optimal steady-state set-point from disturbance estimates and within MPC. This thesis also presents novel variants of the feedback optimizing MPC design robust to parametric uncertainty. It also applied the technique to achieve optimal load frequency control in multi-area power systems. In the following, a chapter by chapter breakdown of our contributions to the field of model predictive control is detailed.

1. Chapter 2 provided a comprehensive review of techniques for achieving steady-state economic optimization in model predictive model control. This review showed that available steady-state economics optimizing MPC techniques are computationally more expensive and more complex than standard tracking MPC. Also, most algorithms rely on disturbance estimation to guarantee closed-loop convergence to the optimal economic steady-state equilibrium. We were motivated by these drawbacks to seek new algorithms that address some of these limitations.

2. In Chapter 3, the tracking economic MPC problem was posed as a feedback optimization problem. We termed this problem the feedback optimzing MPC (FOMPC) problem. It was realized in this chapter that the FOMPC problem is a generalization of the standard tracking MPC to the case of unknown steady-state set-points.

3. In Chapter 4, we integrated feedback optimization and linear quadratic control to achieve a feedback optimizing linear quadratic control (FOLQC) algorithm. The

FOLQC algorithm obtained solves the tracking economic MPC problem when the steady-state cost is quadratic, the system is deterministic and linear, the disturbance is piecewise constant and the inequality constraints are inactive. Our FOLQC algorithm is the first linear quadratic controller capable of autonomous steady-state optimization. Solutions to the FOLQC problem were devised from first principles using dynamic programming and linear matrix inequalities. We also presented simulation results to demonstrate the performance of the algorithm.

4. In Chapter 5, we developed a FOLQC algorithm that accounts for polytopic model uncertainty. This is the first robust LQR controller with autonomous steady-state optimization capabilities.

5. In Chapter 6, we proposed an MPC algorithm capable of autonomously regulating a constrained LTI system to the unknown optimum of a steady-state quadratic program without external set-points or online economic optimization. We achieved this by formulating the control problem as a FOMPC problem. To solve the FOMPC problem, we formulated optimal control problems (OCP) which solved online generate implicit feedback laws that achieve autonomous optimization of a quadratic program with optimal transient performance and recursive feasibility. Also, in this chapter, we presented a detailed analysis of the convergence of FOMPC under both nominal and uncertain conditions. The results confirm that FOMPC is inherently robust against small perturbation in the dynamic model. Finally, we present simulation results to illustrate the performance of the proposed FOMPC algorithm.

6. In Chapter 7, the FOMPC algorithm was developed for uncertain linear systems with polytopic model uncertainty and piecewise constant disturbances. This is the first robust MPC algorithm that optimizes a steady-state cost in feedback. Using results from tube-based robust MPC and min-max MPC, we developed two distinct algorithms that solve the robust FOMPC problem. The tube-based MPC algorithm is computationally efficient, adding very little complexity to the already designed nominal FOMPC . However, it can only guarantee convergence to a neighbourhood of the optimal steady-state solution and also results in conservative closed loop performance due to the tightening of the feasible set. We addressed this limitation by turning to a min-max MPC formulation. Here, the FOMPC is formulated via LMI as a semi-definite program which is solved online

to generate implicit feedback optimizing control laws that robustly drive the uncertain linear system to optimal steady states for a defined quadratic program.

7. In Chapter 8, we developed distributed MPC methods for regulating large-scale constrained linear systems to economically optimal steady-states that are the solution of a steady-state quadratic program. The algorithms developed are based on a distributed solution to the FOMPC problem and achieve optimal steady-state optimization without explicitly solving the steady-state optimization or estimating the disturbances.

10.2 Future Research Directions

Possible direction for future research include:

10.2.1 Feedback optimzing MPC for unreachable set-points

The FOMPC algorithms developed in this thesis only enforce constraints during the transient phase, assuming inactive steady-state constraints instead. Consequently, the presented algorithms can only solve unconstrained steady-state optimization problems. However, economic operation for most practical systems occur at the boundary of the feasible region. Hence, the developed FOMPC algorithms may generate unreachable and suboptimal setpoints in the presence of active steady-state constraints. Therefore, it is necessary to design novel FOMPC algorithms capable of handling inequality constraints both during transients and in steady state. One possible way to achieve this will be to formulate the steady-state optimization as a constrained problem and convert it to a penalized unconstrained optimization problem using penalty methods. The challenge with this approach is that the penalized unconstrained optimization may introduce non-linearities into the FOMPC problem. Developing novel algorithms that can handle these non-linearities without over-complicating the FOMPC algorithms presented so far will be an exciting direction for future research. A possible way to achieve this could be to use the Courant-Beltrami penalty function and pose the FOMPC problem as a hybrid MPC problem. This could allow the wealth of tools available for hybrid/switching MPC to be deployed to solve the FOMPC when the steady-state constraints are active. Another possible solution could involve reformulating the presented FOMPC algorithms to avoid violating the steady-state constraints. However, this approach will not guarantee convergence to the constrained optimal steady-state equilibrium.

10.2.2 Nonlinear feedback optimzing MPC

Although the FOMPC problem formulation is very general, the algorithms developed in this thesis only solve the problem for linear systems with steady-state quadratic objectives. However, most real-world systems are non-linear, and many economic optimization problems admit non-quadratic costs. Therefore, developing efficient algorithms to solve the FOMPC problem for non-linear systems with non-quadratic steady-state costs will be interesting for future research. The key challenge here is finding novel ways to solve the non-linear MPC problem while retaining the computational and algorithmic complexity of the algorithms presented thus far. A gain scheduling approach that uses a collection of locally linearized dynamics at different operating points could be adopted to solve the resulting non-linear FOMPC problem. It will be interesting to investigate how accurately such an approach can approximate the solution to the actual non-linear steady-state optimization problem. Another interesting direction would be to investigate the use of the so-called feedback linearization methods to linearize the non-linear system globally. Then the standard FOMPC algorithm could be applied to the resulting linear dynamics. This approach, however, can complicate the design of the FOMPC algorithm, especially the set computations.

10.2.3 Improved robust feedback optimizing MPC

The robust FOMPC algorithm presented in Chapter 7 has two significant drawbacks. Firstly, the algorithm relies on the quadratic stabilizability of the uncertain system to guarantee the closed-loop stability, which can result in very conservative closed-loop performance. Also, the application of the robust FOMPC algorithm is limited to the small class of quadratically-stabilizable systems. Therefore, developing novel control algorithms that address the conservatism of the robust FOMPC will be a good direction for future research. One way to achieve this could involve using parameter-dependent Lyapunov functions in the problem formulation. Secondly, the robust FOMPC algorithm developed in the thesis can be computationally costly for online implementation. As a result, this limits its application to the control of slow systems. It would be interesting to address this limitation by developing computationally more efficient formulations of the presented algorithm. An excellent place to start would be to develop offline formulations of the robust FOMPC algorithm. Here, most of the computations happen offline to improve the computational efficiency without degrading the control performance.

10.3 Further applications of feedback optimizing MPC

The need for economically optimal long-term operation while achieving real-time control is a feature of many engineering systems. Therefore, applying the FOMPC algorithms developed in this thesis to other engineering systems will be a good research direction. The power system example presented in this thesis utilized a simplified power system model. It would therefore be interesting to study the performance of FOMPC for a detailed model of the power system example. Also, other application problems such as power-sharing in micro-grid and economic process control would be interesting research directions to pursue. Of course the major challenge for real world applications will be the development of linear dynamic models that capture the dominant dynamics of the real-world systems. Also, since the FOMPC designed requires knowledge of the state variables, a key design challenge will be the estimation of the systems state variables under possible model uncertainty.

References

[1] Abass, Y. A., Al-Awami, A. T., and Jamal, T. (2016). Integrating automatic generation control and economic dispatch for microgrid real-time optimization. In *Power and Energy Society General Meeting (PESGM), 2016*, pages 1–5. IEEE.

[2] Abou El-Ela, A. A., Mouwafi, M. T., and Elbaset, A. A. (2022). Modern optimization techniques for smart grids.

[3] Alessio, A., Barcelli, D., and Bemporad, A. (2011). Decentralized model predictive control of dynamically coupled linear systems. *Journal of Process Control*, 21(5):705–714.

[4] Alvarado, F. L., Meng, J., DeMarco, C. L., and Mota, W. S. (2001). Stability analysis of interconnected power systems coupled with market dynamics. *IEEE Transactions on power systems*, 16(4):695–701.

[5] Ariyur, K. B. and Krstic, M. (2003). *Real-time optimization by extremum-seeking control*. John Wiley & Sons.

[6] Arvanitis, K. and Kalogeropoulos, G. (1997). Guaranteed stability margins for discrete-time lq optimal regulators for the performance index with cross-product terms. *Circuits, Systems and Signal Processing*, 16(6):663–701.

[7] Astolfi, A. (2021). Tracking and regulation in linear systems. In *Encyclopedia of Systems and Control*, pages 2329–2335. Springer.

[8] Åström, K. J. and Wittenmark, B. (2013). *Adaptive control*. Courier Corporation.

[9] Asuk, A. and Trodden, P. (2021a). Feedback optimizing linear quadratic control. In *2021 American Control Conference (ACC)*, pages 3800–3805. IEEE.

[10] Asuk, A. and Trodden, P. (2021b). Feedback-optimizing model predictive control for constrained linear systems. *IFAC-PapersOnLine*, 54(6):43–49.

[11] Atic, N., Rerkpreedapong, D., Hasanovic, A., and Feliachi, A. (2003). Nerc compliant decentralized load frequency control design using model predictive control. In *Power Engineering Society General Meeting, 2003, IEEE*, volume 2, pages 554–559. IEEE.

[12] Bacher, R. and Van Meeteren, H. P. (1988). Real-time optimal power flow in automatic generation control. *IEEE Transactions on Power Systems*, 3(4):1518–1529.

[13] Baillieul, J. and Samad, T. (2015). *Encyclopedia of systems and control.* Springer Publishing Company, Incorporated.

[14] Balakrishnan, V. and Vandenberghe, L. (1995). Connections between duality in control theory and convex optimization. In *Proceedings of 1995 American Control Conference-ACC'95*, volume 6, pages 4030–4034. IEEE.

[15] Baocang, D. (2018). *Modern predictive control.* CRC press.

[16] Barbieri, E. and Alba-Flores, R. (2000). On the infinite-horizon lq tracker. *Systems & Control Letters*, 40(2):77–82.

[17] Bayer, F. A., Lorenzen, M., Müller, M. A., and Allgöwer, F. (2016). Robust economic model predictive control using stochastic information. *Automatica*, 74:151–161.

[18] Bazaraa, M. S., Sherali, H. D., and Shetty, C. M. (2013). *Nonlinear programming: theory and algorithms.* John Wiley & Sons.

[19] Bejestani, A. K., Annaswamy, A., and Samad, T. (2014). A hierarchical transactive control architecture for renewables integration in smart grids: Analytical modeling and stability. *IEEE Transactions on Smart Grid*, 5(4):2054–2065.

[20] Bejestani, K. et al. (2013). *Wholesale Energy Market in a Smart Grid: Dynamic modeling, stability, and robustness.* PhD thesis, Technische Universität München.

[21] Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust optimization.* Princeton university press.

[22] Bergen, A. R. (2009). *Power systems analysis.* Pearson Education India.

[23] Bernstein, A. and Dall'Anese, E. (2019). Real-time feedback-based optimization of distribution grids: A unified approach. *IEEE Transactions on Control of Network Systems*, 6(3):1197–1209.

[24] Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334.

[25] Betti, G., Farina, M., and Scattolini, R. (2012a). Distributed predictive control for tracking constant references. In *2012 American Control Conference (ACC)*, pages 6364–6369. IEEE.

[26] Betti, G., Farina, M., and Scattolini, R. (2012b). An mpc algorithm for offset-free tracking of constant reference signals. In *2012 IEEE 51st IEEE conference on decision and control (CDC)*, pages 5182–5187. IEEE.

[27] Betti, G., Farina, M., and Scattolini, R. (2013a). Decentralized predictive control for tracking constant references. In *52nd IEEE Conference on Decision and Control*, pages 5228–5233. IEEE.

[28] Betti, G., Farina, M., and Scattolini, R. (2013b). A robust mpc algorithm for offset-free tracking of constant reference signals. *IEEE Transactions on Automatic Control*, 58(9):2394–2400.

[29] Bevrani, H. and Hiyama, T. (2011). *Intelligent automatic generation control*. CRC press Boca Raton.

[30] Bianchin, G., Vaquero, M., Cortes, J., and Dall'Anese, E. (2021). Online stochastic optimization for unknown linear systems: Data-driven synthesis and controller analysis. *arXiv preprint arXiv:2108.13040*.

[31] Bonnans, J. F. (2019). *Convex and Stochastic Optimization*. Springer.

[32] Boulet, B. and Duan, Y. (2007). The fundamental tradeoff between performance and robustness-a new perspective on loop shaping-classic control revisited part ii. *IEEE Control Systems Magazine*, 27(3):30–44.

[33] Boyd, S., Boyd, S. P., and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

[34] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*. SIAM.

[35] Brdys, M. A. and Tatjewski, P. (2005). *Iterative algorithms for multilayer optimizing control*. World Scientific.

[36] Brunner, F. D., Dürr, H.-B., and Ebenbauer, C. (2012). Feedback design for multi-agent systems: A saddle point approach. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 3783–3789. IEEE.

[37] Cai, D., Mallada, E., and Wierman, A. (2017). Distributed optimization decomposition for joint economic dispatch and frequency regulation. *IEEE Transactions on Power Systems*, 32(6):4370–4385.

[38] Caliskan, S. Y. and Tabuada, P. (2015). Uses and abuses of the swing equation model. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 6662–6667. IEEE.

[39] Camacho, E. F. and Alba, C. B. (2013). *Model predictive control*. Springer science & business media.

[40] Camacho, E. F. and Bordons, C. (2016). *Model predictive control: Classical, robust and stochastic*. Springer International Publishing.

[41] Chen, X. (2022). *Distributed and Data-Driven Decision-Making for Sustainable Power Systems*. PhD thesis, Harvard University.

[42] Cherukuri, A. and Cortés, J. (2015). Asymptotic stability of saddle points under the saddle-point dynamics. In *2015 American Control Conference (ACC)*, pages 2020–2025. IEEE.

[43] Chong, E. K. and Zak, S. H. (2013). *An Introduction to Optimization*, volume 75. John Wiley & Sons.

[44] Cohn, N. (1956). Some aspects of tie-line bias control on interconnected power systems. *Transactions of the American institute of electrical engineers. Part III: power apparatus and systems*, 75(3):1415–1436.

[45] Colombino, M., Dall'Anese, E., and Bernstein, A. (2019a). Online optimization as a feedback controller: Stability and tracking. *IEEE Transactions on Control of Network Systems*.

[46] Colombino, M., Simpson-Porco, J. W., and Bernstein, A. (2019b). Towards robustness guarantees for feedback-based optimization. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 6207–6214. IEEE.

[47] Cortés, J. (2006). Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42(11):1993–2000.

[48] Dall'Anese, E., Dhople, S. V., and Giannakis, G. B. (2015). Regulation of dynamical systems to optimal solutions of semidefinite programs: Algorithms and applications to ac optimal power flow. In *2015 American Control Conference (ACC)*, pages 2087–2092. IEEE.

[49] Dall'Anese, E. and Simonetto, A. (2016). Optimal power flow pursuit. *IEEE Transactions on Smart Grid*, 9(2):942–952.

[50] De Persis, C., Monshizadeh, N., Schiffer, J., and Dörfler, F. (2016). A lyapunov approach to control of microgrids with a network-preserved differential-algebraic model. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 2595–2600. IEEE.

[51] DeHaan, D. and Guay, M. (2005). Extremum-seeking control of state-constrained nonlinear systems. *Automatica*, 41(9):1567–1574.

[52] Di Ruscio, D. (2012). Discrete lq optimal control with integral action: A simple controller on incremental form for mimo systems.

[53] D'jorge, A., Anderson, A., González, A. H., and Ferramosca, A. (2018). A robust economic mpc for changing economic criterion. *International Journal of Robust and Nonlinear Control*, 28(15):4404–4423.

[54] Donglei, S., Xueshan, H., Zhang, B., Mingqiang, W., and Tianyu, D. (2016). Frequency aware robust economic dispatch. *Journal of Modern Power Systems and Clean Energy*, 4(2):200–210.

[55] Dörfler, F., Bolognani, S., Simpson-Porco, J. W., and Grammatico, S. (2019). Distributed control and optimization for autonomous power grids. In *2019 18th European Control Conference (ECC)*, pages 2436–2453. IEEE.

[56] Dörfler, F., Simpson-Porco, J. W., and Bullo, F. (2016). Breaking the hierarchy: Distributed control and economic optimality in microgrids. *IEEE Transactions on Control of Network Systems*, 3(3):241–253.

[57] Driessen, P., Hermans, R. M., and van den Bosch, P. (2012). Distributed economic model predictive control of networks in competitive environments. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 266–271. IEEE.

[58] Duan, Y. (2006). Fundamental tradeoff between performance and robustness in control design: Yingxuan duan.

[59] D'Jorge, A., Ferramosca, A., and González, A. H. (2017). A robust gradient-based mpc for integrating real time optimizer (rto) with control. *Journal of Process Control*, 54:65–80.

[60] Egerstedt, M. B. (2007). Graph-theoretic methods for multi-agent coordination. Georgia Institute of Technology.

[61] Eidson, D. B. and Ilic, M. D. (1995). Advanced generation control with economic dispatch. In *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, volume 4, pages 3450–3458. IEEE.

[62] Elgerd, O. I. and Fosha, C. E. (1970). Optimum megawatt-frequency control of multiarea electric energy systems. *IEEE Transactions on Power Apparatus and Systems*, (4):556–563.

[63] Ellis, M., Durand, H., and Christofides, P. D. (2014). A tutorial review of economic model predictive control methods. *Journal of Process Control*, 24(8):1156–1178.

[64] Ellis, M., M. L. and Christofides, P. (2017). *Economic Model Predictive Control: Theory, Formulations and Chemical Process Applications*. Springer, Berlin.

[65] Engell, S. (2007). Feedback control for optimal process operation. *Journal of process control*, 17(3):203–219.

[66] Ersdal, A. M., Imsland, L., and Uhlen, K. (2016). Model predictive load-frequency control. *IEEE Transactions on Power Systems*, 31(1):777–785.

[67] Farina, M., Betti, G., Giulioni, L., and Scattolini, R. (2013a). An approach to distributed predictive control for tracking–theory and applications. *IEEE Transactions on Control Systems Technology*, 22(4):1558–1566.

[68] Farina, M., Betti, G., and Scattolini, R. (2013b). A solution to the tracking problem using distributed predictive control. In *2013 European Control Conference (ECC)*, pages 4347–4352. IEEE.

[69] Farina, M., Colaneri, P., and Scattolini, R. (2013c). Block-wise discretization accounting for structural constraints. *Automatica*, 49(11):3411–3417.

[70] Farina, M. and Scattolini, R. (2011). Distributed non-cooperative mpc with neighbor-to-neighbor communication. *IFAC Proceedings Volumes*, 44(1):404–409.

[71] Farina, M. and Scattolini, R. (2012). Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*, 48(6):1088–1096.

[72] Farina, M. and Scattolini, R. (2019). Distributed mpc for large-scale systems. In *Handbook of Model Predictive Control*, pages 239–258. Springer.

[73] Faulwasser, T., Grüne, L., Müller, M. A., et al. (2018). Economic nonlinear model predictive control. *Foundations and Trends® in Systems and Control*, 5(1):1–98.

[74] Faulwasser, T. and Pannocchia, G. (2019). Toward a unifying framework blending real-time optimization and economic model predictive control. *Industrial & Engineering Chemistry Research*, 58(30):13583–13598.

[75] Ferramosca, A., González, A. H., and Limon, D. (2015). Economic optimality in mpc: A comparative study. In *2015 American Control Conference (ACC)*, pages 2555–2560. IEEE.

[76] Ferramosca, A., Limon, D., and Camacho, E. F. (2014). Economic mpc for a changing economic criterion for linear systems. *IEEE Transactions on Automatic Control*, 59(10):2657–2667.

[77] Forbes, M. G., Patwardhan, R. S., Hamadah, H., and Gopaluni, R. B. (2015). Model predictive control in industry: Challenges and opportunities. *IFAC-PapersOnLine*, 48(8):531–538.

[78] Forsman, K. (2016). Implementation of advanced control in the process industry without the use of mpc. *IFAC-PapersOnLine*, 49(7):514–519.

[79] Fosha, C. E. and Elgerd, O. I. (1970). The megawatt-frequency control problem: a new approach via optimal control theory. *IEEE Transactions on Power Apparatus and Systems*, (4):563–577.

[80] Gao, B., Hong, J., Yu, S., and Chen, H. (2017). Linear-quadratic output regulator with disturbance rejection: Application to vehicle launch control. In *2017 American Control Conference (ACC)*, pages 1960–1965. IEEE.

[81] Gilbert, E. G. and Tan, K. T. (1991). Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *IEEE Transactions on Automatic control*, 36(9):1008–1020.

[82] Goel, G., Chen, N., and Wierman, A. (2017). Thinking fast and slow: Optimization decomposition across timescales. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 1291–1298. IEEE.

[83] Golub, G. H. and Van Loan, C. F. (2013). *Matrix computations*. JHU press.

[84] Gomez-Exposito, A., Conejo, A. J., and Canizares, C. (2018). *Electric energy systems: analysis and operation*. CRC press.

[85] Häberle, V., Hauswirth, A., Ortmann, L., Bolognani, S., and Dörfler, F. (2020). Non-convex feedback optimization with input and output constraints. *IEEE Control Systems Letters*, 5(1):343–348.

[86] Halter, W., Michalowsky, S., and Allgöwer, F. (2019). Extremum seeking for optimal enzyme production under cellular fitness constraints. In *2019 18th European Control Conference (ECC)*, pages 2159–2164. IEEE.

[87] Hauswirth, A., Bolognani, S., Hug, G., and Dörfler, F. (2021). Optimization algorithms as robust feedback controllers. *arXiv preprint arXiv:2103.11329*.

[88] Hauswirth, A., Bolognani, S., Hug, G., and Drfler, F. (2020). Timescale separation in autonomous optimization. *IEEE Transactions on Automatic Control*.

[89] Hauswirth, A., Zanardi, A., Bolognani, S., Dörfler, F., and Hug, G. (2017). Online optimization in closed loop on the power flow manifold. In *2017 IEEE Manchester PowerTech*, pages 1–6. IEEE.

[90] He, Z., Bolognani, S., He, J., Dörfler, F., and Guan, X. (2022). Model-free nonlinear feedback optimization. *arXiv preprint arXiv:2201.02395*.

[91] Heidarinejad, M., Liu, J., and Christofides, P. D. (2012). Economic model predictive control of nonlinear process systems using lyapunov techniques. *AIChE Journal*, 58(3):855–870.

[92] Hermans, R., Lazar, M., and Jokić, A. (2011). Distributed predictive control of the 7-machine cigré power system. In *American Control Conference (ACC), 2011*, pages 5225–5230. IEEE.

[93] Hermans, R. M., Jokić, A., Lazar, M., Alessio, A., Van den Bosch, P. P., Hiskens, I. A., and Bemporad, A. (2012). Assessment of non-centralised model predictive control techniques for electrical power networks. *International journal of control*, 85(8):1162–1177.

[94] Hernandez, B. and Trodden, P. (2016). Distributed model predictive control using a chain of tubes. In *2016 UKACC 11th International Conference on Control (CONTROL)*, pages 1–6. IEEE.

[95] Hernandez Vicente, B. A. (2018). *Model predictive control for linear systems: adaptive, distributed and switching implementations*. PhD thesis, University of Sheffield.

[96] Ilic, M., Jaddivada, R., Miao, X., and Popli, N. (2019). Toward multi-layered mpc for complex electric energy systems. In *Handbook of Model Predictive Control*, pages 625–663. Springer.

[97] Ilic, M. and Yu, C.-N. (1996). Minimal system regulation and its value in a changing industry. In *Control Applications, 1996., Proceedings of the 1996 IEEE International Conference on*, pages 442–449. IEEE.

[98] Ilic, M. D. (2007). From hierarchical to open access electric power systems. *Proceedings of the IEEE*, 95(5):1060–1084.

[99] Ilic, M. D. and Zaborszky, J. (2000). *Dynamics and control of large electric power systems*. Wiley New York.

[100] Jahn, J. (2020). *Introduction to the theory of nonlinear optimization*. Springer Nature.

[101] Jäschke, J., Cao, Y., and Kariwala, V. (2017). Self-optimizing control–a survey. *Annual Reviews in Control*, 43:199–223.

[102] Jia, D. and Krogh, B. (2002). Min-max feedback model predictive control for distributed control with communication. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, volume 6, pages 4507–4512. IEEE.

[103] Jokic, A. (2007). *Price-based optimal control of electrical power systems*. PhD thesis, Eindhoven University of Technology.

[104] Jokic, A., Lazar, M., and van den Bosch, P. P. (2009). On constrained steady-state regulation: Dynamic kkt controllers. *IEEE Transactions on Automatic Control*, 54(9):2250–2254.

[105] Jokić, A., Lazar, M., and van den Bosch, P. P. (2009). Real-time control of power systems using nodal prices. *International Journal of Electrical Power & Energy Systems*, 31(9):522–530.

[106] Jokic, A., van den Bosch, P. P., and Lazar, M. (2007). Distributed price-based optimal control of power systems. In *2007 IEEE International Conference on Control Applications*, pages 910–915. IEEE.

[107] K. arrow, L. h. and h. uzawa (1960). Studies in linear and non-linear programming. *Economic Review*, 11(3):349–351.

[108] Karimi-Ghartemani, M., Khajehoddin, S. A., Jain, P., and Bakhshai, A. (2011). Linear quadratic output tracking and disturbance rejection. *International Journal of control*, 84(8):1442–1449.

[109] Kawamura, S., Cai, K., and Kishida, M. (2020). Distributed output regulation of heterogeneous uncertain linear agents. *Automatica*, 119:109094.

[110] Kelly, F. P., Maulloo, A. K., and Tan, D. K. (1998). Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research society*, 49(3):237–252.

[111] Khlebnikov, M. V., Shcherbakov, P. S., and Chestnov, V. N. (2015). Linear-quadratic regulator. i. a new solution. *Automation and Remote Control*, 76(12):2143–2155.

[112] Köhler, J. (2017). Distributed economic model predictive control under inexact minimization with application to power systems. Master's thesis.

[113] Köhler, J., Müller, M. A., Li, N., and Allgöwer, F. (2017). Real time economic dispatch for power networks: A distributed economic model predictive control approach. In *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, pages 6340–6345. IEEE.

[114] Kolmanovsky, I. and Gilbert, E. G. (1998). Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical problems in engineering*, 4(4):317–367.

[115] Kong, L. and Xiao, L. (2007). A new model predictive control scheme-based load-frequency control. In *Control and Automation, 2007. ICCA 2007. IEEE International Conference on*, pages 2514–2518. IEEE.

[116] Kothare, M. V., Balakrishnan, V., and Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379.

[117] Krishnamoorthy, D. (2021). A distributed feedback-based online process optimization framework for optimal resource sharing. *Journal of Process Control*, 97:72–83.

[118] Krishnamoorthy, D., Jahanshahi, E., and Skogestad, S. (2018). Feedback real-time optimization strategy using a novel steady-state gradient estimate and transient measurements. *Industrial & Engineering Chemistry Research*, 58(1):207–216.

[119] Krishnamoorthy, D. and Skogestad, S. (2022). Real-time optimization as a feedback control problem-a review. *Computers & Chemical Engineering*, page 107723.

[120] Kumar, P., Kothari, D. P., et al. (2005). Recent philosophies of automatic generation control strategies in power systems. *IEEE transactions on power systems*, 20(1):346–357.

[121] Kundur, P., Balu, N. J., and Lauby, M. G. (1994). *Power system stability and control*, volume 7. McGraw-hill New York.

[122] Kwatny, H. and Athay, T. (1979). Coordination of economic dispatch and load-frequency control in electric power systems. In *Decision and Control including the Symposium on Adaptive Processes, 1979 18th IEEE Conference on*, volume 18, pages 703–714. IEEE.

[123] Kwatny, H., Kalnitsky, K., and Bhatt, A. (1975). An optimal tracking approach to load-frequency control. *IEEE Transactions on Power Apparatus and Systems*, 94(5):1635–1643.

[124] Kwatny, Harry G., M.-M. K. (2016). *Power System Dynamics and Control*. Springer Science+Business Media New York.

[125] Lawrence, L. S., Nelson, Z. E., Mallada, E., and Simpson-Porco, J. W. (2018a). Optimal steady-state control for linear time-invariant systems. *arXiv preprint arXiv:1810.03724*.

[126] Lawrence, L. S., Simpson-Porco, J. W., and Mallada, E. (2018b). Linear-convex optimal steady-state control. *arXiv preprint arXiv:1810.12892*.

[127] Lawrence, L. S., Simpson-Porco, J. W., and Mallada, E. (2018c). The optimal steady-state control problem. *arXiv preprint arXiv:1810.12892*.

[128] Lawrence, L. S. P., Nelson, Z. E., Mallada, E., and Simpson-Porco, J. W. (2018). Optimal steady-state control for linear time-invariant systems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 3251–3257.

[129] Lawrence, L. S. P., Simpson-Porco, J. W., and Mallada, E. (2020). Linear-convex optimal steady-state control. *IEEE Transactions on Automatic Control*.

[130] Lee, J.-O., Kim, Y.-S., Kim, E.-S., and Moon, S.-I. (2018). Generation adjustment method based on bus-dependent participation factor. *IEEE Transactions on Power Systems*, 33(2):1959–1969.

[131] Levine, W. and Athans, M. (1970). On the determination of the optimal constant output feedback gains for linear multivariable systems. *IEEE Transactions on Automatic control*, 15(1):44–48.

[132] Levine, W. S. (2011). *The Control Handbook (three volume set)*. CRC press.

[133] Lewis, F. L., Vrabie, D., and Syrmos, V. L. (2012). *Optimal control*. John Wiley & Sons.

[134] Li, N., Zhao, C., and Chen, L. (2016). Connecting automatic generation control and economic dispatch from an optimization view. *IEEE Transactions on Control of Network Systems*, 3(3):254–264.

[135] Limon, D. and Alamo, T. (2015). Tracking model predictive control. In *Encyclopedia of Systems and Control*, pages 1475–1484. Springer.

[136] Liu, X., Kong, X., and Deng, X. (2012). Power system model predictive load frequency control. In *American Control Conference (ACC), 2012*, pages 6602–6607. IEEE.

[137] Liu, X., Zhan, X., and Qian, D. (2010). Load frequency control considering generation rate constraints. In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pages 1398–1401. IEEE.

[138] Liu, X., Zhang, Y., and Lee, K. Y. (2016). Robust distributed mpc for load frequency control of uncertain power systems. *Control Engineering Practice*, 56:136–147.

[139] Liu, X., Zhang, Y., and Lee, K. Y. (2017). Coordinated distributed mpc for load frequency control of power system with wind farms. *IEEE Transactions on Industrial Electronics*, 64(6):5140–5150.

[140] Low, S. H., Paganini, F., and Doyle, J. C. (2002). Internet congestion control. *IEEE control systems magazine*, 22(1):28–43.

[141] Luenberger, D. (2008). *Linear and nonlinear programming*. Springer Science+ Business Media, LLC.

[142] Ma, M., Chen, H., Liu, X., and Allgöwer, F. (2014). Distributed model predictive load frequency control of multi-area interconnected power system. *International Journal of Electrical Power & Energy Systems*, 62:289–298.

[143] Ma, M., Zhang, C., Liu, X., and Chen, H. (2017). Distributed model predictive load frequency control of the multi-area power system after deregulation. *IEEE Transactions on Industrial Electronics*, 64(6):5129–5139.

[144] Maeder, U., Borrelli, F., and Morari, M. (2009). Linear offset-free model predictive control. *Automatica*, 45(10):2214–2222.

[145] Maestre, J. M., Negenborn, R. R., et al. (2014). *Distributed model predictive control made easy*, volume 69. Springer.

[146] Magni, L. and Scattolini, R. (2006). Stabilizing decentralized model predictive control of nonlinear systems. *Automatica*, 42(7):1231–1236.

[147] Mallada, E., Zhao, C., and Low, S. (2017). Optimal load-side control for frequency regulation in smart grids. *IEEE Transactions on Automatic Control*, 62(12):6294–6309.

[148] Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *Industrial & engineering chemistry research*, 48(13):6022–6033.

[149] Marchetti, A., Chachuat, B., and Bonvin, D. (2010). A dual modifier-adaptation approach for real-time optimization. *Journal of Process Control*, 20(9):1027–1037.

[150] Marchetti, A. G. (2013). A new dual modifier-adaptation approach for iterative process optimization with inaccurate models. *Computers & Chemical Engineering*, 59:89–100.

[151] Marruedo, D. L., Alamo, T., and Camacho, E. (2002). Stability analysis of systems with bounded additive uncertainties based on invariant sets: Stability and feasibility of mpc. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, volume 1, pages 364–369. IEEE.

[152] Mayne, D. Q. (2014). Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986.

[153] Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814.

[154] Mayne, D. Q., Seron, M. M., and Raković, S. (2005). Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224.

[155] Mc Namara, P. and Milano, F. (2018). Efficient implementation of mpc-based agc for real-world systems with low inertia. *Electric Power Systems Research*, 158:315–323.

[156] Menta, S., Hauswirth, A., Bolognani, S., Hug, G., and Dörfler, F. (2018). Stability of dynamic feedback optimization with applications to power systems. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 136–143. IEEE.

[157] Miao, Z. and Fan, L. (2017). Achieving economic operation and secondary frequency regulation simultaneously through local feedback control. *IEEE Transactions on Power Systems*, 32(1):85–93.

[158] Michalowsky, S. and Ebenbauer, C. (2016). Extremum control of linear systems based on output feedback. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 2963–2968. IEEE.

[159] Mohamed, T., Bevrani, H., Hassan, A., and Hiyama, T. (2011). Decentralized model predictive based load frequency control in an interconnected power system. *Energy Conversion and Management*, 52(2):1208–1214.

[160] Molzahn, D. K., Dörfler, F., Sandberg, H., Low, S. H., Chakrabarti, S., Baldick, R., and Lavaei, J. (2017). A survey of distributed optimization and control algorithms for electric power systems. *IEEE Transactions on Smart Grid*, 8(6):2941–2962.

[161] Monasterios, P. R. B. and Trodden, P. (2017). Low-complexity distributed predictive automatic generation control with guaranteed properties. *IEEE Transactions on Smart Grid*, 8(6):3045–3054.

[162] Morari, M., Arkun, Y., and Stephanopoulos, G. (1980). Studies in the synthesis of control structures for chemical processes: Part i: Formulation of the problem. process decomposition and the classification of the control tasks. analysis of the optimizing control structures. *AIChE Journal*, 26(2):220–232.

[163] Morattab, A., Shafiee, Q., and Bevrani, H. (2011). Decentralized model predictive load-frequency control for deregulated power systems in a tough situation. In *PowerTech, 2011 IEEE Trondheim*, pages 1–5. IEEE.

[164] Mudumbai, R. and Dasgupta, S. (2014). Distributed control for the smart grid: The case of economic dispatch. In *Information Theory and Applications Workshop (ITA), 2014*, pages 1–6. IEEE.

[165] Muhanji, S. O., Muzhikyan, A., and Farid, A. M. (2018). Distributed control for distributed energy resources: Long-term challenges and lessons learned. *IEEE Access*, 6:32737–32753.

[166] Nassar, I., AlAli, S., and Weber, H. (2014). Effects of increasing intermittent generation on the frequency control of the european power system. *IFAC Proceedings Volumes*, 47(3):3134–3139.

[167] Nelson, Z. E. and Mallada, E. (2018). An integral quadratic constraint framework for real-time steady-state optimization of linear time-invariant systems. In *2018 Annual American Control Conference (ACC)*, pages 597–603. IEEE.

[168] Nonhoff, M. and Müller, M. A. (2021). Data-driven online convex optimization for control of dynamical systems. *arXiv preprint arXiv:2103.09127*.

[169] Ortmann, L., Hauswirth, A., Caduff, I., Dörfler, F., and Bolognani, S. (2019). Experimental validation of feedback optimization in power distribution grids. *arXiv preprint arXiv:1910.03384*.

[170] Pannocchia, G. (2018). An economic mpc formulation with offset-free asymptotic performance. *IFAC-PapersOnLine*, 51(18):393–398.

[171] Pannocchia, G., Gabiccini, M., and Artoni, A. (2015). Offset-free mpc explained: novelties, subtleties, and applications. *IFAC-PapersOnLine*, 48(23):342–351.

[172] Pannocchia, G. and Rawlings, J. B. (2001). The velocity algorithm LQR: A survey. Technical Report 2001-01, Texas–Wisconsin Modeling and Control Consortium.

[173] Pasquini, M., Colin, K., Chotteau, V., and Hjalmarsson, H. (2021). Robustness of a feedback optimization scheme with application to bioprocess manufacturing. *arXiv preprint arXiv:2103.16567*.

[174] Picallo, M., Ortmann, L., Bolognani, S., and Dörfler, F. (2021). Adaptive real-time grid operation via online feedback optimization with sensitivity estimation. *arXiv preprint arXiv:2110.00954*.

[175] Polyak, B., Shcherbakov, P., and Khlebnikov, M. (2014). Control of linear systems subjected to exogenous disturbances: the linear matrix inequality technique. *Moscow: LENAND*.

[176] Pourmousavi, S. A. and Nehrir, M. H. (2014). Introducing dynamic demand response in the lfc model. *IEEE Transactions on Power Systems*, 29(4):1562–1572.

[177] Qin, S. J. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764.

[178] Raimondo, D., Magni, L., and Scattolini, R. (2007). Decentralized mpc of nonlinear systems: An input-to-state stability approach. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 17(17):1651–1667.

[179] Rakovic, S. V., Kerrigan, E. C., Kouramas, K. I., and Mayne, D. Q. (2005). Invariant approximations of the minimal robust positively invariant set. *IEEE Transactions on automatic control*, 50(3):406–410.

[180] Rakovic, S. V. and Levine, W. S. (2019). *Handbook of Model Predictive Control*. Springer International Publishing AG, 1st edition edition.

[181] Rawlings, J. B. and Mayne, D. Q. (2009). *Model predictive control: Theory and design*. Nob Hill Pub.

[182] Rawlings, J. B., Mayne, D. Q., and Diehl, M. M. (2017). *Model Predictive Control: Theory, Computation, and Design*. Nob Hill, 2nd edition edition.

[183] Rerkpreedapong, D., Atic, N., and Feliachi, A. (2003). Economy oriented model predictive load frequency control. In *Power Engineering, 2003 Large Engineering Systems Conference on*, pages 12–16. IEEE.

[184] Riverso, S., Farina, M., and Ferrari-Trecate, G. (2013). Plug-and-play decentralized model predictive control for linear systems. *IEEE Transactions on Automatic Control*, 58(10):2608–2614.

[185] Riverso, S. and Ferrari-Trecate, G. (2012). Tube-based distributed control of linear constrained systems. *Automatica*, 48(11):2860–2865.

[186] Saadat, H. (1999). *Power System Analysis McGraw-Hill Series in Electrical Computer Engineering*, volume 2. McGraw-hill.

[187] Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control–a review. *Journal of process control*, 19(5):723–731.

[188] Schiffer, J. and Dörfler, F. (2016). On stability of a distributed averaging pi frequency and active power controlled differential-algebraic power system model. In *2016 European Control Conference (ECC)*, pages 1487–1492. IEEE.

[189] Schiffer, J., Dörfler, F., and Fridman, E. (2017). Robustness of distributed averaging control in power systems: Time delays & dynamic communication topology. *Automatica*, 80:261–271.

[190] Shayeghi, H., Shayanfar, H., and Jalili, A. (2009). Load frequency control strategies: A state-of-the-art survey for the researcher. *Energy Conversion and management*, 50(2):344–353.

[191] Shi, X., Hu, J., Yu, J., Yong, T., and Cao, J. (2015). A novel load frequency control strategy based on model predictive control. In *Power & Energy Society General Meeting, 2015 IEEE*, pages 1–5. IEEE.

[192] Shiltz, D. J., Baros, S., Cvetković, M., and Annaswamy, A. M. (2017). Integration of automatic generation control and demand response via a dynamic regulation market mechanism. *IEEE Transactions on Control Systems Technology*, (99):1–16.

[193] Shiltz, D. J., Cvetković, M., and Annaswamy, A. M. (2016). An integrated dynamic market mechanism for real-time markets and frequency regulation. *IEEE Transactions on Sustainable Energy*, 7(2):875–885.

[194] Shiroei, M. and Ranjbar, A. (2014). Supervisory predictive control of power system load frequency control. *International Journal of Electrical Power & Energy Systems*, 61:70–80.

[195] Shiroei, M., Ranjbar, A. M., and Amraee, T. (2013a). A functional model predictive control approach for power system load frequency control considering generation rate constraint. *International Transactions on Electrical Energy Systems*, 23(2):214–229.

[196] Shiroei, M., Toulabi, M. R., and Ranjbar, A. M. (2013b). Robust multivariable predictive based load frequency control considering generation rate constraint. *International Journal of Electrical Power & Energy Systems*, 46:405–413.

[197] Simpson-Porco, J. W. (2020). On area control errors, area injection errors, and textbook automatic generation control. *IEEE Transactions on Power Systems*, 36(1):557–560.

[198] Skogestad, S. (2000a). Plantwide control: The search for the self-optimizing control structure. *Journal of process control*, 10(5):487–507.

[199] Skogestad, S. (2000b). Self-optimizing control: The missing link between steady-state optimization and control. *Computers & Chemical Engineering*, 24(2-7):569–575.

[200] Skogestad, S. (2009). Feedback: Still the simplest and best solution.

[201] Sokoler, L. E., Edlund, K., and Jørgensen, J. B. (2015). Application of economic mpc to frequency control in a single-area power system. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 2635–2642. IEEE.

[202] Stegink, T., Cherukuri, A., De Persis, C., van der Schaft, A., and Cortés, J. (2017). Frequency-driven market mechanisms for optimal dispatch in power networks. *arXiv preprint arXiv:1801.00137*.

[203] Stegink, T., De Persis, C., and van der Schaft, A. (2016). Optimal power dispatch in networks of high-dimensional models of synchronous machines. *arXiv preprint arXiv:1603.06688*.

[204] Stewart, B. T., Venkat, A. N., Rawlings, J. B., Wright, S. J., and Pannocchia, G. (2010). Cooperative distributed model predictive control. *Systems & Control Letters*, 59(8):460–469.

[205] Streiffert, D. (1995). Multi-area economic dispatch with tie line constraints. *IEEE Transactions on Power Systems*, 10(4):1946–1951.

[206] Tang, Y. (2018). Distributed optimal steady-state regulation for high-order multiagent systems with external disturbances. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(11):4828–4835.

[207] Tarbouriech, S., Garcia, G., da Silva Jr, J. M. G., and Queinnec, I. (2011). *Stability and stabilization of linear systems with saturating actuators*. Springer Science & Business Media.

[208] Terpin, A., Fricker, S., Perez, M., de Badyn, M. H., and Dörfler, F. (2021). Distributed feedback optimisation for robotic coordination. *arXiv preprint arXiv:2109.14486*.

[209] Thatte, A. A., Zhang, F., and Xie, L. (2011). Frequency aware economic dispatch. In *North American Power Symposium (NAPS), 2011*, pages 1–7. IEEE.

[210] Trip, S., Bürger, M., and De Persis, C. (2016). An internal model approach to (optimal) frequency regulation in power grids with time-varying voltages. *Automatica*, 64:240–253.

[211] Trip, S. and De Persis, C. (2017). Distributed optimal load frequency control with non-passive dynamics. *IEEE Transactions on Control of Network Systems*, 5(3):1232–1244.

[212] Trodden, P. (2016). A one-step approach to computing a polytopic robust positively invariant set. *IEEE Transactions on Automatic Control*, 61(12):4100–4105.

[213] Trodden, P. and Richards, A. (2010). Distributed model predictive control of linear systems with persistent disturbances. *International Journal of Control*, 83(8):1653–1663.

[214] Turkington, D. A. (2013). *Generalized vectorization, cross-products, and matrix calculus*. Cambridge University Press.

[215] Vaccari, M. and Pannocchia, G. (2016). A modifier-adaptation strategy towards offset-free economic mpc. *Processes*, 5(1):2.

[216] Venkat, A. N., Hiskens, I. A., Rawlings, J. B., and Wright, S. J. (2008). Distributed mpc strategies with application to power system automatic generation control. *IEEE transactions on control systems technology*, 16(6):1192–1206.

[217] Wang, J. and Elia, N. (2011). A control perspective for centralized and distributed convex optimization. In *2011 50th IEEE conference on decision and control and European control conference*, pages 3800–3805. IEEE.

[218] Wang, Z., Liu, F., Low, S. H., Zhao, C., and Mei, S. (2016). Decentralized optimal frequency control of interconnected power systems with transient constraints. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 664–671. IEEE.

[219] Wang, Z., Liu, F., Low, S. H., Zhao, C., and Mei, S. (2017). Distributed frequency control with operational constraints, part i: Per-node power balance. *IEEE Transactions on Smart Grid*.

[220] Weitenberg, E. and De Persis, C. (2018). Robustness to noise of distributed averaging integral controllers in power networks. *Systems & Control Letters*, 119:1–7.

[221] Wood, A. J. and Wollenberg, B. F. (2012). *Power generation, operation, and control*. John Wiley & Sons.

[222] Wu, X., Dörfler, F., and Jovanović, M. R. (2016). Topology identification and design of distributed integral action in power networks. In *2016 American Control Conference (ACC)*, pages 5921–5926. IEEE.

[223] Yu, C.-N. (1996). *Real power and frequency control of large electric power systems under open access*. PhD thesis, Massachusetts Institute of Technology.

[224] Zaborszky, J. (1984). A large system approach toward operating the electric power system by decision and control. In *American Control Conference, 1984*, pages 1143–1155. IEEE.

[225] Zaborszky, J. and Ying Chiang, T. (1975). Economic areawise load frequency control. Technical report, Washington Univ., St. Louis, Mo.(USA). Dept. of Systems Science and Mathematics.

[226] Zhang, X. and Papachristodoulou, A. (2015). A real-time control framework for smart power networks: Design methodology and stability. *Automatica*, 58:43–50.

[227] Zhang, X., Papachristodoulou, A., and Li, N. (2015). Distributed optimal steady-state control using reverse-and forward-engineering. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 5257–5264. IEEE.

[228] Zhang, X., Papachristodoulou, A., and Li, N. (2017). Distributed control for reaching optimal steady state in network systems: An optimization approach. *IEEE Transactions on Automatic Control*, 63(3):864–871.

[229] Zhao, C., Mallada, E., and Dörfler, F. (2015). Distributed frequency control for stability and economic dispatch in power networks. In *2015 American Control Conference (ACC)*, pages 2359–2364. IEEE.

[230] Zhao, C., Mallada, E., Low, S. H., and Bialek, J. (2018). Distributed plug-and-play optimal generator and load control for power system frequency regulation. *International Journal of Electrical Power & Energy Systems*, 101:1–12.

[231] Zhao, C., Topcu, U., Li, N., and Low, S. (2014). Design and stability of load-side primary frequency control in power systems. *IEEE Transactions on Automatic Control*, 59(5):1177–1189.

[232] Zheng, T., Simpson-Porco, J., and Mallada, E. (2019). Implicit trajectory planning for feedback linearizable systems: A time-varying optimization approach. *arXiv preprint arXiv:1910.00678*.

[233] Zhu, Y., Yu, W., Wen, G., and Chen, G. (2018). Projected primal–dual dynamics for distributed constrained nonsmooth convex optimization. *IEEE Transactions on Cybernetics*, 50(4):1776–1782.

Appendix A

Mathematical Background and Preliminaries

This section briefly recalls fundamental concepts used in the thesis.

A.1 Linear Algebra

Definition 6 (Subspace of the vector space $\mathbb{R}^n$)**.** *A subspace of $\mathbb{R}^n$ is a subset that is also a vector space.*

Definition 7 (Linear combination, linear independence and linear dependence)**.** *For the collection of vectors $a_1, \ldots, a_n \in \mathbb{R}^n$ and the elements $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ the following notions are introduced:*

1. ***Linear combinations** of $a_1, \ldots, a_n$ over $\mathbb{R}$ are vectors of the form*

$$\alpha_1 a_1 + \alpha_2 a_2 + \ldots + \alpha_n a_n \tag{A.1}$$

2. *The collection of vectors $a_1, \ldots, a_n \in \mathbb{R}^n$ are **linearly independent** if the vector equation*

$$\alpha_1 a_1 + \alpha_2 a_2 + \ldots + \alpha_n a_n = 0 \tag{A.2}$$

   *implies $\alpha_1 = \alpha_2 = \ldots = \alpha_n = 0$. Otherwise, a non-trivial combination of the a_i is zero and $a_1, \ldots, a_n$ is said to be **linearly dependent**.*

Definition 8 (Span)**.** *Given a collection of vectors $a_1, \ldots, a_n \in \mathbb{R}^n$, the set of all linear combinations of these vectors is a subspace referred to as the span of $a_1, \ldots, a_n$.*

Definition 9 (Basis). *Let the vectors $a_1, \ldots, a_n$ belong to the subspace, $\mathbb{S} \subset \mathbb{R}^n$. A basis of $\mathbb{S}$ is a set of vectors $\{s_1, \ldots, s_m\}$ such that:*

1. *$\mathbb{S} = span\{s_1, \ldots, s_m\}$, and*

2. *the vectors $s_1, \ldots, s_m$ are linearly independent.*

Definition 10 (Dimension of a subspace, $\dim(\mathbb{S})$). *The dimension of a subspace $\mathbb{S}$ is the number of element in the basis vector of the subspace.*

Definition 11 (Range of a matrix). *The range of a matrix $A \in \mathbb{R}^{m\times n}$ is defined by*

$$\mathcal{R}(A) = \{y \in \mathbb{R}^m \mid \ y = Ax \text{ for some } x \in \mathbb{R}^n\}. \tag{A.3}$$

If $A = [a_1 \ldots a_i \ldots a_n]$, where $a_i \in \mathbb{R}^m$ then

$$\mathcal{R}(A) = \text{span}\{a_1 \ldots a_i \ldots a_n\}. \tag{A.4}$$

Definition 12 (Nullspace of a matrix). *The nullspace of a matrix $A \in \mathbb{R}^{m\times n}$ is defined by*

$$\mathcal{N}(A) = \{x \in \mathbb{R}^n \mid \ Ax = 0\}. \tag{A.5}$$

Definition 13 (Rank of a matrix). *The rank of a matrix A is the maximal number of linearly independent columns (or rows) and is defined by*

$$\text{rank}(A) = \dim(\mathcal{R}(A)). \tag{A.6}$$

Definition 14 (Orthogonal complement of a subspace). *The orthogonal complement of a subspace $\mathbb{S} \subseteq \mathbb{R}^m$ is defined by*

$$\mathbb{S}^\perp = \{y \in \mathbb{R}^m \mid \ y^\top x = 0 \ \forall x \in \mathbb{S}\}. \tag{A.7}$$

Theorem A.1.1 (Rank-nullity theorem). *Let $A \in \mathbb{R}^{m\times n}$ be the matrix for the linear transformation $T : \mathbb{R}^n \to \mathbb{R}^m$, then*

$$\text{rank}(A)^\perp = \dim(\mathcal{N}(A^\top)) \tag{A.8}$$

and

$$\dim(\mathcal{N}(A)) + \text{rank}(A) = n. \tag{A.9}$$

We say $A \in \mathbb{R}^{m\times n}$ is rank deficient or not full rank if $\text{rank}(A) < \min\{m, n\}$.

A.2 Convex Analysis

Definition 15 (Ball of radius r). *The ball of radius r centred at the x_0 is the set*

$$\mathscr{B}_r(x_0) \coloneqq \{x \in \mathbb{R}^n \mid \|x - x_0\| \leq r\}. \tag{A.10}$$

Definition 16 (Interior point). *A point $x_0 \in \mathbb{S}$ is an interior point in $\mathbb{S}$ if there is a small ball $\mathscr{B}_r(x_0)$ centred at x_0 that lies entirely in $\mathbb{S}$ i.e.,*

$$x_0 \text{ is an interior point if } \ \mathscr{B}_r(x_0) \subset \mathbb{S} \text{ for some } r > 0. \tag{A.11}$$

If $x_0 \in int\ \mathbb{S}$, we call $\mathbb{S}$ a neighbourhood of x_0.

Definition 17 (Boundary point). *A point $x_0 \in \mathbb{S}$ is a boundary point of $\mathbb{S}$ if any small ball $\mathscr{B}_r(x_0)$ centred at x_0 has non-empty intersections with both $\mathbb{S}$ and its complement, $\mathbb{S}^{'}$. i.e.,*

$$x_0 \text{ is a boundary point if } \ \forall\ r > 0\ \exists x, y \in \mathscr{B}_r(x_0) \colon x \in \mathbb{S}, y \in \mathbb{S}^{'}. \tag{A.12}$$

Definition 18 (Interior of a set). *The interior of a set $\mathbb{S}$, denoted by int $\mathbb{S}$, is the set of all interior points in $\mathbb{S}$. If $x \in int\ \mathbb{S}$, we call $\mathbb{S}$ a neighbourhood of x.*

Definition 19 (Boundary of a set). *The boundary of a set $\mathbb{S}$, denoted by $\partial\mathbb{S}$, is the set of all boundary points of $\mathbb{S}$.*

Definition 20 (Closure of set). *The closure of $\mathbb{S}$, written as cl $\mathbb{S}$, is the union of $\mathbb{S}$ and its boundary $\partial\mathbb{S}$ i.e.,*

$$cl\ \mathbb{S} \coloneqq \mathbb{S} \cup \partial\mathbb{S}. \tag{A.13}$$

Definition 21 (Open set). *A set $\mathbb{S}$ is open if $int\ \mathbb{S} = \mathbb{S}$*
Equivalently, $\mathbb{S}$ is open if $\mathbb{S} \cap\ \partial\mathbb{S} = \emptyset$

Definition 22 (Closed set). *A set $\mathbb{S}$ is closed if $cl\ \mathbb{S} = \mathbb{S}$. Equivalently, $\mathbb{S}$ is closed if $\partial\mathbb{S} \subseteq \mathbb{S}$.*

Definition 23 (Bounded set). *A set $\mathbb{S}$ is bounded if $\mathbb{S} \subset \mathscr{B}_r$ for some $r > 0$.*

Definition 24 (Compact set). *A set is compact if it is closed and bounded*

Definition 25 (Limit of sequence). *A point x is the limit of the sequence of points $x_1, x_2, \ldots,$ written $x_i \to x$, if $\|x_i - x\| \to 0$ as $i \to \infty$.*

Definition 26 (Convex set). *A set $\mathbb{S}$ is convex if, for all elements $(x, y) \in \mathbb{S}$, the line segment joining x and y also lies in the set $\mathbb{S}$ i.e*

$$\forall \alpha \in [0, 1],\ \forall (x, y) \in \mathbb{S} \times \mathbb{S},\ \alpha x + (1 - \alpha) y \in \mathbb{S}. \tag{A.14}$$

Definition 27 (Convex combination). *For a finite number of points $\{x_i, \ldots, x_n\} \in \mathbb{S} \times \ldots \times \mathbb{S}$, a convex combination is defined by the relation,*

$$x = \sum_{i=1}^{n} \alpha_i x_i \in \mathbb{S}, \forall i \in \mathbb{I}_{[1,n]}, \alpha_i \geq 0,\ \sum_{i=1}^{n} \alpha_i = 1. \tag{A.15}$$

Definition 28 (Affine set). *A set $\mathbb{S}$ is affine if, for all elements $(x, y) \in \mathbb{S}$, the line segment joining x and y also lies in the set $\mathbb{S}$ i.e*

$$\forall \alpha \in \mathbb{R},\ \forall (x, y) \in \mathbb{S} \times \mathbb{S},\ \alpha x + (1 - \alpha) y \in \mathbb{S}. \tag{A.16}$$

Definition 29 (Polyhedron). *A polyhedron, $\mathbb{P}$ is the set formed from the intersection of a finite number of closed and/or open half-spaces. Equivalently, a set $\mathbb{P}$ is a polyhedron when there exists a set of affine inequalities represented by*

$$\mathbb{P} = \{x \in \mathbb{R}^n \mid\ P_x x \leq q_x\}. \tag{A.17}$$

Definition 30 (Polytope). *A polytope is a closed and bounded polyhedron*

Definition 31 (Unit simplex). *The unit simplex set of dimension n, Λ_n, is defined by*

$$\Lambda_n := \Big\{\delta \in \mathbb{R}^n \colon\ \sum_{i=1}^{n} \delta_i = 1,\ \delta_i \geq 0, \forall i \in \mathbb{I}_{[1:n]}\Big\}. \tag{A.18}$$

Definition 32 (Ellipsoid). *An ellipsoid $\mathcal{E}$ is defined as follows*

$$\mathcal{E} = \Big\{x \in \mathbb{R}^n \mid (x - x_c)^\top P^{-1} (x - x_c) \leq 1,\ P \succeq 0\Big\} \tag{A.19}$$

where x_c is the centre of the ellipsoid.

Definition 33 (Convex hull). *The convex hull of $\mathbb{S}$ is defined as*

$$Co\{\mathbb{S}\} = \left\{\sum_{i=1}^{n} \alpha_i x_i \mid [\alpha_1, \ldots, \alpha_n] \in \Lambda_n,\ x_i \in \mathbb{S}\right\}. \tag{A.20}$$

The convex hull of $\mathbb{S}$ is the intersection of all convex sets containing $\mathbb{S}$.

Definition 34 (Eigenvalues and eigenvectors)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a square $n \times n$ matrix. Then any nonzero vector $x \in \mathbb{R}^n$ is referred to as an **eigenvector** of the matrix A if*

$$Ax = \lambda x \tag{A.21}$$

*for some value $\lambda \in \mathbb{R}$ called the **eigenvalue** of the matrix A corresponding to the eigenvector x.*

Definition 35 (Spectral radius)**.** *The spectral radius of the matrix $A \in \mathbb{R}^{n\times n}$ is the maximum of the absolute value of its eigenvalues.*

Definition 36 (Schur matrix)**.** *The matrix $A \in \mathbb{R}^{n\times n}$ is Schur if all of its eigenvalues have an absolute value that is less than one.*

Definition 37 (Convex function)**.** *A function $\Phi(\bar{z}) : \mathbb{R}^{n_u+n_y} \to \mathbb{R}$ is convex if for all $\bar{z}_1, \bar{z}_2 \in \mathbb{R}^{n_u+n_y}$ and for all $\alpha \in (0,1)$ the following holds:*

$$\Phi(\alpha\bar{z}_1 + (1-\alpha)\bar{z}_2) \leq \alpha\Phi(\bar{z}_1) + (1-\alpha)\Phi(\bar{z}_2). \tag{A.22}$$

Definition 38 (Strictly convex function)**.** *A convex function $\Phi(\bar{z}) : \mathbb{R}^{n_u+n_y} \to \mathbb{R}$ is strictly convex if the inequality in* (A.22) *holds strictly, i.e.*

$$\Phi(\alpha\bar{z}_1 + (1-\alpha)\bar{z}_2) < \alpha\Phi(\bar{z}_1) + (1-\alpha)\Phi(\bar{z}_2). \tag{A.23}$$

A.3 Matrix Calculus

Definition 39 (Gradient of a scalar function)**.** *The gradient of the scalar function $\Phi(u,y)$ is defined by*

$$\nabla\Phi(u,y) = \begin{bmatrix} \frac{\partial\Phi(.)}{\partial u} & \frac{\partial\Phi(.)}{\partial y} \end{bmatrix}^\top. \tag{A.24}$$

where $u \in \mathbb{R}^m$ and $u \in \mathbb{R}^n$ are vectors.

Definition 40 (Vector differentiation [214])**.** *Let $x \in \mathbb{R}^r$ be a vector and $A \in \mathbb{R}^{m\times n}$ a matrix of constants. Then*

$$\frac{\partial Ax}{\partial x} = A^\top \tag{A.25a}$$

$$\frac{\partial x^\top Ax}{\partial x} = (A + A^\top)x \tag{A.25b}$$

$$\frac{\partial x^\top Ax}{\partial x} = 2Ax, \text{ if } A \text{ is symmetric.} \tag{A.25c}$$

Theorem A.3.1 (The backward chain rule)**.** *Let $x \in \mathbb{R}^r$, $y \in \mathbb{R}^s$ and $z \in \mathbb{R}^t$ be vectors. Suppose z is a vector function of y, which in turn is a vector function of x, so we can write $f = z(y(x))$. Then*

$$\frac{\partial f}{\partial x} = \frac{\partial y}{\partial x}\frac{\partial z}{\partial y}.$$

Proof. see [214] □

A.4 Convex Optimization

This section recalls basic concepts from convex theory and convex optimization adopted throughout this thesis. The following definitions and results are standard and can be found, e.g., in [24, 141, 33]. In optimization, we seek the best possible decision from a set of decisions based on a given selection criterion. In mathematical terms, an optimization problem seeks the decision variables that minimizes a defined cost function subject to certain constraints, if any.
Mathematically, we write an optimization problem as follows:

$$\min_x \Phi(x) \tag{A.26a}$$

$$\text{subject to:} \quad h(x) = 0 \tag{A.26b}$$

$$g(x) \leq 0 \tag{A.26c}$$

where $x \in \mathbb{R}^n$ is the $n-$dimensional vector of decision variables, $\Phi : \mathbb{R}^n \to \mathbb{R}$ is the scalar objective function, $h(x) : \mathbb{R}^n \to \mathbb{R}^m$ the $m-$dimensional vector of equality constraints, and $g(x) : \mathbb{R}^n \to \mathbb{R}^p$ the $p-$dimensional vector of inequality constraints. The goal of the optimization problem (A.26) is to find the decision vector x that minimizes the objective function $\Phi(x)$ while fulfilling the constraints (A.26b) and (A.26c). The set of all possible choices for the decision variable is called the feasible set.
With respect to the optimization problem (A.26), we introduce the following definitions.

Definition 41 (Feasible set)**.** *The feasible set, $\mathbb{C}$ is defined as*

$$\mathbb{C} := \{x \in \mathbb{R}^n \mid h(x) = 0,\ g(x) \leq 0\}. \tag{A.27}$$

Definition 42 (Feasible point)**.** *The point $x \in \mathbb{R}^n$ is feasible for problem* (A.26) *if $x \in \mathbb{C}$.*

Definition 43 (Active and Inactive Constraints)**.** *Given $x \in \mathbb{C}$, the set of* active inequality constraints *at x is denoted by $\mathbb{A} \coloneqq \{i \mid g_i(x) = 0\}$ and the set of inactive inequality constraints at x is denoted by $\bar{\mathbb{A}} \coloneqq \{i \mid g_i(x) < 0\}$.*

Remark 58. *Given the feasible set $\mathbb{C}$, the optimization problem* (A.26) *can be re-stated as:*

$$x^* = \arg\min_{x \in \mathbb{C}} \ \Phi(x) \tag{A.28}$$

where x^* is the optimizer of problem (A.28) *as defined next.*

Definition 44 (Global optimizer)**.** *A feasible point $x^* \in \mathbb{C}$ is a global optimizer of* (A.26) *(equivalently* (A.28)*) if $\Phi(x^*) \leq \Phi(x)$ for all $x \in \mathbb{C}$.*

Definition 45 (Local Optimizer)**.** *A feasible point $x^* \in \mathbb{C}$ is a local optimizer of* (A.26) *(equivalently* (A.28)*) in a non-empty neighbourhood $\acute{\mathbb{C}} \subset \mathbb{C}$ if and only if $x^* \in \acute{\mathbb{C}}$ and $\Phi(x^*) \leq \Phi(x)$ for every $x \in \acute{\mathbb{C}}$.*

Sometimes an optimization problem has just one optimizer, in which case we say that the problem is unique, otherwise the problem is non-unique with multiple optimizers.

Definition 46 (Level set)**.** *A level set of Φ is any non-empty set described by*

$$\mathbb{L}(\alpha) \coloneqq \{x \in \mathbb{R}^n \mid \Phi(x) \leq \alpha\}. \tag{A.29}$$

Definition 47 (Lagrangian)**.** *the Lagrangian associated with the optimization problem* (A.26) *is defined as*

$$L(x, \lambda, \mu) \coloneqq \Phi(x) + \lambda^\top h(x) + \mu^\top g(x) \tag{A.30}$$

where λ, μ are called dual variables (or Lagrange multipliers) associated with the constraints h and g respectively.

Definition 48 (KKT conditions)**.** *A point (x^*, λ^*, μ^*) satisfies the Karush-Kuhn-Tucker (KKT) conditions for* (A.26) *if*

$$\nabla L(x^*, \lambda^*, \mu^*) = \nabla\Phi(x^*) + (\lambda^*)^\top \nabla h(x^*) + (\mu^*)^\top \nabla g(x^*) = \mathbf{0}. \tag{A.31}$$

Under linear independent constraint qualification (LICQ) (see [33] for definition), every optimizer is a KKT point [18]. There exists weaker constraint qualifications than

LICQ that guarantee that a minimizer satisfies the KKT conditions. For example, in convex optimization, Slater's condition which requires the feasible set to have non-empty relative interior is a weaker constraint qualification.

A.5 Dynamical systems

A.5.1 Classical control

Definition 49 (Gain margin, GM)**.** *The gain margin is the maximum additional gain that can be applied to a dynamic system without losing closed loop stability. It is a measure of relative stability and is defined as the reciprocal of the magnitude of the loop transfer function evaluated at* $z = j\omega_\pi$ *where* ω_π *is the frequency at which the phase angle is* $-180°$ *also called the phase crossover frequency.*

Definition 50 (Phase margin, PM)**.** *The phase margin is the maximum additional phase change that can be tolerated by a dynamic system without losing closed loop stability. It is also a measure of relative stability and is defined as* $180°$ *plus the phase angle of the loop transfer function evaluated at* $z = j\omega_1$ *where* ω_1 *is the frequency at which the gain is unity also called the gain crossover frequency.*

A.5.2 Stability and feasibility

Definition 51 (Recursive feasibility)**.** *A discrete-time control system,* $\mathcal{S}(x(k))$ *is said to be recursively feasible if and only if feasibility of the system at the initial conditions i.e.* $\mathcal{S}(x(0))$ *implies feasibility at all subsequent evolution of the system i.e.* $\mathcal{S}(x(k+1))$ *for all* $k > 0$*.*

Definition 52 (Distance of point from set)**.** *The distance of the point* $x \in \mathbb{R}^n$ *from the set* $\mathbb{S}$ *denoted by* $d(x, \mathbb{S})$*, is defined as the greatest lower bound of the set of distances from* x *to a point in* $\mathbb{S}$ *i.e*

$$d(x, \mathbb{S}) \coloneqq \inf_{s \in \mathbb{S}} d(x, s) \tag{A.32}$$

where $d(x, s) \colon \mathbb{R}^n \to \mathbb{R}$ *is a metric function in* $\mathbb{R}^n$*.*

Definition 53 (Radially unbounded function)**.** *A function* $x : \mathbb{R}^n \to \mathbb{R}$ *is said to be radially unbounded if* $\|k\| \to \infty$ *implies* $x(k) \to \infty$*.*

Definition 54 (Lyapunov stability of an equilibrium). *The equilibrium x_e is said to be Lyapunov stable for the system $x(k+1) = f(x(k))$ if $\forall \varepsilon > 0$, there exists $\delta > 0$ such that $d(x(0), x_e) \leq \delta$ implies that $d(x(k), x_e) \leq \epsilon$, $\forall k \geq 0$.*

Definition 55 (Attractivity of an equilibrium). *The equilibrium x_e is asymptotically attractive for the system $x(k+1) = f(x(k))$ with the domain $\mathscr{X}$ if, $\forall x(0) \in \mathscr{X}$, $x_e \in int\ \mathscr{X}$, $d(x(k), x_e) \to x_e$ as $k \to \infty$.*

Definition 56 (Asymptotic stability of an equilibrium). *The equilibrium x_e is said to be asymptotically stable for the system $x(k+1) = f(x(k))$ with the domain $\mathscr{X}$ if it is Lyapunov stable and asymptotically attractive within $\mathcal{X}$.*

Definition 57 (Robust stability of a set $\mathscr{R}$). *A set $\mathscr{R}$ is said to be robustly stable for the dynamics $x(k+1) = f(x(k), w(k))$,$w(k) \in \mathbb{W}$ if, $\forall \varepsilon > 0$, there exists a $\delta > 0$ such that $d(x(0), \mathscr{R}) \leq \delta \implies d(x(k), \mathscr{R}) \leq \varepsilon\ \forall k \geq 0$ and $\forall w \in \mathbb{W}$. Here, w is an unknown disturbance and $\mathbb{W}$ a bounded set.*

Definition 58 (Robust attractivity of a set $\mathscr{R}$). *A set $\mathscr{R}$ is said to be asymptotically attractive for the dynamics $x(k+1) = f(x(k), w(k))$, $w(k) \in \mathbb{W}$ if $d(x(k), \mathscr{R}) \to 0$ as $k \to \infty\ \forall w(k) \in \mathbb{W}$ where w is an unknown disturbance and $\mathbb{W}$ a bounded set.*

Definition 59 ($\mathcal{K}-$ function). *A function $\alpha : \mathbb{R}^+ \to \mathbb{R}^+$ is said to be a $\mathcal{K}-$ function if it is continuous, strictly increasing and $\alpha(0) = 0$.*

Definition 60 ($\mathcal{K}_\infty-$ function). *A function $\alpha : \mathbb{R}^+ \to \mathbb{R}^+$ is said to be $\mathcal{K}_\infty-$ function if it is a $\mathcal{K}-$ function and is radially unbounded.*

Definition 61 ($\mathcal{L}-$ function). *A function $\gamma : \mathbb{R}^+ \to \mathbb{R}^+$ is said to be a $\mathcal{L}-$ function if it is continuous, strictly decreasing and $\lim_{\tau\to\infty} \gamma(\tau) = 0$.*

Definition 62 ($\mathcal{KL}-$ function). *A function $\beta\colon \mathbb{R}^+ \times \mathbb{R} \to \mathbb{R}$ is a $\mathcal{KL}-$ function if $\beta(\cdot, k) \in \mathcal{K}$ for every fixed $k \in \mathbb{R}^+$ and $\beta(r, \cdot) \in \mathcal{L}$ for every fixed $r \in \mathbb{R}^+$.*

Definition 63 (Lyapunov function). *A function $V : \mathbb{R}^n \to \mathbb{R}^+$ is said to be a Lyapunov function for $x(k+1) = f(x(k))$ if there exists functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot) \in \mathcal{K}_\infty$ such that for all $x(k) \in \mathbb{R}^n$:*

- $\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$
- $V(f(x(k))) \leq V(x) \leq V(x) - \alpha_3(\|x\|)$

Definition 64 (Input-to-State (ISS) stability). *The dynamics* $x(k+1) = f(x(k), w(k))$ *is ISS stable if there exists* $\beta \in \mathcal{KL}$ *and* $\gamma \in \mathcal{K}$ *such that for each input* $w(k) \in \mathbb{W} \subseteq \mathbb{R}^q$ *and each initial state* $x(0) = x_0 \in \mathbb{R}^n$

$$\|x(k, x_0)\| \leq \beta(\|x_0\|, k) + \gamma(\|w\|) \tag{A.33}$$

Definition 65 (ISS Lyapunov function). *A function* $V : \mathbb{R}^n \to \mathbb{R}^+$ *is said to be an ISS Lyapunov function for* $x(k+1) = f(x(k), w(k))$ *if there exists functions* $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot) \in \mathcal{K}_\infty$*, and* $\sigma \in \mathcal{K}$ *such that for all* $x(k) \in \mathbb{R}^n$ *:*

- $\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$
- $V(f(x), w(k)) \leq V(x) - \alpha_3(\|x\|) + \sigma(\|w\|)$

A.5.3 Set invariance

Definition 66 (Positive Invariant (PI) set). *A set* $\mathbb{E} \subset \mathbb{R}^n$ *is a PI set for* $x(k+1) = f(x(k))$ *if for any* $x(k) \in \mathbb{E}$*, its successor state,* $x(k+1) \in \mathbb{E}$*.*

Definition 67 (Robust Positive Invariant (RPI) set). *A set* $\mathbb{E} \subset \mathbb{R}^n$ *is a RPI set for* $x(k+1) = f(x(k), w(k))$ *if for any* $x(k) \in \mathbb{E}$ *and for all* $w(k) \in \mathbb{W}$*, its successor state,* $x(k+1) \in \mathbb{E}$*.*

A.5.4 Graph Theory

Definition 68 (Weighted directed graph). *A weighted directed graph of order n is a 3-tuple* $\mathscr{G} := (\mathscr{V}, \mathscr{E}, \Gamma_{\mathscr{G}})$ *where* $\mathscr{V} := \{1, 2, \ldots, n\}$ *is the set of nodes/vertices,* $\mathscr{E} \subset \mathscr{V} \times \mathscr{V}$ *is the set of edges, i.e., ordered pairs of nodes* (i, k) *and* $\Gamma_{\mathscr{G}} \colon \mathscr{E} \to \mathbb{R}^+$ *is a weighting function assigning a positive value to each edge.*

Definition 69 (Complete graph). *A complete graph is a directed graph in which every pair of distinct vertices is connected by a pair of unique edges i.e a graph with* $\mathscr{V}^2$ *set of edges.*

Definition 70 (Subgraph). *A subgraph* $S_{\mathscr{G}}$ *of a graph* $\mathscr{G}$ *is a pair* $(\mathscr{V}, \mathscr{E}')$ *where* $\mathscr{E}' \subset \mathscr{E}$ *and its weighting function is the restriction of* $\Gamma_{\mathscr{G}}$ *to* $\mathscr{E}'$*.*

Definition 71 (Path). *A path in a graph is an ordered sequence of vertices such that any pair of consecutive nodes in the sequence is an edge of the graph. To be more* *precise, a path P in $\mathscr{G}$ linking the vertices $v_0, v_k \in \mathscr{V}$ is a subgraph of $\mathscr{G}$ with the vertex set $\mathscr{V}_P = \{v_0, v_1, \ldots, v_k\}$ such that the set of edges contain $\mathscr{E}_P = \{v_0 : v_1, \ldots, v_{k-1} : v_k\}$*

Definition 72 (Connected graph). *A graph is connected if there exists a path between any two vertices. Precisely, a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ is a connected graph if for every pair of vertices $v_i, v_j \in \mathscr{V}$, there exists a path P_{ij} linking them.*

Definition 73 (Neighbourhood set of a vertex). *The neighbourhood set of the vertex v_i denoted by $\mathcal{N}_i$ is the set of all vertices adjacent to v_i i.e.*

$$\mathcal{N}_i := \{j \in \mathcal{V} \mid (v_i, v_j) \in \mathscr{E}\}. \tag{A.34}$$

Definition 74 (In- and out-neighbours). *In a directed graph $\mathscr{G}$ with an edge $(v_i, v_j) \in \mathscr{E}$, v_i is called an in-neighbour of v_j, and v_j is called an out-neighbour of v_i.*

Definition 75 (Degree of a vertex). *The degree of the vertex v_i is the number of neighbours of v_i or the cardinality of the neighbourhood set $\mathcal{N}_i$*

Definition 76 (In- and out- degree). *The in-degree, $d_{in}(v_i)$, and out-degree $d_{out}(v_i)$ of the vertex v_i are the number of in-neighbours and out-neighbours of v_i , respectively. In a weighted directed graph, the weighted out-degree and the weighted in-degree of the vertex v_i are defined by, respectively,*

$$d_{out}(v_i) = \sum_{j=1}^{n} \gamma_{ij} \tag{A.35}$$

i.e $d_{out}(v_i)$ is the sum of the weights of all the out-edges of v_i and

$$d_{in}(v_i) = \sum_{j=1}^{n} \gamma_{ji} \tag{A.36}$$

i.e $d_{in}(v_i)$ is the sum of the weights of all the in-edges of v_i.

Definition 77 (Adjacency matrix of graph $\mathscr{G}$). *The adjacency matrix, $\mathscr{A}_{\mathscr{G}}$, of the graph $\mathscr{G}$ is an $n \times n$ symmetric matrix with $[\mathscr{A}]_{ij} = 1$ when $(v_i, v_j) \in \mathscr{E}$ and $[\mathscr{A}]_{ij} = 0$ otherwise.*

Definition 78 (Out-degree matrix of graph $\mathscr{G}$). *The out-degree matrix $\mathscr{D}_{\mathscr{G}}^{out}$ of the graph $\mathscr{G}$ is a diagonal matrix of the weighted out-degree of all the vertices in $\mathscr{G}$.*

Definition 79 (Laplacian matrix of graph $\mathscr{G}$)**.** *The Laplacian matrix, $\mathscr{L}_{\mathscr{G}}$ of the graph $\mathscr{G}$ is defined as*

$$\mathscr{L}_{\mathscr{G}} = \mathscr{D}_{\mathscr{G}}^{out} - \mathscr{A}_{\mathscr{G}} \in \mathbb{R}^{n \times n} \tag{A.37}$$

Lemma A.5.1. *(Properties of the graph Laplacian matrix)[60]*
The graph Laplacian $\mathscr{L}_{\mathscr{G}}$ has the following important properties

1. *If the graph $\mathscr{G}$ is undirected, then $\mathscr{L}_{\mathscr{G}}$ is a symmetric positive semi-definite matrix i.e. $\mathscr{L}_{\mathscr{G}} \succeq 0$*

2. *The number of zero eigenvalues of $\mathscr{L}_{\mathscr{G}}$ is equal to the number of connected graph components.*

3. *For the graph $\mathscr{G}$, the relation $e = \mathscr{L}_{\mathscr{G}}\Phi$ can be decomposed for each vertex i to the equation*

$$e_i = -\sum_{j \in \mathscr{N}_i} \gamma_{ij}(\Phi_i - \Phi_j) \tag{A.38}$$

where γ_{ij} is the weight of the edge (v_i, v_j).

Definition 80 (Reduced Laplacian matrix)**.** *The reduced Laplacian matrix, $\mathscr{L}_{\mathscr{G}r}$ is the matrix obtained by deleting the reference column from the Laplacian matrix.*

Appendix B

Proofs and Extensions

This section presents some proofs and extensions to ideas presented in the thesis.

B.1 FOLQC problem for time-varying disturbances

To regulate the system under time-varying disturbances to steady-state equilibria where $e = 0$, we consider the velocity, or incremental, form of the system dynamics (4.1) augmented with the tracking error dynamics, whose output is the tracking error $e(k)$:

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k) + \mathcal{E}\delta w(k) \tag{B.1a}$$

$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k) \tag{B.1b}$$

where

$$\epsilon(k) := \begin{bmatrix} \delta x(k) \\ e(k-1) \end{bmatrix} \quad \text{with} \quad \begin{aligned} \delta x(k) &:= x(k) - x(k-1) \\ \delta u(k) &:= u(k) - u(k-1) \\ \delta w(k) &:= w(k) - w(k-1) \end{aligned} \tag{B.2}$$

and

$$\mathcal{A} = \begin{bmatrix} A & \mathbf{0}_{n_x \times n_y} \\ \Lambda_y C & I_{n_y} \end{bmatrix} \qquad \mathcal{B} = \begin{bmatrix} B \\ \Lambda_u \end{bmatrix}, \tag{B.3a}$$

$$\mathcal{C} = \begin{bmatrix} \Lambda_y C & I_{n_y} \end{bmatrix} \qquad \mathcal{D} = \Lambda_u, \qquad \mathcal{E} = \begin{bmatrix} E \\ \mathbf{0}_{n_y \times n_w} \end{bmatrix}. \tag{B.3b}$$

B.1.1 FOLQC Formulation

Given the tracking error and velocity dynamics, the feedback optimizing linear quadratic control problem is defined, for a state $\epsilon(k)$, as

$$\min_{\delta u(k)} J_\infty(\epsilon(k)) \tag{B.4}$$

subject to,

$$\epsilon(k+1) = \mathcal{A}\epsilon(k) + \mathcal{B}\delta u(k) + \mathcal{E}\delta w(k), \tag{B.5a}$$
$$e(k) = \mathcal{C}\epsilon(k) + \mathcal{D}\delta u(k). \tag{B.5b}$$

In this problem, the decision variable is the control law

$$\delta u(k) = -\mathcal{K}\epsilon(k) \tag{B.6}$$

This control law is chosen to minimize the objective $J_\infty(\epsilon(k))$, which is defined as

$$J_\infty(\epsilon(k)) = \frac{1}{2}\sum_{k=0}^{\infty} l(e(k), \delta u(k))$$

where

$$l(e(k), \delta u(k)) := e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k) \tag{B.7}$$

The matrices Q_e and R_δ are the respective penalty matrices on the squared tracking error e_k and input deviation $\delta u(k)$.

The finite- and infinite-horizon control laws are easily derived by applying the standard arguments of dynamic programming and Bellman's Principle of Optimality to the optimal control problem [133]:

Proposition 7. *The solution to (4.15 s.t. 4.16) is*

$$\delta u^*(k) = -\mathcal{K}(k)\epsilon(k) - \mathcal{K}_\delta(k)\delta w(k) \quad k = 0, \ldots, N-1 \tag{B.8}$$

where, for $k = 0, \dots, N-1$,

$$\mathcal{K}(k) = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}(k)\mathcal{B}\Big)^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}(k)\mathcal{A}\Big) \tag{B.9a}$$

$$\mathcal{K}_\delta(k) = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}(k)\mathcal{B}\Big)^{-1}\mathcal{B}^\top \mathcal{P}(k)\mathcal{E} \tag{B.9b}$$

$$\mathcal{P}(k) = \Big(\mathcal{A} - \mathcal{B}\mathcal{K}(k)\Big)^\top \mathcal{P}(k+1)\Big(\mathcal{A} - \mathcal{B}\mathcal{K}(k)\Big) + \mathcal{K}(k)^\top \mathcal{R}\mathcal{K}(k) + \mathcal{Q} - 2\mathcal{N}\mathcal{K}(k) \tag{B.9c}$$

Moreover, as $N \to \infty$ *then* $\mathcal{K}(N) \to \mathcal{K}$, $\mathcal{K}_\delta(N) \to \mathcal{K}_\delta$ *and* $\mathcal{P}(k+1) \to \mathcal{P}(k)$ *with* $\mathcal{P}(N) \to \mathcal{P}$ *where*

$$\mathcal{K} = (\mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B})^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}\mathcal{A}\Big) \tag{B.10a}$$

$$\mathcal{K}_\delta = \Big(\mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B}\Big)^{-1}\mathcal{B}^\top \mathcal{P}\mathcal{E} \tag{B.10b}$$

$$\mathcal{P} = \mathcal{A}^\top \mathcal{P}\mathcal{A} + \mathcal{Q} - \Big(\mathcal{N} + \mathcal{A}^\top \mathcal{P}\mathcal{B}\Big)(\mathcal{R} + \mathcal{B}^\top \mathcal{P}\mathcal{B})^{-1}\Big(\mathcal{N}^\top + \mathcal{B}^\top \mathcal{P}\mathcal{A}\Big) \tag{B.10c}$$

Proof. In this development, we adopt a backward dynamic programming approach starting from the time interval $k \in [0, N]$. Let $J_k^*(\epsilon(k), \delta\mathbf{u}(k))$ denote the optimum cost (4.15) of transferring the system (4.12) from an initial state $\epsilon(k)$ to the terminal state $\epsilon(N)$. At the terminal state $\epsilon(N)$, let the optimal cost be given by the function

$$J_N^*(\epsilon(N)) = \frac{1}{2}\epsilon(N)^\top \mathcal{P}(N)\epsilon(N) \tag{B.11}$$

where $\mathcal{P}$ is a symmetric positive (semi) definite matrix. Let $J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1))$ be the optimal cost evaluated from time $k+1$ to N. Then at any stage k, using the principle of optimality [133],

$$J_k^*\Big(\epsilon(k), \delta u(k)\Big) = \min_{\delta u(k)} \Big\{ l(e(k), \delta u(k)) + J_{k+1}^*(\epsilon(k+1), \delta\mathbf{u}(k+1)) \Big\} \tag{B.12}$$

By solving (4.28) recursively, we can compute the optimal control law $u^*(k)$ that solves the OCP. To begin, at $k = N-1$,

$$J_{N-1}(\epsilon(N-1), \delta\mathbf{u}(N-1)) = J_N^*(\epsilon(N)) + l(e(N-1), \delta\mathbf{u}(N-1)) \tag{B.13}$$

Using the state equation (4.12) at $k = N - 1$,

$$\epsilon(N) = \mathcal{A}\epsilon(N-1) + \mathcal{B}\delta u(N-1) + \mathcal{E}\delta w(N-1) \tag{B.14}$$

we can eliminate $\epsilon(N)$ from J_{N-1} to obtain

$$\begin{aligned} J_{N-1}(\epsilon(N-1), \delta u(N-1)) = \frac{1}{2}\Big(\mathcal{A}\epsilon(N-1) + \mathcal{B}\delta u(N-1) + \mathcal{E}\delta w(N-1)\Big)^{\top}\mathcal{P}(N) \\ \Big(\mathcal{A}\epsilon(N-1) + \mathcal{B}\delta u(N-1) + \mathcal{E}\delta w(N-1)\Big) + \frac{1}{2}\epsilon(N-1)^{\top}\mathcal{Q}\epsilon(N-1) \\ +\epsilon(N-1)^{\top}\mathcal{N}\delta u(N-1) + \frac{1}{2}\delta u(N-1)^{\top}\mathcal{R}\delta u(N-1) \end{aligned} \tag{B.15}$$

The optimal control law at $k = N - 1$ i.e $\delta u^*(N-1)$ can be obtained by applying the first order necessary optimality condition,

$$\begin{aligned} \frac{\partial J_{N-1}}{\partial \delta u(N-1)} = \Big(\mathcal{R} + \mathcal{B}^{\top}\mathcal{P}(N)\mathcal{B}\Big)\delta u^*(N-1) + \Big(\mathcal{N}^{\top} + \mathcal{B}^{\top}\mathcal{P}(N)\mathcal{A}\Big)\epsilon(N-1) + \\ \mathcal{B}^{\top}\mathcal{P}\mathcal{E}\delta w(N-1) = \mathbf{0} \end{aligned} \tag{B.16}$$

Solving for $\delta u^*(N-1)$, we obtain

$$\delta u^*(N-1) = -\mathcal{K}(N-1)\epsilon(N-1) - \mathcal{K}_{\delta}(N-1)\delta w(N-1) \tag{B.17}$$

where,

$$\mathcal{K}(N-1) = \Big(\mathcal{R} + \mathcal{B}^{\top}\mathcal{P}(N)\mathcal{B}\Big)^{-1}\Big(\mathcal{N}^{\top} + \mathcal{B}^{\top}\mathcal{P}(N)\mathcal{A}\Big)$$

and

$$\mathcal{K}_{\delta}(N-1) = \Big(\mathcal{R} + \mathcal{B}^{\top}\mathcal{P}(N)\mathcal{B}\Big)^{-1}\mathcal{B}^{\top}\mathcal{P}(N)\mathcal{E}$$

With (4.33), we can compute the corresponding optimal cost at $k = N - 1$ as,

$$
\begin{aligned}
J^*_{N-1}(\epsilon(N-1)) = \frac{1}{2}\epsilon(N-1)^\top\Big[\Big(\mathcal{A}-\mathcal{B}\mathcal{K}(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{A}-\mathcal{B}\mathcal{K}(N-1)\Big)+ \\
\mathcal{Q}-2\mathcal{N}\mathcal{K}(N-1)+\mathcal{K}(N-1)^\top\mathcal{R}\mathcal{K}(N-1)\Big]\epsilon(N-1) \\
+\epsilon(N-1)^\top\Big[\Big(\mathcal{A}-\mathcal{B}\mathcal{K}(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{E}-\mathcal{B}\mathcal{K}_\delta(N-1)\Big) \\
-2\mathcal{N}\mathcal{K}_\delta(N-1)+\mathcal{K}(N-1)^\top\mathcal{R}\mathcal{K}_\delta(N-1)\Big]\delta w(N-1)+ \\
\frac{1}{2}\delta w(N-1)^\top\Big[\Big(\mathcal{E}-\mathcal{B}\mathcal{K}_\delta(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{E}-\mathcal{B}\mathcal{K}_\delta(N-1)\Big) \\
+\mathcal{K}_\delta^\top(N-1)\mathcal{R}\mathcal{K}_\delta(N-1)\Big]\delta w(N-1)
\end{aligned} \tag{B.18}
$$

which can also be expressed as

$$
J^*_{N-1}(\epsilon(N-1)) = \frac{1}{2}\epsilon(N-1)^\top\mathcal{P}(N-1)\epsilon(N-1)
$$

resulting in the recursion

$$
\begin{aligned}
\mathcal{P}(N-1) = \Big(\mathcal{A}-\mathcal{B}\mathcal{K}(N-1)\Big)^\top \mathcal{P}(N)\Big(\mathcal{A}-\mathcal{B}\mathcal{K}(N-1)\Big)+ \\
\mathcal{K}(N-1)^\top\mathcal{R}\mathcal{K}(N-1)+\mathcal{Q}-2\mathcal{N}\mathcal{K}(N-1)
\end{aligned} \tag{B.19}
$$

We can repeat this procedure for $k = N-2,\ N-3,\dots$ giving the results (4.24) and (4.25). □

The main result of this chapter—that the infinite-horizon control law characterized by proposition 4 solves Problem 1—immediately follows.

Theorem B.1.1. *Suppose that Assumptions 1 and 2 hold, and also the hypotheses of Proposition 2 and 3. The infinite-horizon control law $\delta u(k) = -\mathcal{K}\epsilon(k) - \mathcal{K}_\delta\delta w(k)$ solves Problem 1, minimizing the infinite-horizon criterion*

$$
J_\infty(\epsilon(0)) := \sum_{k=0}^{\infty}\Big\{e(k)^\top Q_e e(k) + \delta u(k)^\top R_\delta \delta u(k)\Big\}. \tag{B.20}
$$

Remark 59 (Finite- and receding-horizon implementations)**.** *The finite-horizon control gains $\mathcal{K}(N), \mathcal{K}(N-1), \dots, \mathcal{K}(k), \dots \mathcal{K}(1)$, and $\mathcal{K}_\delta(N), \mathcal{K}_\delta(N-1), \dots, \mathcal{K}_\delta(k), \dots \mathcal{K}_\delta(1)$ can also solve Problem 1 if $\mathcal{P}$ satisfies the Lyapunov equation with some known stabilizing $\mathcal{K}(0) = \mathcal{K}$; in a receding-horizon (MPC) implementation, the applied control law is $\delta u(k) = -\mathcal{K}(k)\epsilon(k) - \mathcal{K}_\delta(k)\delta w(k)$. The result is suboptimal. However, such implementations may be useful—indeed necessary—when constraints are present on* *the system. The FOLQC problem subject to such constraints is the subject of following chapter.*

B.1.2 FOLQC with state and disturbance estimation

In the previous section, we presented control algorithms for solving the deterministic feedback-optimizing linear-quadratic control (FOLQC) problem. To implement the control law in (B.8), availability of the velocity state $\epsilon(k)$ and the change in disturbance $\delta w(k)$ is assumed. If the state variables, $x(k)$, of the original system (4.1) are measurable, and the disturbance $w(k)$ is piecewise constant, then $\delta w(k) = 0$ and $\epsilon(k)$ can be computed from the state, input and output measurements via the following

$$\epsilon(k) = \begin{bmatrix} x(k) - x(k-1) \\ \Lambda_y y(k-1) + \Lambda_u u(k-1) + r \end{bmatrix} \tag{B.21}$$

In practice, the disturbance may be time varying (i.e., not piece-wise constant) and obtaining measurements for all state variables may be impossible or simply impractical. Thus, there is a need to estimate the disturbances and the system states using the inputs and output measurements, and the system model. To this end, we present a state estimator for the velocity state $\epsilon(k)$ and the disturbance deviation $\delta w(k)$. Given the system parameters (A, B, E, C) and a model of the disturbance $w(k)$, it is possible to estimate the state and disturbance of the system from input and output measurements if the velocity system augmented with $\delta w(k)$ is observable.

Velocity state and disturbance estimator

Given the velocity system (B.1) and the disturbance model,

$$w(k+1) = A_w w(k) \tag{B.22}$$

we can write the following disturbance augmented velocity dynamics

$$\begin{bmatrix} \epsilon(k+1) \\ \delta w(k+1) \end{bmatrix} = \mathcal{A}_a \begin{bmatrix} \epsilon(k) \\ \delta w(k) \end{bmatrix} + \mathcal{B}_a \delta u(k) \tag{B.23a}$$

$$e(k) = \mathcal{C}_a \begin{bmatrix} \epsilon(k) \\ \delta w(k) \end{bmatrix} + \mathcal{D} \delta u(k) \tag{B.23b}$$

where

$$\mathcal{A}_a = \begin{bmatrix} \mathcal{A} & \mathcal{E} \\ \mathbf{0}_{n_w \times (n_x+n_y)} & A_w \end{bmatrix}, \; \mathcal{B}_a = \begin{bmatrix} \mathcal{B} \\ \mathbf{0}_{n_w \times n_u} \end{bmatrix} \text{ and } \mathcal{C}_a = \begin{bmatrix} \mathcal{C} & \mathbf{0}_{n_y \times n_w} \end{bmatrix}.$$

We make the following assumption on the augmented system

Assumption 30. *The augmented system* (B.23) *is observable*

Proposition 8. *Assumption 30 is satisfied if* $\left(\mathcal{C}, \mathcal{A}\right)$ *is observable and the matrices* $\mathcal{E}$ *and* A_w *are such that*

$$\text{rank} \begin{bmatrix} \mathcal{A} - \lambda I_{n_x+n_y} & \mathcal{E} \\ \mathcal{C} & \mathbf{0}_{n_y \times n_w} \end{bmatrix} = (n_x + n_y) + n_w \quad \forall \lambda \in eig(A_w) \tag{B.25}$$

For the $(n_x + n_y + n_w)-$dimensional augmented system (B.23), we propose a linear state estimator to also be an $(n_x + n_y + n_w)-$dimensional system (B.26) that takes $\delta u(k)$ and $e(k)$ as inputs and whose state represents an estimate of the augmented state variable i.e. $\begin{bmatrix} \hat{\epsilon}(k) \\ \delta\hat{w}(k) \end{bmatrix}$.

The proposed state estimator has the form

$$\begin{bmatrix} \hat{\epsilon}(k+1) \\ \delta\hat{w}(k+1) \end{bmatrix} = \left(\mathcal{A}_a - \mathcal{L}_a\mathcal{C}_a\right) \begin{bmatrix} \hat{\epsilon}(k) \\ \delta\hat{w}(k) \end{bmatrix} + \left(\mathcal{B}_a - \mathcal{L}_a\mathcal{D}\right)\delta u(k) + \mathcal{L}_a e(k) \tag{B.26}$$

which is obtained from a duplicate of the augmented dynamics (B.23) driven by the error term $\left(e(k) - \mathcal{C}_a \begin{bmatrix} \hat{\epsilon}(k) \\ \delta\hat{w}(k) \end{bmatrix} - \mathcal{D}\delta u(k)\right)$ which enters the duplicated dynamics via the estimator gain $\mathcal{L}_a \in \mathbb{R}^{(n_x+n_y+n_w)\times n_y}$. The gain $\mathcal{L}_a$ is designed such that the estimation error dynamics

$$\xi(k+1) = \left(\mathcal{A}_a - \mathcal{L}_a\mathcal{C}_a\right)\xi(k) \tag{B.27}$$

are driven asymptotically to zero for any non-zero value of $\xi(0)$, where

$$\xi(k) = \begin{bmatrix} \epsilon(k) \\ \delta w(k) \end{bmatrix} - \begin{bmatrix} \hat{\epsilon}(k) \\ \delta\hat{w}(k) \end{bmatrix}.$$

Output Feedback-optimizing Linear Quadratic Control

With the disturbance augmented observer above, the FOLQC law (B.8) can be implemented in an output feedback framework without explicit state and disturbance measurements for any arbitrary time varying disturbance $w(k)$. By connecting the state estimator (B.26) in feedback with the LTI dynamics (B.1), we obtain the closed-loop system for the output FOLQC system shown in Fig. B.1 below. Due to the observability of the augmented dynamics, the controllability of the velocity dynamics and the linearity of both dynamical systems, the principle of separation applies, allowing independent design of the state estimator and FOLQC gains. To achieve uniform stability of the closed-loop system, the estimator gain $\mathcal{L}_a$ must be designed such that its eigenvalues are sufficiently faster than those of the FOLQC gain $\begin{bmatrix} -\mathcal{K} & -\mathcal{K}_\delta \end{bmatrix}$.

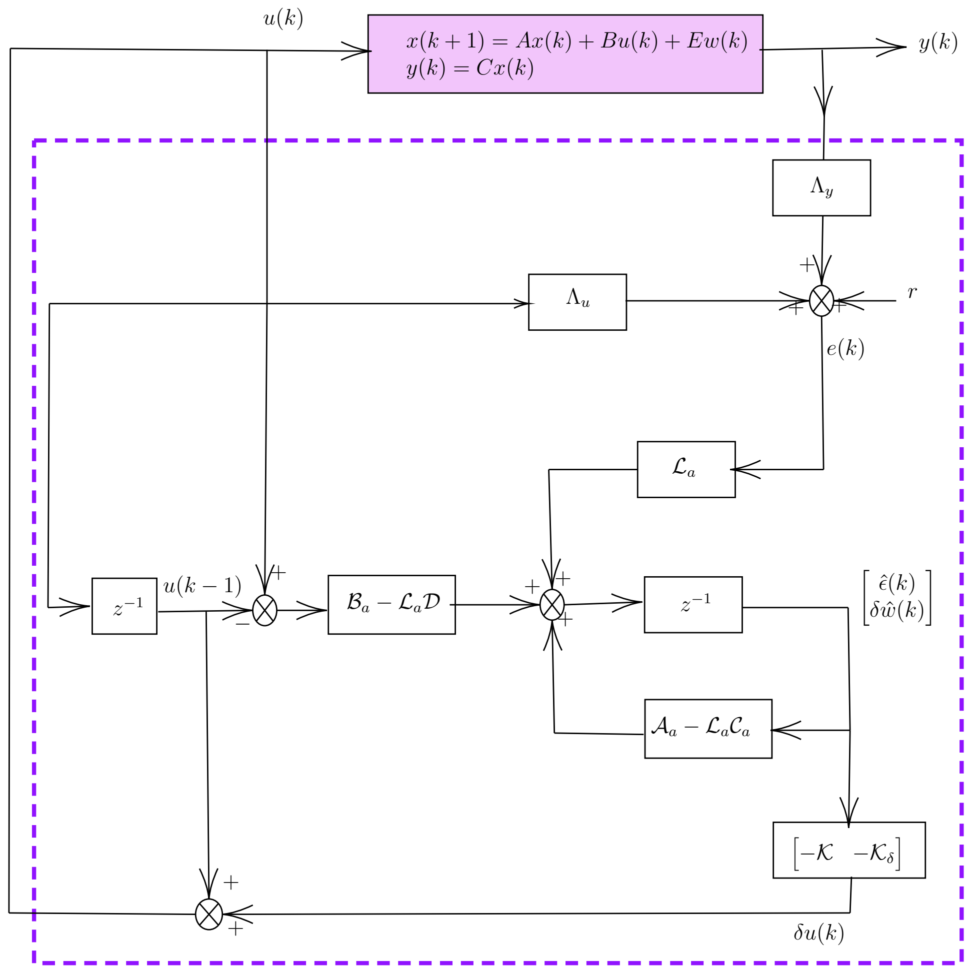

Fig. B.1 Output-feedback FOLQC

B.2 A Feedforward Formulation of FOLQC

The FOLQC formulation based on a velocity model of the system dynamics as presented above is just one approach to solving Problem 4. Our interest in the velocity model formulation of FOLQC stems from its ability to be interpreted as a Proportional-Integral (PI) controller thereby allowing a direct comparison to other feedback-optimizing control algorithms in the literature (e.g optimal steady-state control [128]). Also, an added benefit of a velocity model based FOLQC formulation is that for piecewise constant/ slowly varying disturbances, disturbance estimation is not required to achieve feedback-optimizing control. In this section, we present a feed-forward formulation of FOLQC that is not based on a velocity model of the system dynamics.

B.2.1 Problem Formulation

In this approach, an estimate of the disturbance i.e $\hat{w}(k)$ is used to compute a priori, the unknown optimal steady-states $\bar{x}$ and inputs $\bar{u}$ for the steady-state optimization (4.2).
Let $x(k) \to \bar{x}$, $u(k) \to \bar{u}$ and $w(k) \to \bar{w}$ as $k \to \infty$. Then the system dynamics (4.1) and the steady-state tracking error (4.10) at an optimal equilibrium both satisfy the following equation

$$\bar{x} = A\bar{x} + B\bar{u} + E\bar{w} \tag{B.28a}$$

$$\mathbf{0} = \Lambda_y C\bar{x} + \Lambda_u \bar{u} + r \tag{B.28b}$$

Rearranging the equations and solving for $\begin{bmatrix}\bar{x} & \bar{u}\end{bmatrix}^\top$ yields

$$\begin{bmatrix}\bar{x} \\ \bar{u}\end{bmatrix} = S^{-1}\begin{bmatrix}E \\ \mathbf{0}_{n_u \times n_w}\end{bmatrix}\bar{w} + S^{-1}\begin{bmatrix}\mathbf{0}_{n_x \times n_y} \\ -I_{n_y}\end{bmatrix} r \tag{B.29}$$

where S is given by (4.11). The relation above gives the optimal steady-state set-points $(\bar{x}, \bar{u})$ for the equilibrium optimization (4.2) parametrised by the unknown disturbance $\bar{w}$ and the steady-state cost parameter r. To obtain unique values of $(\bar{x}, \bar{u})$, we assume the matrix S has full rank and $n_y \leq n_u$. Now to formulate the FOLQC problem, we simply translate the origin of the system (4.1) to the optimal set-points $(\bar{x}, \bar{u})$ and apply the standard linear-quadratic regulator (LQR) theory on the translated system.

Let

$$\tilde{x}(k) = x(k) - \bar{x}, \quad \tilde{u}(k) = u(k) - \bar{u} \quad \text{and,} \quad \tilde{w}(k) = w(k) - \bar{w}$$

then if the estimated disturbance $\hat{w}(k)$ approaches the actual disturbance $w(k)$ sufficiently fast and $w(k) \to \bar{w}$ as $k \to \infty$ (i.e, disturbance is asymptotically constant), then the translated system can be represented as

$$\tilde{x}(k+1) = A\tilde{x}(k) + B\tilde{u}(k) \tag{B.30}$$

Regulating the translated variables $\tilde{x}(k)$ and $\tilde{u}(k)$ to the origin achieves tracking of the optimal steady-state set-points $(\bar{x}, \bar{u})$. We achieve this by finding a stabilizing control law $\tilde{u}(k)$ that minimizes the performance index

$$V_N(\tilde{x}(k)) = \frac{1}{2}\tilde{x}(N)^\top \tilde{P}\tilde{x}(N) + \sum_{k=0}^{N-1} l(\tilde{x}(k), \tilde{u}(k)) \tag{B.31}$$

for the translated system (B.30) where

$$l(\tilde{x}(k), \tilde{u}(k)) := \frac{1}{2}\Big(\tilde{x}(k)^\top \tilde{Q}\tilde{x}(k) + \tilde{u}(k)^\top \tilde{R}\tilde{u}(k)\Big) \tag{B.32}$$

The matrices $\tilde{Q} \succeq 0$, $\tilde{R} \succ 0$ and $\tilde{P} \succeq 0$ are the penalties on the translated state, control and the terminal state respectively. The solution to the FOLQC problem based on the feedforward formulation above is summarised in theorem B.2.1 below.

Theorem B.2.1. *Assume* (A, B) *is stabilizable and the matrix* S *is full rank, then the control law,*

$$u(k) = -K_x x(k) + K_w \hat{w}(k) + K_r r \tag{B.33}$$

is a solution to the deterministic FOLQC problem (4)*, where,*

$$K_x = (\tilde{R} + B^\top \tilde{P} B)^{-1} B^\top \tilde{P} A \tag{B.34a}$$

$$K_w = \begin{bmatrix} K_x & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} E \\ \mathbf{0} \end{bmatrix}, \; K_r = \begin{bmatrix} K_x & I_{n_u} \end{bmatrix} S^{-1} \begin{bmatrix} \mathbf{0} \\ -I_{n_y} \end{bmatrix}, \tag{B.34b}$$

and $\tilde{P}$ satisfies the recursion,

$$\tilde{P} = (A - BK_x)^\top \tilde{P}(A - BK_x) + K_x^\top \tilde{R} K_x + \tilde{Q} \tag{B.34c}$$

To implement the control law (B.33), an estimate of the disturbance, $\hat{w}(k)$, is required. To this end, a disturbance augmented state estimator, similar to that presented previously for the velocity based control law is designed to estimate the state and unknown disturbance. The design of this estimator can be easily adapted from [171] and is therefore omitted.

B.3 A Feedforward Formulation of Nominal FOMPC

The FOMPC formulation based on a velocity model of the system dynamics as presented in Chapter 6 is just one approach to solving Problem 6. As previously mentioned, a benefit of the velocity formulation of FOMPC is that for piecewise constant or slowly varying disturbances, disturbance estimation is not explicitly required. In this section, we present a feed-forward formulation of FOMPC that is not based on a velocity model of the system dynamics.

In this approach, an estimate of the disturbance i.e $\hat{w}(k)$ is used to compute a priori, the unknown optimal steady-states $\bar{x}^*$ and inputs $\bar{u}^*$ for the steady-state optimization (6.2).

Let $x(k) \to \bar{x}$, $u(k) \to \bar{u}$ and $w(k) \to \bar{w}$ as $k \to \infty$. Then the system dynamics (6.1) and the steady-state tracking error (6.6) at an optimal equilibrium both satisfy the following equation

$$\bar{x}^* = A\bar{x} + B\bar{u} + E\bar{w} \tag{B.35a}$$

$$\mathbf{0}^* = \Lambda_y C\bar{x} + \Lambda_u \bar{u} + r \tag{B.35b}$$

Rearranging the equations and solving for $\begin{bmatrix}\bar{x}^* & \bar{u}^*\end{bmatrix}^\top$ yields

$$\begin{bmatrix}\bar{x}^* \\ \bar{u}^*\end{bmatrix} = S^{-1}\begin{bmatrix}E \\ \mathbf{0}_{n_u \times n_w}\end{bmatrix}\bar{w} + S^{-1}\begin{bmatrix}\mathbf{0}_{n_x \times n_y} \\ -I_{n_y}\end{bmatrix} r \tag{B.36}$$

The relation above gives the optimal steady-state set-points $(\bar{x}^*, \bar{u}^*)$ for the equilibrium optimization (6.2) parametrised by the unknown disturbance $\bar{w}$ and the steady-state cost parameter r. To obtain unique values of $(\bar{x}^*, \bar{u}^*)$, we assume the matrix S has full rank and $n_y \leq n_u$. Now to formulate the FOMPC problem, we simply translate the origin of the system (6.1) to the optimal set-points $(\bar{x}^*, \bar{u}^*)$ and apply the standard offset-free tracking model predictive control algorithm on the translated system (see

Chapter 2). We refer the reader to [171, 144] for details.
To implement the FOMPC law, an estimate of the disturbance, $\hat{w}(k)$, is required. To this end, a disturbance augmented state estimator, similar to that presented previously for the velocity based control law is designed to estimate the state and unknown disturbance. The design of this estimator can be easily adapted from [171] and is therefore omitted.