University of California
Santa Barbara

Simultaneous Nonlinear Model Predictive Control and State Estimation: Theory and Applications

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Mechanical Engineering

by

David A. Copp

Committee in charge:

      Professor João Hespanha, Chair
      Professor Andrew Teel
      Professor Bassam Bamieh
      Professor Brad Paden

December 2016

The Dissertation of David A. Copp is approved.

Professor Andrew Teel

Professor Bassam Bamieh

Professor Brad Paden

Professor João Hespanha, Committee Chair

November 2016

Simultaneous Nonlinear Model Predictive Control and State Estimation: Theory and
Applications

to Stacy & Lydia

Acknowledgements

I was incredibly privileged and fortunate to interact with and learn from numerous exceptional researchers, teachers, mentors, collaborators, students, and friends during my time at UC Santa Barbara. I certainly learned a lot and made lasting fond memories.

First, I would like to thank my advisor, Prof. João Hespanha, for mentoring me through my PhD. I have learned so much about how to approach research problems, the importance of starting simple, looking at the big picture, and then quickly zooming in and focusing on the details of a problem, as well as the art of crafting effective articles and presentations. In addition, I would like to thank my other PhD committee members and faculty in the Center for Control, Dynamical-Systems, and Computation for their thoughtful teaching, questions, comments, advice, guidance, and patience. I owe many thanks to Prof. Ricardo Sanfelice for giving me the opportunity to perform research as an undergraduate at the University of Arizona and for encouraging me to pursue a PhD. For the people I have had the privilege of seeing on a daily basis, thank you to my lab mates Jason, Steven, Justin, Hari, Henrique, Soheil, Kyriakos, and Farshad, as well as the many people who visited our lab during my time at UCSB, for fun times and fruitful discussions. They truly made my graduate school experience an enjoyable one. I am grateful to my collaborator Ravi Gondhalekar for the opportunity to participate in his stimulating research regarding feedback control of an artificial pancreas and for offering sound advice about research and many other areas of life in general. To the high school students I have had the privilege of mentoring, thank you for your excitement and creativity in tackling research challenges and providing me with new perspectives on the knowledge I hope I imparted to you. Lastly, I am indebted to my family for their love and support throughout this process. In particular to Stacy: I am so thankful that we were able to share this experience of working towards PhDs together at UCSB.

Curriculum Vitæ
David A. Copp

Education

2011-2016	Ph.D. in Mechanical Engineering, University of California, Santa Barbara.
2011-2014	M.S. in Mechanical Engineering, University of California, Santa Barbara.
2007-2011	B.S. in Mechanical Engineering, University of Arizona.

Experience

2012-2016	Graduate Student Researcher, University of California, Santa Barbara.
2011,2015	Graduate Teaching Assistant, University of California, Santa Barbara.
2011	Student Researcher, Air Force Research Labs, Kirtland AFB.
2010-2011	Undergraduate Research Assistant, University of Arizona.

Selected Publications

D. A. Copp and J. P. Hespanha. Simultaneous nonlinear model predictive control and state estimation. Automatica, in press, 2016.

D. A. Copp and J. P. Hespanha. Nonlinear output-feedback model predictive control with moving horizon estimation. In 53rd Conference on Decision and Control (CDC), p. 3511-3517. IEEE, December 2014.

D. A. Copp and J. P. Hespanha. Conditions for saddle-point equilibria in output-feedback MPC with MHE. In Proc. of the American Control Conference (ACC), p. 13-19. IEEE, July 2016.

D. A. Copp and J. P. Hespanha. Addressing adaptation and learning in the context of MPC with MHE. Control of Complex Systems: Theory and Applications. Elsevier, 1st edition, 2016.

S. A. P. Quintero, D. A. Copp, and J. P. Hespanha. Robust UAV coordination for target tracking using output-feedback model predictive control with moving horizon estimation. In Proc. of the American Control Conference (ACC), p. 3758-3764. IEEE, July 2015.

D. A. Copp, R. Gondhalekar, and J. P. Hespanha. Simultaneous Model Predictive Control and Moving Horizon Estimation for Blood Glucose Regulation in Type 1 Diabetes. Submitted, 2016.

Abstract

Simultaneous Nonlinear Model Predictive Control and State Estimation: Theory and Applications

by

David A. Copp

As computational power increases, online optimization is becoming a ubiquitous approach for solving control and estimation problems in both academia and industry. This widespread popularity of online optimization techniques is largely due to their abilities to solve complex problems in real time and to explicitly accommodate hard constraints. In this dissertation, we discuss an especially popular online optimization control technique called model predictive control (MPC). Specifically, we present a novel output-feedback approach to nonlinear MPC, which combines the problems of state estimation and control into a single min-max optimization. In this way, the control and estimation problems are solved simultaneously providing an output-feedback controller that is robust to worst-case system disturbances and noise. This min-max optimization is subject to the nonlinear system dynamics as well as constraints that come from practical considerations such as actuator limits. Furthermore, we introduce a novel primal-dual interior-point method that can be used to efficiently solve the min-max optimization problem numerically and present several examples showing that the method succeeds even for severely nonlinear and non-convex problems.

Unlike other output-feedback nonlinear optimal control approaches that solve the estimation and control problems separately, this combined estimation and control approach facilitates straightforward analysis of the resulting constrained, nonlinear, closed-loop system and yields improved performance over other standard approaches. Under appro-

priate assumptions that encode controllability and observability of the nonlinear process to be controlled, we show that this approach ensures that the state of the closed-loop system remains bounded. Finally, we investigate the use of this approach in several applications including the coordination of multiple unmanned aerial vehicles for vision-based target tracking of a moving ground vehicle and feedback control of an artificial pancreas system for the treatment of Type 1 Diabetes. We discuss why this novel combined control and estimation approach is especially beneficial for these applications and show promising simulation results for the eventual implementation of this approach in real-life scenarios.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Online optimization has become a ubiquitous approach for solving control and estimation problems in both academia and industry. This is largely due to the ability of online optimization techniques to explicitly accommodate nonlinear dynamics, sophisticated disturbance and noise models, as well as hard constraints. Two paramount challenges when considering online optimization techniques are ensuring stability/convergence as well as developing efficient numerical solvers. In this thesis, we provide a new online optimization technique for solving control and estimation problems and address both of these challenges by proving stability of the resulting closed-loop system as well as developing new efficient algorithms for numerically solving these optimizations. Our new technique is motivated by two finite horizon online optimization approaches: model predictive control (MPC) and moving horizon estimation (MHE). Next we give a brief background on MPC and MHE.

1.1   Model Predictive Control and Moving Horizon Estimation

MPC is a particularly popular online optimization technique for solving control problems and involves the solution of an open-loop optimal control problem at each sampling time [72]. This optimization is subject to the dynamics of the process to be controlled as well as constraints on the states and inputs. Classical MPC is formulated with full state feedback. Given the current state of the system to be controlled, MPC results in a sequence of future optimal control actions and a sequence of corresponding future predicted states. The first control action in the sequence is applied to the process, and then the optimization is solved again at each successive sampling time. MPC has historically been popular for problems in which the process dynamics are sufficiently slow so that the optimization can be solved between consecutive sampling times. However, as available computational power increases and optimization algorithms improve in terms of computational speed, MPC can be applied to broader application areas. More than a decade ago MPC was being used in numerous industrial applications [84], and, consequently, much effort has been devoted to developing a stability theory for MPC (see e.g. [79, 16, 95, 43, 92]). Depictions of the MPC problem are shown in Figure 1.1 as a block diagram and Figure 1.2 as a cartoon.
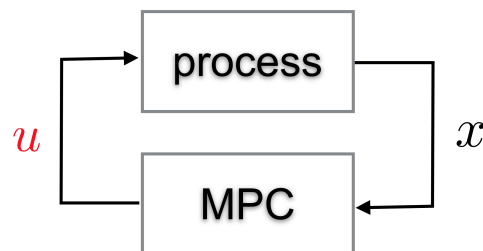
Figure 1.1: Block diagram of the MPC problem.
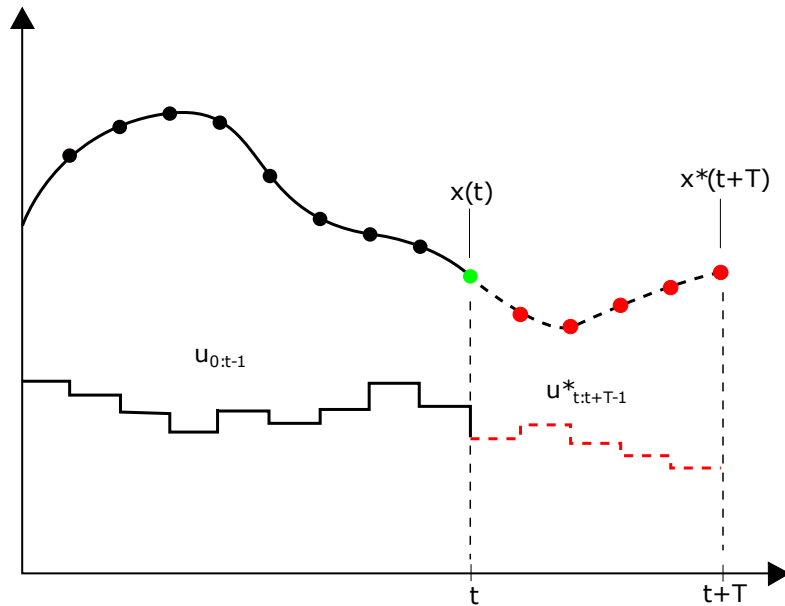
Figure 1.2: Cartoon of the MPC problem. Given a model of the process to be controlled and the current state $x(t)$, a sequence of future control actions $u^*_{t:t+T-1}$ is computed and result in a sequence of future predicted states.

In many practical cases, the full state of the process to be controlled cannot be measured. Therefore output-feedback MPC must be considered which involves the use of additional algorithms for state estimation, including observers, filters, and moving horizon estimation (MHE), several of which are discussed in [94]. MHE stands out among these methods due to its ability to deal with constraints on the states. In fact, neglecting these constraints may lead to increased estimation error or divergence of the estimator [46]. MHE is especially attractive for use with MPC because it can be formulated as a similar online optimization problem.

MHE involves the solution of a finite horizon online optimization problem which, given a model of the process to be estimated, results in a state estimate that is compatible with sets of past measurements and inputs that recede as the current time advances [77, 96, 90]. This estimate is optimal in the sense that it maximizes a criterion that captures the likelihood of the measurements. MHE enjoys desirable asymptotic stability

properties [90], and in many ways is a dual problem to MPC. MHE is depicted in a block
diagram in Figure 1.3, where the process is subject to a disturbance d, and the output
measurements are subjected to noise n. Figure 1.4 shows a cartoon depiction of MHE.
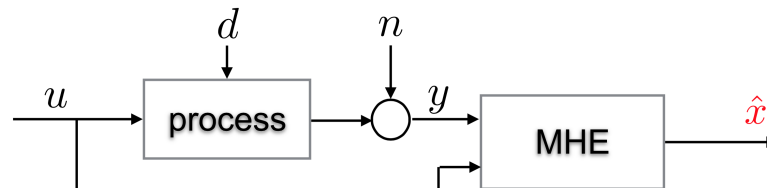
Figure 1.3: Block diagram of the MHE problem.
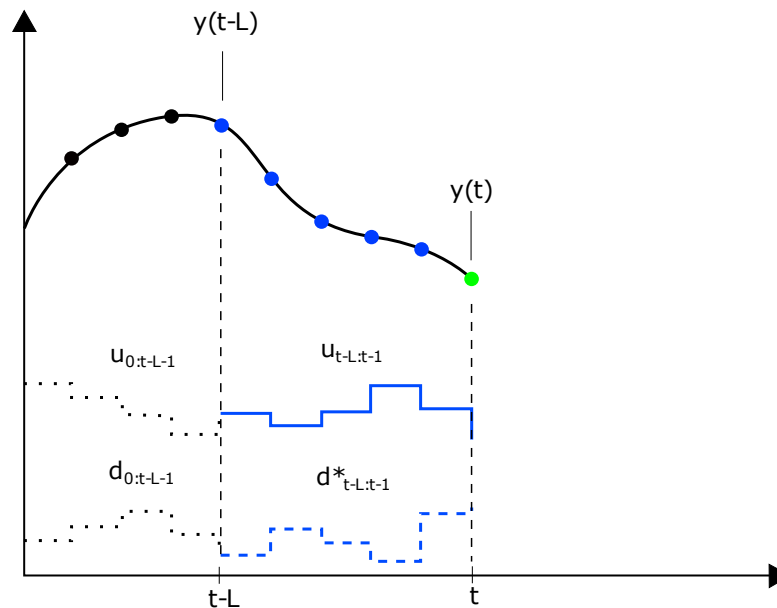
Figure 1.4: Cartoon of the MHE problem. Given a model of the process to be es-
timated and a finite number of past measurements $y_{t-L:t}$ and past inputs $u_{t-L:t-1}$,
sequences of past noises and disturbances $d^*_{t-L:t-1}$ are computed in order to produce
an estimate of the state of the system.

1.2   Statement of Contributions

As noted in a recent survey of MPC and its future directions [72], output-feedback
MPC is a largely an open problem that has many possibilities for future work. The

results presented in this thesis work towards this goal. In this thesis, we propose an approach to combine MPC and MHE into a single optimization that is solved online to construct an output-feedback controller. To account for the uncertainty that results from unmeasured disturbances and measurement noise, we replace the minimization that is used in classical MPC by a min-max optimization. In this case, the minimization is carried out with respect to future control actions, and the maximization is taken with respect to the variables that cannot be measured, namely the system's initial state, the unmeasured disturbances, and the output measurement noise. The criterion for this min-max optimization combines a term that captures the control objective and a term that captures the likelihood of the uncertain variables, resulting in essentially the summation of an MPC criterion with an MHE criterion. A block diagram of this MPC/MHE formulation is shown in Figure 1.5, and Figure 1.6 shows a cartoon depiction.
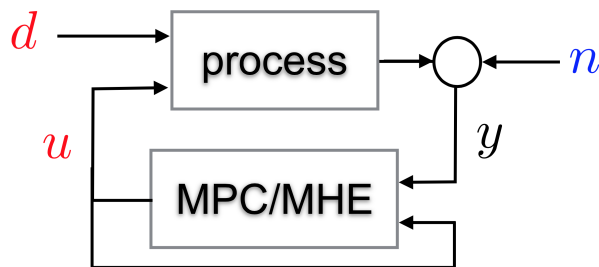
Figure 1.5: Block diagram of the combined MPC/MHE problem investigated in this thesis.

Contributions of this thesis include the theory and design of a new approach to solving MHE and MPC problems simultaneously as a single min-max optimization problem [22, 25, 23] as well as the investigation of using this approach in several applications (see, e.g. [85, 21]). In particular, the contributions of each chapter are given as follows:

Chapter 2 provides the main theoretical contribution of this thesis which addresses stability of the proposed combined MPC/MHE approach. We show that the proposed output-feedback controller results in closed-loop trajectories along which the state of
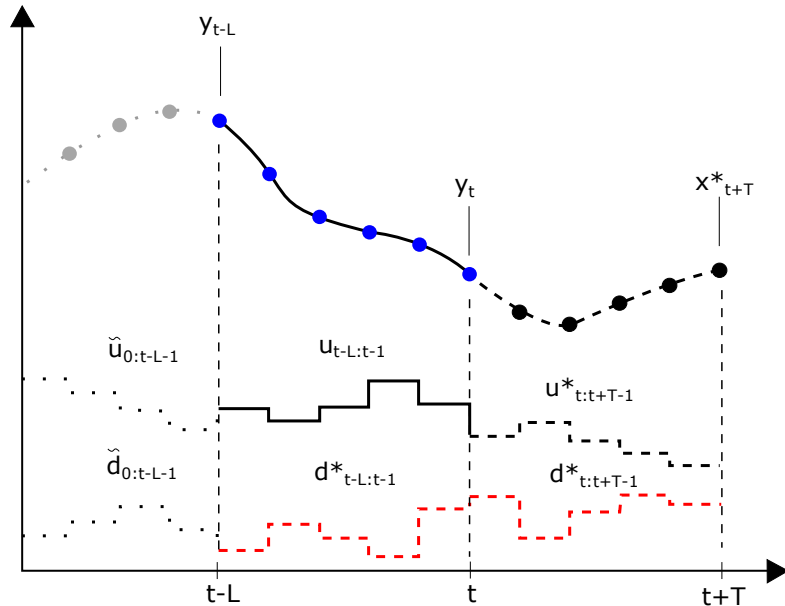
Figure 1.6: Cartoon of the combined MPC/MHE problem. Given a model of the process to be estimated and controlled as well as a finite number of past inputs $u_{t-L:t-1}$ and past measurements $y_{t-L:t}$, solving the combined MPC/MHE results in sequences of past disturbances $d^*_{t-L:t-1}$ and noise that produce a state estimate as well as future sequences of disturbances $d^*_{t:t+T-1}$ and control inputs $u^*_{t:t+T-1}$ that produce a predicted state trajectory $x^*_{t:t+T}$ into the future.

the process remains bounded, and, for tracking problems, our results provide explicit bounds on the tracking error. These results rely on three key assumptions: The first assumption requires the existence of saddle-point equilibria for the min-max optimization, or equivalently, that the min and max commute. In practice, this assumption can be viewed as a form of observability of the process. The second key assumption requires the optimization criterion to include a terminal cost that is a control ISS-Lyapunov function with respect to the disturbance input. This type of assumption is common in classical state-feedback robust MPC. The third and final assumption involves observability of the nonlinear process and essentially requires that the backwards horizon is sufficiently large so that enough information about the initial state is obtained in order to find past estimates that are compatible with the dynamics.

Chapter 3 addresses the first assumption invoked in Chapter 2, i.e., the existence of

saddle-point equilibria. In this chapter, we derive conditions under which a saddle-point solution to the min-max optimization is guaranteed to exist. For linear processes and quadratic costs, these conditions boil down to observability of the linear process and the appropriate choice of weights in the quadratic cost functions.

Chapter 4 presents two numerical algorithms that can be used to solve general min-max optimization problems, including those introduced in Chapter 2. Both algorithms involve primal-dual interior-point methods that rely on Newton's method to solve a relaxed version of the Karush-Kuhn-Tucker (KKT) conditions associated with the coupled optimizations that define a saddle-point equilibrium. The second algorithm specializes this method for formulations with common latent variables. Several numerical examples throughout this thesis demonstrate the capabilities of these algorithms.

Chapter 5 discusses the scenario where the process model includes uncertain or unknown parameters. In this case, the uncertain parameters can be included as optimization variables in the combined MPC/MHE approach and learned online. Furthermore, we show that the results from Chapter 2 still hold. Several examples with parametric uncertainty are considered, and in all of the examples the MPC/MHE approach provides effective control.

This thesis also includes the investigation of this combined MPC/MHE approach for multiple applications including the coordination of unmanned aerial vehicles for vision-based target tracking of a moving ground vehicle, feedback control of an artificial pancreas system for the treatment of Type 1 Diabetes, estimation and control of a DC motor, and distributed control of multi-agents for achieving consensus.

In **Chapter 6**, we consider the coordination of multiple unmanned aerial vehicles (UAVs) for vision-based target-tracking of a moving ground vehicle. This application is conveniently formulated using the combined MPC/MHE approach because it can be formulated as a nonlinear pursuit-evasion two-player game, and, therefore, is naturally

solved using a min-max optimization where the ground vehicle acts as an evasive target who tries to maximum the distance between itself and the UAVs, while the UAVs try to minimize this distance. A novel cost function is used in order to achieve the best vision-based estimate of the target's location, and we show in simulations that the UAVs are able to coordinate their motion to track the target even when the target acts evasively.

Chapter 7 explores the use of the MPC/MHE approach for the feedback control of an artificial pancreas system for the treatment of Type 1 Diabetes. This is an inherently asymmetric problem with many safety considerations due to the possibly serious consequences of poorly regulated blood-glucose. Therefore, we design an asymmetric cost function to facilitate appropriate controller response to high and low blood-glucose levels. We show that the combined MPC/MHE approach is advantageous for this application when compared to another approach using state-feedback MPC and a recursive state estimator.

Finally, in **Chapter 8**, we discuss two additional applications which include output-feedback control of a DC motor and distributed optimization for multi-agent consensus. Armed with the computationally efficient algorithms described in Chapter 4, the combined MPC/MHE approach can be applied to both of these problems in order to solve the online optimizations in real-time. This is important for these two applications in particular because the DC motor exhibits fast dynamics, and the real-world implementation of distributed control of multiple agents may require computation using low-power embedded processors on-board robots, for instance. We show that this MPC/MHE approach is an effective control approach for both of these applications.

1.3   Organization

This thesis is organized into two parts. The first part includes Chapters 2, 3, 4, and 5 and covers the theoretical results and design of our novel simultaneous estimation and control scheme as well as numerical optimization methods than can be used to solve the resulting min-max optimization problem. The second part investigates the application of this new MPC/MHE approach to several problems including: 1) the coordination of multiple unmanned aerial vehicles (UAVs) for vision-based target tracking in Chapter 6, 2) control and estimation for an artificial pancreas system used in the treatment of Type I Diabetes in Chapter 7, and 3) output-feedback control of a DC motor, as well as 4) distributed optimization for multi-agent consensus, in Chapter 8.

Notation: Throughout this thesis, we denote by $\mathbb{R}$ the set of real numbers, by $\mathbb{R}_{\geqslant 0}$ the set of non-negative real numbers, by $\mathbb{R}_+$ the set of positive real numbers, by $\mathbb{Z}$ the set of integers, by $\mathbb{Z}_{\geqslant 0}$ the set of non-negative integers, by $\mathbb{Z}_+$ the set of positive integers, and by $\mathbb{Z}_a^b$ the set of consecutive integers $\{a, ..., b\}$. Given a discrete-time signal $z : \mathbb{Z}_{\geqslant 0} \to \mathbb{R}^n$ and two times t_0, $t \in \mathbb{Z}_{\geqslant 0}$ with $t_0 < t$, we denote by $z_{t_0:t}$ the sequence $\{z_{t_0}, z_{t_0+1}, ..., z_t\}$. With a slight abuse of notation, we write $z_{t_0:t} \in \mathcal{Z}$ to mean that each element of the sequence $z_{t_0:t}$ belongs to the set $\mathcal{Z}$. Given a vector $x \in \mathbb{R}^n$, we denote by x' the transpose of x. Given a vector $x \in \mathbb{R}^n$ and a scalar $a \in \mathbb{R}$, we denote by $x \dot{\geqslant} a$ the proposition that every entry of x is greater than or equal to a. Given an integer M, we denote by $\mathbf{0}_M$ and by $\mathbf{1}_M$ the M-vectors with all entries equal to 0 and 1, respectively. Given two vectors $x, y \in \mathbb{R}^n$ we denote by $x \odot y \in \mathbb{R}^n$ and by $x \oslash y \in \mathbb{R}^n$ the entry-wise product and division of the two vectors, respectively.

Part I

Theory and Design

Chapter 2

Simultaneous Nonlinear Model Predictive Control and State Estimation

Parts of this chapter come from [22] and [25]:

2014 IEEE. Reprinted, with permission, from D. A. Copp and J. P. Hespanha, Nonlinear output-feedback model predictive control and moving horizon estimation, 2014 IEEE 53rd Conference on Decision and Control (CDC), Dec. 2014.

In this chapter we introduce an output-feedback approach to nonlinear model predictive control that combines state estimation and control into a single min-max optimization. Specifically, a chosen criterion is minimized with respect to control input variables and is maximized with respect to the unknown initial state as well as disturbance and measurement noise variables. Under appropriate assumptions that encode controllability and observability of the nonlinear process to be controlled, we prove that the state of the closed-loop system remains bounded and establish bounds on the tracking error for trajectory tracking problems. The results apply both to infinite and finite horizon

optimizations, the latter requiring an additional observability assumption and the use of a terminal cost that is an ISS-control Lyapunov function with respect to a disturbance input. The combined MPC/MHE approach considered here was introduced in [22, 25].

2.1 Introduction

Online optimization has become a ubiquitous approach for solving control and estimation problems in both academia and industry. This is largely due to the ability to explicitly accommodate hard state and input constraints in online optimization techniques. Because of this, an especially popular online optimization control technique called model predictive control (MPC) is used in numerous industrial applications [84], and, consequently, much effort has been devoted to developing a stability theory for MPC (see e.g. [79, 16, 95, 43, 92]).

MPC involves the solution of an open-loop optimal control problem at each sampling time. Each of these optimizations results in a sequence of future optimal control actions and a sequence of corresponding future states. The first control action in the sequence is applied to the plant, and then the optimization is solved again at the next sampling time. MPC has historically been popular for problems in which the plant dynamics are sufficiently slow so that the optimization can be solved between consecutive sampling times. However, as available computational power increases and optimization algorithms improve in terms of speed, MPC can be applied to broader application areas.

MPC is often formulated assuming that the full state of the process to be controlled can be measured. However, this is not possible in many practical cases, so the use of independent algorithms for state-estimation, including observers, filters, and moving horizon estimation (MHE), as discussed, i.e., in [94], is required. Of these methods, MHE is especially attractive for use with MPC because it can be formulated as a similar online

optimization problem that explicitly handles constraints. Solving the MHE problem produces a state estimate that is compatible with a set of past measurements that recedes as the current time advances. This estimate is optimal in the sense that it maximizes a criterion that captures the likelihood of the measurements. By receding the set of measurements considered in the MHE optimization, one maintains a constant computational cost for the optimization.

In this chapter, we propose an approach to combine MPC and MHE into a single optimization that is solved online to construct an output-feedback controller. To account for the uncertainty that results from unmeasured disturbances and measurement noise, we replace the minimization that is used in classical MPC by a min-max optimization. In this case, the minimization is carried out with respect to future control actions, and the maximization is taken with respect to the variables that cannot be measured, namely the system's initial state, the unmeasured disturbances, and the output measurement noise. The criterion for this min-max optimization combines a term that captures the control objective and a term that captures the likelihood of the uncertain variables, resulting in essentially the summation of an MPC criterion with an MHE criterion.

The main technical contribution of this chapter addresses the stability of the proposed combined MPC/MHE approach. We show that the proposed output-feedback controller results in closed-loop trajectories along which the state of the process remains bounded, and, for tracking problems, our results provide explicit bounds on the tracking error. These results rely on three key assumptions: The first assumption requires the existence of saddle-point equilibria for the min-max optimization, or equivalently, that the min and max commute. In practice, this assumption can be viewed as a form of observability of the process. The second key assumption requires the optimization criterion to include a terminal cost that is a control ISS-Lyapunov function with respect to the disturbance input. This type of assumption is common in classical state-feedback robust MPC.

The final observability assumption essentially requires that the backwards horizon is sufficiently large so that enough information about the initial state is obtained in order to find past estimates that are compatible with the dynamics.

2.1.1   Related Work

State-feedback MPC is a mature field with numerous contributions. Particularly relevant to the results in this thesis is the work on the so-called robust or min-max MPC, which considers model uncertainty, input disturbances, and noise [17, 58, 9, 67]. Min-max MPC for constrained linear systems was considered by [99] and [8], and a game theoretic approach for robust constrained nonlinear MPC was proposed by [19]. More recent studies of input-to-state stability of min-max MPC can be found in [57, 62, 88]. These works focused on state-feedback MPC and did not consider robustness with respect to errors in state estimation. A novelty of the work presented in this thesis is the reliance on saddle-point equilibria, rather than a simple min-max optimal, which we found instrumental in establishing our stability results.

Fewer results are available for output-feedback MPC. An overview of nonlinear output-feedback MPC is given by [34] and the references therein. Many of these output-feedback approaches involve designing separate state estimator and MPC schemes. Several of the observers, estimators, and filters that have been proposed for use with nonlinear output-feedback MPC include an extended Kalman filter [49], optimization based moving horizon observers [77], high gain observers [50], extended observers [97], and robust MHE [112]. In contrast to solving the estimation and control problems separately, the formulation of our combined MPC/MHE approach as a single optimization facilitates the stability analysis of the closed-loop without the need for a separation principle for nonlinear systems.

Results on robust output-feedback MPC for constrained, linear, discrete-time systems

with bounded disturbances and measurement noise can be found in [74, 75], where a stable Luenberger observer is employed for state estimation and robustly stabilizing tube-based MPC is performed to control the state of the observer. Alternatively, in [103], MHE is employed for state estimation and is combined with a similar tube-based MPC approach. These approaches first solve the estimation problem and show convergence of the state estimate to a bounded set and then take the uncertainty of the state estimate into account when solving the robust MPC problem. The work of [64] combines an estimation scheme, which provides a guaranteed ellipsoidal error bound on the state estimate, with a min-max MPC scheme for estimation and control of linear systems with bounded disturbances and measurement noise.

During the same time that many important results on MPC were developed, parallel work began on MHE. The work of [4] gives a tutorial overview and background of both MPC and MHE as well as methods that can be used to solve these optimization problems. Useful overviews of constrained linear and nonlinear MHE can be found in [89] and [90] where, with appropriate assumptions regarding observability, continuity, and an approximate arrival cost, the authors prove asymptotic stability as well as bounded stability in the presence of bounded noise.

More recent results regarding MHE for discrete-time nonlinear systems are given by [3], in which the authors minimize a quadratic cost that includes the standard output error term as well as a term penalizing the distance of the current estimated state from its prediction. The authors prove boundedness of the estimation error, when considering bounded disturbances and measurement noise, and convergence of the state estimate to the true value in the noiseless case. Even more recent work on robust MHE for nonlinear systems appeared in [63], where first a high-gain observer is used to bound the estimation error, and then that bound is used to design a constraint for incorporation in an MHE problem. This formulation seems to reduce the sensitivity of the performance of MHE

to the accuracy of the approximate arrival cost, and boundedness of the state estimate
is proven when the noise is bounded.

This chapter is organized as follows. In Section 2.2, we formulate the control problem
we would like to solve and discuss its relationship to MPC and MHE. In Sections 2.3
and 2.4 we state the main closed-loop stability results for the infinite horizon and finite
horizon cases, respectively. Several numerical examples are given in Section 2.5, and we
provide some conclusions in Section 2.6.

2.2   Problem Formulation

In this thesis, we consider the control of time-varying nonlinear discrete-time processes
of the form

$$x_{t+1} = f_t(x_t, u_t, d_t), \qquad\qquad y_t = g_t(x_t) + n_t, \qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0} \qquad (2.1)$$

with *state* x_t taking values in a set $\mathcal{X} \subset \mathbb{R}^{n_x}$. The inputs to this system are the *control
input* u_t that must be restricted to the set $\mathcal{U} \subset \mathbb{R}^{n_u}$, the *unmeasured disturbance* d_t that
is known to belong to the set $\mathcal{D} \subset \mathbb{R}^{n_d}$, and the *measurement noise* n_t belonging to the
set $\mathcal{N} \subset \mathbb{R}^{n_n}$. The signal y_t, belonging to the set $\mathcal{Y} \subset \mathbb{R}^{n_y}$, denotes the *measured output*
that is available for feedback. The *control objective* is to select the control signal $u_t \in \mathcal{U}$,
$\forall t \in \mathbb{Z}_{\geqslant 0}$ so as to minimize a criterion of the form

$$\sum_{t=0}^{\infty} c_t(x_t, u_t, d_t) - \sum_{t=0}^{\infty} \eta_t(n_t) - \sum_{t=0}^{\infty} \rho_t(d_t), \qquad (2.2)$$

for worst-case values of the unmeasured disturbance $d_t \in \mathcal{D}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$ and the measure-
ment noise $n_t \in \mathcal{N}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$. The functions $c_t(\cdot)$, $\eta_t(\cdot)$, and $\rho_t(\cdot)$ in (2.2) are all assumed
to take non-negative values. One can view the terms $\rho_t(\cdot)$ and $\eta_t(\cdot)$ as measures of the

likelihood of specific values for d_t and n_t. Then, the negative signs in front of $\rho_t(\cdot)$ and $\eta_t(\cdot)$ penalize the maximizer for using low likelihood values for the disturbances and noise (low likelihood meaning very large values for $\rho_t(\cdot)$ and $\eta_t(\cdot)$).

To better understand (2.2), it is also useful to note that boundedness of the criterion (2.2) by a constant γ guarantees that

$$\sum_{t=0}^{\infty} c_t(x_t, u_t, d_t) \leqslant \gamma + \sum_{t=0}^{\infty} \eta_t(n_t) + \sum_{t=0}^{\infty} \rho_t(d_t). \tag{2.3}$$

In what follows, we allow the functions $\eta_t(\cdot)$ and $\rho_t(\cdot)$ in the criterion (2.2) to take the value $+\infty$. This provides a convenient formalism to consider bounded disturbances and noise while formally allowing n_t and d_t to take values in the whole spaces $\mathbb{R}^{n_n}$ and $\mathbb{R}^{n_d}$, respectively. Specifically, considering *extended-value extensions* [14] of the form

$$\rho_t(d_t) \coloneqq \begin{cases} \bar{\rho}_t(d_t) & d_t \in \mathcal{D} \\ \infty & d_t \notin \mathcal{D}, \end{cases} \qquad \eta_t(n_t) \coloneqq \begin{cases} \bar{\eta}_t(n_t) & n_t \in \mathcal{N} \\ \infty & n_t \notin \mathcal{N}, \end{cases} \tag{2.4}$$

with $\bar{\rho}_t$ and $\bar{\eta}_t$ bounded in $\mathcal{D}$ and $\mathcal{N}$, respectively, the minimization of (2.2) with respect to the control signal u_t need not consider cases where d_t and n_t take values outside $\mathcal{D}$ and $\mathcal{N}$, respectively, as this would directly lead to the cost $-\infty$ for any control signal u_t that keeps the positive term bounded.

Remark 1 *While the results presented here are general, the reader is encouraged to consider the quadratic case $c_t(x_t, u_t, d_t) \coloneqq \|x_t\|^2 + \|u_t\|^2$, $\eta_t(n_t) \coloneqq \|n_t\|^2$, $\rho_t(d_t) \coloneqq \|d_t\|^2$ to gain intuition on the results. In this case, (2.3) would guarantee that the state x_t and input u_t are ℓ_2, provided that the disturbance d_t and noise n_t are also ℓ_2. This would mean that the closed-loop has a finite ℓ_2-induced gain.* $\qquad\square$

2.3   Infinite Horizon Online Optimization

To overcome the conservativeness of an open-loop control, we use online optimization to generate the control signals. Specifically, at each time $t \in \mathbb{Z}_{\geqslant 0}$, we compute the control u_t so as to minimize

$$\sum_{s=t}^{\infty} c_s(x_s, u_s, d_s) - \sum_{s=0}^{t} \eta_s(n_s) - \sum_{s=0}^{\infty} \rho_s(d_s) \tag{2.5}$$

under worst-case assumptions on the *unknown* system's initial condition x_0, unmeasured disturbances d_t, and measurement noise n_t, subject to the constraints imposed by the system dynamics and the measurements y_t collected up to the current time t. Since the goal is to optimize this cost at the current time t to compute the control inputs at times $s \geqslant t$, there is no point in penalizing the running cost $c_s(x_s, u_s, d_s)$ for past time instants $s < t$, which explains the fact that the first summation in (2.5) starts at time t. There is also no point in considering the values of future measurement noise at times $s > t$, as they will not affect choices made at time t, which explains the fact that the second summation in (2.5) stops at time t. However, we do need to consider all values for the unmeasured disturbance d_s because past values affect the (unknown) current state x_t, and future values affect the future values of the running cost.

The following notation facilitates formalizing the control law proposed: Given a discrete-time signal $z : \mathbb{Z}_{\geqslant 0} \rightarrow \mathbb{R}^n$ and two times $t_0, t \in \mathbb{Z}_{\geqslant 0}$ with $t_0 \leqslant t$, we denote by $z_{t_0:t}$ the sequence $\{z_{t_0}, z_{t_0+1}, ..., z_t\}$. This notation allows us to re-write (2.5) as

$$J_t^{\infty}(x_0, u_{0:\infty}, d_{0:\infty}, y_{0:t}) := \sum_{s=t}^{\infty} c_s(x_s, u_s, d_s) - \sum_{s=0}^{t} \eta_s\big(y_s - g_s(x_s)\big) - \sum_{s=0}^{\infty} \rho_s(d_s), \tag{2.6}$$

which emphasizes the dependence of (2.5) on the unknown initial state x_0, the unknown disturbance input sequence $d_{0:\infty}$, the measured output sequence $y_{0:t}$, and the control input

sequence $u_{0:\infty}$. Regarding the latter, one should recognize that $u_{0:\infty}$ is composed of two distinct sequences: the (known) past inputs $u_{0:t-1}$ that have already been applied, and the future inputs $u_{t:\infty}$ that still need to be selected.

At a given time $t \in \mathbb{Z}_{\geqslant 0}$, we do not know the value of the variables x_0 and $d_{0:\infty}$ on which the value of criterion (2.6) depends, so we optimize this criterion under worst-case assumptions on these variables, leading to the following min-max optimization

$$\min_{\hat{u}_{t:\infty|t} \in \mathcal{U}} \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} J_t^\infty(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}_{t:\infty|t}, \hat{d}_{0:\infty|t}, y_{0:t}), \tag{2.7}$$

where the arguments $u_{0:t-1}, \hat{u}_{t:\infty|t}$ to the function $J_t^\infty(\cdot)$ in (2.7) correspond to the argument $u_{0:\infty}$ in the definition of $J_t^\infty(\cdot)$ in the left-hand side of (2.6). The subscript $._{|t}$ in the (dummy) optimization variables in (2.7) emphasizes that this optimization is repeated at each time step $t \in \mathbb{Z}_{\geqslant 0}$. At different time steps, these optimizations typically lead to different solutions, which generally do not coincide with the real control input, disturbances, and noise. We can view the optimization variables $\hat{x}_{0|t}$ and $\hat{d}_{0:\infty|t}$ as (worst-case) estimates of the initial state and disturbances, respectively, based on the past inputs $u_{0:t-1}$ and outputs $y_{0:t}$ available at time t.

As is common in model predictive control, at each time t, we use as the control input the first element of the sequence

$$\hat{u}_{t:\infty|t}^* = \{\hat{u}_{t|t}^*, \hat{u}_{t+1|t}^*, \hat{u}_{t+2|t}^*, \dots\} \in \mathcal{U}$$

that minimizes (2.7), leading to the following control law:

$$u_t = \hat{u}_{t|t}^*, \quad \forall t \geqslant 0. \tag{2.8}$$

Relationship with Model Predictive Control

When the state of (2.1) can be measured exactly and the maps $d_t \mapsto f_t(x_t, u_t, d_t)$ are injective (for each fixed x_t and u_t), the initial state x_0 and past values for the disturbance $d_{0:t-1}$ are uniquely defined by the "measurements" $x_{0:t}$. In this case, the control law (2.8) that minimizes (2.7) can also be determined by the optimization

$$\min_{\hat{u}_{t:\infty|t} \in \mathcal{U}} \quad \max_{\hat{d}_{t:\infty|t} \in \mathcal{D}} J_t^\infty(x_0, u_{0:t-1}, \hat{u}_{t:\infty|t}, d_{0:t-1}, \hat{d}_{t:\infty|t}),$$

with

$$J_t^\infty(x_t, u_{t:\infty}, d_{t:\infty}) := \sum_{s=t}^\infty c_s(x_s, u_s, d_s) - \sum_{s=t}^\infty \rho_s(d_s),$$

which is essentially the robust model predictive control problem considered in [19, 70].

Remark 2 (Economic MPC) *It is worth noting that our framework is more general than standard forms of MPC where the minimal cost is achieved at the optimal feasible state and input in order to ensure stability of the desired state. It can also apply to economic MPC where the operating cost of the plant is used directly in the objective function, and therefore the cost need not be zero or minimal at the optimal state and input [93].* □

Relationship with Moving-Horizon Estimation

When setting both $c_s(\cdot)$ and $q_{t+T}(\cdot)$ equal to zero in the criterion (2.6), this optimization no longer depends on $u_{t:\infty}$ and $d_{t:\infty}$, so the optimization in (2.7) simply becomes

$$\max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:t-1|t} \in \mathcal{D}}} J_t^\infty(\hat{x}_{0|t}, u_{0:t-1}, \hat{d}_{0:t-1|t}, y_{0:t}),$$

where now the optimization criterion only contains a finite number of terms that recede as the current time t advances:

$$J_t^\infty(x_0, u_{0:t-1}, d_{0:t-1}, y_{0:t}) := -\sum_{s=0}^{t-1} \rho_s(d_s) - \sum_{s=0}^{t} \eta_s\big(y_s - g_s(x_s)\big),$$

which is essentially the moving horizon estimation problem considered in [90, 3].

2.3.1   Closed-Loop Boundedness and Tracking

We now show that the control law (2.8) leads to boundedness of the state of the closed-loop system under appropriate assumptions, which we discuss next.

A necessary condition for the implementation of the control law (2.8) is that the outer minimizations in (2.7) lead to finite values for the optima that are achieved at specific sequences $\hat{u}^*_{t:\infty|t} \in \mathcal{U}$, $t \in \mathbb{Z}_{\geqslant 0}$. However, for the stability results in this section we actually ask for the existence of a saddle-point solution to the min-max optimizations in (2.7), which is a common requirement in game theoretical approaches to control design [6]:

Assumption 1 (Saddle-point) *The min-max optimization* (2.7) *always has a saddle-point solution for which the min and max commute. Specifically, for every time $t \in \mathbb{Z}_{\geqslant 0}$, past control input sequence $u_{0:t-1} \in \mathcal{U}$, and past measured output sequence $y_{0:t} \in \mathcal{Y}$, there exists a finite scalar $J_t^*(u_{0:t-1}, y_{0:t}) \in \mathbb{R}$, an initial condition $\hat{x}^*_{0|t} \in \mathcal{X}$, and sequences $\hat{u}^*_{t:\infty|t} \in \mathcal{U}$, $\hat{d}^*_{0:\infty|t} \in \mathcal{D}$ such that*

$$J_t^{\infty*}(u_{0:t-1}, y_{0:t}) = J_t(\hat{x}^*_{0|t}, u_{0:t-1}, \hat{u}^*_{t:\infty|t}, \hat{d}^*_{0:\infty|t}, y_{0:t})$$

$$= \min_{\substack{\hat{u}_{t:\infty|t} \in \mathcal{U}}} \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} J_t^\infty(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}_{t:\infty|t}, \hat{d}_{0:\infty|t}, y_{0:t})$$

$$= \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} J_t^\infty(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}^*_{t:\infty|t}, \hat{d}_{0:\infty|t}, y_{0:t}) \qquad (2.9a)$$

$$= \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} \min_{\hat{u}_{t:\infty|t} \in \mathcal{U}} J_t^\infty(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}_{t:\infty|t}, \hat{d}_{0:\infty|t}, y_{0:t})$$

$$= \min_{\hat{u}_{t:\infty|t} \in \mathcal{U}} J_t^\infty(\hat{x}_{0|t}^*, u_{0:t-1}, \hat{u}_{t:\infty|t}, \hat{d}_{0:\infty|t}^*, y_{0:t}) \tag{2.9b}$$

$$< \infty. \tag{2.9c}$$

Assumption 1 presumes an appropriate form of *observability/detectability* adapted to the criterion $\sum_{s=t}^\infty c_s(x_s, u_s, d_s)$ because (2.9a) implies that, for every initial condition $\hat{x}_{0|t} \in \mathcal{X}$ and disturbance sequence $\hat{d}_{0:\infty|t} \in \mathcal{D}$,

$$c_t(\hat{x}_{t|t}, \hat{u}_{t|t}^*, \hat{d}_{t|t}) \leqslant J_t^{\infty*}(u_{0:t-1}, y_{0:t}) + \sum_{s=0}^\infty \rho_s(\hat{d}_{s|t}) + \sum_{s=0}^t \eta_s(y_s - g_s(\hat{x}_{s|t})). \tag{2.10}$$

This means that we can bound the size of the *current* state using past outputs and past/future input disturbances. Assumption 1 also presumes an appropriate form of *controllability/stabilizability* adapted to the criterion $\sum_{s=t}^\infty c_s(x_s, u_s, d_s)$ because (2.9a) implies that the *future* control sequence $\hat{u}_{t:\infty|t}^* \in \mathcal{U}$ is able to keep "small" the size of *future* states as long as the noise and disturbance remain "small".

The following theorem is the main result of this section and provides a bound that can be used to prove boundedness of the state when the control signal is constructed using the *infinite horizon* criterion (2.6).

Theorem 1 (Infinite horizon cost-to-go bound) *Suppose that Assumption 1 holds. Then, for every $t \in \mathbb{Z}_{\geqslant 0}$, the trajectories of the process (2.1) with control (2.8) defined by the infinite-horizon optimization (2.7) satisfy*

$$c_t(x_t, u_t, d_t) \leqslant J_0^{\infty*}(y_0) + \sum_{s=0}^t \eta_s(n_s) + \sum_{s=0}^t \rho_s(d_s). \tag{2.11}$$

$\square$

22

Proof

Before proving Theorem 1, we introduce a key technical lemma that establishes the monotonicity of the sequence $\{J_t^{\infty*}(u_{0:t-1}, y_{0:t}) : t \in \mathbb{Z}_{\geqslant 0}\}$.

Lemma 1 *Under Assumption 1 and using the control law (2.8), we have that*

$$J_{t+1}^{\infty*}(u_{1:t}, y_{1:t+1}) - J_t^{\infty*}(u_{0:t-1}, y_{0:t}) \leqslant 0, \quad \forall t \in \mathbb{Z}_{\geqslant 0} \tag{2.12}$$

and consequently

$$J_t^{\infty*}(u_{0:t-1}, y_{0:t}) \leqslant J_0^{\infty*}(y_0), \quad \forall t \in \mathbb{Z}_{\geqslant 0}. \tag{2.13}$$

$\square$

Proof of Lemma 1. We conclude from (2.9b) in Assumption 1 at time $t+1$ that

$$J_{t+1}^{\infty*}(u_{0:t}, y_{0:t+1}) = \min_{\hat{u}_{t+1:\infty|t+1} \in \mathcal{U}} J_{t+1}^{\infty}(\hat{x}_{0|t+1}^*, u_{0:t}, \hat{u}_{t+1:\infty|t+1}, \hat{d}_{0:\infty|t+1}^*, y_{0:t+1})$$

$$\leqslant J_{t+1}^{\infty}(\hat{x}_{0|t+1}^*, u_{0:t}, \hat{u}_{t+1:\infty|t}^*, \hat{d}_{0:\infty|t+1}^*, y_{0:t+1}), \tag{2.14}$$

where the inequality results from the fact that the minimization with respect to $\hat{u}_{t+1:\infty|t+1} \in \mathcal{U}$ must lead to a value no larger than what would be obtained by setting $\hat{u}_{t+1:\infty|t+1} = \hat{u}_{t+1:\infty|t}^*$.

Similarly, we can conclude from (2.9a) in Assumption 1 that

$$J_t^{\infty*}(u_{0:t-1}, y_{0:t}) = \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} J_t^{\infty}(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}_{t:\infty|t}^*, \hat{d}_{0:\infty|t}^*, y_{0:t})$$

$$\geqslant J_t^{\infty}(\hat{x}_{0|t+1}^*, u_{0:t-1}, \hat{u}_{t:\infty|t}^*, \hat{d}_{0:\infty|t+1}^*, y_{0:t})$$

23

$$= J_t^\infty(\hat{x}_{0|t+1}^*, u_{0:t}, \hat{u}_{t+1:\infty|t}^*, \hat{d}_{0:\infty|t+1}^*, y_{0:t}), \qquad (2.15)$$

where the inequality results from the fact that the maximization with respect to $\hat{x}_{0|t}$ and $\hat{d}_{0:\infty|t}$ must lead to a value no smaller than what would be obtained by setting $\hat{x}_{0|t} = \hat{x}_{0|t+1}^*$, $\hat{d}_{0:\infty|t} = \hat{d}_{0:\infty|t+1}^*$. The last equality stems from the control law (2.8). Combining (2.14) and (2.15) leads to

$$J_{t+1}^{\infty*}(u_{0:t}, y_{0:t+1}) - J_t^{\infty*}(u_{0:t-1}, y_{0:t}) \leqslant$$

$$J_{t+1}^\infty(\hat{x}_{0|t+1}^*, u_{0:t}, \hat{u}_{t+1:\infty|t}^*, \hat{d}_{0:\infty|t+1}^*, y_{0:t+1}) - J_t^\infty(\hat{x}_{0|t+1}^*, u_{0:t}, \hat{u}_{t+1:\infty|t}^*, \hat{d}_{0:\infty|t+1}^*, y_{0:t}). \quad (2.16)$$

The crucial observation behind this inequality is that *both* terms $J_{t+1}^\infty(\cdot)$ and $J_t(\cdot)$ in the right-hand side of (2.16) are computed along a trajectory initialized at time 0 with the *same initial state* $\hat{x}_{0|t+1}^*$ and share the *same control input sequence* $\{u_{0:t}, \hat{u}_{t+1:\infty|t}^*\}$ and the *same disturbance input sequence* $\hat{d}_{0:\infty|t+1}^*$ from time 0 to time ∞. Denoting such trajectory by

$$\tilde{x}_{s+1} = f_s(\tilde{x}_s, \tilde{u}_s, \tilde{d}_s), \quad s \in \{0 : \infty\}$$

where

$$\tilde{x}_0 \coloneqq \hat{x}_{0|t+1}^*,$$

$$\tilde{d}_s \coloneqq \hat{d}_{s|t+1}^*, \quad \forall s \in \{0 : \infty\},$$

$$\tilde{u}_s \coloneqq \begin{cases} u_s & s \in \{0 : t\} \\ \hat{u}_{s|t}^* & s \in \{t+1 : \infty\}, \end{cases}$$

we can rewrite the inequality (2.16) and express both terms $J_{t+1}^\infty(\cdot)$ and $J_t^\infty(\cdot)$ in terms

of this trajectory as follows:

$$J_{t+1}^{\infty*}(u_{0:t}, y_{0:t+1}) - J_t^{\infty*}(u_{0:t-1}, y_{0:t}) \leqslant$$

$$\sum_{s=t+1}^{\infty} c_s(\tilde{x}_s, \tilde{u}_s, \tilde{d}_s) - \sum_{s=0}^{t+1} \eta_s(y_s - g_s(\tilde{x}_s)) - \sum_{s=0}^{\infty} \rho_s(\tilde{d}_s)$$

$$- \sum_{s=t}^{\infty} c_s(\tilde{x}_s, \tilde{u}_s, \tilde{d}_s) + \sum_{s=0}^{t} \eta_s(y_s - g_s(\tilde{x}_s)) + \sum_{s=0}^{\infty} \rho_s(\tilde{d}_s)$$

$$= -c_t(\tilde{x}_t, \tilde{u}_t, \tilde{d}_t) - \eta_{t+1}(y_{t+1} - g_{t+1}(\tilde{x}_{t+1})).$$

Equation (2.12) follows from this and the fact that $c_t(\cdot)$ and $\eta_{t+1}(\cdot)$ are both non-negative.

∎

We are now ready to prove the main result of this section.

Proof of Theorem 1. Using (2.9a) in Assumption 1, we conclude that

$$J_t^{\infty*}(u_{0:t-1}, y_{0:t}) = \max_{\substack{\hat{x}_{0|t} \in \mathcal{X}, \\ \hat{d}_{0:\infty|t} \in \mathcal{D}}} J_t^{\infty}(\hat{x}_{0|t}, u_{0:t-1}, \hat{u}_{t:\infty|t}^*, \hat{d}_{0:\infty|t}, y_{0:t})$$

$$\geqslant J_t^{\infty}(x_0, u_{0:t-1}, \hat{u}_{t:\infty|t}^*, d_{0:\infty}, y_{0:t}).$$

Using the definition of $J_t^{\infty}(\cdot)$ in (2.6), this inequality becomes

$$J_t^{\infty*}(u_{0:t-1}, y_{0:t}) \geqslant \sum_{s=t}^{\infty} c_s(x_s, \hat{u}_{s|t}^*, d_s) - \sum_{s=0}^{t} \eta_s(y_s - g_s(x_s)) - \sum_{s=0}^{\infty} \rho_s(d_s).$$

Recalling that $n_s = y_s - g_s(x_s)$, we conclude that

$$\sum_{s=t}^{\infty} c_s(x_s, \hat{u}_{s|t}^*, d_s) \leqslant J_t^{\infty*}(u_{0:t-1}, y_{0:t}) + \sum_{s=0}^{t} \eta_s(n_s) + \sum_{s=0}^{\infty} \rho_s(d_s). \tag{2.17}$$

Combining (2.17) and (2.13), we obtain

$$\sum_{s=t}^{\infty} c_s(x_s, \hat{u}_{s|t}^*, d_s) \leqslant J_0^{\infty *}(y_0) + \sum_{s=0}^{t} \eta_s(n_s) + \sum_{s=0}^{\infty} \rho_s(d_s).$$

Since all the terms in the left-hand side are positive, the first term in the left-hand side summation (the one for $s = t$) must also be upper bounded by the right-hand side, leading to

$$c_t(x_t, u_t, d_t) = c_t(x_t, \hat{u}_{t|t}^*, d_t) \leqslant J_0^{\infty *}(y_0) + \sum_{s=0}^{t} \eta_s(n_s) + \sum_{s=0}^{\infty} \rho_s(d_s), \qquad (2.18)$$

where the left-hand side equality follows from (2.8). Since the left-hand side of (2.18) does not depend on d_s, $\forall s > t$, this bound must hold for arbitrary choices of d_s, $\forall s > t$. Equation (2.11) is obtained by picking $d_s = 0$, $\forall s > t$. ∎

Next we discuss the implications of Theorem 1 in terms of establishing bounds on the state of the closed-loop system, asymptotic stability, and the ability of the closed-loop to asymptotically track desired trajectories.

State boundedness and asymptotic stability

When we select criterion (2.6), for which there exists a class $\mathcal{K}_\infty$ function $\alpha(\cdot)$ and class $\mathcal{K}$ functions[1] $\beta(\cdot), \delta(\cdot)$ such that

$$c_t(x, u, d) \geqslant \alpha(\|x\|), \quad \eta_t(n) \leqslant \beta(\|n\|), \quad \rho_t(d) \leqslant \delta(\|d\|),$$

$$\forall x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, d \in \mathbb{R}^{n_d}, n \in \mathbb{R}^{n_n},$$

---

[1] A function $\alpha : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$ is said to belong to class $\mathcal{K}$ if it is continuous, zero at zero, and strictly increasing and is said to belong to class $\mathcal{K}_\infty$ if it belongs to class $\mathcal{K}$ and is unbounded.

we conclude from (2.11) that, along trajectories of the closed-loop system, the following

inequality holds for all $t \in \mathbb{Z}_{\geqslant 0}$:

$$\alpha(\|x_t\|) \leqslant J_0^{\infty*}(y_0) + \sum_{s=0}^{t} \beta(\|n_s\|) + \sum_{s=0}^{t} \delta(\|d_s\|). \tag{2.19}$$

This provides a bound on the state provided that the noise and disturbances are "vanishing," in the sense that

$$\sum_{s=0}^{\infty} \beta(\|n_s\|) < \infty, \qquad\qquad \sum_{s=0}^{\infty} \delta(\|d_s\|) < \infty.$$

Theorem 1 also provides bounds on the state for non-vanishing noise and disturbances

when we use exponentially time-weighted functions $c_t(\cdot)$, $\eta_t(\cdot)$, and $\rho_t(\cdot)$ that satisfy

$$c_t(x, u, d) \geqslant \lambda^{-t} \alpha(\|x\|), \quad \eta_t(n) \leqslant \lambda^{-t} \beta(\|n\|), \quad \rho_t(d) \leqslant \lambda^{-t} \delta(\|d\|), \tag{2.20}$$

for all $x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, d \in \mathbb{R}^{n_d}, n \in \mathbb{R}^{n_n}$ and some $\lambda \in (0, 1)$. In this case we conclude

from (2.11) that for all $t \in \mathbb{Z}_{\geqslant 0}$,

$$\alpha(\|x_t\|) \leqslant \lambda^t J_0^{\infty*}(y_0) + \sum_{s=0}^{t} \lambda^{t-s} \beta(\|n_s\|) + \sum_{s=0}^{t} \lambda^{t-s} \delta(\|d_s\|).$$

Therefore, x_t remains bounded provided that the measurement noise n_t and the unmeasured disturbance d_t are both uniformly bounded. Moreover, $\|x_t\|$ converges to zero as

$t \to \infty$, when the noise and disturbances vanish asymptotically. We have proved the

following:

Corollary 1 *Suppose that Assumption 1 holds and also that (2.20) holds for a class $\mathcal{K}_\infty$

function $\alpha(\cdot)$, class $\mathcal{K}$ functions $\beta(\cdot), \delta(\cdot)$, and $\lambda \in (0, 1)$. Then, for every initial condition x_0, uniformly bounded measurement noise sequence $n_{0:\infty}$, and uniformly bounded*

disturbance sequence $d_{0:\infty}$, the state x_t remains uniformly bounded along the trajectories of the process (2.1) with control (2.8) defined by the infinite-horizon optimization (2.7). Moreover, when d_t and n_t converge to zero as $t \to \infty$, the state x_t also converges to zero.

□

Remark 3 (Time-weighted criteria) *The exponentially time-weighted functions (2.20) typically arise from a criterion of the form*

$$\sum_{s=t}^{\infty} \lambda^{-s} c(x_s, u_s, d_s) - \sum_{s=0}^{t} \lambda^{-s} \eta(n_s) - \sum_{s=0}^{\infty} \lambda^{-s} \rho(d_s)$$

that weight the future more than the past. In this case, (2.20) holds for functions $\alpha(\cdot)$, $\beta(\cdot)$, and $\delta(\cdot)$ such that $c(x, u, d) \geqslant \alpha(\|x\|)$, $\eta(n) \leqslant \beta(\|n\|)$, and $\rho(d) \leqslant \delta(\|d\|)$, $\forall x, u, d, n$.

□

Reference tracking

When the control objective is for the state x_t to follow a given trajectory z_t, the optimization criterion can be selected of the form

$$\sum_{s=t}^{\infty} \lambda^{-s} c(x_s - z_s, u_s, d_s) - \sum_{s=0}^{t} \lambda^{-s} \eta(n_s) - \sum_{s=0}^{\infty} \lambda^{-s} \rho(d_s),$$

with $c(x, u, d) \geqslant \alpha(\|x\|)$, $\forall x, u, d$ for some class $\mathcal{K}_{\infty}$ function α and $\lambda \in (0, 1)$. In this case, we conclude from (2.11) that, for all $t \in \mathbb{Z}_{\geqslant 0}$,

$$\alpha(\|x_t - z_t\|) \leqslant \lambda^t J_0^{\infty*}(y_0) + \sum_{s=0}^{t} \lambda^{t-s} \eta(n_s) + \sum_{s=0}^{t} \lambda^{t-s} \rho(d_s),$$

which allows us to conclude that x_t converges to z_t as $t \to \infty$ when both n_t and d_t are vanishing sequences, and also that, when these sequences are "ultimately small", the

tracking error $x_t - z_t$ will converge to a small value.

2.4   Finite Horizon Online Optimization

To avoid solving the infinite-dimensional optimization in (2.7) that resulted from the infinite horizon criterion (2.5), we also consider a *finite horizon* version of the criterion (2.5) of the form

$$\sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s) + q_{t+T}(x_{t+T}) - \sum_{s=t-L}^{t} \eta_s(n_s) - \sum_{s=t-L}^{t+T-1} \rho_s(d_s) \qquad (2.21)$$

Again, at each time $t \in \mathbb{Z}_{\geqslant 0}$, we compute the control u_t so as to minimize the criterion (2.21) under worst-case assumptions on the *unknown* system's initial condition x_{t-L}, unmeasured disturbances d_s, and measurement noise n_s, subject to the constraints imposed by the system dynamics and the measurements y_s collected up to the current time t.

For computational tractability, in (2.21) we have replaced the infinite summations that appeared in (2.2) by finite forward and backward horizon lengths. In particular, (2.21) includes $T \in \mathbb{Z}_{\geqslant 1}$ future terms of the running cost $c_s(x_s, u_s, d_s)$, which recede as the current time t advances, and $L + 1 \in \mathbb{Z}_{\geqslant 1}$ past terms of the noise penalty term $\eta_s(n_s)$. The function $q_{t+T}(x_{t+T})$ acts as a terminal cost to penalize the "final" state at time $t+T$.

Since the goal is to optimize (2.21) at the current time t to compute the control inputs at times $s \geqslant t$, there is no point in penalizing the running cost $c_s(x_s, u_s, d_s)$ for past time instants $s < t$, which explains the fact that the first summation in (2.21) starts at time t. There is also no point in considering the values of future measurement noise at times $s > t$, as they will not affect choices made at time t, which explains the fact that the second summation in (2.21) stops at time t. However, we do need to consider all values for the unmeasured disturbance d_s, because past values affect the (unknown)

current state x_t, and future values affect the future values of the running cost.

Therefore, we can define the finite horizon optimization criterion as

$$J_t(x_{t-L}, u_{t-L:t+T-1}, d_{t-L:t+T-1}, y_{t-L:t}) :=$$
$$\sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s) + q_{t+T}(x_{t+T}) - \sum_{s=t-L}^{t} \eta_s\big(y_s - g_s(x_s)\big) - \sum_{s=t-L}^{t+T-1} \rho_s(d_s), \quad (2.22)$$

which emphasizes the dependence of (2.22) on the unknown initial state x_{t-L}, the unknown disturbance input sequence $d_{t-L:t+T-1}$, the measured output sequence $y_{t-L:t}$, and the control input sequence $u_{t-L:t+T-1}$. Regarding the latter, one should note that $u_{t-L:t+T-1}$ is composed of two distinct sequences: the (known) past inputs $u_{t-L:t-1}$ that have already been applied and the future inputs $u_{t:t+T-1}$ that still need to be selected.

At a given time $t \in \mathbb{Z}_{\geqslant L}$, we do not know the value of the variables x_{t-L} and $d_{t-L:t+T-1}$ on which the value of the criterion (2.22) depends, so we optimize this criterion under worst-case assumptions on these variables, leading to the following min-max optimization

$$\min_{\hat{u}_{t:t+T-1|t} \in \mathcal{U}} \quad \max_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} J_t\big(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}\big), \qquad (2.23)$$

where the arguments $u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}$ to the function $J_t(\cdot)$ in (2.23) correspond to the argument $u_{t-L:t+T-1}$ in the definition of $J_t(\cdot)$ in the left-hand side of (2.22). When interpreting (2.23), one should view $\hat{u}_{t:t+T-1|t} \in \mathcal{U}$ as the optimization variables for the (outer) minimization, and $\hat{x}_{t-L|t} \in \mathcal{X}$, $\hat{d}_{t-L:t+T-1|t} \in \mathcal{D}$ as the optimization variables for the (inner) maximization. There are additional optimization variables $\hat{n}_{t-L:t}$, but they are not independent optimization variables as they are uniquely determined by the remaining

optimization variables and the output equation:

$$\hat{n}_{s|t} = y_s - g_s(\hat{x}_{s|t}), \quad \forall s \in \{t - L, t - L + 1, \ldots, t\}.$$

Consequently, a constraint on the noise, such as the condition $\hat{n}_{t-L:t|t} \in \mathcal{N}$, can simply be regarded as a constraint on the remaining optimization variables for the (inner) maximization. Because of this, we do not include the sequence $\hat{n}_{t-L:t|t}$ as explicit optimization variables in (2.23).

The subscript $_{\cdot|t}$ in the optimization variables that appear in (2.23) emphasizes that this optimization is repeated at each time step $t \in \mathbb{Z}_{\geq 0}$. At different time steps these optimizations typically lead to different solutions which generally do not coincide with the real control input, disturbances, and noise. We can view the optimization variables $\hat{x}_{t-L|t}$ and $\hat{d}_{t-L:t+T-1|t}$ as (worst-case) estimates of the initial state and disturbances, respectively, based on the past inputs $u_{t-L:t-1}$ and outputs $y_{t-L:t}$ available at time t.

As is common in MPC, at each time t, we use as the control input the first element of the sequence

$$\hat{u}^*_{t:t+T-1|t} = \{\hat{u}^*_{t|t}, \hat{u}^*_{t+1|t}, \hat{u}^*_{t+2|t}, \ldots \hat{u}^*_{t+T-1|t}\} \in \mathcal{U}$$

that minimizes (2.23), leading to the following control law:

$$u_t = \hat{u}^*_{t|t}, \quad \forall t \geq 0. \tag{2.24}$$

2.4.1 Closed-loop Boundedness and Tracking

To establish state boundedness under the control (2.24) defined by the *finite horizon* optimization criterion (2.22), one requires the same saddle-point Assumption 1 modi-

fied for finite horizon optimization (2.23) as well as additional assumptions regarding observability of the nonlinear process (2.1) and the terminal cost $q_t(\cdot)$.

Assumption 2 (Saddle-point) *The finite horizon min-max optimization in (2.23) always has a saddle-point solution for which the min and max commute. Specifically, for every time $t \in \mathbb{Z}_{\geqslant 0}$, past control input sequence $u_{t-L:t-1} \in \mathcal{U}$, and past measured output sequence $y_{t-L:t} \in \mathcal{Y}$, there exists a finite scalar $J_t^* \in \mathbb{R}$, an initial condition $\hat{x}_{t-L|t}^* \in \mathcal{X}$, and sequences $\hat{u}_{t:t+T-1|t}^* \in \mathcal{U}$, $\hat{d}_{t-L:t+T-1|t}^* \in \mathcal{D}$, and $\hat{n}_{t-L:t|t}^* \in \mathcal{N}$ such that*

$$J_t^* = J_t(\hat{x}_{t-L|t}^*, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}^*, \hat{d}_{t-L:t+T-1|t}^*, y_{t-L:t})$$

$$= \max_{\substack{\hat{x}_{t-L|t}\in\mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t}\in\mathcal{D}}} J_t(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}^*, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}) \tag{2.25a}$$

$$= \min_{\hat{u}_{t:t+T-1|t}\in\mathcal{U}} J_t(\hat{x}_{t-L|t}^*, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}^*, y_{t-L:t}). \tag{2.25b}$$

$\square$

In general, J_t^* depends on the past outputs and control inputs, so we sometimes write $J_t^*(u_{t-L:t-1}, y_{t-L:t})$ to emphasize this dependence.

As in the infinite horizon case, Assumption 2 presumes an appropriate form of *observability/detectability* adapted to the criterion $\sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s)$ because (2.25a) implies that, for every initial condition $\hat{x}_{t-L|t} \in \mathcal{X}$, disturbance sequence $\hat{d}_{t-L:t+T-1|t} \in \mathcal{D}$, and resulting state trajectory $\hat{x}_{t-L:t+T}$,

$$c_t(\hat{x}_t, \hat{u}_{t|t}^*, \hat{d}_{t|t}) \leqslant J_t^*(u_{t-L:t-1}, y_{t-L:t}) + \sum_{s=t-L}^{t+T-1} \rho_s(\hat{d}_{s|t}) + \sum_{s=t-L}^{t} \eta_s(y_s - g_s(\hat{x}_s)).$$

This means that we can essentially bound the size of the *current* state using past outputs and past/future input disturbances. In fact, for linear systems and quadratic costs, Assumption 2 is satisfied if the system is observable and the weights in the cost function

are chosen appropriately [24]. We prove this fact and discuss this assumption further in Chapter 3.

Remark 4 (Alternating min and max) *Assumption 2 also ensures that all of the minimizations and maximizations commute even if (2.23) is written with alternating* min *and* max. *Consider a two-dimensional example. The following inequality holds*

$$\min_{u_1} \min_{u_2} \max_{d_1} \max_{d_2} J(u_1, u_2, d_1, d_2) \geqslant \min_{u_1} \max_{d_1} \min_{u_2} \max_{d_2} J(u_1, u_2, d_1, d_2)$$

$$\geqslant \max_{d_1} \min_{u_1} \max_{d_2} \min_{u_2} J(u_1, u_2, d_1, d_2) \geqslant \max_{d_1} \max_{d_2} \min_{u_1} \min_{u_2} J(u_1, u_2, d_1, d_2) \quad (2.26)$$

because of the fact that $\min_{(.)} \max_{(.)} J \geqslant \max_{(.)} \min_{(.)} J$. *Therefore, since Assumption 2 ensures that the left-hand-side of (2.26) equals the right-hand-side of (2.26), all of the inequalities in between are also equal.*

Assumption 3 (Observability) *There exists a bounded set* $\mathcal{N}_{\mathrm{pre}} \subset \mathbb{R}^{n_n}$ *such that, for every time* $t \in \mathbb{Z}_{\geqslant 0}$, *every state* $\hat{x}_{t-L:t} \in \mathcal{X}$, *and every disturbance and noise sequence,* $\hat{d}_{t-L:t} \in \mathcal{D}$ *and* $\hat{n}_{t-L:t} \in \mathcal{N}$, *that are compatible with the applied control input* u_s, $s \in \mathbb{Z}_{\geqslant 0}$, *and the measured output* y_s, $s \in \mathbb{Z}_{\geqslant 0}$, *in the sense that*

$$\hat{x}_{s+1} = f_s(\hat{x}_s, u_s, \hat{d}_s), \qquad\qquad y_s = g_s(\hat{x}_s) + \hat{n}_s, \qquad\qquad (2.27)$$

$\forall s \in \{t - L, t - L + 1, \ldots, t\}$, *there exists a "predecessor" state estimate* $\hat{x}_{t-L-1} \in \mathcal{X}$, *disturbance estimate* $\hat{d}_{t-L-1} \in \mathcal{D}$, *and noise estimate* $\hat{n}_{t-L-1} \in \mathcal{N}_{\mathrm{pre}}$ *such that (5.11) also holds for time* $s = t - L - 1$. $\qquad\square$

In essence, Assumption 3 requires the past horizon length L to be sufficiently large so that, by observing the system's inputs and outputs over a past time interval $\{t - L, t - L + 1, \ldots, t\}$, one obtains enough information about the initial condition x_{t-L} so that

any estimate $\hat{x}_{t-L}$ that is compatible with the observed input/output data is "precise".

By "precise," we mean that if one were to observe one additional past input/output pair

u_{t-L-1}, y_{t-L-1} just before the original interval, it would be possible to find an estimate

$\hat{x}_{t-L-1}$ for the "predecessor" state x_{t-L-1} that would be compatible with the previous

estimate $\hat{x}_{t-L}$, that is,

$$\hat{x}_{t-L} = f_{t-L-1}(\hat{x}_{t-L-1}, u_{t-L-1}, \hat{d}_{t-L-1}).$$

This "predecessor" state estimate $\hat{x}_{t-L-1}$ would also be compatible with the measured

output at time $t-L-1$ in the sense that the output estimation error lies in the bounded

set $\mathcal{N}_{\mathrm{pre}}$:

$$y_{t-L-1} - g_{t-L-1}(\hat{x}_{t-L-1}) \in \mathcal{N}_{\mathrm{pre}}. \tag{2.28}$$

Note that we do not require the bounded set $\mathcal{N}_{\mathrm{pre}}$ to be the same as the set $\mathcal{N}$ in which

the actual noise is known to lie. In fact, the set $\mathcal{N}_{\mathrm{pre}}$ where the "predecessor" output error

(2.28) should lie may have to be made larger than $\mathcal{N}$ to make sure that Assumption 3

holds. For linear systems, it is straightforward to argue that Assumption 3 holds provided

that the matrix

$$\begin{bmatrix} C \\ CA \\ \vdots \\ CA^L \end{bmatrix}$$

is full column rank and the set $\mathcal{N}_{\mathrm{pre}}$ is chosen sufficiently large. For nonlinear systems,

computing the set $\mathcal{N}_{\mathrm{pre}}$ may be difficult, but fortunately we do not need to compute this

set to implement the controller.

Remark 5 (Choosing length of L) *Although computing the set $\mathcal{N}_{\mathrm{pre}}$ is not required, how large $\mathcal{N}_{\mathrm{pre}}$ needs to be is essentially determined by the length of the backwards horizon L. As the length of L is increased, equation (5.11) provides more constraints on the estimates which leads to better estimates and, therefore, a necessarily smaller set $\mathcal{N}_{\mathrm{pre}}$. In addition, as is discussed later after (2.44), a smaller bound on the norm of the state x may be achieved as L is increased. Therefore, larger L is generally better, but increasing L also increases the computation required to solve (2.23) as the number of optimization variables increases as well. Thus, a heuristic for choosing L is to make it as large as possible given available computational resources.*

Assumption 4 (ISS-control Lyapunov function) *The terminal cost $q_t(\cdot)$ is an ISS-control Lyapunov function, in the sense that, for every $t \in \mathbb{Z}_{\geqslant 0}$, $x \in \mathcal{X}$, there exists a control $u \in \mathcal{U}$ such that for all $d \in \mathcal{D}$*

$$q_{t+1}\big(f_t(x, u, d)\big) - q_t(x) \leqslant -c_t(x, u, d) + \rho_t(d). \tag{2.29}$$

$\square$

Assumption 4 plays the role of the common assumption in MPC that the terminal cost must be a control Lyapunov function for the closed-loop [76]. In the absence of the disturbance d, (2.29) would mean that $q_t(\cdot)$ could be viewed as a control Lyapunov function that decreases along system trajectories for an appropriate control input u [101]. With disturbances, $q_t(\cdot)$ needs to be viewed as an ISS-control Lyapunov function that satisfies an ISS stability condition for the disturbance input d and an appropriate control input u [61]. When, the dynamics are linear and the cost function is quadratic, a terminal

cost $q_t(\cdot)$ satisfying Assumption 4 is typically found by solving a system of linear matrix inequalities.

We are now ready to state the finite horizon counter-part to Theorem 1.

Theorem 2 (Finite horizon cost-to-go bound) *Suppose that Assumptions 2, 3, and 4 hold. Along any trajectory of the closed-loop system defined by the process* (2.1) *and the control law* (2.24)*, we have that*

$$c_t(x_t, u_t, d_t) \leqslant J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \rho_s(\tilde{d}_s) + \sum_{s=0}^{t-L-1} \eta_s(\tilde{n}_s) + \sum_{s=t-L}^{t} \eta_s(n_s) + \sum_{s=t-L}^{t} \rho_s(d_s)$$

$$\forall t \in \mathbb{Z}_{\geqslant L}, \quad (2.30)$$

for appropriate sequences $\tilde{d}_{0:t-L-1} \in \mathcal{D}$, $\tilde{n}_{0:t-L-1} \in \mathcal{N}_{\text{pre}}$. □

The terms $\sum_{s=0}^{t-L-1} \eta_s(\tilde{n}_s) + \sum_{s=0}^{t-L-1} \rho_s(\tilde{d}_s)$ in the right hand side of (5.13) can be thought of as the *arrival cost* that appears in the MHE literature to capture the quality of the estimate at the beginning of the current estimation window [90].

Proof

Before proving Theorem 2, we introduce a key technical lemma that establishes a monotonicity-like property of the sequence $\{J_t^*(u_{t-L:t-1}, y_{t-L:t}) : t \in \mathbb{Z}_{\geqslant 0}\}$ computed along solutions to the closed loop.

Lemma 2 *Suppose that Assumptions 2, 3, and 4 hold. Along any trajectory of the closed-loop system defined by the process* (2.1) *and the control law* (2.24)*, the sequence $\{J_t^*(u_{t-L:t-1}, y_{t-L:t}) : t \in \mathbb{Z}_{\geqslant 0}\}$, whose existence is guaranteed by Assumption 2, satisfies*

$$J_{t+1}^*(u_{t-L+1:t}, y_{t-L+1:t+1}) - J_t^*(u_{t-L:t-1}, y_{t-L:t}) \leqslant \eta_{t-L}(\tilde{n}_{t-L}) + \rho_{t-L}(\tilde{d}_{t-L}), \quad \forall t \in \mathbb{Z}_{\geqslant L}$$

$$(2.31)$$

for appropriate sequences $\tilde{d}_{0:t-L-1} \in \mathcal{D}$, $\tilde{n}_{0:t-L-1} \in \mathcal{N}_{\text{pre}}$. □

The following notation will be used in the proof below to denote the solution to process (2.1): given a control input sequence $u_{t-L:t-1}$ and a disturbance input sequence $d_{t-L:t-1}$, we denote by

$$\varphi(t; t - L, x_{t-L}, u_{t-L:t-1}, d_{t-L:t-1})$$

the state x_t of the system (2.1) at time t for the given inputs and initial condition x_{t-L}.

Proof of Lemma 2. From (2.25b) in Assumption 2 at time $t + 1$, we conclude that there exists an initial condition $\hat{x}^*_{t-L+1|t+1} \in \mathcal{X}$ and sequences $\hat{d}^*_{t-L+1:t+T|t+1} \in \mathcal{D}$, $\hat{n}^*_{t-L+1:t+1|t+1} \in \mathcal{N}$ such that

$$J^*_{t+1}(u_{t-L+1:t}, y_{t-L+1:t+1}) =$$
$$\min_{\hat{u}_{t+1:t+T|t+1} \in \mathcal{U}} J_{t+1}(\hat{x}^*_{t-L+1|t+1}, u_{t-L+1:t}, \hat{u}_{t+1:t+T|t+1}, \hat{d}^*_{t-L+1:t+T|t+1}, y_{t-L+1:t+1}). \quad (2.32)$$

On the other hand, from Assumption 4 at time $t + T$, with $d = \hat{d}^*_{t+T|t+1}$ and

$$x = \hat{x}^*_{t+T|t+1} := \varphi(t + T; t - L + 1, \hat{x}^*_{t-L+1|t+1}, \hat{u}^*_{t+1:t+T|t+1}, \hat{d}^*_{t-L+1:t+T|t+1}),$$

we conclude that there exists a control $\tilde{u}_{t+T} \in \mathcal{U}$ such that

$$q_{t+T+1}\big(f_{t+T}(\hat{x}^*_{t+T|t+1}, \tilde{u}_{t+T}, \hat{d}^*_{t+T|t+1})\big) - q_{t+T}(\hat{x}^*_{t+T|t+1})$$
$$+ c_{t+T}(\hat{x}^*_{t+T|t+1}, \tilde{u}_{t+T}, \hat{d}^*_{t+T|t+1}) - \rho_{t+T}(\hat{d}^*_{t+T|t+1}) \leqslant 0. \quad (2.33)$$

Moreover, we conclude from Assumption 3, that there exist vectors $\tilde{x}_{t-L}$, $\tilde{d}_{t-L} \in \mathcal{D}$,

$\tilde{n}_{t-L} \in \mathcal{N}$ such that

$$\hat{x}^*_{t-L+1|t+1} = f_{t-L}(\tilde{x}_{t-L}, u_{t-L}, \tilde{d}_{t-L}), \tag{2.34}$$

$$y_{t-L} = g_{t-L}(\tilde{x}_{t-L}) + \tilde{n}_{t-L},$$

Using now (2.25a) in Assumption 2 at time t, we conclude that there also exists a finite scalar $J^*_t(u_{t-L:t-1}, y_{t-L:t}) \in \mathbb{R}$ and a sequence $\hat{u}^*_{t:t+T-1|t} \in \mathcal{U}$ such that

$$J^*_t(u_{t-L:t-1}, y_{t-L:t}) = \max_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} J_t(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}^*_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}). \tag{2.35}$$

Going back to (2.32), we then conclude that

$$J^*_{t+1}(u_{t-L+1:t}, y_{t-L+1:t+1}) \leqslant$$
$$J_{t+1}(\hat{x}^*_{t-L+1|t+1}, u_{t-L+1:t}, \hat{u}^*_{t+1:t+T-1|t}, \tilde{u}_{t+T}, \hat{d}^*_{t-L+1:t+T|t+1}, y_{t-L+1:t+1}) \tag{2.36}$$

because the minimization in (2.32) with respect to $\hat{u}_{t+1:t+T|t+1} \in \mathcal{U}$ must lead to a value no larger than what would be obtained by setting $\hat{u}_{t+1:t+T-1|t+1} = \hat{u}^*_{t+1:t+T-1|t}$ and $\hat{u}_{t+T|t+1} = \tilde{u}_{t+T}$.

Similarly, we can conclude from (2.35) that

$$J^*_t(u_{t-L:t-1}, y_{t-L:t}) \geqslant J_t(\tilde{x}_{t-L}, u_{t-L:t-1}, \hat{u}^*_{t:t+T-1|t}, \tilde{d}_{t-L}, \hat{d}^*_{t-L+1:t+T-1|t+1}, y_{t-L:t})$$
$$= J_t(\tilde{x}_{t-L}, u_{t-L:t}, \hat{u}^*_{t+1:t+T-1|t}, \tilde{d}_{t-L}, \hat{d}^*_{t-L+1:t+T-1|t+1}, y_{t-L:t}), \tag{2.37}$$

because the maximization in (2.35) with respect to $\hat{x}_{t-L|t}$ and $\hat{d}_{t-L:t+T-1|t}$ must lead to a value no smaller than what would be obtained by setting $\hat{x}_{t-L|t} = \tilde{x}_{t-L}$, $\hat{d}_{t-L|t} = \tilde{d}_{t-L}$

and $\hat{d}_{t-L+1:t+T-1|t} = \hat{d}^*_{t-L+1:t+T-1|t+1}$. The last equality in (2.37) is obtained by applying the control law (2.24).

Combining (2.36), (2.37), and (2.34) leads to

$$
\begin{aligned}
J^*_{t+1}&(u_{t-L+1:t}, y_{t-L+1:t+1}) - J^*_t(u_{t-L:t-1}, y_{t-L:t}) \\
&\leqslant J_{t+1}\big(f_{t-L}(\tilde{x}_{t-L}, u_{t-L}, \tilde{d}_{t-L}), u_{t-L+1:t}, \hat{u}^*_{t+1:t+T-1|t}, \tilde{u}_{t+T}, \hat{d}^*_{t-L+1:t+T|t+1}, y_{t-L+1:t+1}\big) \\
&\qquad - J_t(\tilde{x}_{t-L}, u_{t-L:t}, \hat{u}^*_{t+1:t+T-1|t}, \tilde{d}_{t-L}, \hat{d}^*_{t-L+1:t+T-1|t+1}, y_{t-L:t}). \quad (2.38)
\end{aligned}
$$

A crucial observation behind this inequality is that *both* terms $J_{t+1}(\cdot)$ and $J_t(\cdot)$ in the right-hand side of (2.38) are computed along a trajectory initialized at time $t-L$ with the *same initial state* $\tilde{x}_{t-L}$ and share the *same control input* sequence $u_{t-L:t}, \hat{u}^*_{t+1:t+T-1|t}$ and the *same disturbance input* sequence $\tilde{d}_{t-L}, \hat{d}^*_{t-L+1:t+T-1|t+1}$. We shall denote this common state trajectory by $\tilde{x}_s$, $s \in \{t - L, \dots, t + T\}$, and the shared control and disturbance sequences by

$$
\tilde{d}_s := \hat{d}^*_{s|t+1}, \quad \forall s \in \{t - L + 1, \dots, t + T - 1\},
$$

$$
\tilde{u}_s := \begin{cases} u_s & s \in \{t - L, \dots, t\} \\ \hat{u}^*_{s|t} & s \in \{t + 1, \dots, t + T - 1\}. \end{cases}
$$

The vectors $\tilde{u}_{t+T}$ and $\tilde{d}_{t-L}$ have been previously defined, but we now also define $\tilde{d}_{t+T} := \hat{d}^*_{t+T|t+1}$, $\tilde{x}_{t+T+1} := f_{t+T}(\tilde{x}_{t+T}, \tilde{u}_{t+T}, \tilde{d}_{t+T})$, and $\tilde{n}_s := y_s - g_s(\tilde{x}_s), s \in \{t - L, \dots, t\}$. All of these definitions enable us to express both terms $J_{t+1}(\cdot)$ and $J_t(\cdot)$ in the right-hand side of (2.38) as follows:

$$
J^*_{t+1}(u_{t-L+1:t}, y_{t-L+1:t+1}) - J^*_t(u_{t-L:t-1}, y_{t-L:t})
$$

$$\leqslant \sum_{s=t+1}^{t+T} c_s(\tilde{x}_s, \tilde{u}_s, \tilde{d}_s) + q_{t+T+1}(\tilde{x}_{t+T+1}) - \sum_{s=t-L+1}^{t+1} \eta_s(\tilde{n}_s) - \sum_{s=t-L+1}^{t+T} \rho_s(\tilde{d}_s)$$

$$- \sum_{s=t}^{t+T-1} c_s(\tilde{x}_s, \tilde{u}_s, \tilde{d}_s) - q_{t+T}(\tilde{x}_{t+T}) + \sum_{s=t-L}^{t} \eta_s(\tilde{n}_s) + \sum_{s=t-L}^{t+T-1} \rho_s(\tilde{d}_s)$$

$$= c_{t+T}(\tilde{x}_{t+T}, \tilde{u}_{t+T}, \tilde{d}_{t+T}) + q_{t+T+1}(\tilde{x}_{t+T+1}) - q_{t+T}(\tilde{x}_{t+T}) - \rho_{t+T}(\tilde{d}_{t+T})$$

$$+ \eta_{t-L}(\tilde{n}_{t-L}) + \rho_{t-L}(\tilde{d}_{t-L}) - c_t(\tilde{x}_t, \tilde{u}_t, \tilde{d}_t) - \eta_{t+1}(\tilde{n}_{t+1}).$$

Equation (2.31) follows from this, (2.33), and the fact that $c_t(\cdot)$ and $\eta_{t+1}(\cdot)$ are both non-negative. ■

With most of the hard work done, we are now ready to prove the main result of this section.

Proof of Theorem 2. Using (2.25a) in Assumption 2, we conclude that

$$J_t^*(u_{t-L:t-1}, y_{t-L:t}) = \max_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} J_t(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}^*, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t})$$

$$\geqslant J_t(x_{t-L}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}^*, d_{t-L:t}, 0_{t+1:t+T-1}, y_{t-L:t})$$

$$= J_t(x_{t-L}, u_{t-L:t}, \hat{u}_{t+1:t+T-1|t}^*, d_{t-L:t}, 0_{t+1:t+T-1}, y_{t-L:t}).$$

The first inequality is a consequence of the fact that the maximum must lead to a value no smaller than what would have been obtained by setting $\hat{x}_{t-L|t}$ equal to the true state x_{t-L}, setting $\hat{d}_{t-L:t}$ equal to the true (past) disturbances $d_{t-L:t}$ and setting $\hat{d}_{t+1:t+T-1}$ equal to zero. The final equality is obtained simply from the use of the control law (2.24).

To proceed, we replace $J_t(\cdot)$ by its definition in (2.22), while dropping all "future" positive

terms in $c_s(\cdot)$, $s > t$ and $q_{t+T}(\cdot)$. This leads to

$$J_t^*(u_{t-L:t-1}, y_{t-L:t}) \geqslant c_t(x_t, u_t, d_t) - \sum_{s=t-L}^{t} \eta_s(n_s) - \sum_{s=t-L}^{t} \rho_s(d_s). \qquad (2.39)$$

Note that the future controls $\hat{u}_{t+1:t+T-1|t}^*$ disappeared because we dropped all the (positive) terms involving the value of the state past time t, and the summation over future disturbances also disappeared since we set all the future $\hat{d}_{t+1:t+T-1}$ to zero.

Adding both sides of (2.31) in Lemma 2 from time L to time $t-1$, leads to

$$J_t^*(u_{t-L:t-1}, y_{t-L:t}) \leqslant J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \rho_s(\tilde{d}_s) + \sum_{s=0}^{t-L-1} \eta_s(\tilde{n}_s), \quad \forall t \in \mathbb{Z}_{\geqslant L}. \quad (2.40)$$

The bound in (2.30) follows directly from (2.39) and (2.40). ∎

Since (2.11) and (5.13) provide nearly identical bounds, the discussion presented after Theorem 1 regarding state boundedness and reference tracking applies also to the finite horizon case with a few minor modifications. In particular, we now discuss the implications of Theorem 2 in terms of establishing bounds on the state of the closed-loop system, practical stability, and the ability of the closed-loop to asymptotically track desired trajectories.

State boundedness and practical stability

When we select penalty functions in the criterion (2.22), for which there exists a class $\mathcal{K}_\infty$ function $\alpha(\cdot)$ and class $\mathcal{K}$ functions $\beta(\cdot), \delta(\cdot)$ such that

$$c_t(x, u, d) \geqslant \alpha(\|x\|), \quad \eta_t(n) \leqslant \beta(\|n\|), \quad \rho_t(d) \leqslant \delta(\|d\|),$$

$$\forall x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, d \in \mathbb{R}^{n_d}, n \in \mathbb{R}^{n_n},$$

we conclude from (5.13) that, along trajectories of the closed-loop system, the following inequality holds for all $t \in \mathbb{Z}_{\geqslant L}$:

$$\alpha(\|x_t\|) \leqslant J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \beta(\|\tilde{n}_s\|) + \sum_{s=0}^{t-L-1} \delta(\|\tilde{d}_s\|) + \sum_{s=t-L}^{t} \beta(\|n_s\|) + \sum_{s=t-L}^{t} \delta(\|d_s\|).$$

(2.41)

Formula (2.41) provides a bound on the state when the future noise and disturbance signals are "vanishing," in the sense that

$$\sum_{s=t-L}^{\infty} \beta(\|n_s\|) < \infty, \qquad\qquad \sum_{s=t-L}^{\infty} \delta(\|d_s\|) < \infty.$$

Theorem 2 also provides bounds on the state for non-vanishing noise and disturbances when we use exponentially time-weighted functions $c_t(\cdot)$, $\eta_t(\cdot)$, and $\rho_t(\cdot)$ that satisfy

$$c_t(x, u, d) \geqslant \lambda^{-t}\alpha(\|x\|), \tag{2.42a}$$

$$\eta_t(n) \leqslant \lambda^{-t}\beta(\|n\|), \tag{2.42b}$$

$$\rho_t(d) \leqslant \lambda^{-t}\delta(\|d\|), \tag{2.42c}$$

for all $x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, d \in \mathbb{R}^{n_d}, n \in \mathbb{R}^{n_n}$ and some $\lambda \in (0, 1)$. In this case, we conclude from (5.13) that for all $t \in \mathbb{Z}_{\geqslant L}$,

$$\alpha(\|x_t\|) \leqslant \lambda^t J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \lambda^{t-s}\beta(\|\tilde{n}_s\|) + \sum_{s=0}^{t-L-1} \lambda^{t-s}\delta(\|\tilde{d}_s\|)$$
$$+ \sum_{s=t-L}^{t} \lambda^{t-s}\beta(\|n_s\|) + \sum_{s=t-L}^{t} \lambda^{t-s}\delta(\|d_s\|). \quad (2.43)$$

Therefore, x_t remains bounded because $n_s \in \mathcal{N}$, $\tilde{n}_s \in \mathcal{N}_{\text{pre}}$, $d_s \in \mathcal{D}$, $\tilde{d}_s \in \mathcal{D}$, and the three sets $\mathcal{N}$, $\mathcal{N}_{\text{pre}}$, and $\mathcal{D}$ are bounded. More specifically, if the noise and disturbances are

uniformly bounded such that, for all $s \geqslant 0$,

$$\beta(\|\tilde{n}_s\|) \leqslant \tilde{a}, \quad \beta(\|n_s\|) \leqslant a, \quad \delta(\|\tilde{d}_s\|) \leqslant \tilde{b}, \quad \delta(\|d_s\|) \leqslant b,$$

where $\tilde{a}$, a, $\tilde{b}$, and b are finite scalars, then an analytical upper bound can be computed for $\alpha(\|x_t\|)$, using the formula for geometric series, and is given by

$$\alpha(\|x_t\|) \leqslant \lambda^t J_L^*(u_{0:L-1}, y_{0:L}) + (\tilde{a} + \tilde{b})\Big(\frac{\lambda^{L+1} - \lambda^{t+1}}{1 - \lambda}\Big) + (a + b)\Big(\frac{1 - \lambda^{L+1}}{1 - \lambda}\Big). \qquad (2.44)$$

Moreover, the terms in the right-hand-side of (2.44) that depend on $\tilde{n}$ and $\tilde{d}$ can be made arbitrarily small by increasing L. The first term in the right-hand-side of (2.44) may initially be large as L is increased, but it exponentially decays to a small value as $t \to \infty$. Finally, if the true noise and disturbances vanish asymptotically, then the terms in the right-hand-side of (2.43) that depend on n_s and d_s converge to zero as $t \to \infty$. Therefore, $\|x_t\|$ converges to a small value as $t \to \infty$ when the true noise and disturbances vanish asymptotically and the backwards horizon is chosen arbitrarily large. We have proved the following:

Corollary 2 *Suppose that Assumptions 2, 3, and 4 hold and also that (2.42) holds for a class $\mathcal{K}_\infty$ function $\alpha(\cdot)$, class $\mathcal{K}$ functions $\beta(\cdot), \delta(\cdot)$, and $\lambda \in (0,1)$. Then, for every initial condition x_0, uniformly bounded measurement noise sequence $n_{0:t}$, and uniformly bounded disturbance sequence $d_{0:t}$, the state x_t remains uniformly bounded along the trajectories of the process (2.1) with control (2.24) defined by the finite-horizon optimization (2.23). Moreover, when d_t and n_t converge to zero as $t \to \infty$, the backwards horizon L can be chosen sufficiently large to ensure that the state x_t converges to an arbitrarily small value as $t \to \infty$.* $\qquad \square$

Remark 6 (Time-weighted criteria) *The exponentially time-weighted functions (2.42)*

typically arise from a criterion of the form

$$\sum_{s=t}^{t+T-1} \lambda^{-s} c(x_s, u_s, d_s) + q_{t+T}(x_{t+T}) - \sum_{s=t-L}^{t} \lambda^{-s} \eta(n_s) - \sum_{s=t-L}^{t+T-1} \lambda^{-s} \rho(d_s)$$

that weight the future more than the past. In this case, (2.42) holds for functions $\alpha(\cdot)$, $\beta(\cdot)$, and $\delta(\cdot)$ such that $c(x, u, d) \geqslant \alpha(\|x\|)$, $\eta(n) \leqslant \beta(\|n\|)$, and $\rho(d) \leqslant \delta(\|d\|)$, $\forall x, u, d, n$.

$\square$

Reference tracking

When the control objective is for the state x_t to follow a given trajectory z_t, the optimization criterion can be selected of the form

$$\sum_{s=t}^{t+T-1} \lambda^{-s} c(x_s - z_s, u_s, d_s) + q_{t+T}(x_{t+T} - z_{t+T}) - \sum_{s=t-L}^{t} \lambda^{-s} \eta(n_s) - \sum_{s=t-L}^{t+T-1} \lambda^{-s} \rho(d_s),$$

with $c(x - z, u, d) \geqslant \alpha(\|x - z\|)$, $\forall x, u, d$ for some class $\mathcal{K}_\infty$ function α and $\lambda \in (0,1)$. In this case, we conclude from (5.13) that, for all $t \in \mathbb{Z}_{\geqslant L}$,

$$\alpha(\|x_t - z_t\|) \leqslant \lambda^t J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \lambda^{t-s} \eta(\tilde{n}_s) + \sum_{s=0}^{t-L-1} \lambda^{t-s} \rho(\tilde{d}_s)$$
$$+ \sum_{s=t-L}^{t} \lambda^{t-s} \eta(n_s) + \sum_{s=t-L}^{t} \lambda^{t-s} \rho(d_s),$$

which allows us to conclude, from Corollary 2, that x_t converges arbitrarily close to z_t as $t \to \infty$ when both n_t and d_t are vanishing sequences and L is chosen sufficiently large. Similarly, if these noise and disturbance sequences are "ultimately small", the tracking error $x_t - z_t$ will converge to a small value.

2.5   Numerical Examples

In this section we discuss several numerical examples using the problem framework introduced in Section 2.2 and the finite horizon estimation and control approach described in Section 2.4. Solutions are found via numerical simulation using the interior-point methods described later in Chapter 4.

Example 1 (Flexible beam) *Consider a single-link flexible beam as depicted in Figure 2.1 similar to the one described in [98], where the control objective is to regulate the mass on the tip of the beam to a desired reference trajectory. The control input is the applied torque at the base, and the outputs are the tip's position $p = l\theta(t) + w(x,t)$ [m], the angle at the base $\theta(t)$ [rad], the angular velocity of the base $\dot{\theta}(t)$ [rad/s], and a strain gauge measurement $w''(x,t)$ [m^{-1}] collected around the middle of the beam at x, respectively.*

$l = 1.13$	link's length [m]
$\rho = .60708$	link's mass per unit length [$\mathrm{Kg\,m^{-1}}$]
$EI = 1.1684$	product of Young's modulus E by the cross-sectional inertia I [$\mathrm{N\,m^2}$]
$I_{\text{base}} = .023$	base's inertia [$\mathrm{Kg\,m^2}$]
$m_{\text{tip}} \geqslant 0$	tip's mass [Kg]
$x \in [0, l]$	strain gauge position [m]
$w(x,t)$	displacement with respect to rigid motion [m]
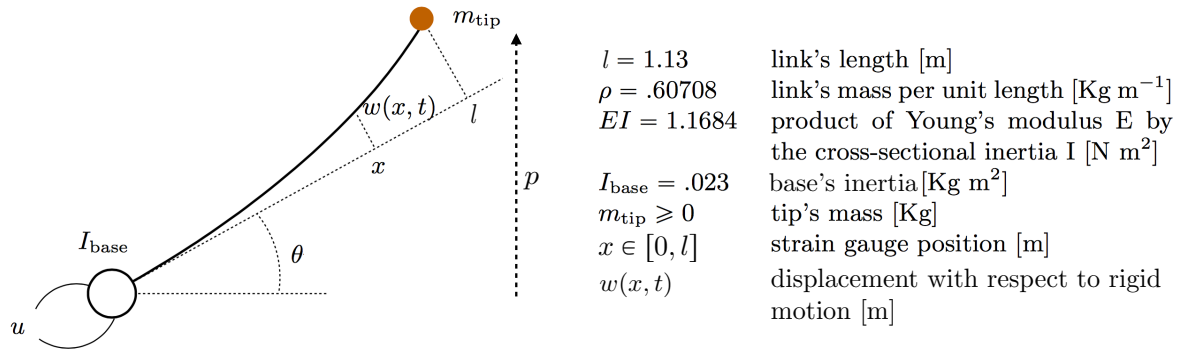
Figure 2.1: Diagram of the Flexible Beam considered in Example 1.

The system is modeled by the following partial differential equation

$$\ddot{w}(x,t) + \frac{EI}{\rho}w''''(x,t) = -x\ddot{\theta}(t), \qquad \forall t \geqslant 0, \qquad x \in [0, l]$$

with boundary conditions

$$u(t) = I_{base}\ddot{\theta}(t) - EIw''(0,t),$$

$$w(0,t) = w'(0,t) = 0,$$

$$w''(l,t) = w'''(l,t) + \frac{m_{tip}}{\rho}w'''(l,t) = 0, \qquad \forall t \geqslant 0.$$

An approximate linearized discrete-time state-space model of the dynamics, with a sampling time $T_s := 1$ second, is given by $x_{t+1} = Ax_t + B(u_t + d_t)$, $y_t = Cx_t + n_t$, where d_t is a disturbance, n_t is measurement noise, and the system matrices are given by

$$A = \begin{bmatrix} 1.0 & 1.016 & -0.676 & -1.084 & 1.0 & 0.585 & 0.233 & 0.032 \\ 0 & -0.665 & 1.241 & 1.783 & 0 & 0.042 & -0.288 & -0.023 \\ 0 & 0.009 & -0.439 & 0.143 & 0 & -0.002 & -0.012 & 0.007 \\ 0 & 0.001 & 0.014 & 0.308 & 0 & -0.000 & 0.001 & 0.001 \\ 0 & 1.264 & -37.070 & 10.581 & 1.0 & 1.016 & -0.676 & -1.084 \\ 0 & -2.109 & 59.920 & -16.883 & 0 & -0.665 & 1.241 & 1.783 \\ 0 & 0.413 & 9.156 & -3.695 & 0 & 0.009 & -0.439 & 0.143 \\ 0 & -0.012 & -0.371 & -3.929 & 0 & 0.001 & 0.014 & 0.308 \end{bmatrix}, \qquad (2.45)$$

$$B = \begin{bmatrix} 0.800 & -0.797 & 0.003 & 0.001 & 1.327 & -1.163 & 0.197 & -0.006 \end{bmatrix}', \qquad (2.46)$$

$$C = \begin{bmatrix} 1.13 & 0.7225 & -0.2028 & 0.1220 & 0 & 0 & 0 & 0 \\ 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ 0 & 0.9282 & -12.001 & -35.294 & 0 & 0 & 0 & 0 \end{bmatrix}. \qquad (2.47)$$

This matrix A has a double eigenvalue at 1 with a single independent eigenvector. Therefore this is an unstable system.

The optimal control input is found by solving the following optimization problem

$$\min_{u_{t:t+T-1|t}\in\mathcal{U}} \quad \max_{x_{t-L}\in\mathcal{X},d_{t-L:t+T-1|t}\in\mathcal{D}}$$

$$\|p_{t:t+T} - r_{t:t+T}\|^2 + \lambda_u\|u_{t:t+T-1}\|^2 - \lambda_d\|d_{t-L:t+T-1}\|^2 - \lambda_n\|n_{t-L:t}\|^2,$$

where $\|\cdot\|$ is the Euclidean norm, $\mathcal{U} := \{u_t \in \mathbb{R}| - u_{max} \leqslant u_t \leqslant u_{max}\}$, $\mathcal{X} := \mathbb{R}^8$, and $\mathcal{D} := \{d_t \in \mathbb{R}| - d_{max} \leqslant d_t \leqslant d_{max}\}$.
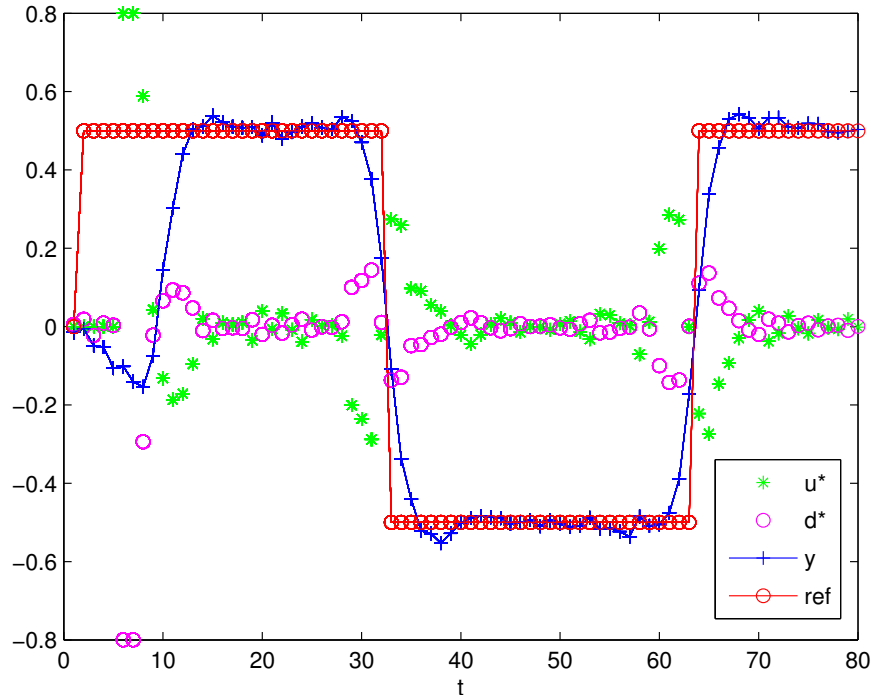
Figure 2.2: Simulation results of Flexible Beam example. The reference [m] is in red, the measured position [m] in blue, the control sequence [Nm] in green, and the disturbance sequence [Nm] in magenta.

The results depicted in Figure 2.2 show the response of the closed loop system under the control law (2.24) when our goal is to regulate the position of the mass at the tip of the beam p to a desired reference $r(t) := \alpha\,\mathrm{sgn}(\sin(\omega t))$ with $\alpha = 0.5$ and $\omega = 0.1$. The other parameters in the optimization have values $\lambda_u = 1$, $\lambda_d = 2$, $\lambda_n = 100$, $L = 5$, $T = 5$, $u_{max} = 0.8$, $d_{max} = 0.8$. The state of the system starts close to zero and evolves

Table 2.1: Numerical Performance of the Flexible Beam Example 1.

# of optimization variables	103
# of inequality constraints	30
# of equality constraints	80
Mean time to compute	3.2 ms
Max. time to compute	8.3 ms
Min. time to compute	2.8 ms

with zero control input and small random disturbance input until time $t = 6$, at which time the optimal control input (2.24) started to be applied along with the optimal worst-case disturbance $d_{t|t}^*$ obtained from the min-max optimization. It can be seen that the optimal control input and disturbance initially hit their respective constraints. The noise process n_t was selected to be a zero-mean Gaussian independent and identically distributed random process with standard deviation of 0.02.

This simulation was performed on a laptop with an Intel® Core^{TM} i7 Processor and used Algorithm 4.4.2 (given in Chapter 4) implemented in C code to compute solutions at each time step. The resulting numerical performance is given in Table 2.1. Even with over 100 optimization variables and 120 combined inequality and equality constraints, the problem can be solved very efficiently using the numerical algorithms given later in Chapter 4, on average in 3.2 ms.   △

Example 2 (Nonlinear Pursuit-Evasion) *In this example a two-player pursuit-evasion game is considered where the pursuer is modeled as a nonholonomic unicycle-type vehicle, and the evader is modeled as a single-integrator. The measured output is the positions of the pursuer and evader. The orientation of the pursuer is not measured. The models can*

be written in discrete-time as follows:

$$x_{t+1}^1 = x_t^1 + v\cos(\theta_t),$$

$$Pursuer: \qquad x_{t+1}^2 = x_t^2 + v\sin(\theta_t), \qquad (2.48\text{a})$$

$$\theta_{t+1} = \theta_t + u_t,$$

$$z_{t+1}^1 = z_t^1 + d_t^1,$$

$$Evader: \qquad \qquad \qquad \qquad \qquad \qquad (2.48\text{b})$$

$$z_{t+1}^2 = z_t^2 + d_t^2,$$

$$Output: \qquad y_t = [x_t' \quad z_t']' + n_t. \qquad (2.48\text{c})$$

The positions of the pursuer and the evader at time t are denoted by $x_t = [x_t^1 \ x_t^2]' \in \mathbb{R}^2$ and $z_t = [z_t^1 \ z_t^2]' \in \mathbb{R}^2$, respectively, and the orientation of the pursuer at time t is denoted by θ_t. The inputs for the pursuer and evader at time t are denoted by $u_t \in \mathbb{R}$ and $d_t = [d_t^1 \ d_t^2]' \in \mathbb{R}^2$, respectively, and are constrained to belong to the sets $\mathcal{U} := \{u_t \in \mathbb{R} : |u_t| \leqslant u_{max}\}$ and $\mathcal{D} := \{d_t \in \mathbb{R}^2 : \|d_t\|_\infty \leqslant d_{max}\}$. The measurement noise is denoted by $n_t \in \mathbb{R}^4$. In the context of the problem described in Section 2.2, we regard the input for the pursuer as the control signal and the input to the evader as a disturbance.

The evader's goal is to make the distance between its position z_t and the position of the pursuer x_t as large as possible, so the evader wants to maximize the value of $\|z_t - x_t\|_2$. The pursuer's goal is to do the opposite, namely, minimize the value of $\|z_t - x_t\|_2$. The pursuer and evader try to achieve these goals by choosing appropriate values for u_t and d_t, respectively. This motivates considering a cost function of the form

$$J_t(\cdot) = \sum_{s=t}^{t+T-1} \|z_s - x_s\|_2^2 + \lambda_u \sum_{s=t}^{t+T-1} \|u_s\|_2^2 - \lambda_n \sum_{s=t-L}^{t} \|n_s\|_2^2 - \lambda_d \sum_{s=t-L}^{t+T-1} \|d_s\|_2^2, \qquad (2.49)$$

where λ_u, λ_n, and λ_d are positive weighting constants.

Figures 2.3 and 2.4 show simulation results from solving the optimization

$$\min_{u_t \in \mathcal{U}} \quad \max_{x_{t-L} \in \mathcal{X}, d_t \in \mathcal{D}} J_t(\cdot) \qquad (2.50)$$

at each time step t, where $J_t(\cdot)$ is the cost function given in (2.49), and the optimization is solved using Algorithm 4.3.1 given in Chapter 4. For this simulation, the parameters for the model (2.48) and the cost function (2.49) are chosen as $L = 8$, $T = 12$, $v = 0.1$, $u_{max} = 0.5$, $d_{max} = 0.06$, $\lambda_u = 8$, $\lambda_d = 100$, and $\lambda_n = 1000$. The output measurements are subjected to normally distributed random noise $n_t \sim \mathcal{N}(0, 0.005^2)$.

Figure 2.3 shows the estimates of the pursuer's and evader's positions computed by solving the optimization (2.50) at every time t. The initial state of the pursuer is $[x_0' \ \theta_0]' = [0 \ 0 \ 0]'$, and the initial state of the evader is $z_0 = [0.5 \ 0.5]'$. The simulation is initialized with zero input for the pursuer (i.e. $u_t = 0$) for the first $L = 8$ time steps after which time the optimization (2.50) is solved at every time step t, and the optimal input u_t^* is applied for the rest of the simulation. The evader applies an input of $d_t = [0.05 \ 0]'$ until time $t = 55$ after which time the optimal computed evader's input d_t^* is applied for every successive time step t. The inputs that are applied are shown in Figure 2.4. We see that several times throughout the simulation the input constraints for both the pursuer and evader are active.

Because the maximum speed of the evader ($d_{max} = 0.06$) is less than the speed of the pursuer ($v = 0.1$), the pursuer is always able to catch up to the evader, but the evader takes advantage of its more agile (integrator) dynamics by making sharp turns and forcing the unicycle-type pursuer to make loops at its maximum turning rate.

This simulation was performed on a laptop with an Intel®Core^{TM} i7 Processor and used Algorithm 4.4.2 (given in Chapter 4) implemented in C code to compute solutions
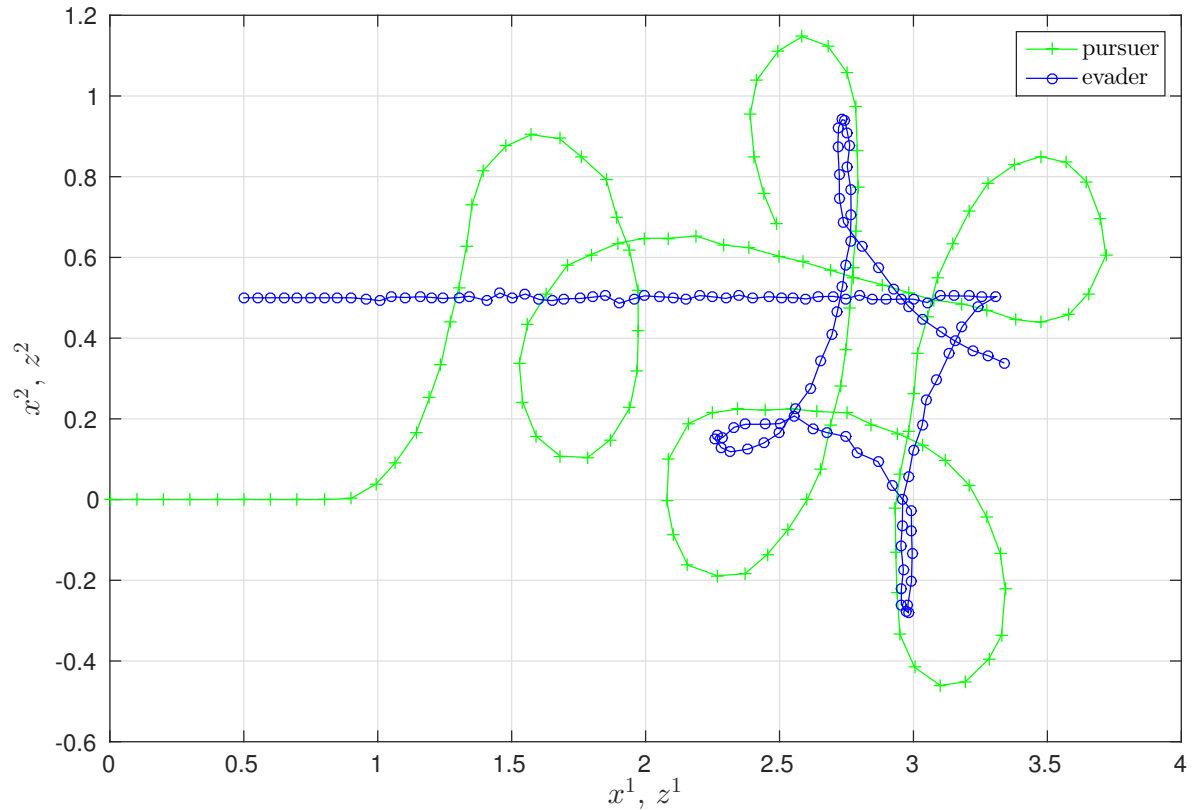
Figure 2.3: Trajectories of the pursuer and evader from Example 2.

Table 2.2: Numerical Performance of the Pursuit-Evasion Example 2.

# of optimization variables	157
# of inequality constraints	104
# of equality constraints	100
Mean time to compute	0.54 ms
Max. time to compute	2.3 ms
Min. time to compute	0.45 ms

at each time step. The resulting numerical performance is given in Table 2.2. The optimization involved 157 optimization variables, 100 equality constraints, and 104 inequality constraints, and the average time to compute the solution at each time step was 0.54 ms. Therefore, solutions can be computed extremely efficiently even for this nonlinear and nonconvex example. △
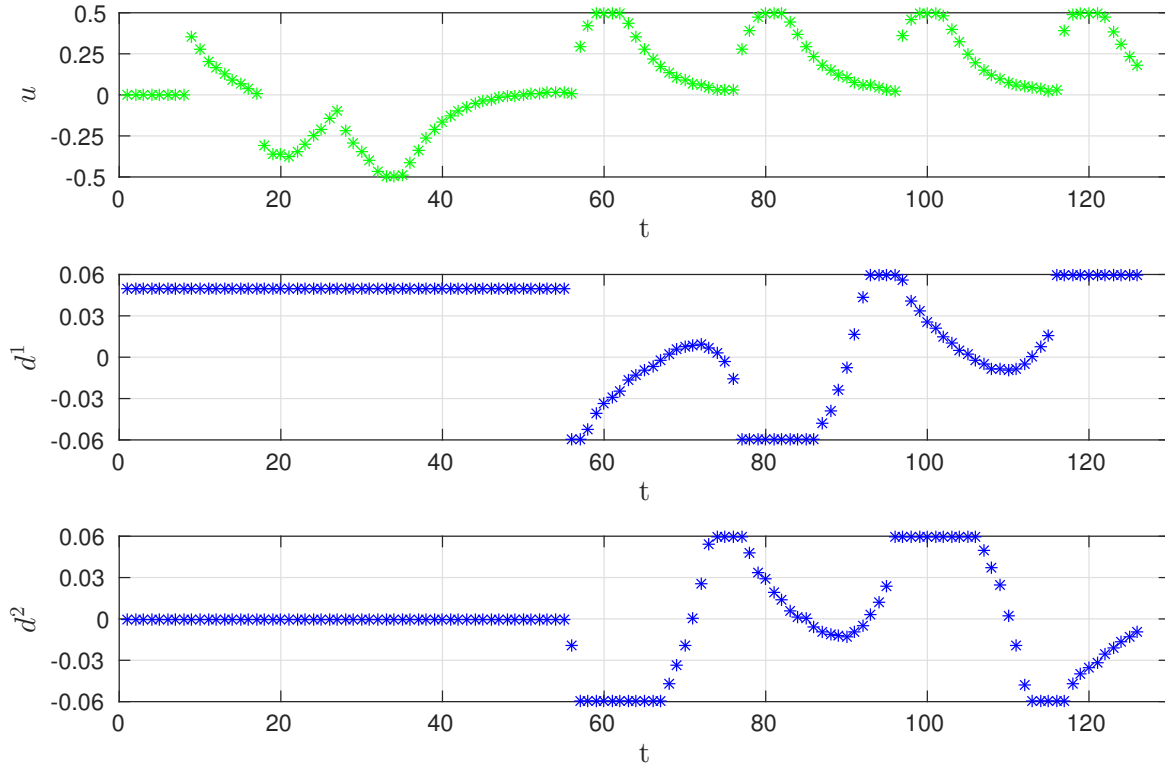
Figure 2.4: Inputs for the pursuer and evader from Example 2.

2.6   Conclusions

We presented an output-feedback approach to nonlinear model predictive involving the simultaneous solution of the corresponding estimation and control problems as a single min-max optimization. Under the assumption that a saddle-point solution exists for the min-max optimization (which presumes appropriate forms of observability and controllability), Theorem 1 ensures bounds on the state of the system and the tracking error for reference tracking problems for the infinite horizon case. Similar results are derived from Theorem 2 for the finite horizon case under the additional assumptions of a form of observability for nonlinear systems and a terminal cost that is an ISS-control Lyapunov function with respect to the disturbance input.

We also presented two numerical examples of estimation and control problems that

can be solved using this combined MPC/MHE approach. The examples included constrained, high-dimensional, nonlinear, and nonconvex problems that were easily formulated as finite horizon min-max optimization problems as in (2.23). The resulting min-max optimization problems were solved very efficiently using the numerical methods described later in Chapter 4.

Chapter 3

Existence of Saddle-Point Equilibria

Some of the content in this chapter comes from [24]:

In Chapter 2, we showed that under the main assumption that a saddle-point solution exists for the min-max optimization problem, in addition to standard observability and controllability assumptions, practical stability of the combined MPC/MHE approach can be established in the presence of noise and disturbances. In this chapter, we derive sufficient conditions for the existence of a saddle-point solution to this min-max optimization problem. For the specialized linear-quadratic case, we show that a saddle-point solution exists if the system is observable and weights in the cost function are chosen appropriately. A numerical example is given to illustrate the effectiveness of this combined control and estimation approach.

3.1 Introduction

Classical MPC has been a prominent control technique in academia and industrial applications for decades because of its ability to handle complex multivariable systems and hard constraints. MPC, which is classically formulated with state-feedback and involves the repeated solution of an open-loop optimal control problem online in order to find a sequence of future control inputs, has a well-developed theory as evidenced by [79, 76, 95], and has been shown to be effective in practice [84]. However, with more efficient methods and continued theoretical work, more recent advances in MPC include the incorporation of disturbances, uncertainties, faster dynamics, distributed systems, and output-feedback.

Related to these advances is the recent work combining nonlinear output-feedback MPC with MHE into a single min-max optimization problem [22, 25] as described in Chapter 2. This combined approach simultaneously solves an MHE problem, which involves the repeated solution of a similar optimization problem over a finite-horizon of past measurements in order to find an estimate of the current state [90, 3], and an MPC problem. In order to be robust to "worst-case" disturbances and noise, this approach involves the solution of a min-max optimization where an objective function is minimized with respect to control input variables and maximized with respect to disturbance and noise variables, similar to game-theoretic approaches to MPC considered in [19, 56].

The motivation for the combined MPC/MHE approach is proving joint stability of the combined estimation and control problems, and the results in Chapter 2 guarantee boundedness of the state, bounds on the tracking error for trajectory tracking problems, and practical stability in the presence of noise and disturbances. Besides standard assumptions regarding observability and controllability of the nonlinear process, the main assumption required for these results to hold is that there exists a saddle-point solution

to the min-max optimization problem at every time step.

The analysis of the min-max problem that appears in the forward horizon of the combined MPC/MHE approach is closely related to the analysis of two-player zero-sum dynamic games as in [6] and to the dynamic game approach to H^∞ optimal control as in [7]. In these analyses the control is designed to guard against the worst-case unknown disturbances and model uncertainties, and in both of these references, saddle-point equilibria and conditions under which they exist are analyzed. The problem proposed in [22, 25] differs, however, in that a backwards finite horizon is also considered in order to incorporate the simultaneous solution of an MHE problem, which also allows the control to be robust to worst-case estimates of the initial state.

In this chapter we derive conditions under which a saddle-point solution exists for the combined MPC/MHE min-max optimization problem proposed in [22] and specialize those results for discrete linear time-invariant (DLTI) systems and quadratic cost functions. We show that in the linear-quadratic case, if the system is observable, simply choosing appropriate weights in the cost function is enough to ensure that a saddle-point solution exists. A numerical example discussed at the end of the chapter shows that, even for unconstrained linear-quadratic problems, better regulation performance may be achieved using this MPC/MHE approach with shorter finite horizons.

The chapter is organized as follows. In Section 3.2, we formulate the control problem and discuss the main stability assumption regarding the existence of a saddle-point. In Section 3.3, we describe a method that can be used to compute a saddle-point solution and give conditions under which this method succeeds. A numerical example is presented in Section 3.4. Finally, we provide some conclusions in Section 3.5.

3.2   Problem Formulation

As in the previous chapter, we consider the control of a time-varying discrete-time process of the form

$$x_{t+1} = f_t(x_t, u_t, d_t), \qquad y_t = g_t(x_t) + n_t \tag{3.1}$$

$\forall t \in \mathbb{Z}_{\geqslant 0}$, with *state* x_t taking values in a set $\mathcal{X} \subset \mathbb{R}^{n_x}$. The inputs to this system are the *control input* u_t that must be restricted to a set $\mathcal{U} \subset \mathbb{R}^{n_u}$, the *unmeasured disturbance* d_t that is known to belong to a set $\mathcal{D} \subset \mathbb{R}^{n_d}$, and the *measurement noise* $n_t \in \mathbb{R}^{n_n}$. The signal $y_t \in \mathcal{Y} \subset \mathbb{R}^{n_y}$ denotes the *measured output* that is available for feedback. The *control objective* is to select the control signal $u_t \in \mathcal{U}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$, so as to minimize a finite-horizon criterion of the form[1]

$$J_t(x_{t-L}, u_{t-L:t+T-1}, d_{t-L:t+T-1}, y_{t-L:t}) :=$$
$$\sum_{k=t}^{t+T-1} c_k(x_k, u_k) + q_{t+T}(x_{t+T}) - \sum_{k=t-L}^{t} \eta_k(n_k) - \sum_{k=t-L}^{t+T-1} \rho_k(d_k) \tag{3.2}$$

for worst-case values of the unmeasured disturbance $d_t \in \mathcal{D}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$ and the measurement noise $n_t \in \mathbb{R}^{n_n}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$. The functions $c_k(\cdot)$, $\eta_k(\cdot)$, and $\rho_k(\cdot)$ in (3.2) are all assumed to take non-negative values. One can view the terms $\rho_t(\cdot)$ and $\eta_t(\cdot)$ as measures of the likelihood of specific values for d_t and n_t. Then, the negative signs in front of $\rho_t(\cdot)$ and $\eta_t(\cdot)$ penalize the maximizer for using low likelihood values for the disturbances and noise (low likelihood meaning very large values for $\rho_t(\cdot)$ and $\eta_t(\cdot)$).

The optimization criterion includes $T \in \mathbb{Z}_{\geqslant 1}$ terms of the running cost $c_t(x_t, u_t)$, which recede as the current time t advances, $L+1 \in \mathbb{Z}_{>1}$ terms of the measurement cost $\eta_t(n_t)$,

---

[1]Given a discrete-time signal $z : \mathbb{Z}_{\geqslant 0} \to \mathbb{R}^n$, and two times $t_0, t \in \mathbb{Z}_{\geqslant} 0$ with $t_0 \leqslant t$, we denote by $z_{t_0:t}$ the sequence $\{z_{t_0}, z_{t_0+1}, \dots, z_t\}$.

and $L + T \in \mathbb{Z}_{>1}$ terms of the cost on the disturbances $\rho_t(d_t)$. We also include a terminal cost $q_{t+T}(x_{t+T})$ to penalize the "final" state at time $t + T$.

Just as in a two-player zero-sum dynamic game, player 1 (the controller) desires to minimize this criterion while player 2 (the noise and disturbance) would like to maximize it. This leads to a control input that is designed for the worst-case disturbance input, measurement noise, and initial state. This motivates the following finite-dimensional optimization

$$\min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \quad \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}}} J_t(\hat{x}_{t-L}, u_{t-L:t-1}, \hat{u}_{t:t+T-1}, \hat{d}_{t-L:t+T-1}, y_{t-L:t}). \tag{3.3}$$

The measurement noise variables $n_{t-L:t}$ do not explicitly show up in (3.3) because they are not independent optimization variables as they are uniquely defined by the remaining optimization variables and the output equation (3.1). In this formulation, we use a control law of the form

$$u_t = \hat{u}_t^*, \quad \forall t \geqslant 0, \tag{3.4}$$

where $\hat{u}_t^*$ denotes the first element of the sequence $\hat{u}_{t:t+T-1}^*$ computed at each time t that minimizes (3.3).

For the implementation of the control law (3.4), the outer minimizations in (3.3) must lead to finite values for the optima that are achieved at specific sequences $\hat{u}_{t:t+T-1}^* \in \mathcal{U}$, $t \in \mathbb{Z}_{\geqslant 0}$. However, for the stability results given in [22, 25], we actually ask for the existence of a saddle-point solution to the min-max optimization in (3.3) as follows:

Assumption 5 (Saddle-point [Assumption 2, Chapter 2) *The min-max optimization (3.3) with cost given as in (3.2) always has a saddle-point solution for which the min and max commute. Specifically, for every time $t \in \mathbb{Z}_{\geqslant 0}$, past control input sequence*

$u_{t-L:t-1} \in \mathcal{U}$, and past measured output sequence $y_{t-L:t} \in \mathcal{Y}$, there exists a finite scalar $J_t^* \in \mathbb{R}$, an initial condition $\hat{x}_{t-L}^* \in \mathcal{X}$, and sequences $\hat{u}_{t:t+T-1}^* \in \mathcal{U}$, $\hat{d}_{t-L:t+T-1}^* \in \mathcal{D}$ such that

$$J_t^* = J_t(\hat{x}_{t-L}^*, u_{t-L:t-1}, \hat{u}_{t:t+T-1}^*, \hat{d}_{t-L:t+T-1}^*, y_{t-L:t})$$

$$= \max_{\substack{\hat{x}_{t-L}\in\mathcal{X}, \\ \hat{d}_{t-L:t+T-1}\in\mathcal{D}}} J_t(\hat{x}_{t-L}, u_{t-L:t-1}, \hat{u}_{t:t+T-1}^*, \hat{d}_{t-L:t+T-1}, y_{t-L:t}) \tag{3.5a}$$

$$= \min_{\hat{u}_{t:t+T-1}\in\mathcal{U}} J_t(\hat{x}_{t-L}^*, u_{t-L:t-1}, \hat{u}_{t:t+T-1}, \hat{d}_{t-L:t+T-1}^*, y_{t-L:t}). \tag{3.5b}$$

$\square$

In general, J_t^* depends on the past control and outputs, so we can alternatively write $J_t^*(u_{t-L:t-1}, y_{t-L:t})$.

In the next section we derive conditions under which a saddle-point solution exists for the general nonlinear case and then specialize those results for DLTI systems and quadratic cost functions.

3.3   Main Results

Before presenting the main results, for convenience we define the following sets of time sequences for the forward and backward horizons, respectively, $\mathcal{T} := \{t, t+1, ..., t+T-1\}$ and $\mathcal{L} := \{t - L, t - L + 1, ..., t - 1\}$ and use them in the sequel.

3.3.1   Nonlinear systems

Theorem 3 *(Existence of saddle-point)*

Suppose there exist recursively computed functions $V_k(\cdot)$, for all $k \in \mathcal{T}$, and $V_j(\cdot)$, for

all $j \in \mathcal{L}$, such that for all $y_{t-L:t} \in \mathcal{Y}$, and $u_{t-L:t-1} \in \mathcal{U}$,

$$V_{t+T}(x_{t+T}) := q_{t+T}(x_{t+T}), \tag{3.6a}$$

$$V_k(x_k) := \min_{\hat{u}_k \in \mathcal{U}} \max_{\hat{d}_k \in \mathcal{D}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k) + V_{k+1}(x_{k+1}) \right)$$

$$= \max_{\hat{d}_k \in \mathcal{D}} \min_{\hat{u}_k \in \mathcal{U}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k) + V_{k+1}(x_{k+1}) \right), \qquad \forall k \in \mathcal{T} \backslash t, \tag{3.6b}$$

$$V_t(x_t, y_t) := \min_{\hat{u}_t \in \mathcal{U}} \max_{\hat{d}_t \in \mathcal{D}} \left(l_t(x_t, \hat{u}_t, \hat{d}_t, y_t) + V_{t+1}(x_{t+1}) \right)$$

$$= \max_{\hat{d}_t \in \mathcal{D}} \min_{\hat{u}_t \in \mathcal{U}} \left(l_t(x_t, \hat{u}_t, \hat{d}_t, y_t) + V_{t+1}(x_{t+1}) \right), \tag{3.6c}$$

$$V_j(x_j, u_{j:t-1}, y_{j:t}) := \max_{\hat{d}_j \in \mathcal{D}} \left(l_j(x_j, u_j, \hat{d}_j, y_j) + V_{j+1}(x_{j+1}, u_{j+1:t-1}, y_{j+1:t}) \right),$$

$$\forall j \in \mathcal{L} \backslash t - L, \tag{3.6d}$$

$$V_{t-L}(u_{t-L:t-1}, y_{t-L:t}) := \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L} \in \mathcal{D}}} \left(l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}, y_{t-L}) \right.$$

$$\left. + V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t}) \right), \tag{3.6e}$$

where

$$l_k(x_k, \hat{u}_k, \hat{d}_k) := c_k(x_k, \hat{u}_k) - \rho_k(\hat{d}_k), k \in \mathcal{T} \backslash t, \tag{3.7a}$$

$$l_t(x_t, \hat{u}_t, \hat{d}_t, y_t) := c_t(x_t, \hat{u}_t) - \eta_t(n_t) - \rho_t(\hat{d}_t), \tag{3.7b}$$

$$l_j(x_j, u_j, \hat{d}_j, y_j) := -\eta_j(n_j) - \rho_j(\hat{d}_j), \quad j \in \mathcal{L}. \tag{3.7c}$$

Then the solutions $\hat{u}_k^*$, $\hat{d}_k^*$, $\hat{d}_j^*$, and $\hat{x}_{t-L}^*$ defined as follows, for all $k \in \mathcal{T}$ and $j \in \mathcal{L}$, satisfy the saddle-point Assumption 5.

$$\hat{u}_k^* := \arg \min_{\hat{u}_k \in \mathcal{U}} \max_{\hat{d}_k \in \mathcal{D}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k, y_k) + V_{k+1}(x_{k+1}) \right), \tag{3.8a}$$

$$\hat{d}_k^* := \arg \max_{\hat{d}_k \in \mathcal{D}} \min_{\hat{u}_k \in \mathcal{U}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k, y_k) + V_{k+1}(x_{k+1}) \right), \tag{3.8b}$$

$$\hat{d}_j^* := \underset{\hat{d}_j \in \mathcal{D}}{\arg\max} \left(l_j(x_j, u_j, \hat{d}_j, y_j) + V_{j+1}(x_{j+1}, u_{j+1:t-1}, y_{j+1:t}) \right), \tag{3.8c}$$

$$\hat{x}_{t-L}^* := \underset{\hat{x}_{t-L} \in \mathcal{X}}{\arg\max} \left(l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t}) \right). \tag{3.8d}$$

Moreover, the saddle-point value is equal to $J_t^*(u_{t-L:t-1}, y_{t-L:t}) = V_{t-L}(u_{t-L:t-1}, y_{t-L:t}).$

$\square$

Proof. We begin by proving equation (3.5b) in Assumption 5. Let $\hat{u}_k^*$ be defined as in (3.8a), and let $\hat{u}_k$ be another arbitrary control input. To prove optimality, we need to show that the latter trajectory cannot lead to a cost lower than the former. Since $V_k(x_k)$ satisfies (3.6b) and $\hat{u}_k^*$ achieves the minimum in (3.6b), for every $k \in \mathcal{T} \backslash t$,

$$V_k(x_k) = \min_{\hat{u}_k \in \mathcal{U}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k^*) + V_{k+1}(x_{k+1}) \right) = l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) + V_{k+1}(x_{k+1}). \tag{3.9}$$

However, since $\hat{u}_k$ does not necessarily achieve the minimum, we also have that

$$V_k(x_k) = \min_{\hat{u}_k \in \mathcal{U}} \left(l_k(x_k, \hat{u}_k, \hat{d}_k^*) + V_{k+1}(x_{k+1}) \right) \leqslant l_k(x_k, \hat{u}_k, \hat{d}_k^*) + V_{k+1}(x_{k+1}). \tag{3.10}$$

Summing both sides of (3.9) from $k = t + 1$ to $k = t + T - 1$, we conclude that

$$\sum_{k=t+1}^{t+T-1} V_k(x_k) = \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) + \sum_{k=t+1}^{t+T-1} V_{k+1}(x_{k+1})$$

$$\iff \sum_{k=t+1}^{t+T-1} V_k(x_k) - \sum_{k=t+1}^{t+T-1} V_{k+1}(x_{k+1}) = \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*)$$

$$\iff V_{t+1}(x_{t+1}) = \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*).$$

Next, summing both sides of (3.10) from $k = t + 1$ to $k = t + T - 1$, we conclude that

$$\sum_{k=t+1}^{t+T-1} V_k(x_k) \leqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, \hat{d}_k^*) + \sum_{k=t+1}^{t+T-1} V_{k+1}(x_{k+1})$$

$$\iff \sum_{k=t+1}^{t+T-1} V_k(x_k) - \sum_{k=t+1}^{t+T-1} V_{k+1}(x_{k+1}) \leqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, \hat{d}_k^*)$$

$$\iff V_{t+1}(x_{t+1}) \leqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, \hat{d}_k^*),$$

from which we conclude that

$$V_{t+1}(x_{t+1}) = \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) \leqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, d_k^*). \tag{3.11}$$

Similarly, since $V_t(x_t, y_t)$ satisfies (3.6c) and $\hat{u}_t^*$ achieves the minimum in (3.6c), we can conclude that

$$V_t(x_t, y_t) = l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t) + V_{t+1}(x_{t+1}) \leqslant l_t(x_t, \hat{u}_t, \hat{d}_t^*, y_t) + V_{t+1}(x_{t+1}). \tag{3.12}$$

Then from (3.11) and (3.12), we conclude that

$$V_t(x_t, y_t) = l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t) + \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) \leqslant l_t(x_t, \hat{u}_t, \hat{d}_t^*, y_t) + \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, d_k^*).$$

$$\tag{3.13}$$

Next, since $V_j(x_j, u_{j:t-1}, y_{j:t})$ satisfies (3.6d) and $\hat{d}_j^*$ achieves the maximum in (3.6d), we can conclude that

$$V_j(x_j, u_{j:t-1}, y_{j:t}) = l_j(x_j, u_j, \hat{d}_j^*, y_j) + V_{j+1}(x_{j+1}, u_{j:t-1}, y_{j:t}). \tag{3.14}$$

Summing both sides of (3.14) from $j = t - L + 1$ to $j = t - 1$, and using (3.13), we conclude that

$$\sum_{j=t-L+1}^{t-1} V_j(x_j, u_{j:t-1}, y_{j:t}) = \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + \sum_{j=t-L+1}^{t-1} V_{j+1}(x_{j+1}, u_{j:t-1}, y_{j:t})$$

$$\iff V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t}) = \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + V_t(x_t, y_t)$$

$$\iff V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t}) = \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t)$$

$$+ \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) \leqslant \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_t(x_t, \hat{u}_t, \hat{d}_t^*, y_t) + \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, \hat{d}_k^*).$$

Finally, from this and the facts that $V_{t-L}(u_{t-L:t-1}, y_{t-L:t})$ satisfies (3.6e) and $\hat{d}_{t-L}^*$ and $\hat{x}_{t-L}^*$ achieve the maximum in (3.6e), we can conclude that

$$V_{t-L}(u_{t-L:t-1}, y_{t-L:t}) = l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t})$$

$$= \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) + l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t) + \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L})$$

$$\leqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k, \hat{d}_k^*) + l_t(x_t, \hat{u}_t, \hat{d}_t^*, y_t) + \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}).$$

Therefore $\hat{u}_k^*$ is a minimizing policy, for all $k \in \mathcal{T}$, and (3.5b) is satisfied with $J_t^*(u_{t-L:t-1}, y_{t-L:t}) = V_{t-L}(u_{t-L:t-1}, y_{t-L:t})$.

To prove (3.5a), let $\hat{d}_k^*$ be defined as in (3.8b), $\hat{d}_j^*$ be defined as in (3.8c), and let $\hat{d}_k$ and $\hat{d}_j$ be other arbitrary disturbance inputs. Similarly, let $\hat{x}_{t-L}^*$ be defined as in (3.8d), and let $\hat{x}_{t-L}$ be another arbitrary initial condition. Then, since $V_k(x_k)$ satisfies (3.6b), $V_t(x_t, y_t)$ satisfies (3.6c), $V_j(x_j, u_{j:t-1}, y_{j:t})$ satisfies (3.6d), and $\hat{d}_k^*$ achieves the maximum in (3.6b), $\hat{d}_t^*$ achieves the maximum in (3.6c), and $\hat{d}_j^*$ achieves the maximum in (3.6d),

we can use a similar argument as in the proof of (3.5b) to conclude that

$$V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t}) = \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t)$$

$$+ \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) \geqslant \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j, y_j) + l_t(x_t, \hat{u}_t^*, \hat{d}_t, y_t) + \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k).$$

Finally, from (3.6e), (3.8c), and (3.8d), we have

$$V_{t-L}(u_{t-L:t-1}, y_{t-L:t}) = \max_{\hat{x}_{t-L} \in \mathcal{X}} \max_{\hat{d}_{t-L} \in \mathcal{D}} \left(l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}, y_{t-L}) \right.$$

$$\left. + V_{t-L+1}(x_{t-L+1}, u_{t-L:t-1}, y_{t-L:t}) \right)$$

$$= l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L:t-1}, y_{t-L:t})$$

$$\geqslant l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L:t-1}, y_{t-L:t}),$$

and

$$V_{t-L}(u_{t-L:t-1}, y_{t-L:t}) = l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L:t-1}, y_{t-L:t})$$

$$\geqslant l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L:t-1}, y_{t-L:t}).$$

Then, (3.5a) follows because

$$V_{t-L}(u_{t-L:t-1}, y_{t-L:t}) = l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L}) + V_{t-L+1}(x_{t-L+1}, u_{t-L+1:t-1}, y_{t-L+1:t})$$

$$= \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k^*) + l_t(x_t, \hat{u}_t^*, \hat{d}_t^*, y_t) + \sum_{j=t-L}^{t-1} l_j(x_j, u_j, \hat{d}_j^*, y_j) + l_{t-L}(\hat{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*, y_{t-L})$$

$$\geqslant \sum_{k=t+1}^{t+T-1} l_k(x_k, \hat{u}_k^*, \hat{d}_k) + l_t(x_t, \hat{u}_t^*, \hat{d}_t, y_t) + \sum_{j=t-L+1}^{t-1} l_j(x_j, u_j, \hat{d}_j, y_j) + l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}, y_{t-L}).$$

Therefore $\hat{d}_k^*$, for all $k \in \mathcal{T}$, and $\hat{d}_j^*$, for all $j \in \mathcal{L}$, are maximizing policies, $\hat{x}_{t-L}^*$ is a

64

maximizing policy, and (3.5a) is satisfied with $J_t^*(u_{t-L:t-1}, y_{t-L:t}) = V_{t-L}(u_{t-L:t-1}, y_{t-L:t})$.

Thus, Assumption 5 is satisfied.   ∎

Next we specialize these results for DLTI systems and quadratic cost functions.

3.3.2   LTI systems and quadratic costs

Consider the following discrete linear time-invariant system, for all $t \in \mathbb{Z}_{\geqslant 0}$,

$$x_{t+1} = Ax_t + Bu_t + Dd_t, \qquad y_t = Cx_t + n_t, \tag{3.15}$$

with $x_t \in \mathcal{X} = \mathbb{R}^{n_x}$, $u_t \in \mathcal{U} = \mathbb{R}^{n_u}$, $d_t \in \mathcal{D} = \mathbb{R}^{n_d}$, $n_t \in \mathcal{N} = \mathbb{R}^{n_n}$, and $y_t \in \mathcal{Y} = \mathbb{R}^{n_y}$. Also consider the quadratic cost function

$$J_t(x_{t-L}, u_{t-L:t+T-1}, d_{t-L:t+T-1}, y_{t-L:t}) :=$$
$$\sum_{k=t}^{t+T-1} \left(x_k' Q x_k + \lambda_u u_k' u_k \right) + x_{t+T}' Q x_{t+T} - \sum_{k=t-L}^{t} \lambda_n (y_k - Cx_k)'(y_k - Cx_k) - \sum_{k=t-L}^{t+T-1} \lambda_d d_k' d_k$$
$$\tag{3.16}$$

where $Q = Q' \geqslant 0$ is a weighting matrix, and λ_u, λ_d, λ_n are positive constants that can be tuned to impose "soft" constraints on the variables x_k, u_k, d_k, and n_k, respectively, or to increase or decrease the penalty for choosing low likelihood values for the disturbances and noise. The positive scalar weights λ_u, λ_d, and λ_n could be replaced with positive-definite matrices, and the following results would still hold with minor adjustments. We use λ_u, λ_d, and λ_n here for simplicity.

Again, the control objective is to solve for a control input u_t^* that minimizes the criterion (3.16) in the presence of the worst-case disturbance d_t^* and initial state x_{t-L}^*. This motivates solving the optimization problem (3.3) with the cost given as in (3.16)

65

subject to the dynamics given in (3.15). Then the control input as defined in (3.4) is selected and applied to the plant.

The following Theorem gives conditions under which a saddle-point solution exists for problem (3.3) with cost (3.16), thereby satisfying Assumption 5, as well as a description of the resulting saddle-point solution.

Theorem 4 *(Existence of saddle-point for linear systems with quadratic costs) Let M_k and Λ_k, for all $k \in \mathcal{T}$, and P_j and Z_j, for all $j \in \mathcal{L}$, be matrices of appropriate dimensions defined by[2]*

$$M_k = Q + A'M_{k+1}\Lambda_k^{-1}A; \quad M_{t+T} = Q, \tag{3.17a}$$

$$\Lambda_k := I + \left(\frac{1}{\lambda_u}BB' - \frac{1}{\lambda_d}DD'\right)M_{k+1}, \tag{3.17b}$$

$$P_j = A'P_{j+1}A + A'P_{j+1}DZ_j^{-1}D'P_{j+1}A - \lambda_n C'C;$$

$$P_t = M_t - \lambda_n C'C, \tag{3.17c}$$

$$Z_j := \lambda_d I - D'P_{j+1}D. \tag{3.17d}$$

Then, if the following conditions are satisfied,

$$\lambda_u I + B'M_{k+1}B > 0, \tag{3.18a}$$

$$\lambda_d I - D'M_{k+1}D > 0, \tag{3.18b}$$

$$\lambda_d I - D'P_{j+1}D > 0, \tag{3.18c}$$

$$\begin{bmatrix} \lambda_n C'C - A'P_{t-L+1}A & -A'P_{t-L+1}D \\ -D'P_{t-L+1}A & \lambda_d I - D'P_{t-L+1}D \end{bmatrix} > 0, \tag{3.18d}$$

the min-max optimization (3.3) with quadratic costs (3.16) subject to the linear dynamics

---

[2]I denotes the identity matrix with appropriate dimensions.

(3.15) *admits a unique saddle-point solution that satisfies Assumption 5.*   □

Proof. This proof shows that for the linear system (3.15) and quadratic cost function (3.16), there exist solutions as in (3.8) that satisfy functions as in (3.6), and therefore, a saddle-point exists by Theorem 3.

In this linear-quadratic case, the functions (3.6) can be solved for explicitly, beginning with $V_{t+T-1}(\cdot)$, then $V_{t+T-2}(\cdot)$, etc., and continuing recursively backwards in time until V_{t-L}, by recognizing that, for all $k \in \mathcal{T}$ and $j \in \mathcal{L}$,

$$q_{t+T}(x_{t+T}) := x'_{t+T}Qx_{t+T},$$

$$c_k(x_k, \hat{u}_k) := x'_k Q x_k + \lambda_u \hat{u}'_k \hat{u}_k,$$

$$\rho_k(\hat{d}_k) := -\lambda_d \hat{d}'_k \hat{d}_k,$$

$$\rho_j(\hat{d}_j) := -\lambda_d \hat{d}'_j \hat{d}_j,$$

$$\eta_j(n_j) := -\lambda_n (y - Cx_j)'(y - Cx_j)$$

and then computing the solutions to (3.6). This results in functions $V_k(\cdot)$ and $V_j(\cdot)$, for all $k \in \mathcal{T}$ and $j \in \mathcal{L}$, given as follows

$$V_{t+T}(x_{t+T}) = x'_{t+T}Qx_{t+T}, \tag{3.19a}$$

$$V_k(x_k) = x'_k M_k x_k, \qquad\qquad \forall k \in \mathcal{T} \backslash t, \tag{3.19b}$$

$$V_t(x_t, y_t) = x'_t P_t x_t, \tag{3.19c}$$

$$V_j(x_j, y_{j:t}, u_{j:t-1}) = x'_j P_j x_j + 2w'_j x_j + c_j, \qquad \forall j \in \mathcal{L} \backslash t - L, \tag{3.19d}$$

$$V_{t-L}(y_{t-L:t}, u_{t-L:t-1}) = w'_{t-L} P^{-1}_{t-L} w_{t-L} - 2w'_{t-L} P^{-1}_{t-L} w_{t-L} + c_{t-L}, \tag{3.19e}$$

where the vectors w_j, for all $j \in \mathcal{L}$, are defined by

$$w_j = A' R'_j (P_{j+1} B u_j + w_{j+1}) + \lambda_n C' y_j;$$

$$w_t = \lambda_n C' y_t,$$

with the matrices R_j and scalars c_j defined by

$$R_j := I + D Z_j^{-1} D' P_{j+1},$$

$$c_j = w'_{j+1} D Z_j^{-1} D' w_{j+1} - \lambda_n y'_j y_j + c_{j+1} + (u'_j B' P_{j+1} + 2 w'_{j+1}) R_j B u_j;$$

$$c_t = -\lambda_n y'_t y_t.$$

Then conditions (3.18) come directly from the second-order conditions for strict-convexity/concavity of a quadratic function. Specifically, for the quadratic cost (3.16), the costs $l_k(\cdot)$, $l_t(\cdot)$, and $l_j(\cdot)$ in (3.6) are given as

$$l_k(x_k, \hat{u}_k, \hat{d}_k) := x'_k Q x_k + \lambda_u \hat{u}'_k \hat{u}_k - \lambda_d \hat{d}'_k \hat{d}_k, \qquad k \in \mathcal{T} \backslash t, \qquad \text{(3.20a)}$$

$$l_t(x_t, \hat{u}_t, \hat{d}_t, y_t) := x'_t Q x_t + \lambda_u \hat{u}'_t \hat{u}_t - \lambda_n n'_t n_t - \lambda_d \hat{d}'_t \hat{d}_t, \qquad \text{(3.20b)}$$

$$l_j(x_j, u_j, \hat{d}_j, y_j) := -\lambda_n n'_j n_j - \lambda_d \hat{d}'_j \hat{d}_j, \qquad j \in \mathcal{L}, \qquad \text{(3.20c)}$$

and condition (3.18a) comes from computing the Hessian matrix (the matrix of second-order partial derivatives) of $l_k(x_k, \hat{u}_k, \hat{d}_k) + V_{k+1}(f(x_k, \hat{u}_k, \hat{d}_k))$ with respect to $\hat{u}_k$ as well as the Hessian matrix of $l_t(x_t, \hat{u}_t, \hat{d}_t) + V_{t+1}(f(x_t, \hat{u}_t, \hat{d}_t))$ with respect to $\hat{u}_t$ and requiring these Hessian matrices to be positive definite. Similarly, condition (3.18b) comes from computing the Hessian matrix of $l_k(x_k, \hat{u}_k, \hat{d}_k) + V_{k+1}(f(x_k, \hat{u}_k, \hat{d}_k))$ with respect to $\hat{d}_k$ as well as the Hessian matrix of $l_t(x_t \hat{u}_t, \hat{d}_t) + V_{t+1}(f(x_t, \hat{u}_t, \hat{d}_t))$ with respect to $\hat{d}_t$ and requiring these Hessian matrices to be negative definite. Condition (3.18c) comes from comput-

ing the Hessian matrix of $l_j(x_j, u_j, \hat{d}_j) + V_{j+1}(f(x_j, u_j, \hat{d}_j), u_{j:t-1}, y_{j:t})$ with respect to $\hat{d}_j$ and requiring it to be negative definite. Finally, condition (3.18d) comes from computing the Hessian matrix of $l_{t-L}(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}) + V_{t-L+1}(f(\hat{x}_{t-L}, u_{t-L}, \hat{d}_{t-L}), u_{t-L:t-1}, y_{t-L:t})$ with respect to $[\hat{x}'_{t-L} \; \hat{d}'_{t-L}]'$ and requiring it to be negative definite. Therefore, if conditions (3.18) are satisfied, the optimization (3.3) with cost (3.16) is strictly convex with respect to $\hat{u}_{t:t+T-1}$ and is strictly concave with respect to $\hat{d}_{t-L:t+T-1}$ and $\hat{x}_{t-L}$, and therefore, the functions (3.19) satisfy the equations in (3.6). Thus a saddle point solution exists because of Theorem 3.

In this case, the solutions (3.8) can be found analytically and are given by

$$\hat{u}_k^* = -\frac{1}{\lambda_u} B' M_{k+1} \Lambda_k^{-1} A x_k^*, \tag{3.21a}$$

$$\hat{d}_k^* = \frac{1}{\lambda_d} D' M_{k+1} \Lambda_k^{-1} A x_k^*, \tag{3.21b}$$

$$\hat{d}_j^* = Z_j^{-1} D' (P_{j+1}(A x_j + B u_j) + w_{j+1}), \tag{3.21c}$$

$$\hat{x}_{t-L}^* = -P_{t-L}^{-1} w_{t-L}, \tag{3.21d}$$

where the corresponding state trajectory is determined from

$$x_{k+1}^* = \Lambda_k^{-1} A x_k^*, \tag{3.22a}$$

$$x_{j+1}^* = R_j (A x_j^* + B u_j) + D Z_j^{-1} D' w_{j+1}, \tag{3.22b}$$

$$x_{t-L}^* = \hat{x}_{t-L}^*. \tag{3.22c}$$

The state trajectory (3.22) is found by plugging the saddle-point solutions (3.21) into the dynamics (3.15). Finally, as a consequence of the argument in the proof of Theorem

3, the corresponding saddle-point value is

$$J_t^*(y_{t-L:t}, u_{t-L:t-1}) = V_{t-L}(y_{t-L:t}, u_{t-L:t-1}).$$

∎

Remark 7 *For times $k \in \mathcal{T} \backslash t$, the result given in Theorem 4 is very close to the result derived in [6] for the affine-quadratic two-person zero-sum game because the equations (3.6a) and (3.6b) equivalently describe a linear-quadratic two-person zero-sum game.*

Corollary 3 *If the discrete-time linear time-invariant system given in (3.15) is observable, then the scalar weights λ_n and λ_d can be chosen sufficiently large such that the conditions (3.18a)-(3.18d) are satisfied. Therefore, according to Theorem 4, there exists a saddle-point solution for the optimization problem (3.3) with cost (3.16). Therefore, also, Assumption 5 is satisfied.*   □

Proof. Condition (3.18a) is trivially satisfied for all $k \in \mathcal{T}$ as long as we choose $\lambda_u > 0$ and weighting matrix $Q \geqslant 0$ because $Q \geqslant 0 \implies M_k \geqslant 0, \forall k \in \mathcal{T}$. [3]

Condition (3.18b) is satisfied if the scalar weight λ_d is chosen sufficiently large. To show this, we take the limit of the sequence of matrices M_k, as given in (3.17a), as $\lambda_d \to \infty$ and notice that $M_k \to \bar{M}_k$, where $\bar{M}_k$ is described by

$$\bar{M}_k = Q + A'(\bar{M}_{k+1}[I + \frac{1}{\lambda_u} B B' \bar{M}_{k+1}]^{-1})A;$$

$$\bar{M}_{t+T} = M_{t+T},$$

for all $k \in \mathcal{T}$. Then, as $\lambda_d \to \infty$, λ_d is greater than the largest eigenvalue of $D'\bar{M}_{t+1}D$, and therefore, condition (3.18b) is satisfied when λ_d is chosen sufficiently large.

---

[3]Note that $M_k = M_k'$ due to the fact that $Q = Q'$ and the matrix identity in [100] which says that $A(I + BA)^{-1} = (I + AB)^{-1}A$.

Next we prove that conditions (3.18c) and (3.18d) are satisfied, for all $j \in \mathcal{L}$, when λ_n and λ_d are chosen sufficiently large and the system (3.15) is observable. We first take the limit of the sequence of matrices P_j, as given in (3.17c), as $\lambda_d \to \infty$ and notice that $P_j \to \bar{P}_j$ where $\bar{P}_j$ is described by

$$\bar{P}_j = -\lambda_n \Theta'_j \Theta_j + A'^{t-j} \bar{M}_t A^{t-j},$$

for all $j \in \mathcal{L} \cup t$, and Θ_j is defined as

$$\Theta_j := [C' \quad A'C' \quad A'^2 C' \quad \ldots \quad A'^{t-j} C']'.$$

The matrix Θ_j looks similar to the observability matrix, and therefore, $\Theta'_j \Theta_j > 0$ if the system given in (3.15) is observable.

Then, the scalar weight λ_n can be chosen large enough to ensure that $\lambda_n \Theta'_j \Theta_j > A'^{t-j} \bar{M}_t A^{t-j}$, for all $j \in \mathcal{L}$. It then follows that $\bar{P}_j < 0$ for all $j \in \mathcal{L}$. Therefore, condition (3.18c) becomes $\lambda_d I - D' \bar{P}_{j+1} D > 0$ in the limit as $\lambda_d \to \infty$ and is trivially satisfied if system (3.15) is observable and λ_n is chosen sufficiently large.

Finally, consider condition (3.18d). Using the Schur Complement, condition (3.18d) is satisfied if, and only if,

$$\lambda_d I - D' P_{t-L+1} D > 0$$

and

$$\lambda_n C'C - A'P_{t-L+1}A - A'P_{t-L+1}D(\lambda_d I - D'P_{t-L+1}D)^{-1}D'P_{t-L+1}A > 0.$$

We just proved that $\lambda_d I - D' P_{t-L+1} D > 0$ if the system (3.15) is observable and the

weights λ_d and λ_n are chosen sufficiently large. Then, in the limit as $\lambda_d \to \infty$, the second inequality becomes

$$\lambda_n C'C - A'\bar{P}_{t-L+1}A > 0.$$

Therefore, if the system (3.15) is observable, λ_n can be chosen sufficiently large such that this inequality is satisfied, and therefore, condition (3.18d) is satisfied.   ∎

3.4   Simulation

Various choices for the parameters in the cost function (3.16) lead to different control inputs that may all satisfy the saddle-point assumption but that produce very different closed-loop performance. For instance, there are examples for which a short finite-horizon approach performs better than a quasi-infinite-horizon approach even for unconstrained linear-quadratic problems. Specifically, better disturbance attenuation can be achieved for the following unconstrained linear-quadratic example, where the system is subjected to impulsive disturbances, using the combined MPC/MHE approach with shorter finite-horizon lengths.

The following example can be solved numerically using the methods described in Chapter 4.

Example 3 *(Stabilizing a riderless bicycle.) Consider the following second order continuous-time linearized bicycle model in state-space form:*

$$\dot{x}_t = Ax_t + B(u_t + d_t), \qquad y_t = Cx_t + n_t \tag{3.23}$$

The state is given by $x_t = \begin{bmatrix} \phi & \delta & \dot{\phi} & \dot{\delta} \end{bmatrix}'$ where ϕ is the roll angle of the bicycle, δ is the

72

steering angle of the handlebars, and $\dot\phi$ and $\dot\delta$ are the corresponding angular velocities. The control input u_t is the steering torque applied to the handlebars. The matrices defining the linearized dynamics are, as described in [18],

$$
A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 13.67 & 0.225 - 1.319v^2 & -0.164v & -0.552v \\ 4.857 & 10.81 - 1.125v^2 & 3.621v & -2.388v \end{bmatrix},
$$

$$
B = \begin{bmatrix} 0 \\ 0 \\ -0.339 \\ 7.457 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},
$$

where v is the bicycle's forward velocity. Only the roll and steering angles (and not their corresponding angular velocities) are measured and available for feedback. In this example, we fix the forward velocity at $v = 2$ m/s, which results in an unstable system, and discretize the system using a 0.1 second zero-order-hold.

The control objective is to stabilize the bicycle in the upright position, i.e. around a zero roll angle ($\phi = 0$), by applying a steering torque to the handlebars. The disturbance d_t acts on the input and can be thought of as jolting the steering due to sharp bumps in the bicycle's path or similar environmental perturbations. We solve this problem by solving the optimization given in (3.3) with cost (3.16) at each time t and apply the resulting $\hat{u}_t^*$ as the control input. The measurement noise is a random variable $n_t \sim \mathcal{N}(0, 0.001^2)$. The disturbance d_t is nominally a random variable $d_t \sim \mathcal{N}(0, 0.01^2)$ but with occasional large, impulsive values.

Because the system (3.23) is observable, we are able to choose λ_d and λ_n so that

conditions (3.18) *are satisfied. Therefore, a saddle-point solution to* (3.3) *with cost* (3.16)

exists for the riderless bicycle example according to Theorem 4. However, conditions

(3.18) *are only sufficient conditions and may lead to unnecessarily conservative weights.*

In this example, it is possible to achieve better performance by choosing weights that do

not satisfy conditions (3.18) *but that still ensure the existence of a saddle-point which*

can be verified numerically.

In this example, we compare results for long horizon lengths ($L = T = 200$) *to*

results for short horizon lengths ($L = 2$, $T = 7$) *and tune the weights* λ_d *and* λ_n *in*

order to achieve the best performance as determined by minimizing the tracking error

$\|\phi\|$*. Table 3.1 shows four simulation scenarios. Rows #1 and #2 of Table 3.1 show*

the weights λ_d *and* λ_n *that satisfy the conditions* (3.18) *and provide the best performance*

for both the long and short horizon lengths. Rows #3 and #4 of Table 3.1 show the best

possible weights λ_d *and* λ_n *for performance that still ensure the existence of a saddle-point*

(verified numerically) but that do not satisfy the conditions (3.18). *For all four scenarios,*

the weighting matrix Q *in the cost* (3.16) *is chosen as a 4x4 matrix with the element in*

upper left corner equal to one and all other elements equal to zero, and λ_u *is chosen as*

0.001.

Table 3.1: Tuning Parameters and Performance

	L	T	λ_d	λ_n	$\|\phi\|$ [deg]	$\|u\|$ [Nm]
#1	200	200	1500	10^7	29.0	19.3
#2	2	7	90	10^7	27.1	19.2
#3	200	200	0.02	15000	22.1	18.0
#4	2	7	0.002	15000	14.6	35.5

Figures 3.1 and 3.2 show results for the scenarios given in Table 3.1. The top plot of

each figure shows the measured output ϕ*, the middle plot shows the measured output* δ*,*

and the bottom plot shows the applied control input u^* *as well as the true disturbance* d

that is the same for all four of the scenarios in Table 3.1. In this case, the disturbance

is impulsive which means that it is usually zero but occasionally becomes a fairly large

nonzero value as shown in the bottom plots of Figures 3.1 and 3.2.
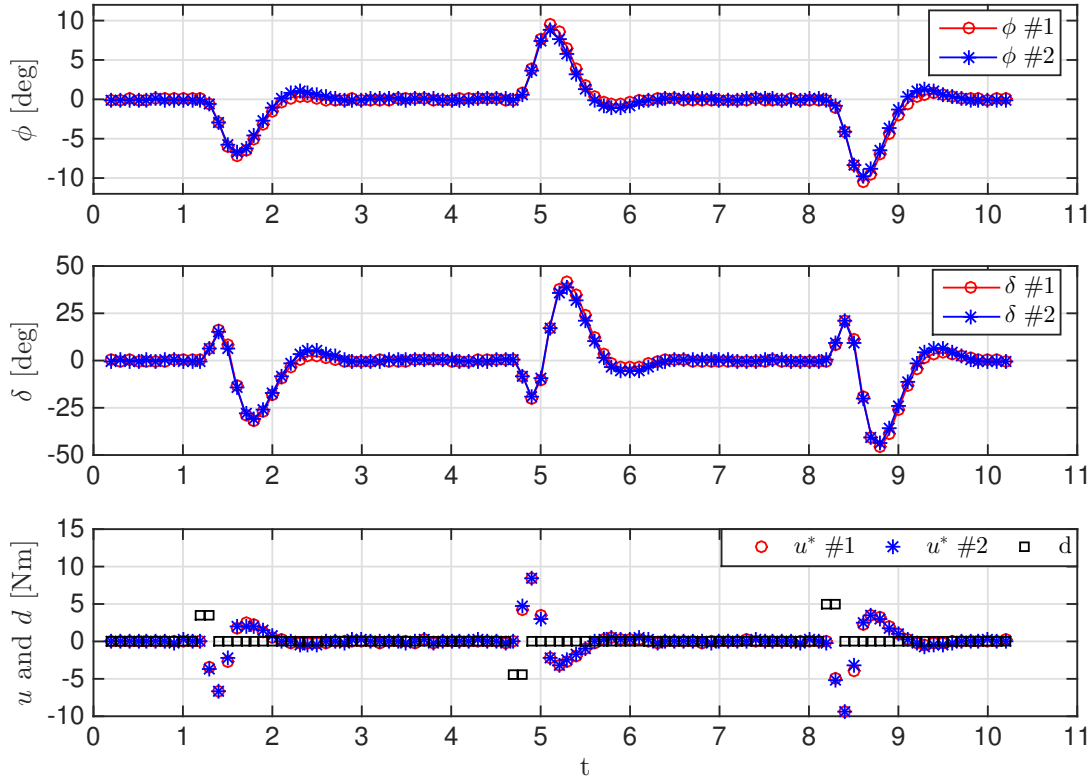
Figure 3.1: Comparing results for longer horizons (red o's) versus shorter horizons (blue *'s) with weights given in rows #1 and #2 of Table 3.1.

The control input computed using the shorter horizons is able to regulate the roll angle

ϕ back to zero without as much oscillation as the control input computed using the longer

horizons. This is because a larger λ_d is required with long horizons to satisfy the saddle-

point assumption while a smaller λ_d can be used with short horizons. In this case, a

smaller λ_d results in a less conservative control input that better attenuates the large im-

pulsive disturbances. Therefore, it may be beneficial to use the finite-horizon MPC/MHE

approach over other standard infinite-horizon control techniques for particular types of

unconstrained linear-quadratic problems.
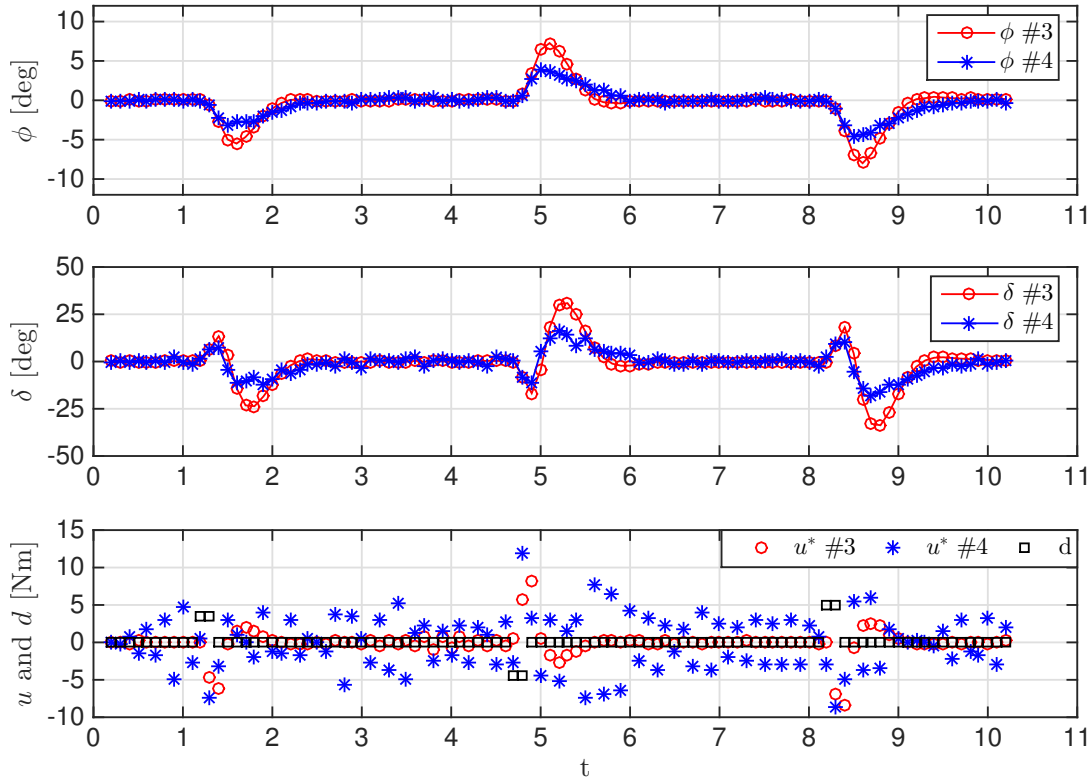
Figure 3.2: Comparing results for longer horizons (red o's) versus shorter horizons (blue *'s) with weights given in rows #3 and #4 of Table 3.1.

3.5 Conclusions

In this chapter, we discussed the main assumption of our combined MPC/MHE approach that a saddle-point solution exists for the min-max optimization problem formulated in Chapter 2. First we gave conditions for the existence of a saddle-point solution when considering a general discrete-time nonlinear system and a general cost function. Next we specialized those results for DLTI systems and quadratic cost functions. For this case, we showed that observability of the linear system and large weights λ_d and λ_n in the cost function are sufficient conditions for a saddle-point solution to exist.

In a numerical example of a linearized riderless bicycle system subjected to impulsive disturbances, we showed results for four different scenarios. We first considered cost function weights that satisfy the derived sufficient conditions for a saddle-point solution

and then considered cost function weights that still result in a saddle-point, but do not satisfy the sufficient conditions, and result in better performance. For each choice of cost function weights, we considered long and short horizon lengths. The results showed that it is straightforward to satisfy the sufficient conditions for the existence of a saddle-point solution and illustrated the importance of carefully choosing tuning parameters in order to achieve desirable performance.

Future work may involve relaxing the requirement of a saddle-point solution to that of being ε-close to a saddle-point solution. This could be related to results for ε-Nash equilibria.

Chapter 4

Numerical Optimization

In this chapter, we discuss numerical optimization methods for solving the min-max optimization problem formulated in Chapter 2. Specifically, we develop new primal-dual interior-point algorithms that can be used to compute the saddle-point equilibrium that needs to be solved for online at each sampling time. These algorithms rely on the use of Newton's method to solve a relaxed version of the Karush-Kuhn-Tucker (KKT) conditions associated with the coupled optimizations that define the saddle-point equilibrium. As in classical primal-dual methods, we replace the equality to zero of the complementary slackness conditions by equality to a positive constant μ that we force to converge to zero as the Newton iterations progress. In practice, the algorithms will stop with a positive value for μ, but we show that this still leads to an ϵ-saddle-point, where ϵ can be explicitly computed and made arbitrarily small through the selection of an appropriate stopping criterion.

The optimization algorithms proposed here are heavily inspired by primal-dual interior-point methods [111] that have been very successful in solving convex optimizations [14]. The use of interior-point algorithms to solve MPC problems is discussed in [91], and additional early work on efficient numerical methods for solving MPC problems can be

found in [12, 11, 110]. An overview of the numerical methods available for solving the optimization problems that arise in nonlinear MPC and MHE is given by [29], whereas the more recent work [109] is focused on the development of fast dedicated solvers for MPC problems. In [80], the authors specifically consider numerical methods for solving min-max MPC as a quadratic program, and robust dynamic programming for min-max MPC of constrained uncertain systems is considered in [28]. Finally, the methods that are described in Sections 4.3 and 4.4 are directly inspired by the primal-dual interior-point method for a single optimization described in [107].

4.1   Review of Chapter 2

To keep this chapter self-contained, we recall some details about the problem formulation from Chapter 2.

We consider time-varying nonlinear discrete-time processes of the form

$$x_{t+1} = f_t(x_t, u_t, d_t), \qquad\qquad y_t = g_t(x_t) + n_t, \qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0} \qquad (4.1)$$

At each time $t \in \mathbb{Z}_{\geqslant L}$, our objective is to compute the control u_t so as to minimize a desired criterion $J_t(x_{t-L}, u_{t-L:t+T-1}, d_{t-L:t+T-1}, y_{t-L:t})$ which depends on the *unknown* initial state $x_{t-L} \in \mathcal{X}$ and a finite number of past inputs $u_{t-L:t-1}$, past output measurements $y_{t-L:t}$, future control inputs $u_{t:t+T-1}$ constrained to belong to the set $\mathcal{U}$, and past and future disturbances $d_{t-L:t+T-1}$ constrained to belong to the set $\mathcal{D}$. Since we do not know the value of the variables x_{t-L} and $d_{t-L:t+T-1}$, we also optimize this criterion under worst-case assumptions on these variables, leading to the following finite horizon

min-max optimization

$$\min_{\substack{\hat{u}_{t:t+T-1|t} \in \mathcal{U}}} \max_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} J_t\big(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}\big). \tag{4.2}$$

which is subject to the process dynamics (4.1). At each time t, we use as the control
input the first element of the sequence

$$\hat{u}^*_{t:t+T-1|t} = \{\hat{u}^*_{t|t}, \hat{u}^*_{t+1|t}, \hat{u}^*_{t+2|t}, \ldots \hat{u}^*_{t+T-1|t}\} \in \mathcal{U}$$

that minimizes (4.2), leading to the following control law:

$$u_t = \hat{u}^*_{t|t}, \quad \forall t \geqslant 0. \tag{4.3}$$

For the theoretical results given in Chapter 2 for this combined estimation and control
approach, we require the following assumption:

Assumption 6 (Saddle-point [Assumption 2, Chapter 2]) *The min-max optimiza-
tion (4.2) always has a saddle-point solution for which the min and max commute. Specif-
ically, for every time $t \in \mathbb{Z}_{\geqslant 0}$, past control input sequence $u_{t-L:t-1} \in \mathcal{U}$, and past mea-
sured output sequence $y_{t-L:t} \in \mathcal{Y}$, there exists a finite scalar $J_t^* \in \mathbb{R}$, an initial condition
$\hat{x}^*_{t-L|t} \in \mathcal{X}$, and sequences $\hat{u}^*_{t:t+T-1|t} \in \mathcal{U}$, $\hat{d}^*_{t-L:t+T-1|t} \in \mathcal{D}$ such that*

$$J_t^* = J_t\big(\hat{x}^*_{t-L|t}, u_{t-L:t-1}, \hat{u}^*_{t:t+T-1|t}, \hat{d}^*_{t-L:t+T-1|t}, y_{t-L:t}\big)$$

$$= \max_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} J_t\big(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}^*_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}\big) \tag{4.4a}$$

$$= \min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} J_t\big(\hat{x}^*_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}, \hat{d}^*_{t-L:t+T-1|t}, y_{t-L:t}\big). \tag{4.4b}$$

□

The remainder of this chapter discusses numerical approaches for computing a saddle-point solution to the min-max optimization (4.2).

4.2   Computation of Control by Solving a Pair of Coupled Optimizations

To implement the control law (4.3) we need to find the control sequence $\hat{u}^*_{t:t+T-1|t} \in \mathcal{U}$ that achieves the outer minimization in (4.2). In view of Assumption 6, the desired control sequence must be part of the saddle-point defined by (4.4a)–(4.4b). From the perspective of numerically computing this saddle-point, it is more convenient to use the following equivalent characterization of the saddle-point:

$$-J^*_t(u_{t-L:t-1}, y_{t-L:t}) = \min_{\substack{\hat{x}_{t-L|t} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1|t} \in \mathcal{D}}} -J_t\big(\hat{x}_{t-L|t}, u_{t-L:t-1}, \hat{u}^*_{t:t+T-1|t}, \hat{d}_{t-L:t+T-1|t}, y_{t-L:t}\big)$$

(4.5a)

$$J^*_t(u_{t-L:t-1}, y_{t-L:t}) = \min_{\hat{u}_{t:t+T-1|t} \in \mathcal{U}} J_t\big(\hat{x}^*_{t-L|t}, u_{t-L:t-1}, \hat{u}_{t:t+T-1|t}, \hat{d}^*_{t-L:t+T-1|t}, y_{t-L:t}\big) \quad (4.5b)$$

where we introduce the "$-$" sign in (4.5a) simply to obtain two minimizations, instead of a maximization and one minimization, which will somewhat simplify the presentation.

Since the process dynamics (4.1) has a unique solution for any given initial condition, control input, and unmeasured disturbance, the coupled optimizations in (4.5) can be re-written as

$$-J^*_t(u_{t-L:t-1}, y_{t-L:t}) = \min_{(\hat{d}_{t-L:t+T-1|t}, \bar{x}_{t-L:t+T|t}) \in \bar{\mathcal{D}}[u_{t-L:t-1}, \hat{u}^*_{t:t+T-1}]}$$

81

$$-\sum_{s=t}^{t+T-1} c_s(\bar{x}_s, \hat{u}_s^*, \hat{d}_s) - q_{t+T}(\bar{x}_{t+T}) + \sum_{s=t-L}^{t} \eta_s(y_s - g_s(\bar{x}_s)) + \sum_{s=t-L}^{t+T-1} \rho_s(\hat{d}_s), \quad (4.6a)$$

$$J_t^*(u_{t-L:t-1}, y_{t-L:t}) = \min_{(\hat{u}_{t:t+T-1|t}, \tilde{x}_{t-L+1:t+T|t}) \in \bar{\mathcal{U}}[\bar{x}_{t-L}^*, \hat{d}_{t-L:t+T-1}^*]}$$

$$\sum_{s=t}^{t+T-1} c_s(\tilde{x}_s, \hat{u}_s, \hat{d}_s^*) + q_{t+T}(\tilde{x}_{t+T}) - \sum_{s=t-L}^{t} \eta_s(y_s - g_s(\tilde{x}_s)) - \sum_{s=t-L}^{t+T-1} \rho_s(\hat{d}_s^*) \quad (4.6b)$$

where

$$\bar{\mathcal{D}}[u_{t-L:t-1}, \hat{u}_{t:t+T-1}^*] := \Big\{ (\hat{d}_{t-L:t+T-1|t}, \bar{x}_{t-L:t+T|t}) :$$

$$\hat{d}_{t-L:t+T-1|t} \in \mathcal{D}, \ \bar{x}_{t-L:t+T|t} \in \mathcal{X},$$

$$\bar{x}_{s+1} = f_s(\bar{x}_s, u_s, \hat{d}_s), \forall s \in \{t-L, ..., t-1\},$$

$$\bar{x}_{s+1} = f_s(\bar{x}_s, \hat{u}_s^*, \hat{d}_s), \forall s \in \{t, ..., t+T-1\} \Big\} \quad (4.7a)$$

$$\bar{\mathcal{U}}[\bar{x}_{t-L}^*, \hat{d}_{t-L:t+T-1}^*] := \Big\{ (\hat{u}_{t:t+T-1|t}, \tilde{x}_{t-L+1:t+T|t}) :$$

$$\hat{u}_{t:t+T-1|t} \in \mathcal{U}, \ \tilde{x}_{t-L+1:t+T|t} \in \mathcal{X},$$

$$\tilde{x}_{t-L+1} = f_{t-L}(\bar{x}_{t-L}^*, u_{t-L}, \hat{d}_{t-L}^*),$$

$$\tilde{x}_{s+1} = f_s(\tilde{x}_s, u_s, \hat{d}_s^*), \forall s \in \{t-L+1, ..., t-1\},$$

$$\tilde{x}_{s+1} = f_s(\tilde{x}_s, \hat{u}_s, \hat{d}_s^*), \forall s \in \{t, ..., t+T-1\} \Big\}. \quad (4.7b)$$

To obtain the optimizations in (4.6), we introduce the values of the state from time $t - L + 1$ to time $t + T$, that are constrained by the system dynamics, as additional optimization variables in each of the optimizations in (4.5). While this introduces additional optimization variables, it avoids the need to explicitly evaluate the solution $\varphi(t; t - L, x_{t-L}, u_{t-L:t-1}, d_{t-L:t-1})$ that appears in the original optimizations (4.5) and that can be numerically poorly conditioned, e.g., for systems with unstable dynamics.

While the numerical method discussed in the next section can be used to solve either

(4.5) or (4.6), we prefer the latter because it generally leads to simpler optimization problems. Therefore we focus our discussion on that approach.

4.3   Interior-Point Method for Minimax Problems

The coupled optimizations in (4.5) or (4.6) can be viewed as a special case of the following more general problem: Find a pair $(u^*, d^*) \in \mathcal{U}[d^*] \times \mathcal{D}[u^*]$ that simultaneously solves the two coupled optimizations

$$f(u^*, d^*) = \min_{u \in \mathcal{U}[d^*]} f(u, d^*), \tag{4.8a}$$

$$g(u^*, d^*) = \min_{d \in \mathcal{D}[u^*]} g(u^*, d), \tag{4.8b}$$

with

$$\mathcal{U}[d] := \{u \in \mathbb{R}^{N_u} : F_u(u, d) \geqslant 0, \; G_u(u, d) = 0\}, \tag{4.9a}$$

$$\mathcal{D}[u] := \{d \in \mathbb{R}^{N_d} : F_d(u, d) \geqslant 0, \; G_d(u, d) = 0\}, \tag{4.9b}$$

for given functions $f : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $g : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $F_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_u}$, $F_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_d}$, $G_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_u}$, $G_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_d}$. To map (4.6) to (4.8), one would associate the vectors $u \in \mathbb{R}^{N_u}$ and $d \in \mathbb{R}^{N_d}$ in (4.8) with the sequences in the sets $\bar{\mathcal{D}}[\cdot]$ and $\bar{\mathcal{U}}[\cdot]$ in (4.7). In this case, the equality constraints in (4.9) would typically correspond to the system dynamics, and the inequality constraints in (4.9) would enforce that the state, control, and disturbance signals belong, respectively, to the sets $\mathcal{X}, \mathcal{U}$, and $\mathcal{D}$.

Remark 8 *The optimization in (4.8) is more general than the one in (4.6) in that the function being minimized in (4.6a) is the symmetric of the function being minimized*

in (4.6b), whereas in (4.8), f and g need not be the symmetric of each other. While this generalization does not appear to be particularly useful for our output-feedback MPC application, all the results that follow do apply to general functions f and g and can be useful for other applications. □

The following duality-like result provides the motivation for a primal-dual-like method to solve the coupled minimizations in (4.8). It provides a set of conditions, involving an unconstrained optimization, that provide an approximation to the solution of (4.8).

Lemma 3 (Approximate equilibrium) *Suppose that we have found* primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}$ *and* dual variables $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}, \hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}$ *that simultaneously satisfy all of the following conditions*[1]

$$G_u(\hat{u}, \hat{d}) = 0, \qquad G_d(\hat{u}, \hat{d}) = 0, \tag{4.10a}$$

$$\hat{\lambda}_{fu} \dot{\geqslant} 0, \quad \hat{\lambda}_{gd} \dot{\geqslant} 0, \quad F_u(\hat{u}, \hat{d}) \geqslant 0, \quad F_d(\hat{u}, \hat{d}) \geqslant 0, \tag{4.10b}$$

$$L_f(\hat{u}, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) = \min_{u \in \mathbb{R}^{N_u}} L_f(u, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}),$$

$$L_g(\hat{u}, \hat{d}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) = \min_{d \in \mathbb{R}^{N_d}} L_g(\hat{u}, d, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) \tag{4.10c}$$

where, for all u, d, λ, and ν,

$$L_f(u, d, \lambda_{fu}, \nu_{fu}) \coloneqq f(u, d) - \lambda'_{fu} F_u(u, d) + \nu'_{fu} G_u(u, d),$$

$$L_g(u, d, \lambda_{gd}, \nu_{gd}) \coloneqq g(u, d) - \lambda'_{gd} F_d(u, d) + \nu'_{gd} G_d(u, d).$$

Then $(\hat{u}, \hat{d})$ approximately satisfy (4.8) in the sense that

$$f(\hat{u}, \hat{d}) \leqslant \epsilon_f + \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}), \tag{4.11a}$$

_____

[1]Given a vector $x \in \mathbb{R}^n$ and a scalar $a \in \mathbb{R}$, we denote by $x \dot{\geqslant} a$ the proposition that every entry of x is greater than or equal to a.

$$g(\hat{u}, \hat{d}) \leqslant \epsilon_g + \min_{d \in \mathcal{D}[\hat{u}]} g(\hat{u}, d), \tag{4.11b}$$

with

$$\epsilon_f := \hat{\lambda}_{fu} F_u(\hat{u}, \hat{d}), \qquad \qquad \epsilon_g := \hat{\lambda}_{gd} F_d(\hat{u}, \hat{d}). \tag{4.12}$$

$\square$

Proof of Lemma 3. The proof is a direct consequence of the following sequence of inequalities that start from the equalities in (4.10c) and use the conditions (4.10a)-(4.10b), and the definitions (4.12) to arrive at (4.11):

$$f(\hat{u}, \hat{d}) - \epsilon_f = L_f(\hat{u}, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) - \hat{\nu}'_{fu} G_u(\hat{u}, \hat{d})$$

$$= \min_{u \in \mathbb{R}^{N_u}} L_f(u, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu})$$

$$= \min_{u \in \mathbb{R}^{N_u}} f(u, \hat{d}) - \hat{\lambda}'_{fu} F_u(u, \hat{d}) + \hat{\nu}'_{fu} G_u(u, \hat{d})$$

$$\leqslant \max_{\substack{\lambda_{fu} \geqslant 0, \\ \nu_{fu}}} \min_{u \in \mathbb{R}^{N_u}} f(u, \hat{d}) - \lambda'_{fu} F_u(u, \hat{d}) + \nu'_{fu} G_u(u, \hat{d})$$

$$\leqslant \max_{\substack{\lambda_{fu} \geqslant 0, \\ \nu_{fu}}} \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}) - \lambda'_{fu} F_u(u, \hat{d}) + \nu'_{fu} G_u(u, \hat{d})$$

$$= \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}),$$

$$g(\hat{u}, \hat{d}) - \epsilon_g = L_g(\hat{u}, \hat{d}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) - \hat{\nu}'_{gd} G_d(\hat{u}, \hat{d})$$

$$= \min_{d \in \mathbb{R}^{N_d}} L_g(\hat{u}, d, \hat{\lambda}_{gd}, \hat{\nu}_{gd})$$

$$= \min_{d \in \mathbb{R}^{N_d}} g(\hat{u}, d) - \hat{\lambda}'_{gd} F_d(\hat{u}, d) + \hat{\nu}'_{gd} G_d(\hat{u}, d)$$

$$\leqslant \max_{\substack{\lambda_{gd} \geqslant 0, \\ \nu_{gd}}} \min_{d \in \mathbb{R}^{N_d}} g(\hat{u}, d) - \lambda'_{gd} F_d(\hat{u}, d) + \nu'_{gd} G_d(\hat{u}, d)$$

85

$$\leqslant \max_{\substack{\lambda_{gd} \geqslant 0, \\ \nu_{gd}}} \min_{d \in \mathcal{D}[\hat{u}]} \ g(\hat{u}, d) - \lambda'_{gd} F_d(\hat{u}, d) + \nu'_{gd} G_d(\hat{u}, d)$$

$$= \min_{d \in \mathcal{D}[\hat{u}]} \ g(\hat{u}, d). \qquad\qquad\qquad \blacksquare$$

4.3.1   Interior-point primal-dual equilibria algorithm

The proposed method consists of using Newton iterations to solve a system of nonlinear equations on the primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}$ and dual variables $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}, \hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}$ introduced in Lemma 3. Throughout this section, we ask that L_f and L_g are continuously differentiable with respect to u and d, respectively (see Remark 9 below). The specific system of equations consists of:

1. the first-order optimality conditions for the unconstrained minimizations in $(4.10c)$[2]:

$$\nabla_u L_f(\hat{u}, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) = \mathbf{0}_{N_u}, \qquad\qquad (4.13a)$$

$$\nabla_d L_g(\hat{u}, \hat{d}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) = \mathbf{0}_{N_d}; \qquad\qquad (4.13b)$$

2. the equality conditions (4.10a); and

3. the equations[3]

$$F_u(\hat{u}, \hat{d}) \odot \hat{\lambda}_{fu} = \mu \mathbf{1}_{M_u}, \qquad\qquad (4.14a)$$

$$F_d(\hat{u}, \hat{d}) \odot \hat{\lambda}_{gd} = \mu \mathbf{1}_{M_d}, \qquad\qquad (4.14b)$$

---

[2]Given an integer M, we denote by $\mathbf{0}_M$ and by $\mathbf{1}_M$ the M-vectors with all entries equal to 0 and 1, respectively.

[3]Given two vectors $x, y \in \mathbb{R}^n$ we denote by $x \odot y \in \mathbb{R}^n$ and by $x \oslash y \in \mathbb{R}^n$ the entry-wise product and division of the two vectors, respectively.

for some $\mu > 0$, which leads to

$$\epsilon_f = M_u\mu, \qquad\qquad\qquad \epsilon_g = M_d\mu.$$

Since our goal is to find primal variables $\hat{u}, \hat{d}$ for which (4.11) holds with $\epsilon_f = \epsilon_g = 0$, we shall make the variable μ converge to zero as the Newton iterations progress. This is done in the context of an interior-point method, meaning that all variables are initialized so that the inequality constraints (4.10b) hold, and the progression along the Newton direction at each iteration is selected so that these constraints are never violated.

The specific steps of the algorithm that follows are based on the primal-dual interior-point method for a single optimization, as described in [107]. To describe this algorithm, we define

$$z := [\hat{u}' \ \hat{d}']', \quad \lambda := [\hat{\lambda}'_{fu} \ \hat{\lambda}'_{gd}]', \quad \nu := [\hat{\nu}'_{fu} \ \hat{\nu}'_{gd}]',$$

$$G(z) := \begin{bmatrix} G_u(\hat{u}, \hat{d}) \\ G_d(\hat{u}, \hat{d}) \end{bmatrix}, \quad F(z) := \begin{bmatrix} F_u(\hat{u}, \hat{d}) \\ F_d(\hat{u}, \hat{d}) \end{bmatrix},$$

which allows us to re-write (4.13), (4.10a), and (4.14) as

$$\nabla_u L_f(z, \lambda, \nu) = \mathbf{0}_{N_u}, \quad \nabla_d L_g(z, \lambda, \nu) = \mathbf{0}_{N_d}, \tag{4.15a}$$

$$G(z) = \mathbf{0}_{K_u+K_d}, \qquad \lambda \odot F(z) = \mu \mathbf{1}_{M_u+M_d}, \tag{4.15b}$$

and (4.10b) as

$$\lambda \geqslant \mathbf{0}_{M_u+M_d}, \qquad F(z) \geqslant \mathbf{0}_{M_u+M_d}. \tag{4.15c}$$

Primal-dual optimization algorithm:

Step 1. Start with estimates z_0, λ_0, ν_0 that satisfy the inequalities $\lambda_0 \geqslant \mathbf{0}_{M_u+M_d}$, $F(z_0) \geqslant$ $\mathbf{0}_{M_u+M_d}$ in (4.15c), and set $\mu_0 = 1$ and $k = 0$. It is often a good idea to start with estimates that also satisfy the equality constraint $G(z_0) = \mathbf{0}_{K_u+K_d}$, and $\lambda_0 = \mu_0 \mathbf{1}_{M_u+M_d} \oslash F(z_0)$, which guarantees that we initially have $\lambda_0 \odot F(z_0) = \mu_0 \mathbf{1}_{M_u+M_d}$.

Step 2. Linearize the equations in (4.15a) around a current estimate z_k, λ_k, ν_k, and μ_k leading to

$$
\begin{bmatrix}
\nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\lambda}L_f(z_k,\lambda_k,\nu_k) \\
\nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\lambda}L_g(z_k,\lambda_k,\nu_k) \\
\nabla_z G(z_k) & 0 & 0 \\
\mathrm{diag}(\lambda_k)\nabla_z F(z_k) & 0 & \mathrm{diag}[F(z_k)]
\end{bmatrix}
\begin{bmatrix}
\Delta z \\
\Delta \nu \\
\Delta \lambda
\end{bmatrix}
$$

$$
= -
\begin{bmatrix}
\nabla_u L_f(z_k,\lambda_k,\nu_k) \\
\nabla_d L_g(z_k,\lambda_k,\nu_k) \\
G(z_k) \\
F(z_k) \odot \lambda_k - \mu_k \mathbf{1}
\end{bmatrix}. \quad (4.16)
$$

Since the vectors $F(z_k)$ and λ_k have positive entries, we can solve this system of equations by first eliminating

$$
\mathrm{diag}(\lambda_k)\nabla_z F(z_k)\Delta z + \mathrm{diag}[F(z_k)]\Delta\lambda = -F(z_k)\odot\lambda_k + \mu_k\mathbf{1} \quad \Leftrightarrow
$$

$$
\Delta\lambda = -\lambda_k - \mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z + \mu_k \mathbf{1}\oslash F(z_k) \quad (4.17a)
$$

which leads to

$$\begin{bmatrix} \nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k) \\ \nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix}\begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda}L_f(z_k)\Delta\lambda \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda}L_g(z_k)\Delta\lambda \\ G(z_k) \end{bmatrix}.$$

$$\text{(4.17b)}$$

Step 3. For additional computational efficiency, find an *affine scaling direction* $\begin{bmatrix} \Delta z_a' & \Delta \nu_a' & \Delta \lambda_a' \end{bmatrix}'$ by solving (4.17) for $\mu_k = 0$:

$$\Delta\lambda_a = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z_a,$$

$$\begin{bmatrix} \nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k) \\ \nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix}\begin{bmatrix} \Delta z_a \\ \Delta \nu_a \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda}L_f(z_k)\Delta\lambda_a \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda}L_g(z_k)\Delta\lambda_a \\ G(z_k) \end{bmatrix}.$$

Step 4. Select scalings so that the inequalities in (4.15c) are not violated along the affine scaling direction:

$$\alpha_a := \min\{\alpha_{\text{primal}}^a, \alpha_{\text{dual}}^a\},$$

where

$$\alpha_{\text{primal}}^a := \max\left\{\alpha \in [0,1] : F(z_k + \alpha\Delta z_a) \geqslant 0\right\},$$

$$\alpha_{\text{dual}}^a := \max\left\{\alpha \in [0,1] : \lambda_k + \alpha\Delta\lambda_a \geqslant 0\right\}.$$

Define the following estimate for the "quality" of the affine scaling direction

$$\sigma := \left(\frac{F(z_k + \alpha_a \Delta z_a) \odot (\lambda_k + \alpha_a \Delta \lambda_a)}{F(z_k) \odot \lambda_k} \right)^{\delta},$$

where δ is a parameter typically selected equal to 2 or 3. Note that the numerator $F(z_k + \alpha_a \Delta z_a) \odot (\lambda_k + \alpha_a \Delta \lambda_a)$ is the value one would obtain for $F(z) \odot \lambda$ by moving purely along the affine scaling directions. A small value for σ thus indicates that a significant reduction in μ_k is possible.

Step 5. Find the *search direction* $\begin{bmatrix} \Delta z_s' & \Delta \nu_s' & \Delta \lambda_s' \end{bmatrix}'$ by solving (4.16) for $\mu_k = \sigma \frac{F(z_k) \odot \lambda_k}{M_u + M_d}$:

$$\begin{bmatrix} \nabla_{uz} L_f(z_k, \lambda_k, \nu_k) & \nabla_{u\nu} L_f(z_k) & \nabla_{u\lambda} L_f(z_k) \\ \nabla_{dz} L_g(z_k, \lambda_k, \nu_k) & \nabla_{d\nu} L_g(z_k) & \nabla_{d\lambda} L_g(z_k) \\ \nabla_z G(z_k) & 0 & 0 \\ \mathrm{diag}(\lambda_k) \nabla_z F(z_k) & 0 & \mathrm{diag}[F(z_k)] \end{bmatrix} \begin{bmatrix} \Delta z_s \\ \Delta \nu_s \\ \Delta \lambda_s \end{bmatrix}$$

$$= - \begin{bmatrix} \nabla_u L_f(z_k, \lambda_k, \nu_k) \\ \nabla_d L_g(z_k, \lambda_k, \nu_k) \\ G(z_k) \\ F(z_k) \odot \lambda_k + (\nabla_z F(z_k) \Delta z_a) \odot \Delta \lambda_a - \sigma \frac{F(z_k) \odot \lambda_k}{M_u + M_d} \mathbf{1} \end{bmatrix},$$

where the additional term $(\nabla_z F(z_k) \Delta z_a) \odot \Delta \lambda_a$ comes from a 2nd order expansion of the left-hand side of the last equality in (4.15a) [65]. Since the vectors $F(z_k)$ and λ_k have positive entries, we can solve this system of equations by first eliminating

$$\mathrm{diag}(\lambda_k) \nabla_z F(z_k) \Delta z_s + \mathrm{diag}[F(z_k)] \Delta \lambda_s$$

$$= -F(z_k) \odot \lambda_k - (\nabla_z F(z_k) \Delta z_a) \odot \Delta \lambda_a + \sigma \frac{F(z_k) \odot \lambda_k}{M_u + M_d} \mathbf{1} \quad \Leftrightarrow$$

$$\Delta\lambda_s = -\lambda_k - (\nabla_z F(z_k)\Delta z_a) \odot \Delta\lambda_a \oslash F(z_k) - \mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z_s$$

$$+ \sigma\frac{F(z_k) \odot \lambda_k}{M_u + M_d}\mathbf{1} \oslash F(z_k)$$

which leads to

$$\begin{bmatrix} \nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k) \\ \nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix}\begin{bmatrix} \Delta z_s \\ \Delta\nu_s \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda}L_f(z_k)\Delta\lambda_s \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda}L_g(z_k)\Delta\lambda_s \\ G(z_k) \end{bmatrix}.$$

Step 6. Update the estimates along the search direction so that the inequalities in (4.15c) hold strictly:

$$z_{k+1} = z_k + \alpha_s\Delta z_s, \qquad \nu_{k+1} = \nu_k + \alpha_s\Delta\nu_s, \qquad \lambda_{k+1} = \lambda_k + \alpha_s\Delta\lambda_s$$

where $\alpha_s := 0.99 \times \min\{\alpha_{\mathrm{primal}}, \alpha_{\mathrm{dual}}\}$, and

$$\alpha_{\mathrm{primal}} := \max\big\{\alpha \in [0,1] : F(z_k + \alpha\Delta z_s) \geqslant 0\big\},$$

$$\alpha_{\mathrm{dual}} := \max\big\{\alpha \in [0,1] : \lambda_k + \alpha\Delta\lambda_s \geqslant 0\big\}.$$

Step 7. Repeat from Step 2 with an incremented value for k until

$$\|\nabla_u L_f(z_k,\lambda_k,\nu_k)\| \leqslant \epsilon_u, \quad \|\nabla_d L_g(z_k,\lambda_k,\nu_k)\| \leqslant \epsilon_d, \tag{4.18a}$$

$$\|G(z_k)\| \leqslant \epsilon_G, \quad \lambda_k \odot F(z_k) \leqslant \epsilon_{\mathrm{gap}}, \tag{4.18b}$$

for sufficiently small tolerances ϵ_u, ϵ_d, ϵ_G, ϵ_{gap}. $\square$

When the functions L_f and L_g that appear in the unconstrained minimizations in (4.10c) have a single stationary point that corresponds to their global minimum, termination of the Algorithm 4.3.1 guarantees that the assumptions of Lemma 3 hold [up to the tolerances in (4.18)], and we obtain the desired solution to (4.8).

The desired uniqueness of the stationary point holds, e.g., when the function $f(u, d)$ is convex in u, $g(u, d)$ is convex in d, $F_u(u, d)$ is concave in u, $F_d(u, d)$ is concave in d, and $G_u(u, d)$ is linear in u, and $G_d(u, d)$ is linear in d. However, in practice the Algorithm 4.3.1 can find solutions to (4.8) even when these convexity assumptions do not hold. For problems for which one cannot be sure whether the Algorithm 4.3.1 terminated at a global minimum of the unconstrained problem, one may run several instances of the algorithm with random initial conditions. Consistent results for the optimizations across multiple initializations provides an indication that a global minimum has been found.

Remark 9 (Smoothness) *Algorithm 4.3.1 requires all the functions f, g, F_u, F_d, G_u, G_d to be twice differentiable for the computation of the matrices that appear in (4.16). However, this does not preclude the use of this algorithm in many problems where these functions are not differentiable because it is often possible to re-formulate non-smooth optimizations into smooth ones by appropriate transformations that often introduce additional optimization variables. Common examples of these transformations include the minimization of criteria involving ℓ_p norms, such as the "non-differentiable ℓ_1 optimization"*

$$\min \left\{ \|A_{m \times n} x - b\|_{\ell_1} + \cdots : x \in \mathbb{R}^n, \ldots \right\}$$

which is equivalent to the following constrained smooth optimization

$$\min \left\{ v' \mathbf{1}_m + \cdots : x \in \mathbb{R}^n, v \in \mathbb{R}^m, -v \leqslant Ax - b \leqslant v, \ldots \right\};$$

and the "non-differentiable ℓ_2 optimization"

$$\min \left\{ \|A_{m \times n} x - b\|_{\ell_2} + \cdots : x \in \mathbb{R}^n, \ldots \right\}$$

which is equivalent to

$$\min \left\{ v + \cdots : x \in \mathbb{R}^n, v \geqslant 0, v^2 \geqslant (Ax - b)'(Ax - b), \ldots \right\}.$$

More examples of such transformations can be found, e.g., in [41, 81, 33, 55, 36]. □

4.4 Interior-Point Method for Minimax Problems with Common Latent Variables

In this section, we develop an algorithm very similar to Algorithm 4.3.1 that takes advantage of the fact that the equality constraints in (4.2), ensuring that the dynamics (4.1) are satisfied, are repeated for both the minimizer and the maximizer. More generally, we consider the case where the minimization and the maximization in minimax problems both contain common optimization variables that can instead be incorporated as latent variables.

Like in Section 4.3, our goal is to find a pair $(u^*, d^*) \in \bar{\mathcal{U}}[d^*] \times \bar{\mathcal{D}}[u^*]$ that simultaneously solves two coupled optimizations

$$\bar{f}(u^*, d^*) = \min_{u \in \bar{\mathcal{U}}[d^*]} \bar{f}(u, d^*), \qquad \bar{g}(u^*, d^*) = \min_{d \in \bar{\mathcal{D}}[u^*]} \bar{g}(u^*, d), \qquad (4.19)$$

with

$$\bar{\mathcal{U}}[d] := \{u \in \mathbb{R}^{N_u} : \bar{F}_u(u, d) \geqslant 0, \ \bar{G}_u(u, d) = 0\}, \tag{4.20a}$$

$$\bar{\mathcal{D}}[u] := \{d \in \mathbb{R}^{N_d} : \bar{F}_d(u, d) \geqslant 0, \ \bar{G}_d(u, d) = 0\}, \tag{4.20b}$$

for given functions $\bar{f} : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $\bar{g} : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $\bar{F}_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_u}$, $\bar{F}_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_d}$, $\bar{G}_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_u}$, $\bar{G}_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_d}$. However, we are now interested in cases where these functions can be expressed in terms of common *latent variables*. Specifically, these functions can be expressed as

$$\bar{f}(u, d) = f\big(u, d, \chi(u, d)\big), \qquad\qquad \bar{g}(u, d) = g\big(u, d, \chi(u, d)\big),$$

$$\bar{F}_u(u, d) = F_u\big(u, d, \chi(u, d)\big), \qquad\qquad \bar{G}_u(u, d) = G_u\big(u, d, \chi(u, d)\big),$$

$$\bar{F}_d(u, d) = F_d\big(u, d, \chi(u, d)\big), \qquad\qquad \bar{G}_d(u, d) = G_d\big(u, d, \chi(u, d)\big),$$

$\forall u \in \mathbb{R}^{N_u}, d \in \mathbb{R}^{N_d}$, for a function $\chi : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{N_x}$ whose value $\chi(u, d)$ is defined implicitly by a function $H : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_x} \to \mathbb{R}^{K_x}$ and an equation of the form

$$H(u, d, x) = 0. \tag{4.21}$$

The function H is assumed to be such that (4.21) has a unique solution x for every $u \in \mathbb{R}^{N_u}, d \in \mathbb{R}^{N_d}$. Corollary 4 (a corollary of Lemma 3) given below is useful in situations where it is difficult (or impossible) to find an explicit form for χ.

4.4.1   Primal-dual method

The following duality-like result provides the motivation for a primal-dual-like method to solve the coupled minimizations in (4.19). It provides a set of conditions, involving two

unconstrained optimizations, that provide an approximation to the solution of (4.19). It improves upon a direct application of Lemma 3 to (4.19), in that the implictely defined function χ does not appear in the conditions.

Corollary 4 (Approximate equilibrium) *Consider the coupled optimizations in* (4.19) *and assume that for every* $u \in \mathbb{R}^{N_u}$, $d \in \mathbb{R}^{N_d}$, *the equation* (4.21) *has a unique solution* x. *Suppose that we have found* primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}, \hat{x} \in \mathbb{R}^{N_x}$ and dual variables $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{fx} \in \mathbb{R}^{K_x}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ *that simultaneously satisfy all of the following conditions*

$$G_u(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad G_d(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad H(\hat{u}, \hat{d}, \hat{x}) = 0, \tag{4.22a}$$

$$\hat{\lambda}_{fu} \dot{\geqslant} 0, \qquad \hat{\lambda}_{gd} \dot{\geqslant} 0, \qquad F_u(\hat{u}, \hat{d}, \hat{x}) \geqslant 0, \qquad F_d(\hat{u}, \hat{d}, \hat{x}) \geqslant 0, \tag{4.22b}$$

$$L_f(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \min_{\substack{u \in \mathbb{R}^{N_u}, \\ x \in \mathbb{R}^{N_x}}} L_f(u, \hat{d}, x, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}), \tag{4.22c}$$

$$L_g(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \min_{\substack{d \in \mathbb{R}^{N_d}, \\ x \in \mathbb{R}^{N_x}}} L_g(\hat{u}, d, x, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) \tag{4.22d}$$

where

$$L_f(u, d, x, \lambda_{fu}, \nu_{fu}, \nu_{fx}) \coloneqq f(u, d, x) - \lambda'_{fu} F_u(u, d, x) + \nu'_{fu} G_u(u, d, x) + \nu'_{fx} H(u, d, x),$$

$$L_g(u, d, x, \lambda_{gd}, \nu_{gd}, \nu_{gx}) \coloneqq g(u, d, x) - \lambda'_{gd} F_d(u, d, x) + \nu'_{gd} G_d(u, d, x) + \nu'_{gx} H(u, d, x).$$

Then $(\hat{u}, \hat{d})$ *approximately satisfy* (4.19) *in the sense that*

$$\bar{f}(\hat{u}, \hat{d}) \leqslant \epsilon_f + \min_{u \in \bar{\mathcal{U}}[\hat{d}]} \bar{f}(u, \hat{d}), \qquad\qquad \bar{g}(\hat{u}, \hat{d}) \leqslant \epsilon_g + \min_{d \in \bar{\mathcal{D}}[\hat{u}]} \bar{g}(\hat{u}, d), \tag{4.23}$$

with

$$\epsilon_f := \hat{\lambda}'_{fu} F_u(\hat{u}, \hat{d}, \hat{x}), \qquad\qquad \epsilon_g := \hat{\lambda}'_{gd} F_d(\hat{u}, \hat{d}, \hat{x}). \qquad \square$$

Note that while Corollary 4 utilizes a single primal (latent) variable $\hat{x}$, it requires two dual variables $\hat{\nu}_{fx}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ associated with the equality constraint $H(\hat{u}, \hat{d}, \hat{x}) = 0$.

Proof of Corollary 4. Since the equation in (4.21) has a unique solution x, the optimizations in (4.19), can be re-written as

$$\bar{f}(u^*, d^*) = \min_{(u,x) \in \mathcal{U}[d^*]} f(u, d^*, x), \qquad \bar{g}(u^*, d^*) = \min_{(d,z) \in \mathcal{D}[u^*]} g(u^*, d, z), \qquad (4.24)$$

with

$$\mathcal{U}[d] := \{(u, x) \in \mathbb{R}^{N_u} \times \mathbb{R}^{N_x} : F_u(u, d, x) \geqslant 0, \; G_u(u, d, x) = 0, \; H(u, d, x) = 0\}, \quad (4.25)$$

$$\mathcal{D}[u] := \{(d, z) \in \mathbb{R}^{N_d} \times \mathbb{R}^{N_x} : F_d(u, d, z) \geqslant 0, \; G_d(u, d, z) = 0, \; H(u, d, z) = 0\}, \quad (4.26)$$

which is again of the form considered in Section 4.3, but for optimization variables (u, x) and (d, z) in higher dimensional spaces.

Applying Lemma 3 to the new formulation in (4.24), we conclude that if we find *primal variables* $\hat{u} \in \mathbb{R}^N, \hat{x} \in \mathbb{R}^N, \hat{d} \in \mathbb{R}^{N_d}, \hat{z} \in \mathbb{R}^{N_x}$ and *dual variables* $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}$, $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{fx} \in \mathbb{R}^{K_x}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ that simultaneously satisfy all of the following conditions

$$G_u(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad H(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad G_d(\hat{u}, \hat{d}, \hat{z}) = 0, \qquad H(\hat{u}, \hat{d}, \hat{z}) = 0, \quad (4.27a)$$

$$\hat{\lambda}_{fu} \dot{\geqslant} 0, \qquad \hat{\lambda}_{gd} \dot{\geqslant} 0, \qquad F_u(\hat{u}, \hat{d}, \hat{x}) \geqslant 0, \qquad F_d(\hat{u}, \hat{d}, \hat{z}) \geqslant 0, \qquad (4.27b)$$

$$L_f(\hat{u}, \hat{x}, \hat{d}, \hat{z}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \min_{\substack{u \in \mathbb{R}^{N_u}, \\ x \in \mathbb{R}^{N_x}}} L_f(u, x, \hat{d}, \hat{z}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}), \tag{4.27c}$$

$$L_g(\hat{u}, \hat{x}, \hat{d}, \hat{z}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \min_{d \in \mathbb{R}^{N_d}, z \in \mathbb{R}^{N_x}} L_g(\hat{u}, \hat{x}, d, z, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) \tag{4.27d}$$

where

$$L_f(u, x, d, z, \lambda_{fu}, \nu_{fu}, \nu_{fx}) := f(u, d, x) - \lambda'_{fu} F_u(u, d, x) + \nu'_{fu} G_u(u, d, x) + \nu'_{fx} H(u, d, x),$$

$$L_g(u, x, d, z, \lambda_{gd}, \nu_{gd}, \nu_{gx}) := g(u, d, z) - \lambda'_{gd} F_d(u, d, z) + \nu'_{gd} G_d(u, d, z) + \nu'_{gx} H(u, d, z);$$

then

$$\bar{f}(\hat{u}, \hat{d}) = f(\hat{u}, \hat{d}, \hat{x}) \leqslant \epsilon_f + \min_{(u,x) \in \mathcal{U}[\hat{d}]} f(u, \hat{d}, x) = \epsilon_f + \min_{u \in \bar{\mathcal{U}}[\hat{d}]} \bar{f}(u, \hat{d}),$$

$$\bar{g}(\hat{u}, \hat{d}) = g(\hat{u}, \hat{d}, \hat{z}) \leqslant \epsilon_g + \min_{(d,z) \in \mathcal{D}[\hat{u}]} g(\hat{u}, d, z) = \epsilon_g + \min_{d \in \bar{\mathcal{D}}[\hat{u}]} \bar{g}(\hat{u}, d)$$

with

$$\epsilon_f := \hat{\lambda}'_{fu} F_u(\hat{u}, \hat{d}, \hat{x}), \qquad\qquad \epsilon_g := \hat{\lambda}'_{gd} F_d(\hat{u}, \hat{d}, \hat{z}).$$

The result follows from this, together with the observation that $\hat{x} = \hat{z}$ because the equations $H(\hat{u}, \hat{d}, \hat{x}) = 0$ and $H(\hat{u}, \hat{d}, \hat{z}) = 0$ in (4.27a) must have exactly the same solution $\hat{x} = \hat{z}$. ∎

4.4.2 Interior-point primal-dual equilibria algorithm

As before, this new proposed method consists of using Newton iterations to solve a system of nonlinear equations on the primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}, \hat{x} \in \mathbb{R}^{N_x}$ and dual variables $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}, \hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{fx} \in \mathbb{R}^{K_x}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ introduced in

Corollary 4. The specific system of equations consists of:

1. the first-order optimality conditions for the unconstrained minimizations in (4.22c)–(4.22d):

$$\nabla_u L_f(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \mathbf{0}_{N_u}, \quad \nabla_x L_f(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \mathbf{0}_{N_u}, \qquad \text{(4.28a)}$$

$$\nabla_d L_g(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \mathbf{0}_{N_d}, \quad \nabla_x L_g(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \mathbf{0}_{N_d}; \qquad \text{(4.28b)}$$

2. the equality conditions (4.22a); and

3. the equations

$$F_u(\hat{u}, \hat{d}, \hat{x}) \odot \hat{\lambda}_{fu} = \mu \mathbf{1}_{M_u}, \qquad F_d(\hat{u}, \hat{d}, \hat{x}) \odot \hat{\lambda}_{gd} = \mu \mathbf{1}_{M_d}, \qquad \text{(4.29)}$$

for some $\mu > 0$, which leads to

$$\epsilon_f = M_u \mu, \qquad\qquad \epsilon_g = M_d \mu.$$

Since our goal is to find primal variables $\hat{u}, \hat{d}, \hat{x}$ for which (4.23) holds with $\epsilon_f = \epsilon_g = 0$, we shall make the variable μ converge to zero as the Newton iterations progress. This is done in the context of an interior-point method, meaning that all variables are initialized so that the inequality constraints (4.27b) hold and the progression along the Newton direction at each iteration are selected so that these constraints are never violated.

The specific steps of the algorithm that follows are also inspired by the primal-dual interior-point method for a single optimization, as described in [107]. To describe this

algorithm, we define

$$
z := \begin{bmatrix} \hat{u} \\ \hat{d} \\ \hat{x} \end{bmatrix}, \quad \lambda := \begin{bmatrix} \hat{\lambda}_{fu} \\ \hat{\lambda}_{gd} \end{bmatrix}, \quad \nu := \begin{bmatrix} \hat{\nu}_{fu} \\ \hat{\nu}_{fx} \\ \hat{\nu}_{gd} \\ \hat{\nu}_{gx} \end{bmatrix}, \quad G(z) := \begin{bmatrix} G_u(\hat{u}, \hat{d}, \hat{x}) \\ G_d(\hat{u}, \hat{d}, \hat{x}) \\ H(\hat{u}, \hat{d}, \hat{x}) \end{bmatrix}, \quad F(z) := \begin{bmatrix} F_u(\hat{u}, \hat{d}) \\ F_d(\hat{u}, \hat{d}) \end{bmatrix},
$$

which allows us to re-write (4.28), (4.22a), and (4.29) as

$$
\nabla_u L_f(z, \lambda, \nu) = \mathbf{0}_{N_u}, \qquad \nabla_x L_f(z, \lambda, \nu) = \mathbf{0}_{N_x}, \tag{4.30a}
$$

$$
\nabla_d L_g(z, \lambda, \nu) = \mathbf{0}_{N_d}, \qquad \nabla_x L_g(z, \lambda, \nu) = \mathbf{0}_{N_x}, \tag{4.30b}
$$

$$
G(z) = \mathbf{0}_{K_u + K_d + K_x}, \qquad \lambda \odot F(z) = \mu \mathbf{1}_{M_u + M_d}, \tag{4.30c}
$$

and (4.22b) as

$$
\lambda \geqslant \mathbf{0}_{M_u + M_d}, \qquad F(z) \geqslant \mathbf{0}_{M_u + M_d}. \tag{4.30d}
$$

Primal-dual optimization algorithm with common latent variables:

Step 1. Start with estimates z_0, λ_0, ν_0 that satisfy the inequalities $\lambda_0 \geqslant \mathbf{0}_{M_u + M_d}$, $F(z_0) \geqslant$ $\mathbf{0}_{M_u + M_d}$ in (4.30d), and set $\mu_0 = 1$ and $k = 0$. It is often a good idea to start with a value for z_0 that satisfies the equality constraint $G(z_0) = \mathbf{0}_{K_u + K_d + K_x}$, and $\lambda_0 = \mu_0 \mathbf{1}_{M_u + M_d} \oslash$ $F(z_0)$, which guarantees that we initially have $\lambda_0 \odot F(u_0) = \mu_0 \mathbf{1}_{M_u + M_d}$.

Step 2. Linearize the equations in (4.30a)–(4.30c) around a current estimate z_k, λ_k, ν_k,

and μ_k leading to

$$
\begin{bmatrix}
\nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k) & \nabla_{u\lambda}L_f(z_k) \\
\nabla_{xz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{x\nu}L_f(z_k) & \nabla_{x\lambda}L_f(z_k) \\
\nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k) & \nabla_{d\lambda}L_g(z_k) \\
\nabla_{xz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{x\nu}L_g(z_k) & \nabla_{x\lambda}L_g(z_k) \\
\nabla_z G(z_k) & 0 & 0 \\
\nabla_z F(z_k) & 0 & \mathrm{diag}[F(z_k)\oslash\lambda_k]
\end{bmatrix}
\begin{bmatrix}
\Delta z \\
\Delta\nu \\
\Delta\lambda
\end{bmatrix}
= -
\begin{bmatrix}
\nabla_u L_f(z_k,\lambda_k,\nu_k) \\
\nabla_x L_f(z_k,\lambda_k,\nu_k) \\
\nabla_d L_g(z_k,\lambda_k,\nu_k) \\
\nabla_x L_g(z_k,\lambda_k,\nu_k) \\
G(z_k) \\
F(z_k) - \mu_k\mathbf{1}\oslash\lambda_k
\end{bmatrix}.
$$

Since the vectors $F(z_k)$ and λ_k have positive entries, we can solve this system of equations by first eliminating

$$
\nabla_z F(z_k)\Delta z + \mathrm{diag}[F(z_k)\oslash\lambda_k]\Delta\lambda = -F(z_k) + \mu_k\mathbf{1}\oslash\lambda_k \quad\Leftrightarrow
$$

$$
\Delta\lambda = -\lambda_k - \mathrm{diag}[\lambda_k\oslash F(z_k)]\nabla_z F(z_k)\Delta z + \mu_k\mathbf{1}\oslash F(z_k)
$$

which leads to

$$
\begin{bmatrix}
\nabla_{uz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu}L_f(z_k) \\
\nabla_{xz}L_f(z_k,\lambda_k,\nu_k) & \nabla_{x\nu}L_f(z_k) \\
\nabla_{dz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu}L_g(z_k) \\
\nabla_{xz}L_g(z_k,\lambda_k,\nu_k) & \nabla_{x\nu}L_g(z_k) \\
\nabla_z G(z_k) & 0
\end{bmatrix}
\begin{bmatrix}
\Delta z \\
\Delta\nu
\end{bmatrix}
= -
\begin{bmatrix}
\nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda}L_f(z_k)\Delta\lambda \\
\nabla_x L_f(z_k,\lambda_k,\nu_k) + \nabla_{x\lambda}L_f(z_k)\Delta\lambda \\
\nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda}L_g(z_k)\Delta\lambda \\
\nabla_x L_g(z_k,\lambda_k,\nu_k) + \nabla_{x\lambda}L_g(z_k)\Delta\lambda \\
G(z_k)
\end{bmatrix}
$$

We omit the remaining steps as they are identical to those in Algorithm 4.3.1. Similarly, the discussions regarding finding a global solution and considering non-differentiable functions that follow Algorithm 4.3.1 also apply here. Algorithm 4.4.2 works very well in practice, and it is used for solving all of the numerical examples in this thesis.

4.5   Conclusions and Future Work

In this chapter we presented two primal-dual interior-point algorithms that can be used to solve min-max optimization problems by solving two coupled optimizations. The first algorithm includes values of the state as optimization variables so as to not require the solution of the potentially poorly conditioned nonlinear dynamics (4.1) and can be used to solve general nonlinear min-max optimization problems. The second algorithm utilizes latent variables that are present in both of the coupled optimizations in order to reduce the total number of optimization variables and solve min-max problems with common latent variables more efficiently. This second algorithm is particularly useful when considering min-max problems as formulated in Chapter 2 as the state variables can be included as latent variables.

Future work may involve a convergence analysis of Algorithms 4.3.1 and 4.4.2. The development of similar algorithms to solve these types of optimization problems and trade offs between methods should be investigated. For example, a Barrier interior-point algorithm could be developed which may be more robust than the primal-dual algorithm for non-convex poorly conditioned problems.

Chapter 5

Adaptation and Learning

This chapter considers the estimation and control of systems with parametric uncertainty. An approach that combines moving horizon estimation and model predictive control into a single min-max optimization is employed to estimate past and current values of the state, compute a sequence of optimal future control inputs, predict future values of the state, as well as estimate current values of uncertain parameters. This is done by including the state, inputs, and uncertain parameters as optimization variables. Learning the true values of the uncertain parameters requires a sufficiently large number of past measurements and that the system is persistently excited. The true values of the uncertain parameters may change over time, and the optimization computes future control inputs that adapt to changing estimates of the uncertain parameters in order to better control the uncertain system. Several linear and nonlinear examples with parametric uncertainty are discussed and effectively controlled using this combined moving horizon estimation and model predictive control approach. Some of the content in this chapter comes from [23].

5.1   Introduction

Having an accurate model of a system to be controlled is often vital for effective control of that system. This is certainly true for a model predictive control (MPC) approach in which a finite-horizon online optimization problem is solved in order to determine an optimal control input given the system's dynamics and a desired control objective [95]. However, in most practical applications, there are unknown parameters in the model of a system or, at least, uncertain parameters that are known only to be within some set of values. These uncertainties may include uncertain model parameters, input disturbances, and measurement noise. Because of this, much work on MPC approaches have involved investigating robustness to model parameter uncertainty, input disturbances, and measurement noise. This work is known as robust MPC [9, 67] which also includes worst-case, or min-max MPC [58].

An attractive, and perhaps less conservative, approach to controlling systems with parameter uncertainties is to update the model of the system with new estimates of the parameters as they become available, which is the underlying idea behind indirect adaptive control (see, e.g. [51, 5]). Very little work has been done on adaptive MPC, but there are a few proposed approaches. The authors of [73] propose an adaptive MPC scheme that uses a standard estimator and certainty equivalence to update the model with the current estimates of the parameters. The authors of [2] investigate nonlinear systems that are affine with respect to unknown parameters and perform adaptive control by combining a parameter adjustment mechanism with robust MPC algorithms such as min-max MPC. A cost function is minimized with respect to feedback control policies and maximized with respect to the unknown parameters so that the MPC approach is robust to the worst-case values of the unknown parameters. For both of these approaches, it is assumed that the full state is available for feedback. This is often the case for

MPC approaches in order to alleviate issues that arise from uncertainties, noise, and disturbances.

Unfortunately, in most practical applications, the full state is not known or necessarily available for feedback. Because of this, output-feedback MPC should be considered, and an independent algorithm for estimating the state is needed. A convenient estimation algorithm for use with MPC is moving-horizon estimation (MHE). MHE can be used for estimating the state of constrained nonlinear systems and similarly involves the solution of a finite-horizon online optimization problem where a criterion based on a finite number of past output measurements is minimized in order to find the best estimate of the state [90, 3]. It is straightforward to also incorporate parameter estimation into the formulation of MHE, so the state and parameters can both be estimated using the same estimator [96].

In [22, 25], a framework for solving the output-feedback MPC problem with MHE is presented that solves both the control and estimation problems as a single min-max optimization problem. This framework already incorporates input disturbances and measurement noise. In this chapter, we further incorporate uncertain model parameters into this framework and obtain parameter estimates by including the uncertain parameters as optimization variables. In this way, we solve simultaneously the control problem and the state and parameter estimation problems, resulting in effective control of uncertain systems. Our approach can be likened to an indirect model reference adaptive control approach as described in the adaptive control literature [51, 5] in that, at each time step, new estimates of the uncertain parameters are computed and used to update the model while a new sequence of future control inputs that minimize an objective criterion is also simultaneously computed.

Because MPC and MHE involve the solution of an online optimization problem, this approach lends itself to adapting to both constant and time-varying parameters

because, at each time step, a new estimate is computed, and the model can be updated accordingly. We show in examples that when the system is sufficiently excited, this MPC with MHE approach is able to learn the true values of the uncertain parameters, but even if the system is not sufficiently excited to learn the true values of the parameters, this approach finds estimates that are consistent with the dynamics and often still enables effective control and disturbance rejection.

The main assumption for this work is that a saddle-point solution exists for the min-max optimization problem at each sampling time. This assumption presumes an appropriate form of observability for the closed-loop system and is a common requirement in game theoretical approaches to control design [6]. For controllability, we additionally require that there exists a terminal cost that is an ISS-control Lyapunov function with respect to a disturbance input, which is a common assumption in MPC [76].

The rest of the chapter is organized as follows. In Section 5.2 we formulate the adaptive MPC with MHE problem that we would like to solve. Stability results that can be used to prove state boundedness and reference tracking are given in Section 5.3. In Section 5.4 we discuss several linear and nonlinear systems with parameter uncertainty and show that using our MPC with MHE scheme we are able to not only stabilize the system, but also estimate the correct values of the uncertain parameters. Finally, we provide some conclusions and directions for future work in Section 5.5.

5.2   Problem Formulation

In the formulation of standard MPC and MHE problems, a time-varying nonlinear discrete-time process of the form

$$x_{t+1} = f_t(x_t, u_t, d_t), \qquad\qquad y_t = g_t(x_t) + n_t, \qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0} \qquad (5.1)$$
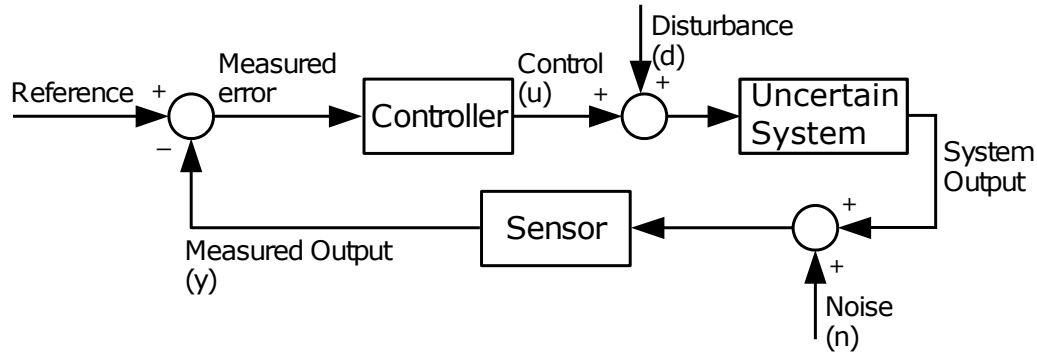
Figure 5.1: Block diagram of the process given in (5.2).

is considered with *state* x_t taking values in a set $\mathcal{X} \subset \mathbb{R}^{n_x}$. The inputs to this system are the *control input* u_t that must be restricted to the set $\mathcal{U} \subset \mathbb{R}^{n_u}$, the *unmeasured disturbance* d_t that is assumed to belong to the set $\mathcal{D} \subset \mathbb{R}^{n_d}$, and the *measurement noise* $n_t \in \mathbb{R}^{n_n}$. The signal $y_t \in \mathbb{R}^{n_y}$ denotes the *measured output* that is available for feedback.

In this chapter, we investigate MPC and MHE of processes with uncertain model parameters. These uncertain parameters are denoted by the vector θ whose elements are known to belong to the set $\Theta \subset \mathbb{R}^{n_\theta}$. In this formulation, the process dynamics depend explicitly on the uncertain parameter θ, so we redefine the process dynamics in (5.1) to include the uncertain parameters as

$$x_{t+1} = f_t(x_t, \theta, u_t, d_t), \qquad y_t = g_t(x_t, \theta) + n_t, \qquad \forall t \in \mathbb{Z}_{\geqslant 0}. \qquad (5.2)$$

We assume that θ is a constant parameter, i.e. $\theta = \theta_t$ for all $t \in \mathbb{Z}_{\geqslant 0}$, but as will be shown later, we are still able to adapt to and learn changing parameter values.

A block diagram depicting the process (5.2) is shown in Figure 5.1.

5.2.1   Moving Horizon Estimation

In MHE, the current state of the system x_t at time t is estimated by solving a finite-horizon online optimization problem using a finite number of past measurements [90]. If we consider a finite horizon of L time steps, then the objective of the MHE problem is to find an estimate of the current state x_t so as to minimize a criterion of the form

$$\sum_{s=t-L}^{t} \eta_s\big(y_s - g_s(x_s)\big) + \sum_{s=t-L}^{t-1} \rho_s(d_s), \tag{5.3}$$

given the system dynamics (5.1). The functions $\eta_s(\cdot)$ and $\rho_s(\cdot)$ are assumed to take non-negative values. This is similar to the MHE criterion considered in [90, 3].

If the system dynamics also include uncertain model parameters, as in (5.2), the MHE problem can be formulated so as to estimate both the current state x_t and the uncertain parameter θ. Then the MHE problem can be written as

$$\min_{\substack{\hat{x}_{t-L}\in\mathcal{X},\\ \hat{d}_{t-L:t-1}\in\mathcal{D},\\ \hat{\theta}\in\Theta}} \sum_{s=t-L}^{t} \eta_s\big(y_s - g_s(\hat{x}_s,\hat{\theta})\big) + \sum_{s=t-L}^{t-1} \rho_s(\hat{d}_s), \tag{5.4}$$

where the initial state x_{t-L} is constrained to belong to the set $\mathcal{X}$, each element of the input disturbance sequence $d_{t-L:t-1}$ is assumed to belong to the set $\mathcal{D}$, and the uncertain parameter θ is known to belong to the set Θ. Throughout the chapter, given two times t_1 and t_2 with $t_1 < t_2$, we use the notation $x_{t_1:t_2}$ to denote the time series $x_{t_1}, x_{t_1+1}, \ldots, x_{t_2-1}, x_{t_2}$. An estimate of the current state is then determined from the dynamics (5.2) given the known past control inputs applied $u_{t-L:t-1}$ and estimates of the initial state $\hat{x}_{t-L}$, the input disturbance sequence $\hat{d}_{t-L:t-1}$, and the uncertain parameter $\hat{\theta}$. The optimization (5.4) is re-solved at each time t in a receding horizon fashion.

5.2.2   Model Predictive Control

In MPC, a sequence of future control inputs that achieve a desired control objective is computed by solving a finite-horizon online optimization problem using an estimate of the current state $\hat{x}_t$ and the system dynamics [95]. If we consider a finite-horizon of T time steps, then the objective of the MPC problem is to find a sequence of future control inputs $u_{t:t+T-1}$ that minimizes a criterion of the form

$$\sum_{s=t}^{t+T-1} \big(c_s(x_s, u_s, d_s) - \rho_s(d_s)\big) + q_{t+T}(x_{t+T}), \tag{5.5}$$

given the system dynamics (5.1). The functions $c_s(\cdot)$, $\rho_s(\cdot)$, and $q_{t+T}(\cdot)$ are all assumed to take non-negative values. The negative sign in front of $\rho_t(\cdot)$ penalizes the maximizer for using large values of d_t. The function $q_{t+T}(x_{t+T})$ is a terminal cost that penalizes the "final" state x_{t+T} and is needed for proving stability (see Assumption 9 below). The criterion (5.5) is similar to the closed-loop min-max MPC criterion considered in [70, 88].

If the system dynamics also include uncertain model parameters, as in (5.2), the MPC criterion (5.5) can be reformulated in order to incorporate worst-case values of the uncertain parameters θ, and the MPC problem can be written as

$$\min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \; \max_{\substack{\hat{d}_{t:t+T-1} \in \mathcal{D}, \\ \hat{\theta} \in \Theta}} \; \sum_{s=t}^{t+T-1} \big(c_s(x_s, \hat{u}_s, \hat{d}_s) - \rho_s(\hat{d}_s)\big) + q_{t+T}(x_{t+T}), \tag{5.6}$$

where each element of the future control input sequence $u_{t:t+T-1}$ is constrained to belong to the set $\mathcal{U}$, each element of the future disturbance sequence $d_{t:t+T-1}$ is assumed to belong to the set $\mathcal{D}$, and the uncertain parameter θ is known to belong to the set Θ. In order to overcome the conservativeness of open-loop control, at each time step t, the first element $\hat{u}_t^*$ of the future control input sequence $\hat{u}_{t:t+T-1}^*$ that is the solution to (5.6) is applied to the system, and the optimization (5.6) is solved again at each time step in a

receding horizon fashion. This is similar to the adaptive MPC problem with exogenous inputs considered in [27].

5.2.3   Adaptive MPC combined with MHE

Next we show how both the MPC problem (5.6) and MHE problem (5.4) can be formulated and solved simultaneously as a single min-max optimization problem.

Taking the criterion (5.5) and subtracting the criterion (5.3) gives a criterion of the form

$$J_t := \sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s) + q_{t+T}(x_{t+T}) - \sum_{s=t-L}^{t} \eta_s(n_s) - \sum_{s=t-L}^{t+T-1} \rho_s(d_s), \qquad (5.7)$$

which contains $T \in \mathbb{Z}_{\geqslant 1}$ terms of the running cost $c_s(x_s, u_s, d_s)$, which recede as the current time t advances, $L + 1 \in \mathbb{Z}_{\geqslant 1}$ terms of the measurement cost $\eta_s(n_s)$, and $L + T \in \mathbb{Z}_{\geqslant 1}$ terms of the cost on the input disturbance $\rho_s(d_s)$. Again, the function $q_{t+T}(x_{t+T})$ acts as a terminal cost in order to penalize the "final" state at time $t + T$. The functions $c_t(\cdot)$, $q_{t+T}(\cdot)$, $\eta_t(\cdot)$, and $\rho_t(\cdot)$ in (5.7) are all assumed to take non-negative values. We use finite-horizons into the past and into the future in order to decrease the computational complexity of the optimization problem, and we use online optimization to generate closed-loop solutions.

The *control objective* is to select the control signal $u_t \in \mathcal{U}$, $\forall t \in \mathbb{Z}_{\geqslant 0}$, so as to minimize the criterion defined in (5.7) under worst-case assumptions on the *unknown* system's initial condition $x_{t-L} \in \mathcal{X}$, unmeasured disturbances $d_t \in \mathcal{D}$, measurement noise $n_t \in \mathbb{R}^{n_n}$, and uncertain parameter $\theta \in \Theta$, for all $t \in \mathbb{Z}_{\geqslant 0}$, subject to the constraints imposed by the system dynamics (5.2) and the measurements $y_{t-L:t}$ collected up to the current time t.

Because the objective is to optimize the criterion (5.7) at the current time t in order to compute control inputs u_s for times $s \geqslant t$, there is no reason to penalize other irrelevant

terms. For instance, the first summation in (5.7) starts at time t because there is no reason to penalize the running cost $c_s(x_s, u_s, d_s)$ for past time instants $s < t$. There is also no reason to consider the values of future measurement noise at times $s > t$ as they will not affect choices made at time t. Thus, the second summation in (5.7) ends at time t. However, all values of the unmeasured disturbance d_s for $t - L \leqslant s \leqslant t + T - 1$ need to be considered because past values affect the (unknown) current state x_t, and future values affect the future values of the running cost.

Boundedness of (5.7) by a constant γ guarantees that

$$\sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s) + q_{t+T}(x_{t+T}) \leqslant \gamma + \sum_{s=t-L}^{t} \eta_s(n_s) + \sum_{s=t-L}^{t+T-1} \rho_s(d_s). \tag{5.8}$$

This shows that we can bound the running and final costs involving the future states $x_{t:t+T}$ in terms of bounds on the noise and disturbance.

Remark 10 (Quadratic case) *While the results presented here are general, it may be easier to gain intuition on the results when considering a quadratic criterion for (5.7) such as $c_t(x_t, u_t, d_t) := \|x_t\|^2 + \|u_t\|^2$, $\eta_t(n_t) := \|n_t\|^2$, $\rho_t(d_t) := \|d_t\|^2$. For this choice of the criterion, boundedness of (5.7) guarantees that the state x_t and input u_t are ℓ_2 provided that the disturbance d_t and noise n_t are also ℓ_2 [c.f. (5.8)].* □

With the objective of optimizing the criterion (5.7) at a given time $t \in \mathbb{Z}_{\geqslant 0}$ for the future control inputs $u_{t:t+T-1}$ and worst-case estimates of x_{t-L}, $d_{t-L:t+T-1}$, and θ, the combined adaptive MPC with MHE problem amounts to solving the following min-max optimization

$$J_t^* = \min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \quad \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}, \\ \hat{\theta} \in \Theta}}$$

$$\sum_{s=t}^{t+T-1} c_s(\hat{x}_s, \hat{u}_s, \hat{d}_s) + q_{t+T}(\hat{x}_{t+T}) - \sum_{s=t-L}^{t} \eta_s\big(y_s - g_s(\hat{x}_s, \hat{\theta})\big) - \sum_{s=t-L}^{t+T-1} \rho_s(\hat{d}_s), \quad (5.9)$$

with the understanding that

$$\hat{x}_{s+1} = \begin{cases} f_s(\hat{x}_s, \hat{\theta}, u_s, \hat{d}_s) & \text{for } t - L \leqslant s < t, \\[2mm] f_s(\hat{x}_s, \hat{\theta}, \hat{u}_s, \hat{d}_s) & \text{for } t \leqslant s < t + T. \end{cases}$$

We can view the optimization variables $\hat{x}_{t-L}$, $\hat{d}_{t-L:t+T-1}$, and $\hat{\theta}$ as (worst-case) estimates of the initial state, disturbances, and uncertain parameter, respectively, based on the past inputs $u_{t-L:t-1}$ and outputs $y_{t-L:t}$ available at time t.

Just as in the MPC problem, after solving this optimization problem at each time t, we use as the control input the first element of the sequence

$$\hat{u}^*_{t:t+T-1} = \{\hat{u}^*_t, \hat{u}^*_{t+1}, \hat{u}^*_{t+2}, \ldots, \hat{u}^*_{t+T-1}\} \in \mathcal{U}$$

that minimizes (5.9), leading to the control law

$$u_t = \hat{u}^*_t, \quad \forall t \geqslant 0. \tag{5.10}$$

A depiction of an example solution to the combined finite-horizon control and estimation problem is shown in Figure 5.2.

5.3   Stability Results

Next we discuss under what appropriate assumptions the control law (5.10) leads to boundedness of the state of the closed-loop system resulting from the finite-horizon optimization introduced in Section 5.2.3.
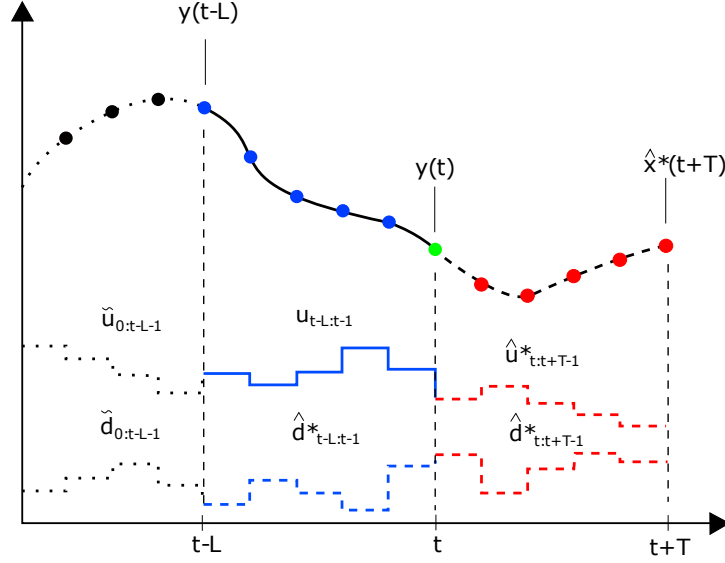
Figure 5.2: Example solution to the combined finite-horizon control and estimation problem. The elements from $t - L$ to t correspond to the MHE problem, and the elements from t to $t + T$ correspond to the MPC problem. The variables denoted as $\hat{\cdot}$ are optimization variables, the variables denoted as $\tilde{\cdot}$ are not relevant for the optimization, and the other variables are known.

In order to implement the control law (5.10), the outer minimization in (5.9) must lead to a finite value for the optimum. For the stability results in this section, we require the existence of a finite-valued saddle-point solution to the min-max optimization in (5.9), which is a common requirement in game theoretical approaches to control design [6]. The following assumptions are the same as those in Chapter 2 just augmented to include the unknown parameters θ.

Assumption 7 (Saddle-point) *The min-max optimization (5.9) always has a finite-valued saddle-point solution for which the min and max commute. Specifically, for all $t \in \mathbb{Z}_{\geqslant 0}$, $u_{t-L:t-1} \in \mathcal{U}$, $y_{t-L:t}$, there exists $J_t^* \in \mathbb{R}$, $\hat{x}_{t-L}^* \in \mathcal{X}$, $\hat{u}_{t:t+T-1}^* \in \mathcal{U}$, $\hat{d}_{t-L:t+T-1}^* \in \mathcal{D}$,*

and $\hat{\theta}^* \in \Theta$ such that

$$J_t^* = \min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}, \\ \hat{\theta} \in \Theta}} J_t = \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}, \\ \hat{\theta} \in \Theta}} \min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} J_t < \infty.$$

$\square$

Assumption 7 presumes an appropriate form of *observability/detectability* adapted to the criterion $\sum_{s=t}^{t+T-1} c_s(x_s, u_s, d_s)$. In particular, it implies that the size of the *current* state can be bounded using past outputs and past/future input disturbances, regardless of the value of $\theta \in \Theta$.

To ensure controllability and to establish state boundedness under the control (5.10) defined by the finite-horizon optimization (5.9), we require additional assumptions regarding the dynamics and the terminal cost $q_t(\cdot)$.

Assumption 8 (Observability) *There exists a bounded set $\mathcal{N}_{\mathrm{pre}} \subset \mathbb{R}^{n_n}$ such that, for every time $t \in \mathbb{Z}_{\geqslant 0}$, every state $\hat{x}_{t-L:t} \in \mathcal{X}$, every uncertain parameter $\hat{\theta} \in \Theta$, and every disturbance and noise sequence, $\hat{d}_{t-L:t} \in \mathcal{D}$ and $\hat{n}_{t-L:t} \in \mathcal{N}$, that are compatible with the applied control input u_s, $s \in \mathbb{Z}_{\geqslant 0}$, and the measured output y_s, $s \in \mathbb{Z}_{\geqslant 0}$, in the sense that*

$$\hat{x}_{s+1} = f_s(\hat{x}_s, \hat{\theta}, u_s, \hat{d}_s), \qquad\qquad y_s = g_s(\hat{x}_s, \hat{\theta}) + \hat{n}_s, \qquad (5.11)$$

$\forall s \in \{t - L, t - L + 1, \ldots, t\}$, there exists a "predecessor" state estimate $\hat{x}_{t-L-1} \in \mathcal{X}$, disturbance estimate $\hat{d}_{t-L-1} \in \mathcal{D}$, and noise estimate $\hat{n}_{t-L-1} \in \mathcal{N}_{\mathrm{pre}}$ such that (5.11) also holds for time $s = t - L - 1$. $\square$

In essence, Assumption 8 requires the past horizon length L to be sufficiently large. The discussion following Assumption 3 in Chapter 2 is also relevant as a further discussion of this assumption.

Assumption 9 (ISS-control Lyapunov function) *The terminal cost $q_t(\cdot)$ is an ISS-control Lyapunov function, in the sense that, for every $t \in \mathbb{Z}_{\geqslant 0}$, $x \in \mathcal{X}$, $d \in \mathcal{D}$, and $\theta \in \Theta$, there exists a control $u \in \mathcal{U}$ such that*

$$q_{t+1}\big(f_t(x, \theta, u, d)\big) - q_t(x) \leqslant -c_t(x, u, d) + \rho_t(d). \tag{5.12}$$

$\square$

Assumption 9 plays the role of the common assumption in MPC that the terminal cost must be a control Lyapunov function for the closed-loop [76]. Without the disturbance d_t, (5.12) would imply that $q_t(\cdot)$ could be viewed as a control Lyapunov function that decreases along system trajectories for an appropriate control input u_t [101]. With the disturbance d_t, $q_t(\cdot)$ should be viewed as an ISS-control Lyapunov function that satisfies an ISS stability condition for the disturbance input d_t and an appropriate control input u_t [61]. In the case of linear dynamics and a quadratic cost function, a terminal cost $q_t(\cdot)$ can typically be found by solving a system of linear matrix inequalities.

5.3.1   State Boundedness

The following theorem provides a bound that can be used to prove boundedness of the state when the control signal is computed by solving the finite-horizon optimization (5.9).

Theorem 5 (Finite horizon cost-to-go bound) *Suppose that Assumptions 7, 8, and 9 hold. Along any trajectory of the closed-loop system defined by the process (5.2) and the control law (5.10), we have that*

$$c_t(x_t, u_t, d_t) \leqslant J_L^*(u_{0:L-1}, y_{0:L}) + \sum_{s=0}^{t-L-1} \rho_s(\tilde{d}_s) + \sum_{s=0}^{t-L-1} \eta_s(\tilde{n}_s) + \sum_{s=t-L}^{t} \eta_s(n_s) + \sum_{s=t-L}^{t} \rho_s(d_s)$$

114

$$\forall t \in \mathbb{Z}_{\geqslant L}, \quad (5.13)$$

for appropriate sequences $\tilde{d}_{0:t-L-1} \in \mathcal{D}$, $\tilde{n}_{0:t-L-1} \in \mathcal{N}_{\mathrm{pre}}$. $\qquad\qquad\qquad\qquad\qquad$ $\square$

Proof. This result is an extension of Theorem 2 presented in Chapter 2. If the state is augmented such that $\bar{x}_t = [x_t \; \theta]'$, and the process is defined as $\bar{x}_{t+1} = [\bar{f}_t(\bar{x}_t, u_t, d_t) \; \theta]'$, then the same proof used for Theorem 2 in Chapter 2 can be applied here using $\bar{x}_t$ in place of x_t. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

The discussion after Theorem 2 presented in Chapter 2 regarding state boundedness, practical stability, and reference tracking can be applied here as well.

5.4   Simulation Study

In this section we consider several examples of systems with parametric uncertainty and present closed-loop simulations using the control approach described in Section 5.2. For all of the following examples, we use a cost function of the form

$$J_t = \sum_{s=t}^{t+T-1} \|h_s(x_s)\|_2^2 + \lambda_u \sum_{s=t}^{t+T-1} \|u_s\|_2^2 - \lambda_n \sum_{s=t-L}^{t} \|n_s\|_2^2 - \lambda_d \sum_{s=t-L}^{t+T-1} \|d_s\|_2^2. \qquad (5.14)$$

where $h_s(x_s)$ is a function of the state x_s that is especially relevant for the example under consideration, and λ_u, λ_n, and λ_d are positive weighting constants.

Given the optimization criterion (5.14), the following examples involve optimizing this criterion with respect to the future control inputs $u_{t:t+T-1}$ under worst-case assumptions on x_{t-L}, $d_{t-L:t+T-1}$, and θ by solving the following min-max optimization problem:

$$\min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \; \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1}, \\ \hat{\theta} \in \Theta}} \; J_t. \qquad (5.15)$$

115

The first time this optimization is solved, guesses for the initial values of the uncertain parameter θ, initial state x_{t-L}, past control inputs $u_{t-L:t-1}$ and input disturbances $d_{t-L:t-1}$ need to be made. Then values for the past states $x_{t-L+1:t}$ that are consistent with the dynamics are picked. These states can be used to determine the output measurements $y_{t-L:t}$, and then the optimization (5.15) can be solved for the first time. At subsequent times, all of the variables from the solution of (5.15) at the previous time step (after moving away from the constraints) can be used as a "warm start" for solving (5.15) at the current time step. In order to solve the optimization (5.15), the primal-dual-like interior-point method described in Chapter 4 can again be used.

Example 4 (Linear System - uncertain gain and poles) *Consider a discrete-time linear system described by the transfer function*

$$G(z) = \frac{b}{(z - p_1)(z - p_2)}, \tag{5.16}$$

where $p = [p_1 \ p_2]^\top$ denotes the uncertain pole locations p_1 and p_2 that are assumed to belong to the set $\mathcal{P} := \{p \in \mathbb{R}^2 : 0 \leqslant p_i \leqslant 2, i = 1, 2\}$, so they may be stable or unstable. The parameter b is an uncertain gain assumed to belong in the interval $\mathcal{B} := \{b \in \mathbb{R} : 1 \leqslant b \leqslant 5\}$.

The transfer function (5.16) can be rewritten in state space controllable canonical form as

$$
\begin{aligned}
x_{t+1} &= \begin{bmatrix} p_1 + p_2 & -p_1 p_2 \\ 1 & 0 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ 0 \end{bmatrix} (u_t + d_t), \\
y_t &= \begin{bmatrix} 0 & b \end{bmatrix} x_t + n_t, \qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0},
\end{aligned}
\tag{5.17}
$$

where y_t is the measured output at time t with noise n_t, and d_t is an additive input

disturbance. *For all $t \in \mathbb{Z}_{\geqslant 0}$, the control input u_t is constrained to belong in the set*

$\mathcal{U} := \{u_t \in \mathbb{R} : \|u_t\|_\infty \leqslant 8\}$, *and the input disturbance d_t is assumed to belong to the set*

$\mathcal{D} := \{d_t \in \mathbb{R} : \|d_t\|_\infty \leqslant 0.1\}$.

By defining $a_1 = p_1 + p_2$ and $a_2 = p_1 p_2$, the state space model (5.17) can be reparametrized as

$$x_{t+1} = \begin{bmatrix} a_1 & -a_2 \\ 1 & 0 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ 0 \end{bmatrix} (u_t + d_t),$$

$$y_t = \begin{bmatrix} 0 & b \end{bmatrix} x_t + n_t, \qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0}. \tag{5.18}$$

Letting $a = [a_1 \ a_2]^\top$, the uncertain parameter a is assumed to belong to $\mathcal{A} := \{a \in \mathbb{R}^2 :$

$0 \leqslant a_i \leqslant 4, i = 1, 2\}$. This set $\mathcal{A}$ is conservative, and a tighter non-convex set could be

used. Now the model (5.18) is linear in the uncertain parameters. This is a standard

problem that can be solved using classical adaptive control techniques. We will show that

our MPC with MHE approach can solve this problem, and in following examples, we will

see that our approach does not require reparametrization such that the system is linear in

the uncertain parameters.

The uncertain parameters (a and b) can be estimated by including them as optimiza-

tion variables in the following problem

$$\min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \ \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}, \\ \hat{a} \in \mathcal{A}, \\ \hat{b} \in \mathcal{B}}} \ \sum_{s=t}^{t+T} \|y_s - r_s\|_2^2 + \lambda_u \sum_{s=t}^{t+T-1} \|\hat{u}_s\|_2^2 - \lambda_n \sum_{s=t-L}^{t} \|n_s\|_2^2 - \lambda_d \sum_{s=t-L}^{t+T-1} \|\hat{d}_s\|_2^2,$$

$$\tag{5.19}$$

where r_t is a desired reference signal for the output of the system to follow. Figures 5.3

and 5.4 show simulations of the resulting closed-loop system for a square-wave reference

defined as $r_t = 10 \, \text{sgn}(\sin(0.4t))$ and the backward and forward horizon lengths chosen as $L = 10$, and $T = 10$, respectively. The weights in the cost function are chosen to be $\lambda_u = 0.1$, $\lambda_n = 1000$, and $\lambda_d = 1000$. In this simulation, the actual input disturbance d_t and measurement noise n_t are unmeasured Gaussian independently and identically distributed (i.i.d.) random variables with zero mean and standard deviations of 0.001 and 0.005, respectively.
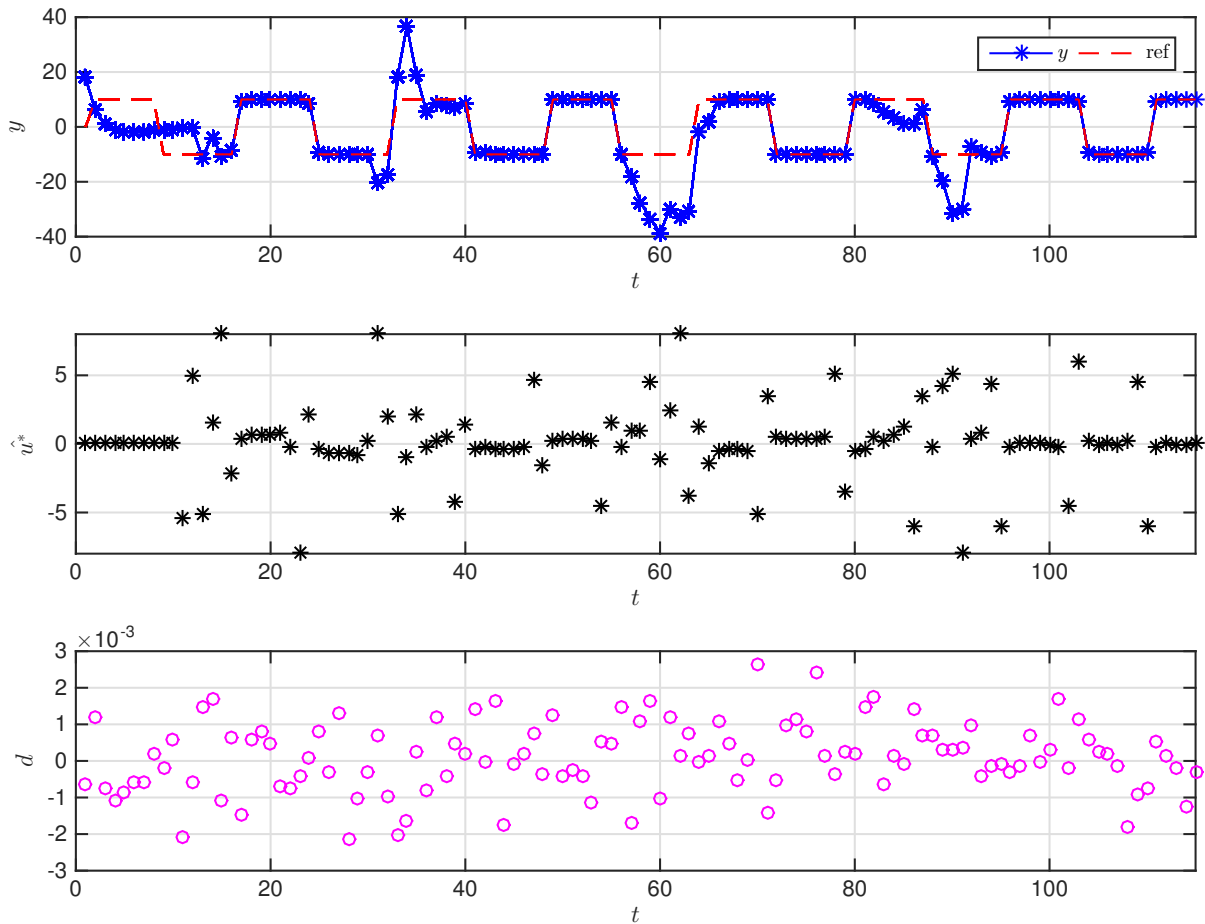
Figure 5.3: Linear System: output and inputs. The top plot shows the measured output (denoted by o's) and the reference signal (denoted by -'s). The second plot shows the input $\hat{u}_t^*$ applied to the system, and the third plot shows the unmeasured disturbance input d_t that the system is subjected to.

Figure 5.3 shows the output of the system successfully following the given square-wave reference trajectory. The system is initialized with incorrect guesses for the initial values
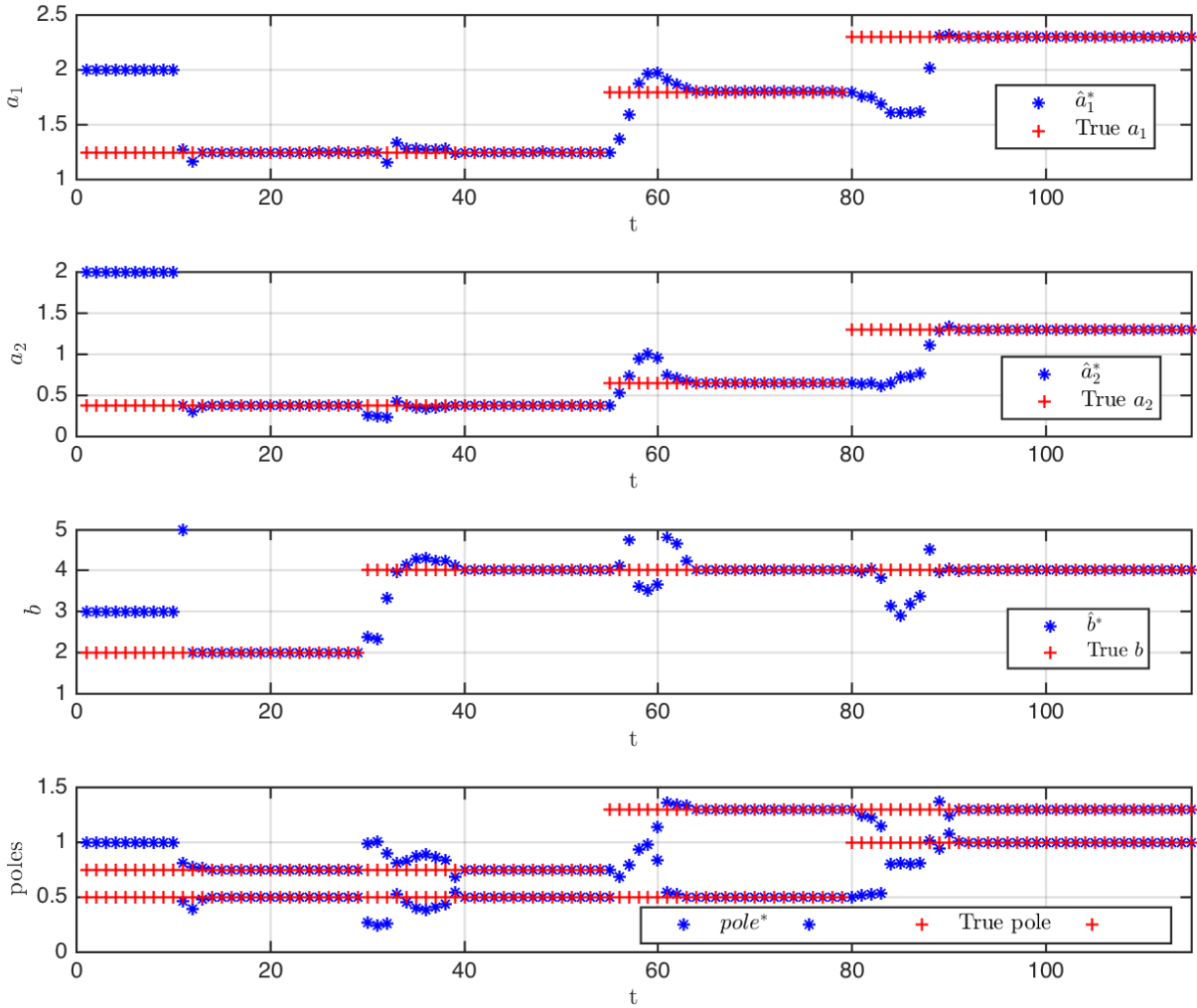
Figure 5.4: Linear System: parameters. The top two plots show the true values of a_1 and a_2 (denoted by +'s) and their estimated values (denoted by *'s). The third plot shows the true value of the gain b (denoted by +'s) and its estimated value (denoted by *'s). The bottom plot shows the true values of the poles p_1 and p_2 (denoted by +'s) and their estimated values (denoted by *'s) computed from $\hat{a}_1^*$ and $\hat{a}_2^*$.

of the uncertain parameters, and zero control input (i.e. $u_t = 0$) is applied for the first $L = 10$ time steps. After that point, starting at time $t = 11$, the optimization problem (5.19) is solved at each time step, and the computed control input $\hat{u}_t^*$ is applied in a receding horizon fashion.

At several times throughout the simulation, the true model of the system (5.17) is altered by changing the value of the gain or the poles (which can be seen in Figure 5.4).

The estimates of the gain and pole locations shown in Figure 4 are obtained from the estimates of the uncertain parameters a and b from the solution of the optimization (5.19). Figure 5.4 shows that the estimates of the gain and pole locations converge to their true values. Even after the true values of the gain and pole locations are changed during the simulation, the combined MPC and MHE scheme is able to adapt to the changing system, effectively regulating the system to the reference trajectory and correctly learning the new parameters of the model. □

Example 5 (Inverted Pendulum - uncertain mass and friction) *Consider an inverted pendulum actuated by a torque at the base as shown in Figure 5.5 and described by the model*

$$ml^2\ddot{\phi} = mgl\sin(\phi) - b\dot{\phi} + \tau,$$

where m is the mass at the end of the pendulum, l is the length of the link, ϕ is the angle from vertical, g is the gravitational constant, b is the coefficient of friction, and τ is the torque applied at the base.
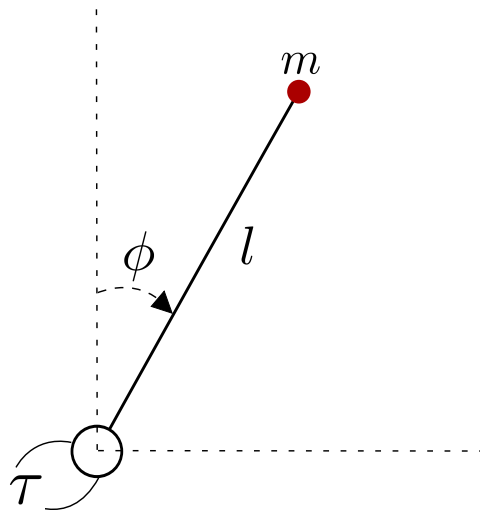
Figure 5.5: Diagram of the pendulum considered in Examples 5 and 6.

We can rewrite this model in state space form as

$$\dot{x}_1 = x_2,$$
$$\dot{x}_2 = \frac{g}{l}\sin(x_1) - \frac{b}{ml^2}x_2 + \frac{1}{ml^2}u, \qquad (5.20)$$

where $x_1 = \phi$, $x_2 = \dot{\phi}$, and $u = \tau$.

By letting $l = 1$, $g = 9.81$, and $a = 1/m$, adding an input disturbance d, and discretizing using Euler's Method with time step Δt, the system (5.20) becomes

$$x_{1,t+1} = x_{1,t} + \Delta t\, x_{2,t},$$
$$x_{2,t+1} = x_{2,t} + \Delta t\big(9.81\sin(x_{1,t}) - abx_{2,t} + a(u_t + d_t)\big), \qquad (5.21)$$
$$y_t = x_{1,t} + n_t, \qquad\qquad\qquad \forall t \in \mathbb{Z}_{\geqslant 0},$$

where y_t is the measurement available at time t with noise n_t. According to this model, a noisy measurement of the angle x_1 is available at each time t.

The inverse of the mass and the coefficient of friction (a and b, respectively) are uncertain but assumed to belong to the sets $\mathcal{A} := \{a \in \mathbb{R} : 1/2 \leqslant a \leqslant 1\}$ and $\mathcal{B} := \{b \in \mathbb{R} : 0.2 \leqslant b \leqslant 0.7\}$, respectively. The control input u_t is constrained to the set $\mathcal{U} := \{u_t \in \mathbb{R} : \|u_t\|_\infty \leqslant 5\}$, and the disturbance input d_t is assumed to belong to $\mathcal{D} := \{d_t \in \mathbb{R} : \|d_t\|_\infty \leqslant 0.3\}$ for all $t \in \mathbb{Z}_{\geqslant 0}$.

The control objective is to regulate the output (the noisy measurement of the angle ϕ) to a desired reference. The uncertain mass and coefficient of friction can be determined using estimates of the parameters a and b in (5.21). These parameters can be estimated

by including them as optimization variables in the following problem.

$$\min_{\hat{u}_{t:t+T-1}\in\mathcal{U}} \max_{\substack{\hat{x}_{t-L}\in\mathcal{X}, \\ \hat{d}_{t-L:t+T-1}\in\mathcal{D}, \\ \hat{a}\in\mathcal{A}, \\ \hat{b}\in\mathcal{B}}} \sum_{s=t}^{t+T} \|y_s - r_s\|_2^2 + \lambda_u \sum_{s=t}^{t+T-1} \|\hat{u}_s\|_2^2 - \lambda_n \sum_{s=t-L}^{t} \|n_s\|_2^2 - \lambda_d \sum_{s=t-L}^{t+T-1} \|\hat{d}_s\|_2^2.$$

$$(5.22)$$

A noteworthy challenge in this nonlinear problem is that the unknown parameters a and b appear multiplied by the unmeasurable state x_2 in the system dynamics (5.21).

Figure 5.6 shows a simulation of the closed-loop system with the discrete time-step chosen as $\Delta t = 0.2$ and a square-wave reference given as $r_t = 5(\pi/180)\operatorname{sgn}\big(\sin(0.5t)\big)$. The backward and forward horizon lengths are chosen to be $L = 6$, and $T = 7$, respectively. The weights in the cost function are chosen to be $\lambda_u = 0$ (i.e. the control input is not penalized), $\lambda_n = 1000$, and $\lambda_d = 10$. In this simulation, the actual input disturbance d_t and measurement noise n_t are unmeasured Gaussian i.i.d. random variables with zero mean and standard deviations of 0.001 and 0.0001, respectively. The system is initialized with incorrect guesses for the initial values of the uncertain parameters, and zero control input (i.e. $u_t = 0$) is applied for the first $L = 6$ time steps. After that point, starting at time $t = 7$, the optimization problem (5.22) is solved at each time step, and the computed control input $\hat{u}_t^*$ is applied in a receding horizon fashion.

As in the previous example, the true values of the uncertain parameters (the mass and coefficient of friction) are changed several times throughout the simulation. Even as the true values of the mass and coefficient of friction change, the control input $\hat{u}_t^*$, computed by solving the optimization (5.22), is able to successfully regulate the output of the system to the reference trajectory, and the estimates of the uncertain mass m and coefficient of friction b converge to their true values. □

Example 6 (Inverted Pendulum - stabilization and disturbance rejection) *This*
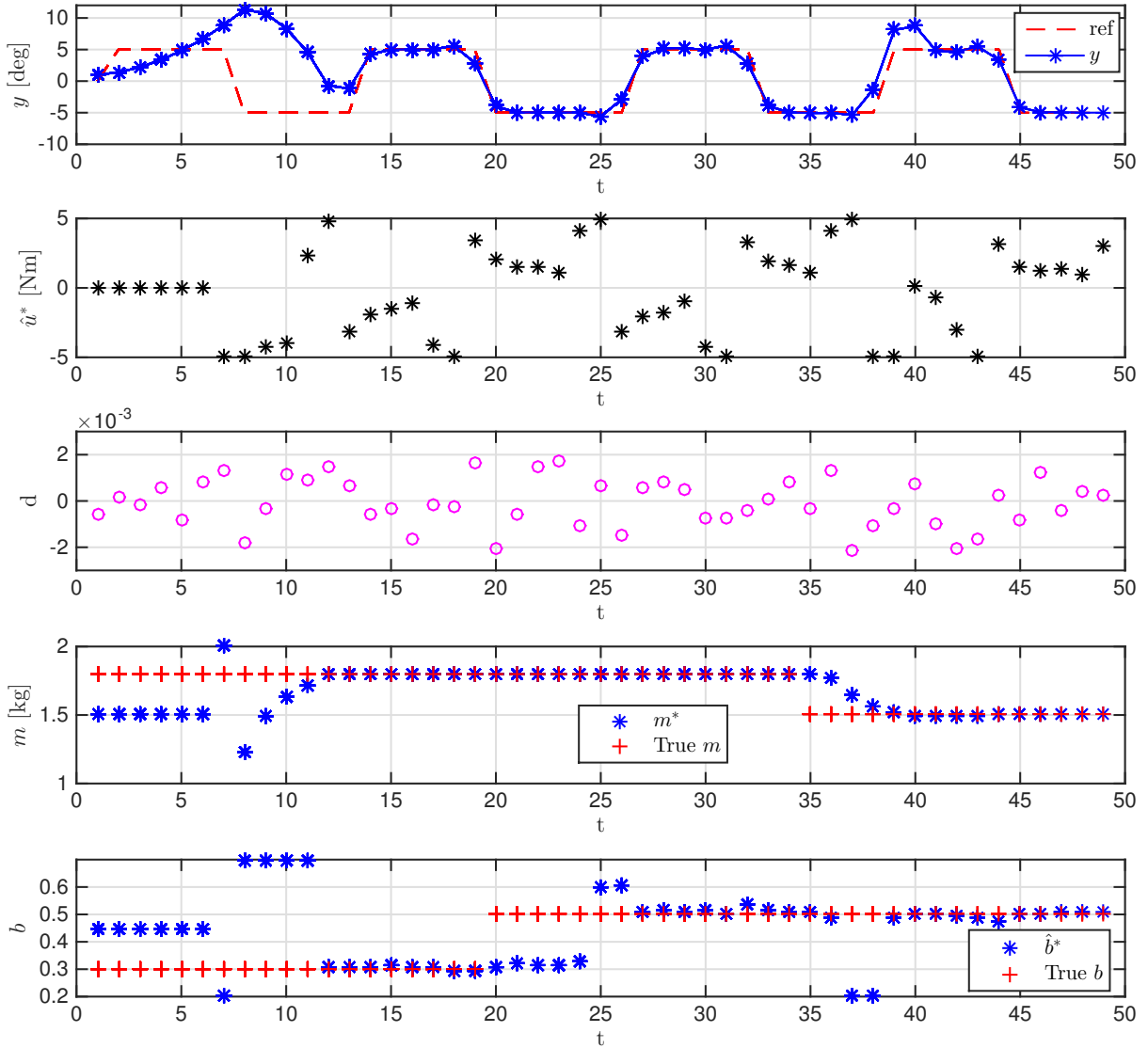
Figure 5.6: Inverted Pendulum: uncertain mass and friction. The top plot shows the measured output (denoted by *'s) tracking the square reference signal (denoted by -'s). The second plot shows the control input $\hat{u}_t^*$ that is applied. The third plot shows the unmeasured input disturbance d_t that the system is subjected to. The bottom two plots show the true values of the mass and coefficient of friction (denoted by +'s) and the estimated values of the mass m^* (computed from $\hat{a}^*$) and coefficient of friction $\hat{b}^*$ (denoted by *'s).

example shows that this adaptive MPC with MHE approach can stabilize uncertain systems even when the systems are not persistently excited and the true values of the uncertain parameters are not learned. Furthermore, the results of this example show that this

estimation and control approach is not only robust to the model uncertainty but is also able to reject large input disturbances.

Again we consider an inverted pendulum as depicted in Figure 5.5 and described using a discretized model of the form (5.21). This time, rather than following a reference trajectory, the control objective is to stabilize the system at the unstable equilibrium $x_1 = 0$. This means that the same optimization given in (5.22) is solved but with $r_t = 0$. Figure 5.7 shows a simulation of the resulting closed-loop system.

For this example, the parameters in the optimization (5.22) are chosen the same as in Example 5 except with respect to the input disturbance d_t. In this example, the unmeasured disturbance d_t is larger and assumed to belong to the set $\mathcal{D} \coloneqq \{d_t \in \mathbb{R} : \|d_t\|_\infty \leqslant 0.5\}$. The weight on the input disturbance in (5.22) is chosen as $\lambda_d = 1$. The actual input disturbance that the system is subjected to is a Gaussian i.i.d. random variable with zero mean and a standard deviation of 0.15. Again, the system is initialized with guesses for the initial values of the uncertain parameters, and zero control input (i.e. $u_t = 0$) is applied for the first $L = 6$ time steps. Starting at time $t = 7$, the optimization problem (5.22) is solved at each time step, and the computed control input $\hat{u}_t^*$ is applied in a receding horizon fashion.

Figure 5.7 shows that the system is not sufficiently excited in order to correctly learn the true values of the mass m and coefficient of friction b. However, the control input $\hat{u}_t^*$ computed by solving the optimization (5.22) is nonetheless able to stabilize the system (even as the true values of the uncertain parameters change) and reject the large unmeasured disturbance input. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Example 7 (Nonlinear Pursuit-Evasion - uncertain wind) *In this example, we consider a two-player pursuit-evasion game where the pursuer is modeled as a unicycle vehicle, and the evader is modeled as a double-integrator. The pursuer is an aerial vehicle*
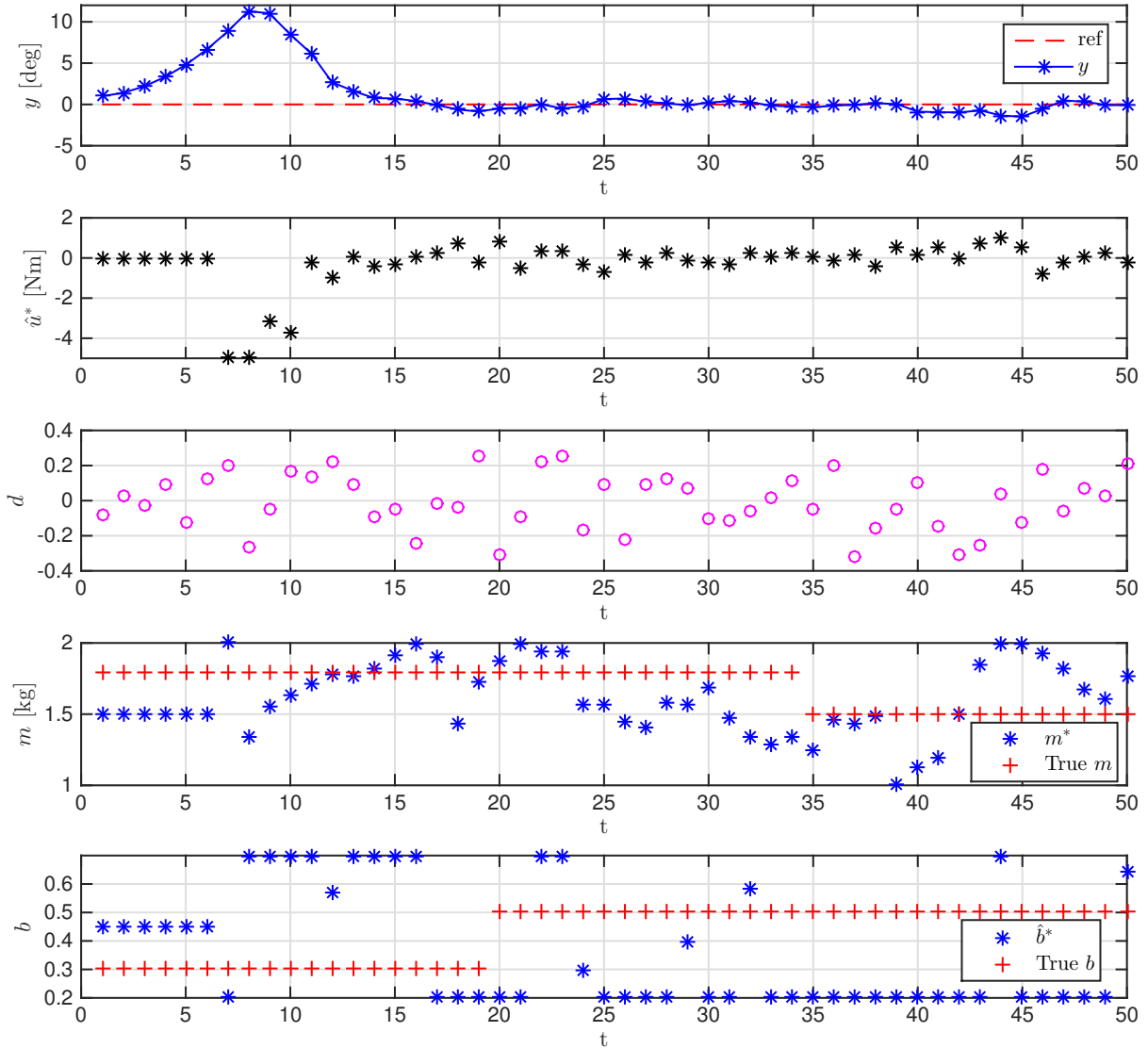
Figure 5.7: Inverted Pendulum: stabilization and disturbance rejection. The top plot shows the output (denoted by *'s) converging to the unstable equilibrium $\phi = 0$ (denoted by -'s). The second plot shows the control input $\hat{u}_t^*$ that is applied. The third plot shows the large unknown input disturbance d_t that the system is subjected to. The fourth and fifth plots show the true values of the mass and coefficient of friction (denoted by +'s) and the estimated values of the mass m^* (computed from $\hat{a}^*$) and coefficient of friction $\hat{b}^*$ (denoted by *'s).

that is subject to wind disturbances, and the evader is a ground vehicle that is not susceptible to the wind. A nonlinear discrete-time model of the overall system is given as follows:

$$x_{t+1} = x_t + \begin{bmatrix} v\cos\phi_t + w_1 \\ v\sin\phi_t + w_2 \\ u_t \\ d_t \end{bmatrix}, \qquad y_t = x_t + n_t, \qquad \forall t \in \mathbb{Z}_{\geqslant 0}. \qquad (5.23)$$

The state of the system is denoted by $x_t = [p_t \ \phi_t \ z_t]^\top$, where $p_t = [p_1 \ p_2]_t^\top$ denotes the planar position of the pursuer, ϕ_t denotes the orientation of the pursuer, and $z_t = [z_1 \ z_2]_t^\top$ denotes the planar position of the evader. The planar wind speed is denoted by $w = [w_1 \ w_2]^\top$ where w_1 is the component of the wind speed in the x-direction, w_2 is the component of the wind speed in the y-direction, and w is known to belong to the set $\mathcal{W} := \{w \in \mathbb{R}^2 : \|w\|_\infty \leqslant 0.05\}$. The control input u_t is constrained to belong in the set $\mathcal{U} := \{u_t \in \mathbb{R} : \|u_t\|_\infty \leqslant 0.35\}$. The evader's velocity is given by $d_t = \begin{bmatrix} d_1 \ d_2 \end{bmatrix}_t^\top$ and is constrained to the set $\mathcal{D} := \{d_t \in \mathbb{R}^2 : \|d_t\|_\infty \leqslant 0.05\}$, and $n_t \in \mathbb{R}^{n_n}$ is measurement noise.

The pursuer's objective is to make the distance between its position p_t and the position of the evader z_t as small as possible, so the pursuer wants to minimize the value of $\|p_t - z_t\|$. The evader's objective is to do the opposite, namely, maximize the value of $\|p_t - z_t\|$. The pursuer and evader try to achieve these objectives by choosing appropriate values for u_t and d_t, respectively. The wind speed is unknown, but both the pursuer and evader would benefit from learning the wind speed. Therefore, the optimal solution will involve each player adapting his or her action (choice of u_t and d_t) to the current estimate of the wind speed. These considerations motivate solving the following problem

$$\min_{\hat{u}_{t:t+T-1} \in \mathcal{U}} \ \max_{\substack{\hat{x}_{t-L} \in \mathcal{X}, \\ \hat{d}_{t-L:t+T-1} \in \mathcal{D}, \\ \hat{w} \in \mathcal{W}}} \ \sum_{s=t}^{t+T} \|p_s - z_s\|_2^2 + \lambda_u \sum_{s=t}^{t+T-1} \|\hat{u}_s\|_2^2 - \lambda_n \sum_{s=t-L}^{t} \|n_s\|_2^2 - \lambda_d \sum_{s=t-L}^{t+T-1} \|\hat{d}_s\|_2^2,$$

$$(5.24)$$

where the pursuer's future actions $u_{t:t+T-1}$, the unknown evader's actions $d_{t-L:t+T-1}$, the unknown initial state x_{t-L}, and the unknown wind speed w are included as optimization variables. A simulation of the resulting closed-loop system is shown in Figures 5.8, 5.9, and 5.10.
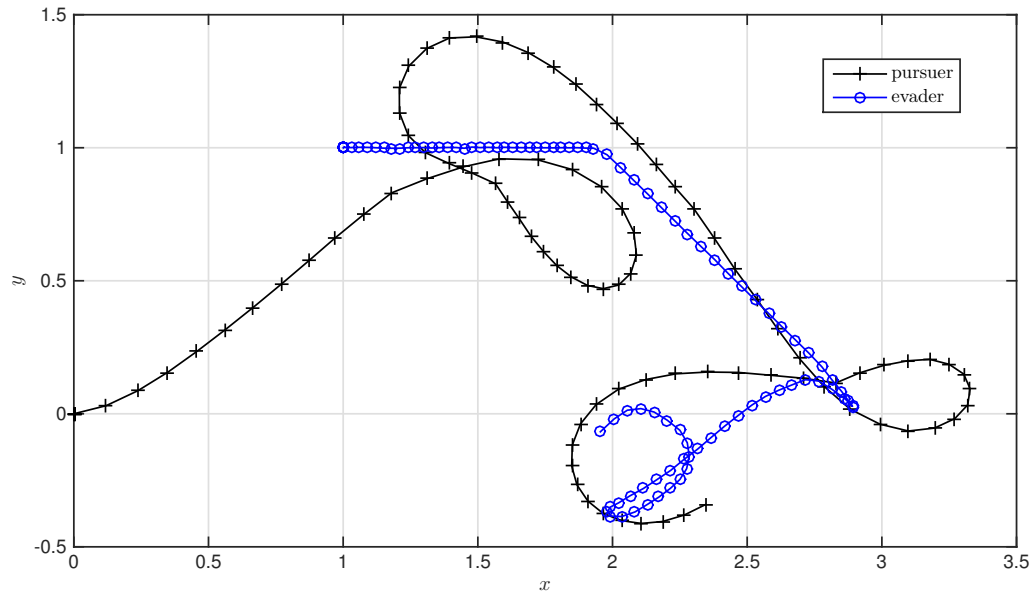
Figure 5.8: Pursuit-evasion: trajectories. The pursuer's trajectory (denoted by +'s) begins at (0,0), and the evader's trajectory (denoted by o's) begins at (1,1).

Parameters chosen for the model (5.23) and the optimization (5.24) are given as follows. The pursuer moves with constant velocity $v = 0.1$. The backward and forward horizon lengths are chosen to be $L = 8$, and $T = 12$, respectively. The weights in the cost function in (5.24) are chosen to be $\lambda_u = 10$, $\lambda_n = 10000$, and $\lambda_d = 100$. The actual measurement noise n_t is an unmeasured Gaussian i.i.d. random variable with zero mean and a standard deviation of 0.001.

The trajectories that each player follow are shown in Figure 5.8. The evader moves with constant velocity in the positive x-direction until time $t = 40$ when the optimal $\hat{d}_t^*$ begins to be applied. The pursuer applies $\hat{u}_t^*$ throughout the entire simulation. Rapidly the pursuer catches up to the evader and is forced to make a loop due to its nonholonomic

127
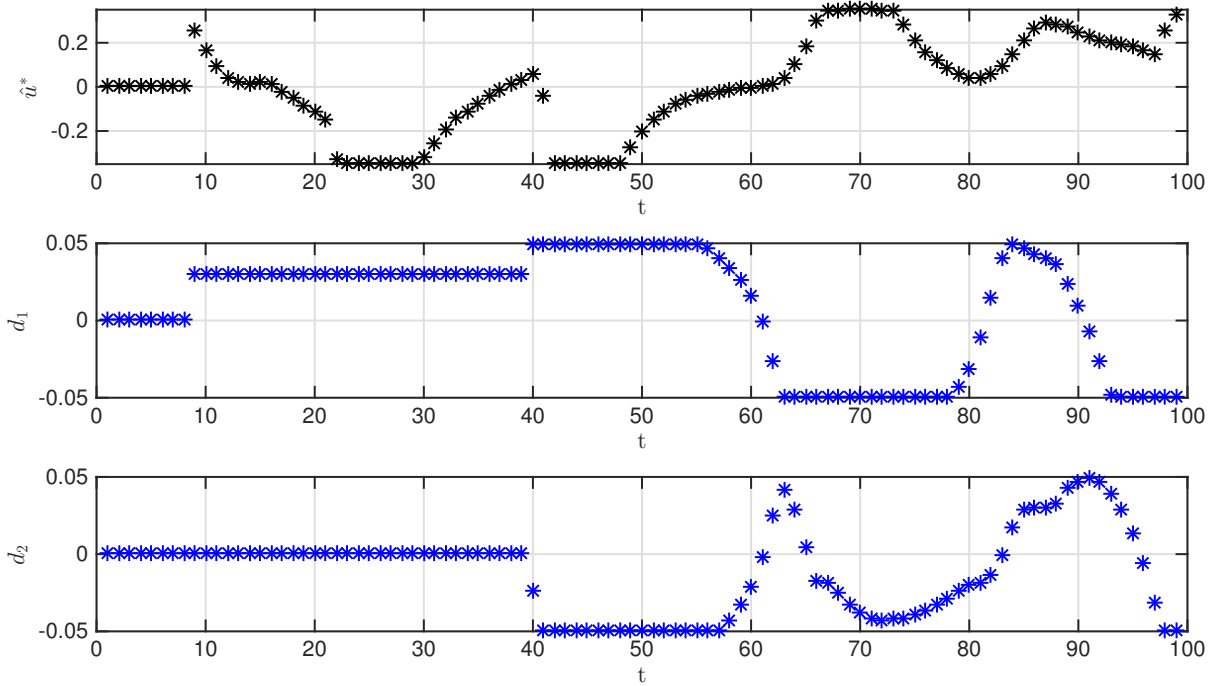
Figure 5.9: Pursuit-evasion: inputs. The top plot shows the pursuer's input $\hat{u}_t^*$ that is applied. The lower two plots show the evader's input d_t that is applied. The evader applies constant velocity until time $t = 40$ at which time the optimal $\hat{d}_t^*$ is applied for the remainder of the simulation.

dynamics. The evader, on the other hand, is able to make sharp maneuvers due to its double-integrator dynamics. The inputs that each player applies are shown in Figure 5.9. Figure 5.10 shows that the estimates of the uncertain wind speed converge to their true values even as they change throughout the simulation. □

5.5   Conclusions

In this chapter, we addressed adaptation and learning in the context of output-feedback MPC with MHE. Often the MPC and MHE problems are formulated with a known model of the dynamics. However, in this chapter, we investigated solving the MPC and MHE problems using a model with uncertain parameters. This was done by simultaneously solving the MPC and MHE problems as a single min-max optimization
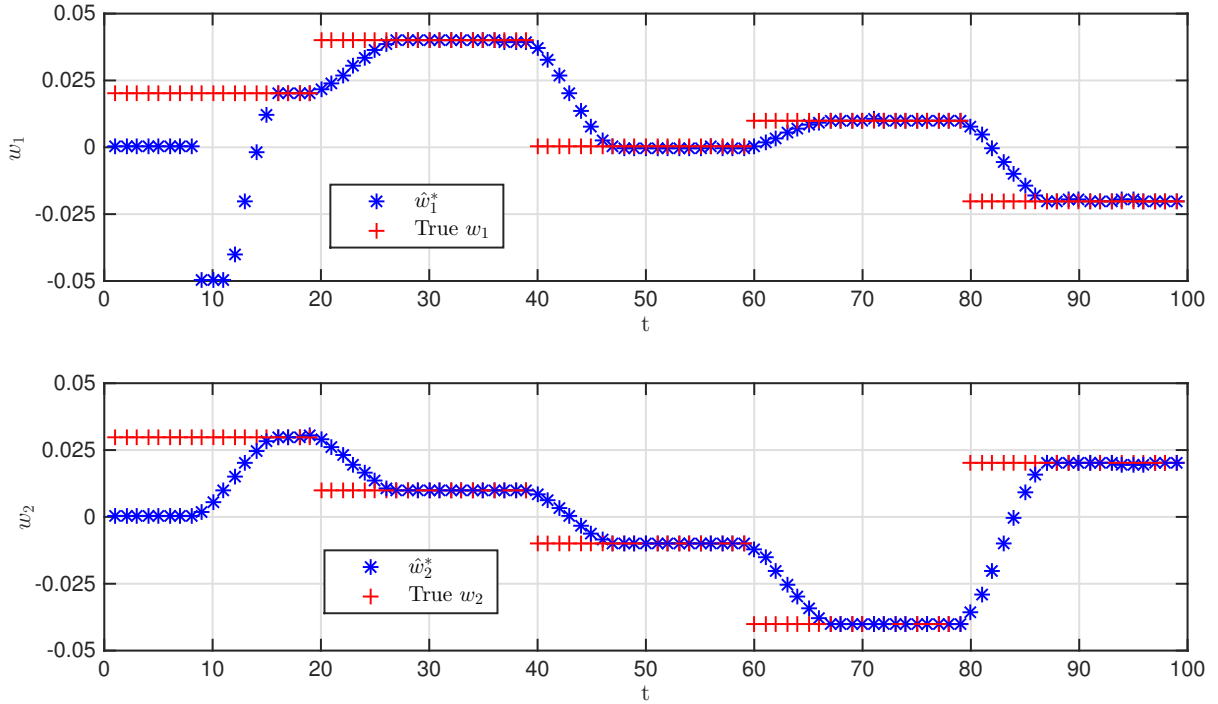
Figure 5.10: Pursuit-evasion: wind. The top and bottom plots show the true values of the wind speed (denoted by +'s) and the estimated values $\hat{w}_1^*$ and $\hat{w}_2^*$ of the wind speed (denoted by *'s) in the x- and y-directions, respectively.

problem and including the uncertain model parameters as optimization variables to be estimated. Under appropriate assumptions ensuring controllability and observability, Theorem 5 provides bounds on the state of the system.

In a simulation study, we showed that the combined control and estimation approach effectively controls linear and nonlinear systems with model parameter uncertainty, adapts to changing model parameters, and also learns the uncertain model parameters when the system is sufficiently excited. However, even when the system is not sufficiently excited to learn the true values of the uncertain model parameters, the computed control law is still able to effectively reject disturbances and stabilize the system. Using a primal-dual-like interior point method, solutions to this MPC with MHE approach can be found even for severely non-convex examples.

Future work may involve investigating under what specific conditions the estimates

of the uncertain parameters are guaranteed to converge to their true values.

Part II

Applications

Chapter 6

UAV Coordination for Vision-Based Target Tracking

Some of the content in this chapter comes from [85]:

2015 IEEE. Reprinted, with permission, from S. A. P. Quintero, D. A. Copp, and J. P. Hespanha, Robust UAV coordination for target tracking using output-feedback model predictive control with moving horizon estimation, 2015 American Control Conference (ACC), July 2015.

In this chapter we consider the control of two UAVs tracking an evasive moving ground vehicle. The UAVs are small fixed-wing aircraft equipped with gimbaled cameras and must coordinate their control actions so that at least one UAV is always close to the target. The control actions of the UAVs are computed based on noisy measurements of the UAVs' current state and vision-based measurements of the target's position corrupted by state-dependent noise. We propose a novel approach for solving this problem in which the state estimate and optimal control are computed simultaneously online using model predictive control with moving horizon estimation. The efficacy of this approach is demonstrated in simulation results using realistic vision-based measurements of the

target's position. These results show that while using noisy, partial information about the target state, the UAVs are able to coordinate their distances to the target in the ideal case of constant target velocity as well as perform robustly when the target becomes evasive.

6.1 Introduction

Small unmanned aerial vehicles (UAVs) are comparatively inexpensive mobile sensing platforms that are becoming ubiquitous due to their ability to autonomously perform tasks that would be either too demanding, dangerous, or mundane/repetitive for a human operator. Such tasks include agricultural monitoring, exploration and mapping, search and rescue, and surveillance and tracking, to name a few. One particularly challenging problem of interest is that of performing *vision-based target tracking* with a small fixed-wing UAV traveling at a constant airspeed and fixed altitude. In this problem, a camera-equipped UAV is responsible for measuring and tracking the position of a vehicle moving unpredictably in the ground plane.

In vision-based target tracking, image processing software is responsible for determining the centroid pixel coordinates of the ground target moving in the image frame. Using these pixel coordinates, along with the intrinsic and extrinsic camera parameters and terrain data, one can estimate the three-dimensional location of the target in inertial coordinates and compute the associated error covariance [71]. This vision-based measurement of the target's position is also referred to as the *geolocation* estimate. The error associated with the geolocation estimate is highly sensitive to the UAV's position relative to that of the target. As the UAV's planar distance from the target increases, the associated error covariance grows and becomes significantly elongated in the viewing direction. When a UAV is directly above the target, the measurement error is smallest, as the

corresponding error ellipse is circular. Thus, a UAV would ideally hover directly above the target, but the relative dynamics between a fixed-wing UAV and a moving ground target typically preclude this viewing position from being maintained over a period of time. To mitigate a single UAV's inability to maintain close proximity to the target, one can employ multiple UAVs to gather measurements, which are then fused to obtain an improved geolocation estimate. This is referred to as cooperative (or coordinated) target tracking.

Considerable work has been done in the general area of coordinated target tracking, with coordinated standoff tracking comprising the greatest body of work in this area. In standoff tracking, two UAVs orbit the target at a nominal standoff distance while maintaining orthogonal viewing angles. This practice minimizes the joint/fused geolocation error covariance at the fixed nominal standoff distance, as the individual measurement error ellipses are orthogonal [35]. Standoff tracking with perfect knowledge of the target state has been studied in [66] and [53] where the most prevalent control strategies involve the use of vector fields and nonlinear feedback. Approaches with only partial information of the target state are presented in [83], [104], and [52]. The authors of these works utilize observers, adaptive control, and extended Kalman filtering to estimate the full target state. Note that [52] utilizes nonlinear model predictive control to achieve the desired standoff configuration for a target that accelerates but is not necessarily evasive.

The preceding works have designed UAV coordination policies that attempt to improve the estimate of the target state without directly solving a dynamic optimization that minimizes some metric of the estimation error. However, a number of works have employed optimal control to achieve this objective. Miller et al. utilize the framework of partially observable Markov decision processes (POMDPs) in [78] to enable two UAVs to track a moving ground target and present a new approximate solution, as nontrivial POMDP problems are typically intractable to solve exactly [105]. Stachura et al. [102]

employ online receding horizon control to enable two variable-airspeed UAVs to track a stochastic ground target using bearing-only sensors in the presence of packet losses when communicating with the base station where target state estimation takes place. In [30], Ding et al. study the problem of optimally controlling two Dubins vehicles and their pan-tilt-zoom cameras to maximize the geolocation information of a stochastic ground target and show that maintaining orthogonal viewing angles is essential in the case of terrestrial pursuit vehicles and less pronounced for airborne vehicles. While the preceding optimization-based methods consider short planning horizons, e.g., $2-3$ seconds, Quintero et al. consider the optimal coordination of two Dubins vehicles to gather the best joint vision-based measurements of a constant-velocity target over long planning horizons of at least one minute [87], where no restrictions are placed on the vehicles other than kinematics. The results show that coordination of the distances to target is more effective for achieving the said goal than the traditional practice of solely coordinating viewing angles.

In this work, there are a number of real-world conditions to which we wish to be explicitly robust. First, we assume that the only information available for feedback is noisy measurements of each full UAV state and noisy vision-based measurements of the target's position. Thus, the target's velocity is unmeasured and must be estimated. Second, unlike any of the previous approaches on cooperative target tracking, we consider evasive target motion. Since the UAVs are fixed-wing aircraft, and are therefore subject to a minimum turning radius, they must carefully consider the impact of current control actions on future tracking performance in light of state estimation error and evasive target maneuvers. Moreover, robustness to both measurement noise and evasive target motion is crucial to the success of vision-based target tracking.

An output-feedback control approach that can be used to achieve the desired robustness was recently introduced by Copp and Hespanha in [22, 25] and combines robust

model predictive control (MPC) with moving horizon estimation (MHE). As described in [9] and [19], robust MPC involves an online dynamic optimization aimed at minimizing a cost function over a finite planning horizon in light of worst-case disturbances on a dynamical system. MHE also involves an online optimization problem but for the purpose of state estimation of nonlinear systems, and it has been shown to have advantages over state-of-the-art alternatives [94]. While the two optimizations have traditionally been done separately, in the framework of [22, 25], the two are combined into a single min-max optimization. More specifically, a desired cost function is maximized with respect to disturbance and measurement noise variables and minimized with respect to control input variables. The min-max optimization provides state estimates over a fixed, finite window into the past and an optimal control input sequence into the future that is simultaneously robust to worst-case estimates of the state as well as worst-case disturbances to the plant. This combined robust MPC/MHE approach is demonstrated here as a viable, practical solution for the present particularly challenging nonlinear problem of autonomous vehicle coordination.

The remainder of the chapter is organized as follows. Section 6.2 describes the dynamics and measurement model that compose the problem of vision-based target tracking. Section 6.3 discusses the cost function and the robust output-feedback MPC/MHE solution. Section 6.4 presents and discusses simulation results for multiple scenarios. Finally, Section 6.5 provides conclusions and plans for future work.

6.2   Problem Formulation

Consider two camera-equipped UAVs tasked with estimating the state of a target vehicle moving evasively in the ground plane. The UAVs fly at a fixed airspeed and constant altitude and are subject to a minimum turning radius. The target vehicle moves

136

in the ground plane and is subject to a maximum acceleration and maximum speed that is less than the UAVs' groundspeed, which is the same as its airspeed in the ideal case of no wind. Each UAV makes measurements of the target's position using a gimbaled video camera, and we assume that the target is detected at all times and kept in the center of the camera's field of view by onboard software. We first discuss the dynamical models for each type of vehicle and then proceed to describe their measurement models.

6.2.1　UAV Dynamics

The Dubins vehicle is a planar vehicle that moves forward at a fixed speed and has a bounded turning radius. It is commonly used to provide a simple model for UAVs flying at a fixed altitude. We assume that UAV j, where $j \in \{1, 2\}$, flies at a constant speed s_j and at a fixed altitude h_j, and it has a bounded turning rate u_j with maximum absolute upper bound $\bar{u} \in \mathbb{R}_{>0}$, which we take to be the same for both UAVs. Accordingly, $u \in \mathcal{U} := [-\bar{u}, \bar{u}] \times [-\bar{u}, \bar{u}]$. We denote by $\xi^{(j)} = [\xi_1^{(j)} \ \xi_2^{(j)} \ \xi_3^{(j)}]^\top \in \mathbb{R}^3$ the state of UAV j, which comprises its planar position $\mathfrak{p}_j := (\xi_1^{(j)}, \xi_2^{(j)})$ and its heading $\psi_j := \xi_3^{(j)}$, all of which are measured in a local East-North-Up coordinate frame. The kinematics of UAV j are given by

$$\frac{d\xi^{(j)}(t)}{dt} = F(\xi^{(j)}, u_j) := \begin{pmatrix} s_j \cos \xi_3^{(j)} \\ s_j \sin \xi_3^{(j)} \\ u_j, \end{pmatrix}. \tag{6.1}$$

While the majority of work on target tracking treats the problem in continuous time, this work addresses the problem in discrete time since measurements of the target's position are available at discrete time instances $t = kT_s$ seconds, where $k \in \mathbb{Z}_{\geqslant 0}$ and $T_s > 0$ is the measurement sampling period. Accordingly, we assume a zero-order hold

137

(ZOH) of T_s seconds on each UAV's control input. The discrete-time equations of motion provided in [87] are given by

$$\xi^{(j)^+} = f_a(\xi^{(j)}, u_j),$$ (6.2)

where the subscript "a" refers to the fact that the discrete-time dynamics are those of an air vehicle.

While the equations of motion for the Dubins vehicle in discrete time are readily derived in [87], they involve a $1/u_j$ term that becomes problematic for numerical optimizations and is hence avoided. Instead, we approximate the Dubins vehicle model using a second order Lie series:

$$f_a(\xi^{(j)}, u_j) \approx \xi^{(j)} + T_s F(\xi^{(j)}, u_j) + \cdots + \frac{T_s^2}{2} \frac{\partial F}{\partial \xi^{(j)}} F(\xi^{(j)}, u_j),$$ (6.3)

where $\partial F / \partial \xi^{(j)}$ denotes the Jacobian of (6.1) that does not involve division by the control input. The Lie series is a good approximation to the Dubins vehicle dynamics since the heading dynamics are exact and, with the simulation parameters considered in this work, the position error from (6.3) corresponding to $u_j = \bar{u}$ is less than 4% of the total distance traveled. In what follows, the approximation of (6.3) is used only in the numerical optimization of Section 6.3 while the simulation results of Section 6.4 utilize (6.2) to propagate the discrete-time dynamics forward in time.

6.2.2   Target Dynamics and Overall State Space

We place no nonholonomic constraints on the ground vehicle and simply model the target as a double integrator moving in the ground plane. The state of the target is denoted by $\eta = [\eta_1 \ \eta_2 \ \eta_3 \ \eta_4]^\top \in \mathbb{R}^4$, where $\mathfrak{p}_g := (\eta_1, \eta_2)$ refers to the target's planar posi-

tion in the same local East-North-Up coordinate frame as the UAVs. The corresponding target velocity is given by $\mathfrak{v} := (\eta_3, \eta_4)$, and the acceleration inputs of the target are denoted by $d \in \mathbb{R}^2$. We assume a T_s- second ZOH on the target's acceleration input synchronized with that of both UAVs, yielding the straightforward discrete-time linear dynamics

$$\eta^+ = f_g(\eta, d) = A\eta + Bd, \tag{6.4}$$

where

$$A = \begin{bmatrix} I_2 & T_s I_2 \\ 0 \cdot I_2 & I_2 \end{bmatrix} \text{ and } B = \begin{bmatrix} (T_s^2/2)I_2 \\ T_s I_2 \end{bmatrix}.$$

Here, I_n is an $n \times n$ identity matrix. To keep the problem realistic and well-posed, we take the target's acceleration input d to belong to

$$\mathcal{D} := \{d \in \mathbb{R}^2 | \|\mathfrak{v} + dT_s\|_2 \leqslant \bar{v}, \|d\|_\infty \leqslant \bar{d}\}, \tag{6.5}$$

where $\bar{v}$ is the maximum allowable target speed and $\bar{d}$ is the maximum absolute acceleration along either the East or North directions. Typically, we take $\bar{v}$ to be less than the smaller of the two UAV airspeeds so that the problem is well-posed.

Now that we have presented all vehicle models, we define the overall state as $x := (\xi^{(1)}, \xi^{(2)}, \eta) \in \mathbb{R}^{10}$. The overall dynamics are thus given by

$$x^+ = f(x, u, d) := \begin{pmatrix} f_a(\xi^{(1)}, u_1) \\ f_a(\xi^{(2)}, u_2) \\ f_g(\eta, d) \end{pmatrix}. \tag{6.6}$$

139

6.2.3   Measurement Error Models

We turn our attention to the overall measurement model in vision-based tracking. The measurement vector associated with the state of UAV j is denoted by $y_a^{(j)} \in \mathbb{R}^3$ and is given by

$$y_a^{(j)} = \xi^{(j)} + w_a^{(j)}, \qquad w_a^{(j)} = [w_{a,1}^{(j)} \; w_{a,2}^{(j)} \; w_{a,3}^{(j)}]^\top, \tag{6.7}$$

where $(w_{a,1}^{(j)}, w_{a,2}^{(j)}) \sim \mathcal{N}(0, \sigma_\mathfrak{p}^2 I_2)$, $w_{a,3}^{(j)} \sim \mathcal{N}(0, \sigma_\psi^2)$, $\sigma_\mathfrak{p}^2$ is the variance of the uncorrelated noise on the UAV's North and East position coordinates, and σ_ψ^2 is the variance on the UAV's heading angle.

Each UAV's camera makes image-plane measurements of the target. The dominant source of geolocation error arises from the error in the sensor attitude matrix $T_S^T(\boldsymbol{\theta}_j)$ that relates the coordinates of the line-of-sight vector $\mathfrak{u}_j^S$ from the UAV to the target in the North-East-Down sensor frame (centered at UAV j's position) to the coordinates of the same vector in the local East-North-Up topographic coordinate frame. This transformation is a nonlinear function of the 3-2-1 Euler-angle sequence of yaw, pitch, and roll denoted by $\boldsymbol{\theta}_j \in \mathbb{R}^3$. Image tracking software controls the camera's gimbal platform to keep the target in the center of the camera's field of view and reports the Euler angles of the camera sensor as well as the line-of-sight vector $\mathfrak{u}_j^S$. Here, a superscript "S" denotes a quantity in the sensor coordinate frame while the absence thereof indicates a quantity in the topographic coordinate frame.

The 3-dimensional target position measured by UAV j with 3D position $\mathfrak{s}_j = [\mathfrak{p}_j^\top, h_j]^\top$ is denoted by $\mathfrak{o}_j$. Its estimate is given by

$$\hat{\mathfrak{o}}_j = \hat{\mathfrak{s}}_j + \hat{r}_j T_S^T(\hat{\boldsymbol{\theta}}_j)\mathfrak{u}_j^S = \hat{\mathfrak{s}}_j + \hat{r}_j \hat{\mathfrak{u}}_j, \tag{6.8}$$

where

$$\hat{\mathfrak{s}}_j = \mathfrak{s}_j + \tilde{\mathfrak{s}}, \quad \tilde{\mathfrak{s}} \sim \mathcal{N}(0, \operatorname{diag}(\sigma_{\mathfrak{p}}^2 I_2, \sigma_a^2)),$$
$$\hat{\boldsymbol{\theta}}_j = \boldsymbol{\theta}_j + \tilde{\boldsymbol{\theta}}, \quad \tilde{\boldsymbol{\theta}} \sim \mathcal{N}(0, \sigma_\psi^2 I_3),$$

(6.9)

and σ_a^2 denotes the variance of the measurement noise on the UAVs' altitude h_j. Also, the 3D distance from UAV j to the target is denoted by $r_j = \|\mathfrak{o}_j - \mathfrak{s}_j\|_2$, and its estimate $\hat{r}_j$ is provided by the flat-Earth approximation $\hat{r}_j = (h_0 - \hat{\mathfrak{s}}_{j,3})/\hat{u}_{j,3}$, where h_0 is the height of the ground plane in the topographic coordinate frame and is taken to be zero in this work without loss of generality. Since all camera angles are measured with respect to the UAV attitude, we take the noise on the estimate of the camera's attitude to be the same as that on the estimate of the UAV's attitude. Thus, the noise on the estimate of UAV j's heading angle is the same as that on the estimate of its camera's yaw angle, i.e., the first element of $\boldsymbol{\theta}_j$.

From the preceding measurement equation, one can show that the covariance $P_{\mathfrak{o},j} \in \mathbb{R}^{3\times3}$ associated with the error $\tilde{\mathfrak{o}}_j := \hat{\mathfrak{o}}_j - \mathfrak{o}_j$ in the three dimensional position of the target is proportional to the product of r_j^2 and the covariance of the Euler-angle sequence estimate $\hat{\boldsymbol{\theta}}_j$ given in (6.9). The exact analytic expression for $P_{\mathfrak{o},j}$ is derived in [87] and is omitted here for brevity. Since we are tracking in the ground plane, only the upper left 2×2 submatrix of $P_{\mathfrak{o},j}$ is relevant and is denoted by P_j.

Since the UAVs collect independent measurements of the target, the fused measurement y_g of the target's true position $\mathfrak{p}_g$ can be computed using the best linear unbiased estimate, which is as follows:

$$y_g = \mathcal{P}(P_1^{-1}\hat{\mathfrak{p}}_g^{(1)} + P_2^{-1}\hat{\mathfrak{p}}_g^{(2)}) = [I_2 \; \mathbf{0}_{2\times2}]\eta + w_g = \mathfrak{p}_g + w_g \,, \qquad (6.10)$$

where $\mathcal{P} = (P_1^{-1} + P_2^{-1})^{-1}$, $w_g \sim \mathcal{N}(0, \mathcal{P})$, $\hat{\mathfrak{p}}_g^{(j)} = [I_2 \; \mathbf{0}_{2\times1}]\hat{\mathfrak{o}}_j$, and $\mathbf{0}_{m\times n}$ denotes the

$m \times n$ zeros matrix. The confidence ellipse corresponding to the fused geolocation error covariance (GEC) $\mathcal{P}$ has the property that it is small when at least one UAV is close to the target and only slightly less when both aircraft are directly above the target. Therefore, it is advantageous for at least one UAV to be near the target at any given time.

Finally, the measurement model corresponding to the overall state x is given by combining (6.7) and (6.10) as follows:

$$y \coloneqq (y_a^{(1)}, y_a^{(2)}, y_g) = Cx + w, \tag{6.11}$$

where $w \coloneqq (w_a^{(1)}, w_a^{(2)}, w_g)$ and $C \coloneqq [I_8 \ \mathbf{0}_{8\times2}]$. Since the target velocity is not measured directly, the control law used in this framework will be based on output feedback.

6.3   Robust Output-Feedback MPC/MHE

When considering only one UAV, the target tracking problem can be regarded as a two-player zero-sum game in which the UAV tries to minimize its 3D distance to the target, r, and the target tries to maximize r. In the two-UAV case, the UAVs ideally coordinate their movements in order to ensure that at least one UAV is close to the target to keep the fused GEC comparatively low. Additionally, the UAVs should keep their individual distances to the target sufficiently small to maintain adequate resolution of the target in the camera's image plane for effective visual detection. This motivates us to choose the following criterion

$$g(x) \coloneqq \beta_1 \frac{r_1^2 r_2^2}{r_1^2 + r_2^2} + \beta_2(r_1^2 + r_2^2), \tag{6.12}$$

142

where β_1 and β_2 are positive weighting constants, that the UAVs (Player 1) would like to minimize and the target (Player 2) would like to maximize. The first term in (6.12) is motivated by noting that the size of the confidence ellipse associated with P_j is proportional to r_j^2 and that the fused GEC has the form $\mathcal{P} = (P_1^{-1} + P_2^{-1})^{-1}$. Moreover, the previous matrix expression is simplified to one that is scalar and more compatible with numerical optimization by replacing the individual covariances with the respective 3D distances. This term enforces distance coordination so that one UAV is always close to the target to improve measurement quality, just as in [87]. The second term in (6.12) penalizes the individual UAV distances to the target to ensure that the size of the target in each UAV's image plane is sufficiently large for reliable detection by image processing software. While other optimality criterion may be considered, we aim to utilize a simpler expression than those found in [30] and [87] that achieves similar behavior and lends itself to efficient numerical computation. We shall see that the distance coordination of [87] is indeed induced by choosing the criterion (6.12).

For our control approach, we use the output-feedback MPC with MHE approach described in [22, 25]. This requires us to solve a finite-horizon online optimization problem at each time k. Solving this online optimization problem uses output measurements from the last L steps into the past in order to give us an estimate of the current state at time k (the MHE problem), and from that, give us policies for both Players 1 and 2 to use for the next K steps into the future (the MPC problem).

Specifically, the control objective is to select the control signal $u_k \in \mathcal{U}$, $\forall k \in \mathbb{Z}_{\geqslant 0}$ so as to minimize a criterion of the form

$$\sum_{\ell=k}^{k+K} g(x_\ell) - \sum_{\ell=k-L}^{k} \lambda_n \|w_\ell\|_2^2, \tag{6.13}$$

where $g(x)$ is given by (6.12) and λ_n is a positive scalar, for worst-case values of $d_k \in \mathcal{D}$

143

and measurement noise $w_k \in \mathbb{R}^{n_w}$, $\forall k \in \mathbb{Z}_{\geqslant 0}$.

Given a discrete-time signal $z : \mathbb{Z}_{\geqslant 0} \to \mathbb{R}^n$ and two times $k_0, k \in \mathbb{Z}_{\geqslant 0}$ with $k_0 \leqslant k$, we denote by $z_{k_0:k}$ the sequence $\{z_{k_0}, z_{k_0+1}, ..., z_k\}$. This notation allows us to re-write (6.13) and define our cost function as

$$J_k(x_{k-L}, u_{k-L:k+K-1}, d_{k-L:k+K-1}, y_{k-L:k}) \coloneqq \sum_{\ell=k}^{k+K} g(x_\ell) - \sum_{\ell=k-L}^{k} \lambda_n \|y_\ell - Cx_\ell\|_2^2 \quad (6.14)$$

which emphasizes the dependence of (6.13) on the unknown initial state x_{k-L}, the unknown input sequence for the target $d_{k-L:k+K-1}$, the measured output sequence $y_{k-L:k}$, and the control input sequence $u_{k-L:k+K-1}$. This control input sequence $u_{k-L:k+K-1}$ comprises two distinct sequences: the (known) past inputs $u_{k-L:k-1}$ that have already been applied and the future inputs $u_{k:k+K-1}$ that still need to be selected.

At a given time $k \in \mathbb{Z}_{\geqslant 0}$, we do not know the values x_{k-L} and $d_{k-L:k+K-1}$ on which the criterion (6.14) depends, so we optimize this criterion under worst-case assumptions on these variables, leading to the following finite-dimensional min-max optimization

$$\min_{\hat{u}_{k:k+K-1|k} \in \mathcal{U}} \quad \max_{\substack{\hat{x}_{k-L|k} \in \mathcal{X}, \\ \hat{d}_{k-L:k+K-1|k} \in \mathcal{D}}} \quad J_k(\hat{x}_{k-L|k}, u_{k-L:k-1}, \hat{u}_{k:k+K-1|k}, \hat{d}_{k-L:k+K-1|k}, y_{k-L:k}), \quad (6.15)$$

where the arguments $u_{k-L:k-1}, \hat{u}_{k:k+K-1|k}$ correspond to the sequence $u_{k-L:k+K-1}$ in the definition of $J_k(\cdot)$ in (6.14). The subscript $\cdot_{|k}$ in the (dummy) optimization variables in (6.15) emphasizes that this optimization is repeated at each time step $k \in \mathbb{Z}_{\geqslant 0}$. At different time steps, these optimizations typically lead to different solutions, which generally do not coincide with the real control input, target input, and noise. We can view the optimization variables $\hat{x}_{k-L|k}$ and $\hat{d}_{k-L:k+K-1|k}$ as (worst-case) estimates of the initial state and target input, respectively, based on the past inputs $u_{k-L:k-1}$ and outputs $y_{k-L:k}$ available at time k.

Inspired by model predictive control, at each time k, we use as the control input the first element of the sequence

$$\hat{u}^*_{k:k+K-1|k} = \{\hat{u}^*_{k|k}, \hat{u}^*_{k+1|k}, \hat{u}^*_{k+2|k}, \ldots, \hat{u}^*_{k+K-1|k}\} \in \mathcal{U}$$

that minimizes (6.15), leading to the following control law:

$$u_k = \hat{u}^*_{k|k}, \quad \forall k \geqslant 0. \tag{6.16}$$

6.4   Simulation Results

We now demonstrate the effectiveness of the proposed robust MPC/MHE control approach to the problem of vision-based target tracking with two UAVs. In particular, we simulate two scenarios with realistic levels of noise in the models presented in Section 6.2. Firstly, we quantify the performance of the proposed approach in an ideal scenario wherein the target is traveling at a constant velocity and secondly in a scenario in which the target follows its worst-case strategy determined by (6.15).

The parameters pertaining to both simulation scenarios are provided in Table 6.1. Given the maximum UAV turn rate, the total time it takes a UAV to make a full loop is $2\pi/\bar{u} \approx 13.66$ seconds, and hence the future planning horizon of $KT_s = 7$ seconds allows the UAVs to consider the impact of beginning to loop around the target. Additionally, in the cost function (6.12), the coefficients β_1 and β_2 were chosen to place a greater emphasis on distance coordination than on keeping individual distances small.

To solve the min-max optimization problem (6.15), we use Algorithm 4.4.2 described in Chapter 4. Typical execution times for solving the optimization problem at each time step using the C programming language on a laptop with a 2.3 GHz Intel® Core™

Table 6.1: Simulation Parameters

Parameter	Description	Value	Units
$\bar{u}$	Max UAV turn rate	0.46	rad./s
(s_1, s_2)	UAV speeds	$(15, 15)$	m/s
$\bar{d}$	Max target accel.	$3/\sqrt{2}$	m/s^2
$\bar{v}$	Max target speed	10	m/s
$\sigma_{\mathfrak{p}}^2$	N/E position variance	2.5^2	m^2
$\sigma_{\mathfrak{a}}^2$	Altitude variance	4^2	m^2
σ_{ψ}^2	Euler angle variance	3^2	deg.^{2}
(h_1, h_2)	UAV altitudes	$(40, 45)$	m
β_1	Coord. coefficient	$4 \cdot 10^{-2}$	-
β_2	Dist. coefficient	$4 \cdot 10^{-3}$	-
λ_n	Noise coefficient	10	-
T_s	Sampling period	1	s
L	Backward horizon	7	-
K	Forward horizon	7	-

i7 processor are near 5 milliseconds with a maximum execution time not exceeding 20 milliseconds. Since these execution times are much less than the $T_s = 1$ second sampling period, the approach presented here is suitable for online real-world implementation.

Because the simulations incorporate stochastic measurement noise, we quantify performance based on $M = 1,000$ Monte Carlo simulations to determine the steady-state tracking performance of each scenario. More specifically, 3 minutes of steady-state behavior are considered, where the effects of initial conditions have been removed by discarding 30 seconds of initial data. We initialize the problem with the UAVs circling a stationary target until the past measurement buffer is filled. Figure 6.1 depicts an instance of the output of the optimization (6.15) at time k for the parameters given in Table 6.1. This plot illustrates the past positions, measurements, and estimates of the UAVs and target as well as the planned future trajectory for each agent generated at the current time k. In this instance, we see that the target just executed a sharp turn in order to take advantage of the UAVs' minimum turning radius of $\underline{r} \approx 32.6$ [m].

146
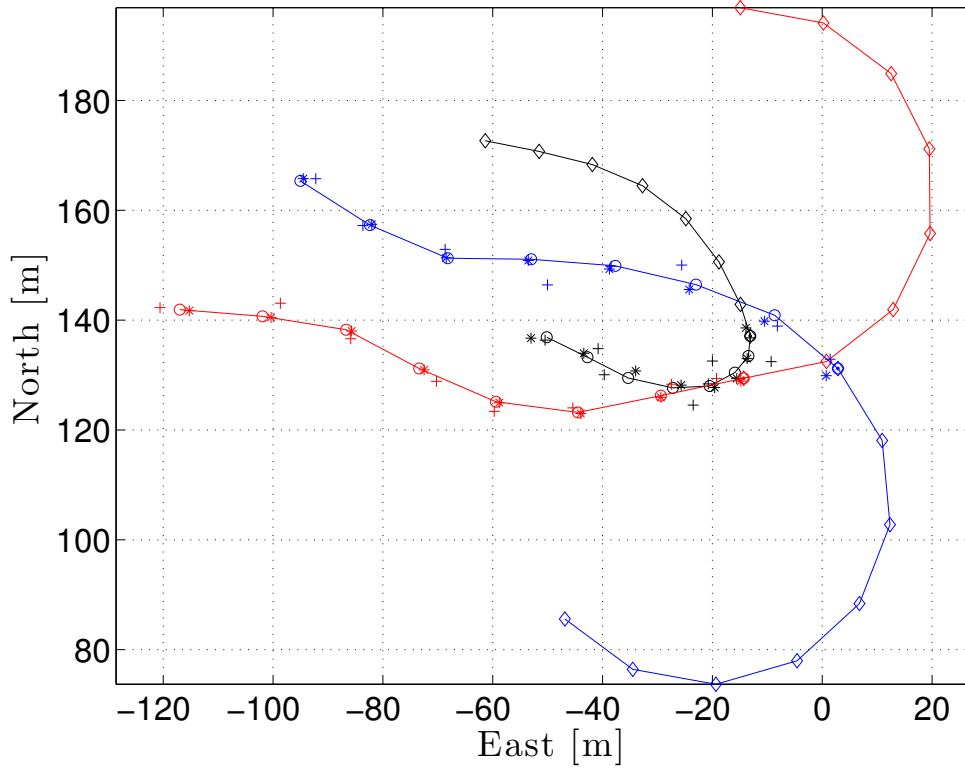
Figure 6.1: The output of the robust MPC/MHE optimization. Red and blue markers depict quantities related to the UAVs while black markers indicate quantities related to the target. The "+" markers illustrate the vehicles' noisy position measurements while the "∗" markers depict ground truth. The "○" markers indicate state estimates from the min-max optimization while "◇" markers illustrate future positions corresponding to the optimal control sequence determined by the same optimization. A "·" marker indicates the estimate of each vehicle's current position. The target trajectory begins near the center of the plot and ends heading Northwest while the leftmost UAV markers indicate their starting positions.
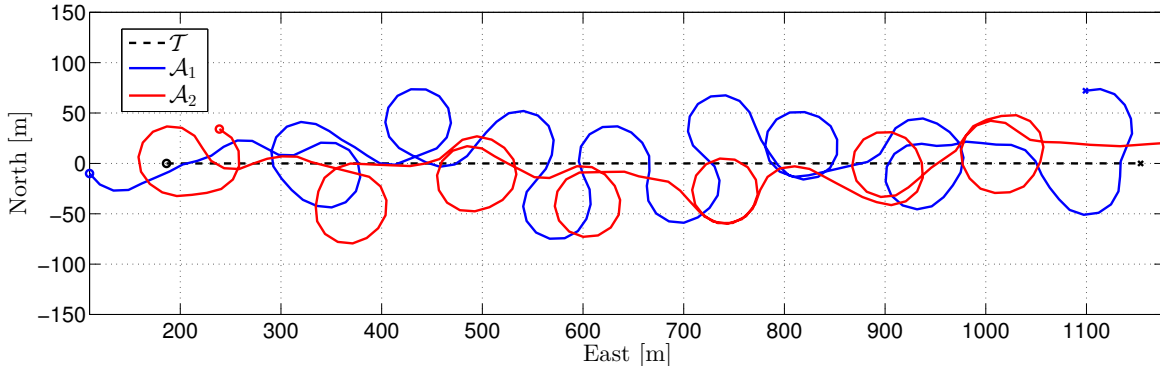
Figure 6.2: Actual trajectories of two UAVs tracking a constant-velocity target over a 3-minute window. The starting positions of all vehicles are denoted by an "∘" while the ending positions are indicated by an "×". In the legend, $\mathcal{T}$ corresponds to the target while $\mathcal{A}_1$ and $\mathcal{A}_2$ refer to the UAVs.

We first consider a constant-velocity target, i.e., $d_k \equiv 0$ in equation (6.4), and the target travels at just over one third of the UAVs' fixed speed. The UAVs are still performing the min-max optimization of (6.15) and hence are planning for an evasive target. The results for one of the Monte Carlo simulations are provided in Figures 6.2 and 6.3. In Figure 6.2, one can see how the UAVs make loops and "S" turns so that their average speed matches that of the target. Figure 6.3 indicates that this is done in a coordinated fashion so that at least one UAV is never very far from the target, as indicated by the dashed cyan curve depicting $\min\{r_1, r_2\}$. We shall soon verify that coordination is consistently achieved across all runs for a constant-velocity target using the robust MPC/MHE approach with the given stage cost.

We now consider an evasive target, i.e., it is using the optimal worst-case d_k^* computed from the min-max optimization (6.15). Results for one of the Monte Carlo simulations are provided in Figures 6.4 and 6.5. By observing the vehicle trajectories in Figure 6.4, one can see that the optimal trajectory for the target is quite erratic. Indeed, the target takes advantage of the UAVs' kinematic constraints by making sharp turns and forcing the UAVs to make loops at their maximum allowable turn rate. This is corroborated
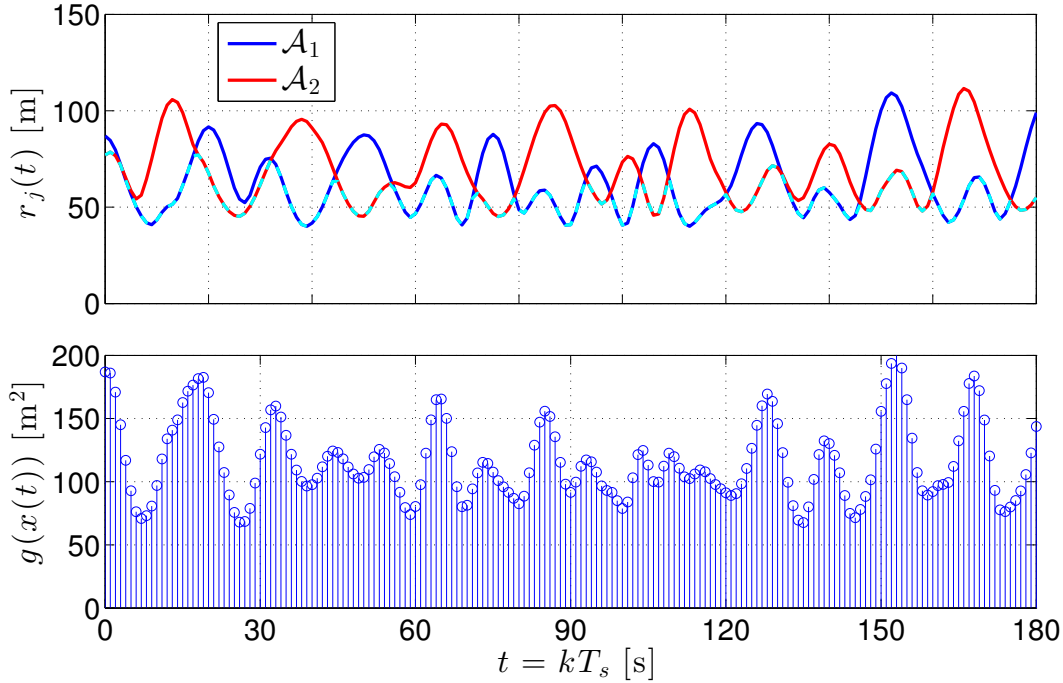
Figure 6.3: 3D distances r_j and stage cost $g(x)$ for two UAVs tracking a constant velocity target.

by the fact that over the $1{,}000$ runs, the modes of the sample distributions (not shown) of the absolute control input values for all vehicles are concentrated at their maximum allowable values. From Figure 6.5, one can see that distance coordination has diminished, and the stage cost peaks over 300 $[\mathrm{m}^2]$ twice, whereas it never exceeded 200 $[\mathrm{m}^2]$ in the case of a constant-velocity target.

A quantitative summary of the $M = 1{,}000$ Monte Carlo runs is provided in Table 6.2, where sample statistics are computed over both time and samples. From the data, one should notice the detrimental effect that adversarial target motion has on the coordination effort of the UAVs. This is indicated by the sample Pearson correlation coefficient ϱ for the UAV distance pairs r_1 and r_2, where ϱ is in general a measure of the linear correlation between two random variables and belongs to the interval $[-1, 1]$. A more negative value for ϱ indicates stronger anti-correlation, which in the present setting implies that when
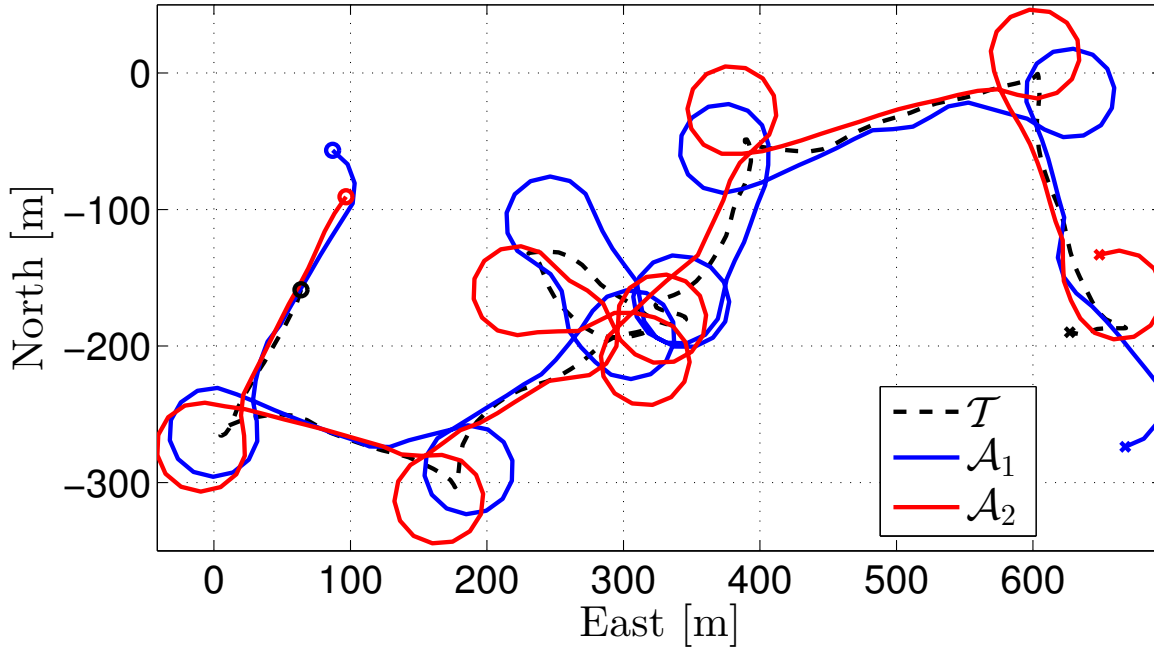
Figure 6.4: Actual trajectories of two UAVs tracking an evasive target over a 3-minute window. The same notation as Figure 6.2 is employed here.

Table 6.2: Statistics for 1,000 3-minute Monte Carlo Simulations

Statistic	Const. Vel.	Evasive	Units
avg $g(x)$	118.4	139.3	m^2
var $g(x)$	1,163.8	3,439.4	m^4
max$\{r_1, r_2\}$	154.2	169.7	m
avg $\|\mathfrak{p}_g - \hat{\mathfrak{p}}_g\|$	4.92	4.66	m
avg $\|w_g\|$	4.72	5.24	m
ϱ	-0.307	0.206	N/A

one UAV is relatively far from the target, the other is likely to be rather close. Thus, one can see that the constant velocity scenario has substantial distance coordination across all simulation runs; however, the evasive target is able to disrupt the UAV coordination and induce positive correlation. Nonetheless, the average cost sees only a moderate increase of nearly 18% when the target becomes evasive while the maximum value of the individual UAV distances from the target only experiences an increase of about 10%.

Concerning estimation of the target 's position, one should first observe that average
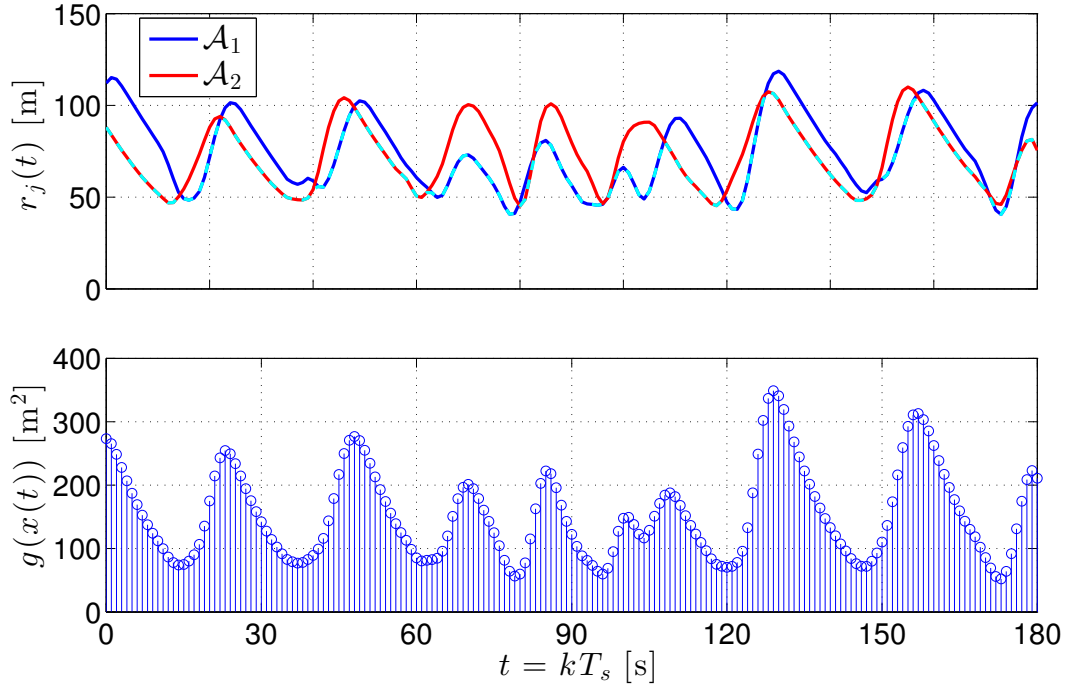
Figure 6.5: 3D distances r_j and stage cost $g(x)$ for two UAVs tracking an evasive target.

measurement error (w_g) on the target's position increases when the target is evasive, as the fused error covariance increases with the UAVs' inability to keep at least one aircraft close at all times. However, the estimation error becomes smaller, which shows that the target's deviation from its optimal worst-case evasion leads to a smaller cost for (6.14). This means that in the evasive case, the first term of (6.14) that quantifies tracking performance has increased and the magnitude of the second (negative) term regarding state estimation has decreased, which can be observed in the first and fourth rows of Table 6.2.

Overall, the target's evasive maneuvers hinder UAV coordination efforts and thereby increase measurement noise. Nonetheless, the UAVs are still able to robustly track the target in the sense that their maximum 3D distance from the target is not only bounded, but also only slightly larger than in the case of constant target velocity.

6.5   Conclusion and Future Work

We considered two UAVs performing vision-based target tracking of a moving ground vehicle. We showed that a novel approach based on min-max MPC combined with MHE is viable for solving this high-dimensional, very nonlinear (non convex) problem. Using this approach with the commonly assumed case of having a constant-velocity target, the UAVs coordinate their distances from the target and achieve very small tracking cost, even though only noisy, partial information about the target state is available. In practice, not only is the full state of the target and UAVs not precisely known, but sharp changes in target velocity occur when the target vehicle makes turns and moves erratically or evasively. Though past work on UAV target tracking does not often consider this case, we have shown that our approach is robust not only to evasive target motion, but also to noisy measurements in the output-feedback setting of vision-based target tracking.

The interested reader can find further work on this problem in the book chapter [86]. This reference includes the extension of this proposed approach utilizing a more realistic UAV model that addresses roll dynamics. In addition, results from a high fidelity flight simulator for the UAVs with actual target trajectories captured from real-world tracking experiments are given. Finally, the extensions in [86] consider unmeasured wind disturbances. These wind disturbances are included as optimization variables as in the examples described in Chapter 5, so the controller adapts to and learns the unknown wind disturbance online.

6.6   Acknowledgments

We would like to thank Dr. Steven Quintero for collaborating on this work.

Chapter 7

Asymmetric MPC/MHE for the Treatment of Type 1 Diabetes

A new estimation and control approach for the feedback control of an Artificial Pancreas to treat Type 1 Diabetes Mellitus is proposed. In particular, we present a new output feedback predictive control approach that simultaneously solves the state estimation and control problems as a single min-max optimization. This involves optimizing a cost function with both finite forward and backward horizons with respect to the unknown initial state, unmeasured disturbances and noise, and future control inputs, and is similar to simultaneously solving a Model Predictive Control (MPC) problem and a Moving Horizon Estimation (MHE) problem. We incorporate a novel asymmetric output cost in order to penalize dangerous low blood-glucose values more severely than less harmful high blood-glucose values. We compare this combined MPC/MHE approach to a control strategy that uses state-feedback MPC preceded by a Luenberger Observer for state estimation. *In-silico* results showcase several advantages of this new simultaneous MPC/MHE approach, including fewer hypoglycemic events without increasing the number of hyperglycemic events, faster insulin delivery in response to meal consumption,

and fewer and shorter insulin pump suspensions, resulting in smoother blood-glucose trajectories. The results in this chapter may also be found in [21].

7.1   Introduction

Type 1 Diabetes Mellitus (T1DM) is a metabolic auto-immune disease that destroys pancreatic β-cells, making it impossible for the pancreas to produce insulin, a hormone the body uses to regulate glucose levels in the blood stream and to facilitate the absorption of glucose into many types of cells. Because of this, people with T1DM require monitoring of blood-glucose (BG) levels and delivery of insulin from an external source. If BG levels are not regulated well, people with T1DM suffer from hyperglycemia and hypoglycemia (high and low BG levels, respectively), which can cause severe health problems. An individual who experiences hyperglycemia over long periods of time may, for example, eventually experience cardiovascular disease, kidney failure, and retinal damage, possibly after many years. Hypoglycemia, on the other hand, may have immediate consequences ranging from dizziness and unconsciousness to possibly even coma or death.

Much recent research has been devoted to the feedback control of an Artificial Pancreas (AP) in order to reduce the burden and improve the effectiveness of T1DM treatment by automating the dosing and delivery of insulin [44, 20, 31]. In the AP, it is the job of a feedback controller to determine appropriate amounts of insulin to be delivered given measurements of BG levels. This work focuses on control of an AP that delivers insulin using a Continuous Subcutaneous Insulin Infusion (CSII) pump and receives BG measurements based on a Continuous Glucose Monitor (CGM) [47], as is the case with AP units destined for outpatient use. Given the potentially severe consequences of excessive or insufficient insulin delivery, the algorithms for feedback control of an AP are crucial for the successful treatment of T1DM [31, 10].

One of the most popular control approaches for the delivery of insulin using an AP is Model Predictive Control (MPC) [82, 48, 68]. Given a model of the plant to be controlled and the current state of the plant, MPC involves solving an online optimization problem over a future time horizon. This yields a sequence of optimal control inputs to be applied to the plant in the future as well as predicted states of the plant based on these inputs [79, 95]. Only the first element of the computed input sequence is applied as an input to the plant, and at each sampling time, this technique is repeated. MPC has been one of the most successful advanced control methods in many industries, including the feedback control of an AP, because of its ability to explicitly handle hard state and input constraints. For a survey of MPC applications in industry, see [84].

Classic MPC is formulated assuming full-state feedback. When considering an AP, however, we only have noisy measurements of a person's BG from a CGM, thus when using a state-feedback control strategy one must implement a state-estimator prior to invoking the control law. Examples of algorithms for state estimation include observers, filters, and moving horizon estimation, some of which are discussed in [94]. Some of the authors' past work has involved the use of a Luenberger Observer for state estimation with an AP [39, 37, 38, 40]. The performance of this Luenberger Observer was compared to that of Moving Horizon Estimation (MHE) in [59], and it was found that MHE provided better state estimates and allowed faster insulin delivery in response to meal consumption. In a sense, MHE [90] is the dual of MPC and is attractive because it explicitly handles constraints and computes estimates of the state by solving an online optimization problem given a fixed number of past measurements and inputs.

In this work, we utilize a novel combination of MPC with MHE recently proposed in [22, 25]. Specifically, we consider output feedback using a model that explicitly includes additive measurement noise and input disturbances, and formulate the combined MPC and MHE problem as a single min-max optimization over both control inputs (min) and

the unknown initial state and input disturbances (max). In this way, we solve both the MPC and MHE problems simultaneously, which gives us an optimal (in a certain sense) control input sequence at each sampling time for worst-case (in a certain sense) estimates of the current state, disturbances, and noise.

The BG regulation problem exhibits inherent asymmetry, which poses a challenge when designing a controller. The asymmetry stems from the fact that hypoglycemia has more immediate and dire consequences than hyperglycemia, insulin can only be delivered, not removed, and, in the single-hormone AP considered here, there is no control action available for increasing BG. Therefore, responding assertively to hyperglycemia and commanding corrective insulin, but not over-correcting and thereby inducing subsequent hypoglycemia, is a paramount and difficult challenge. Many AP controllers utilize supervisory control or additional safety logic to address this challenge. In this work, however, we propose an appropriate choice of the MPC cost function to address this challenge without the need for additional ad hoc safeguards. Specifically, we consider an asymmetric output cost that penalizes "riskier" low BG values more severely than high BG values. Asymmetric output costs have been considered by others in works such as [68, 13] but with different implementations.

Our proposed output cost function is not only asymmetric but also assigns very low cost to BG values within a safe range in order to regulate BG values to be within that range, rather than tracking a particular set-point, similar to the AP controllers deployed in clinical trials [39, 42, 40]. In this way, our output cost function enforces a small penalty for BG values within a desired range and sharply (asymmetrically) penalizes excursions outside of that range. This approach has proven useful in AP applications as there exists a set of BG values generally considered to be safe, and it is extremely difficult to obtain accurate physiological models [106]. There is often large plant-model mismatch due to the significant variability in the physiology of a single individual over time as well as of

different individuals. It is also difficult to accurately model the noise and delays that are present in the BG measurements provided by a CGM. Therefore, regulating to a range of BG values is one method to prevent excessive response to changes in the measurements when the measured BG level is safely within the desired range.

To demonstrate the differences and benefits of the combined MPC/MHE strategy, in this paper the results are compared to results from a method that has performed successfully in outpatient clinical trials. The method is a state-feedback MPC strategy that utilizes asymmetric input costs and a Luenberger Observer for state estimation and is described in [39]. Throughout the paper, we refer to this approach as the MPC/LO method. Importantly, for simplicity and to more clearly demonstrate the utility of the cost function within the novel MPC/MHE control law, in this work we do not include many of the safety features necessary for a controller to be deployed in clinical trials. Thus the results presented here are not representative of expected results in trials. Specifically, we do not consider diurnal zones or constraints [39], feed-forward control action following user-initiated meal announcement, nor do we consider insulin-on-board constraints [32]. The purpose of this work is to investigate the benefits of the proposed MPC/MHE method with the simplest possible comparison to an existing successful AP control method. To this end, we demonstrate the benefits of this MPC/MHE approach by presenting *in-silico* studies based on the commercially available 10-subject UVA/Padova U.S. Food and Drug Administration (FDA) accepted metabolic simulator [54]. The *in-silico* results showcase several advantages of the MPC/MHE approach, including fewer hypoglycemic events without increasing the number of hyperglycemic events, faster insulin delivery in response to meal consumption, and fewer and shorter insulin pump suspensions, resulting in smoother BG trajectories.

This chapter is organized as follows: We present the control-relevant model, desired BG range, and input constraints in Section 7.2. In Section 7.3 we describe our combined

MPC/MHE estimation and control approach, and compare it to a simplified version of the approach from [39] utilizing state-feedback MPC with a Luenberger Observer as state estimator. In Section 7.4 we compare the results of the two estimation and control approaches and discuss the advantages of the MPC/MHE approach. Finally, we conclude with closing remarks in Section 7.5.

7.2   Problem Formulation

7.2.1   Insulin-glucose transfer function

Because it is difficult to derive accurate models, and because there are long delays and significant noise in the CGM measurements, accurate estimation and effective control for an AP is exceptionally challenging. We use the control-relevant model proposed in [106], which has successfully been employed in AP controller design in [39, 40]. The model is a discrete-time linear time-invariant (LTI) system with sample period $T_s = 5$ minutes. Denoting the current time by t, the scalar plant input is the administered insulin bolus $u_{\text{IN},t}$ [U] delivered per sample-period, and the scalar plant output is the subject's BG value $y_{\text{BG},t}$ [mg/dL]. The plant is linearized around a steady-state that is assumed to result in a BG output $y_s = 110$ [mg/dL] when applying the subject-specific, basal input rate u_{BASAL} [U/hour].

The input u_t and output y_t of the LTI model are defined as

$$u_t := u_{\text{IN},t} - u_{\text{BASAL}} \times T_s, \qquad y_t := y_{\text{BG},t} - y_s.$$

Denoting z^{-1} as the backwards shift operator, we write $\mathcal{U}(z^{-1})$ and $\mathcal{Y}(z^{-1})$ for the z-transforms of the time-domain signals of input u_t and output y_t, respectively. The

transfer function from u to y is given by

$$\frac{\mathcal{Y}(z^{-1})}{\mathcal{U}(z^{-1})} = \frac{1800g}{u_{\text{TDI}}} \times \frac{z^{-3}}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})^2} \tag{7.1}$$

with poles $p_1 = 0.98$, $p_2 = 0.965$, the subject-specific *total daily insulin* amount u_{TDI} [U], and with the constant

$$g := -90(1 - p_1)(1 - p_2)^2$$

employed to set the correct gain and for unit conversion. The number 1800 comes from the "1800 rule" to estimate BG decrease with respect to delivering rapid-acting insulin [108].

7.2.2   State-space model

For control, we utilize a state-space model of the form

$$x_{t+1} = Ax_t + Bu_t + Dd_t \qquad y_t = Cx_t + n_t, \tag{7.2}$$

with

$$A \; := \; \begin{bmatrix} p_1 + 2p_2 & -2p_1 p_2 - p_2^2 & p_1 p_2^2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3}$$

$$B \; := \; \frac{1800g}{u_{\text{TDI}}} \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^{\top} \in \mathbb{R}^3$$

$$D \; := \; -B/10 \in \mathbb{R}^3$$

$$C \; := \; \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{1 \times 3}.$$

The *state* is $x_t \in \mathbb{R}^3$. The inputs are the *control input* u_t that belongs to the set $\mathcal{U} \subset \mathbb{R}$, and the *unmeasured disturbance* d_t that is assumed to belong to a set $\mathcal{D} \subset \mathbb{R}$. The *measurement noise* n_t belongs to the set $\mathcal{N} \subset \mathbb{R}$. The nominal system ((7.2) without d_t and n_t) is an equivalent realization of (7.1). We have included noise and disturbance terms d_t and n_t, respectively, in order to explicitly account for model uncertainty, additive disturbances, and sensor noise. The matrix D is chosen to allow the disturbance to affect the state an order of magnitude smaller, and in the opposite way, than the insulin delivered as the control input does. Therefore, a positive disturbance d induces a rise in modeled BG output, akin to the consumption of carbohydrates.

7.2.3   Desired blood-glucose range

In the MPC/LO approach that we compare with, a range of desired BG values, i.e., the BG values generally considered safe and for which delivering the insulin basal rate is appropriate, is considered. This range is $[\underline{r}, \overline{r}]$ mg/dL, where $\underline{r} = 80$ and $\overline{r} = 140$, the same as in [42, 106]. For simplicity, the range is time-invariant, in contrast to [39, 40]. In order to implement this range in the controller, BG values are penalized according to the range excursion $Z : \mathbb{R} \to \mathbb{R}$ defined, as in [39], as

$$Z(y) := \arg \min_{\alpha \in \mathbb{R}} \{\alpha^2 \,|\, y + y_s - \alpha \in [\underline{r}, \overline{r}]\}. \tag{7.3}$$

For the MPC/MHE approach, we do not strictly penalize BG values outside of the desired range but rather approximate the range by penalizing BG values according to the following functions $h : \mathbb{R} \to \mathbb{R}$ and $c : \mathbb{R} \to \mathbb{R}$:

$$h(y) := (\arctan(0.1y) + \pi/2)y + 10, \qquad c(y) := \check{\lambda}h(-y + \underline{r}) + \hat{\lambda}h(y - \overline{r} + 20) \tag{7.4}$$

where $c(y)$ is parameterized by $\check{\lambda}$, the weight on low BG, and $\hat{\lambda}$, the weight on high BG. These weights can be chosen to separately tune the control response to hypoglycemia and hyperglycemia. In this work, we choose $\check{\lambda} = 0.02$ and $\hat{\lambda} = 0.005$. The remaining numbers in (7.4), such as 0.1, $\pi/2$, 10, and 20, are chosen to shift the nominal cost as desired. Plots of these functions h and c are shown in Figures 7.1 and 7.2, respectively. The output cost used in the MPC/MHE cost function (7.6) below is $c(y)^2$. Therefore, the output cost is asymmetric with respect to the desired BG range, roughly penalizing lower BG values 4^2 times more heavily than higher BG values.
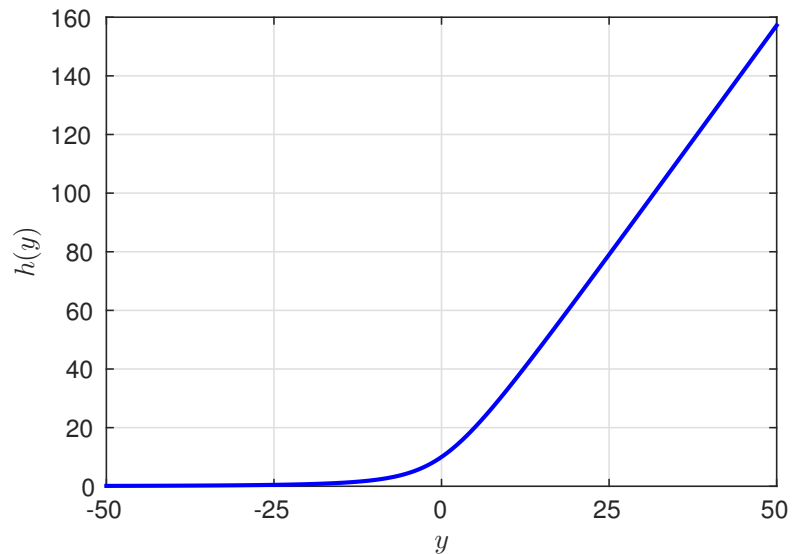

Figure 7.1: arctan function h used in defining asymmetric output penalty as given in (7.4).

7.2.4   Insulin delivery constraints

At each time t the control input is restricted to the set $\mathcal{U}$ defined as

$$\mathcal{U} := \{u_t \in \mathbb{R} | 0 \leqslant u_t + u_{\text{BASAL}} \times T_s \leqslant u_{\text{MAX}}\}, \tag{7.5}$$
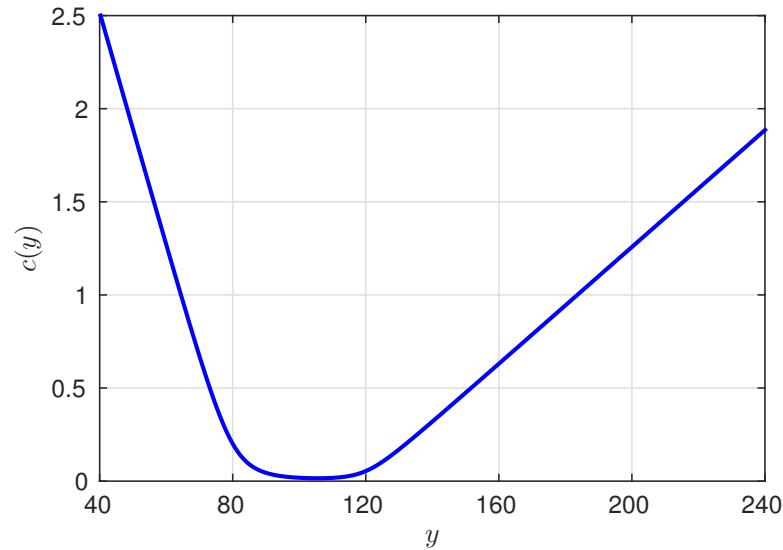
Figure 7.2: Asymmetric function c used in defining asymmetric output penalty as given in (7.4).

where $u_{\text{MAX}} = 25$ [U] is the maximum bolus size the CSII pump is allowed to command, and is selected in this work to be so large that it is practically impossible for the upper bound to become active.

7.3   Estimation and Control Approach

In this section, we discuss our novel MPC/MHE estimation and control approach and compare it to a control strategy that is based on state-feedback MPC with a Luenberger Observer (MPC/LO). A more complicated version of this MPC/LO approach has been used in clinical trials, but we have simplified it for use as a benchmark-controller in this study. First we define some notation: We denote by $\mathbb{Z}_{\geqslant 0}$ the set of non-negative integers, by $\mathbb{Z}_+$ the set of positive integers, and by $\mathbb{Z}_a^b$ the set of consecutive integers $\{a, ..., b\}$. Given a discrete-time signal $w : \mathbb{Z}_{\geqslant 0} \to \mathbb{R}^n$ and two times $t_0, t \in \mathbb{Z}_{\geqslant 0}$ with $t_0 < t$, we denote by $w_{t_0:t}$ the sequence $\{w_{t_0}, w_{t_0+1}, ..., w_t\}$. The prediction horizon is denoted by $T \in \mathbb{Z}_+$, the control horizon is denoted by $M \in \mathbb{Z}_1^T$, and the estimation horizon is denoted

by $L \in \mathbb{Z}_+$. In this work, we choose $T = 9$, $M = 5$, and $L = 3$ (where T and M are chosen the same as in [39], and L was chosen based on experimentation).

7.3.1   Combined MPC and MHE

Our estimation and control approach combines MPC and MHE as discussed in [22, 25], where the MPC and MHE optimization objectives are incorporated within a single min-max optimization problem, the solution of which simultaneously characterizes solutions to the MHE and MPC sub-problems. We formulate the MPC/MHE problem as a finite-horizon min-max optimization problem, to be solved at each time t, of the form

$$\min_{u_{t:t+M-1}} \quad \max_{\substack{x_{t-L}, \\ d_{t-L:t+T-1}}} \quad J_t(x_{t-L}, u_{t-L:t+M-1}, d_{t-L:t+T-1}, y_{t-L:t}) \tag{7.6}$$

with cost function

$$J_t(\cdot) := \sum_{k=t+1}^{t+T} c(Cx_k)^2 + \sum_{k=t}^{t+M-1} \lambda_u u_k^2 - \sum_{k=t-L}^{t+T-1} \lambda_d d_k^2 - \sum_{k=t-L}^{t} \lambda_n n_k^2, \tag{7.7}$$

and subject to

$$x_{k+1} = Ax_k + Bu_k + Dd_k \qquad\qquad \forall k \in \mathbb{Z}_{t-L}^{t+T-1} \tag{7.8a}$$

$$y_k = Cx_k + n_k \qquad\qquad \forall k \in \mathbb{Z}_{t-L}^{t} \tag{7.8b}$$

$$d_k \in \mathcal{D} \qquad\qquad \forall k \in \mathbb{Z}_{t-L}^{t+T-1} \tag{7.8c}$$

$$u_k \in \mathcal{U} \qquad\qquad \forall k \in \mathbb{Z}_{t}^{t+M-1} \tag{7.8d}$$

$$u_k = 0 \qquad\qquad \forall k \in \mathbb{Z}_{t+M}^{t+T-1} \tag{7.8e}$$

where λ_u, λ_d, and λ_n are positive scalar weights on the control input u, disturbance d, and measurement noise n, respectively. In this work, we choose these weights to be $\lambda_u = 2$,

$\lambda_d = 2$, and $\lambda_n = 300$. The running cost $c(y)$ is given in (7.4) and shown in Figure 7.2.

Equations (7.8a)-(8.11d) enforce the dynamics of the model (7.2). Equations (8.11e) and (7.8d) enforce that the input disturbance and control input belong to the constraint sets $\mathcal{D}$ and $\mathcal{U}$ (as defined in (7.5)), respectively. In this work, we define the constraint set $\mathcal{D} := \{d \in \mathbb{R} | 0 \leqslant d \leqslant 0.5\}$. A disturbance of $d = 0.5$ would counteract the effects of delivering 0.05 [U] insulin (which turns out to be slightly more than the standard deviation of the mean insulin delivered using the MPC/MHE approach shown later in Figure 7.5). Lastly, equation (7.8e) ensures that, beyond the control horizon M, the basal rate u_{BASAL} is delivered.

The criterion defined in (7.7) depends on the unknown initial state x_{t-L}, the unknown disturbance input sequence $d_{t-L:t+T-1}$, the measured output sequence $y_{t-L:t}$ via the cost on $n_{t-L:t}$, and the control input sequence $u_{t-L:t+M-1}$. The control input sequence is composed of two distinct sequences: the (known) past inputs $u_{t-L:t-1}$ that have already been applied, and the future inputs $u_{t:t+M-1}$ that need to be characterized. To select the future inputs $u_{t:t+M-1}$, we optimize (minimize) the criterion (7.7) with respect to these variables. At a given time $t \in \mathbb{Z}_{\geqslant 0}$, we do not know the values of x_{t-L} and $d_{t-L:t+T-1}$ (and $n_{t-L:t}$ which depends on these), so we optimize the criterion under worst-case assumptions on these variables (i.e., maximize (7.7)). The optimization (7.6) is repeated at each time step, i.e., every $T_s = 5$ minutes.

As is common in MPC, at each time t, we use as the control input the first element of the sequence

$$u^*_{t:t+M-1} = \{u^*_t, u^*_{t+1}, u^*_{t+2}, \ldots, u^*_{t+M-1}\}$$

164

that minimizes (7.6), leading to the following control law:

$$u_t = u_t^*, \quad \forall t \in \mathbb{Z}_{\geqslant 0}. \tag{7.9}$$

At each time t, after the solution to (7.6) is computed, the control command given to the pump is the value of u_t^* rounded down to the nearest integer multiple of the CSII pump-discretization of 0.05 [U] [39]. The portion that is removed when rounding down is then added to the control command given at time $t + 1$ in a so-called *carry-over* scheme, which is precisely described in [39].

More details and theoretical results regarding this approach to combine MPC with MHE can be found in [22, 25]. A particular primal-dual-like interior-point method was developed by the authors to numerically solve these optimization problems, and details about this method can be found in [25].

7.3.2   State-feedback MPC and Luenberger Observer

For completeness, we now give a brief overview of the simplified asymmetric MPC/LO approach that we use for comparison. The full MPC/LO approach used in clinical trials is described in detail in [39]. The purpose of this work is to investigate the benefits of the proposed MPC/MHE method with the simplest possible comparison to an existing AP control method; therefore, the version of this MPC/LO approach with which we compare does not include several important safety features that are included in [39] for implementation in clinical trials. These additional features are diurnal zones and constraints, feedforward control action following user-initiated meal-announcement, and insulin-on-board constraints. For more details on these additional features of the MPC/LO approach and its performance in trials, we refer the interested reader to [39, 26].

The main conceptual difference between this MPC/LO approach and the MPC/MHE

approach described above is that the MPC/MHE approach employs output-feedback, whereas the MPC/LO approach is based on state-feedback. State-feedback control is dependent upon the state estimator whose function is independent of, and indifferent to, the control design. In the MPC/MHE approach, the state estimate and control are computed simultaneously, so they directly affect each other. Another difference is that the asymmetric cost is on the control input rather than the predicted output. This is meant to facilitate decoupled design of the control response to hypoglycemia and hyperglycemia, whereas the asymmetry on the output in the MPC/MHE approach allows both the minimizer and the maximizer to affect the result asymmetrically because the predicted output depends on both u and d. Finally, neither disturbances nor noise are explicitly considered in this MPC/LO approach.

The state estimate $\hat{x}_t$ that is used for control in the MPC/LO approach is found using a linear recursive state estimator known as a Luenberger Observer (see e.g. [60]). This state estimator is given by

$$\hat{x}_{t+1} = A\hat{x}_t + K(y_t - \hat{y}_t) + Bu_t, \qquad \hat{y}_t = C\hat{x}_t, \tag{7.10}$$

where the gain K is designed as in [39].

We formulate the asymmetric MPC/LO problem as a finite-horizon optimization problem, to be solved at each time t, of the form

$$\min_{u_{t:t+M-1}} J_t(x_t, u_{t:t+M-1}) \tag{7.11}$$

with cost function

$$J_t(\cdot) := \sum_{k=t+1}^{t+T} z_k^2 + \sum_{k=t}^{t+M-1} \left(\hat{R}\hat{u}_k^2 + \check{R}\check{u}_k^2 \right), \tag{7.12}$$

and subject to

$$x_t = \hat{x}_t \tag{7.13a}$$

$$x_{k+1} = Ax_k + Bu_k \qquad \forall k \in \mathbb{Z}_t^{t+T-1} \tag{7.13b}$$

$$y_k = Cx_k \qquad \forall k \in \mathbb{Z}_t^{t+T} \tag{7.13c}$$

$$u_k \in \mathcal{U} \qquad \forall k \in \mathbb{Z}_t^{t+M-1} \tag{7.13d}$$

$$u_k = 0 \qquad \forall k \in \mathbb{Z}_{t+M}^{t+T-1} \tag{7.13e}$$

$$\hat{u}_k = \max(u_k, 0) \qquad \forall k \in \mathbb{Z}_t^{t+M-1} \tag{7.13f}$$

$$\check{u}_k = \min(u_k, 0) \qquad \forall k \in \mathbb{Z}_t^{t+M-1} \tag{7.13g}$$

$$z_k = Z(y_k) \qquad \forall k \in \mathbb{Z}_{t+1}^{T}, \tag{7.13h}$$

where $\hat{R}$ and $\check{R}$ are positive scalar weights on the non-negative and non-negative control inputs, respectively. As in [39], we choose the weights to be $\hat{R} = 7000$ and $\check{R} = 100$ in order to conservatively respond to hyperglycemia while encouraging pump attenuation in response to predicted hypoglycemia. As before, only the first element u_t^* of the predicted optimal insulin trajectory is commanded to the pump, and the optimization (7.11) is repeated at each time step, i.e., every $T_s = 5$ minutes.

Equation (7.13a) enforces that the initial state is defined as the state estimate from the Luenberger Observer (7.10). Equations (7.13b)-(7.13c) enforce the dynamics of the model (7.2) without considering disturbances or noise. Equation (7.13d) enforces that the control input belongs to the constraint set $\mathcal{U}$ (as defined in (7.5)). Equation (7.13e) ensures that, beyond the control horizon M, the basal rate u_{BASAL} is delivered. Equations (7.13f)-(7.13g) facilitate an asymmetric input cost and provide positive and negative deviations of the input u_k from u_{BASAL}. Finally, (7.13h) provides the cost for output excursions from the desired BG range, where $Z(y_k)$ is defined in (7.3).

Figures 7.3 and 7.4 show the input and output costs used in both the MPC/MHE approach and the MPC/LO approach, respectively.

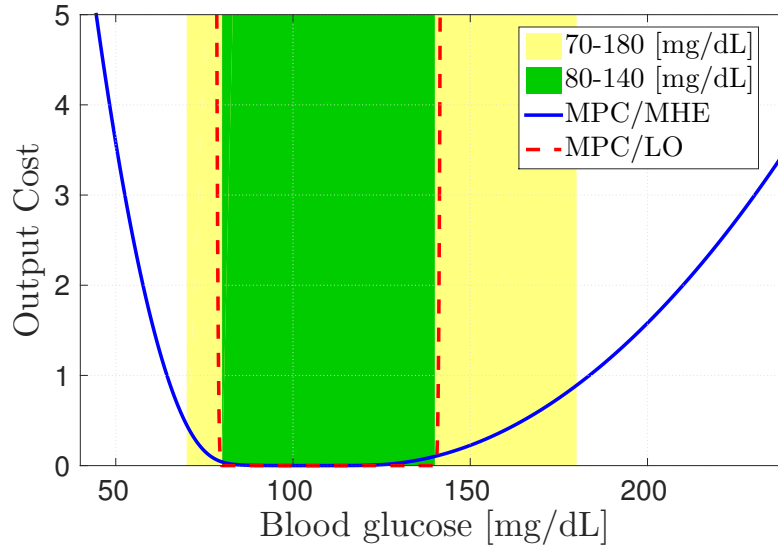Figure 7.3: Output costs for MPC/MHE and MPC/LO as given in the first terms of (7.7) and (7.12), respectively. The blood-glucose target and admissible zones are shown in green and yellow, respectively, for reference.
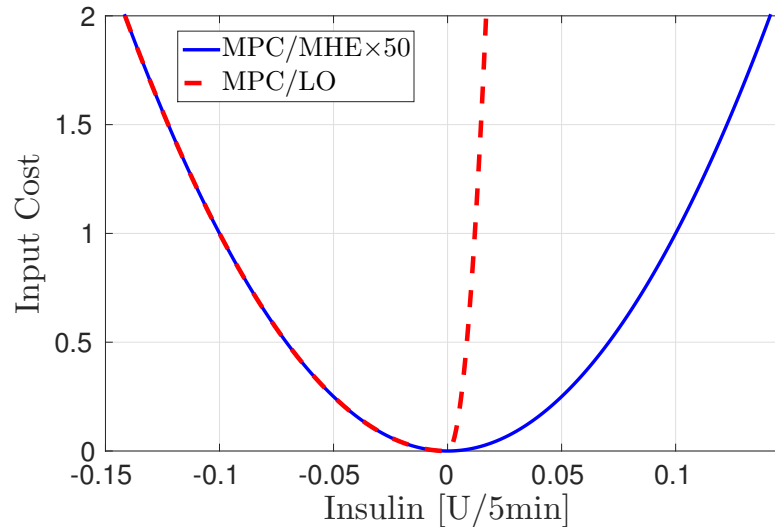

Figure 7.4: Input costs for MPC/MHE and MPC/LO as given in the second terms of (7.7) and (7.12), respectively. The MPC/MHE input cost is multiplied by 50 for comparison because $\check{R}/\lambda_u = 50$.

7.4 Simulation Study

The efficacy of this combined MPC/MHE estimation and control approach is demonstrated via *in-silico* trials of 10 subjects using the commercially available UVA/Padova US Food and Drug Administration (FDA) accepted metabolic simulator [54]. Simulations start at 14:00 and are 28 hours in duration. The simulations begin with two hours of open-loop until 16:00 when the feedback controller is turned on, and simulations run in closed-loop until 18:00 the next day. Every simulation includes three 90 gram carbohydrate (gCHO) meals consumed at 18:30, 07:00, and 13:00, respectively. For most people, 90 gCHO constitutes a large meal, and it happens to be the largest meal allowed in the clinical protocol of [26]. The parameters of both the simulator and controller are time-invariant, so the time of day of meal consumption is irrelevant. We only consider the most challenging case of unannounced meals, meaning the controller has no information about when a meal will be consumed, or how large any consumed meal is, and therefore the meal-disturbance is rejected based on CGM feedback only. To further stress the controller, the UVA/Padova simulator includes subjects with parameter values that may be considered slightly physiologically implausible.

We consider a total of 110 simulations. All 10 subjects in the commercially available UVA/Padova metabolic simulator are simulated 11 times: 10 times with different seeds for random CGM additive sensor noise, and 1 time with no additive sensor noise. Even without additive sensor noise, the CGM measurements are subject to dynamics and delays that cause it to differ from the true (simulated) BG values. For more details, see [54]. We make no assumptions on the CGM measurement noise, which is, in general, neither zero-mean nor Gaussian. Therefore, we do not introduce constraints on the measurement noise but simply penalize it according to the last term in (7.7).

7.4.1   Comparison of MPC/MHE and MPC/LO

The aggregate results for all 110 simulations are shown in Figure 7.5. A zoomed version of these results for the first meal is shown in Figure 7.6. To yield a meaningful comparison, the MPC/MHE approach is tuned specifically to achieve similar hyperglycemic peaks after meal consumption as those achieved by the MPC/LO approach. This facilitates a comparison between the proposed MPC/MHE and benchmark MPC/LO approach, with respect to the other glucose control metrics. In particular, the aggregate results in Figure 7.5 show that the MPC/MHE approach is able to prevent the extremely low BG values that are within the min/max envelope of the MPC/LO approach. In addition, the MPC/MHE approach does not cause a large rebound in BG just before 03:00 as the MPC/LO approach does. However, we see that, on average, BG values are not regulated back to within the desired BG range as soon after a meal is consumed using the MPC/MHE approach. Finally, the hyperglycemic peak after the third meal is noticeably lower for the MPC/MHE approach; this is likely due to the fact that the MPC/MHE approach enforces fewer and shorter pump suspensions, resulting in more insulin-on-board, which helps to attenuate the BG response due to the meal. We discuss this further later.

The insulin profiles, shown in the bottom plots of Figures 7.5 and 7.6, are significantly different. The amount of insulin delivered using the MPC/MHE approach peaks at a lower value after a meal is consumed, but it continues delivering insulin above the basal rate for longer than the MPC/LO approach. Another significant difference is that, after a meal is consumed, the MPC/MHE approach responds more quickly by delivering insulin above the basal rate before the MPC/LO approach does. This is most easily seen in Figure 7.6. This benefit is likely due to the difference in estimation schemes; the Luenberger Observer is recursive, which may cause its state estimate to lag slightly when
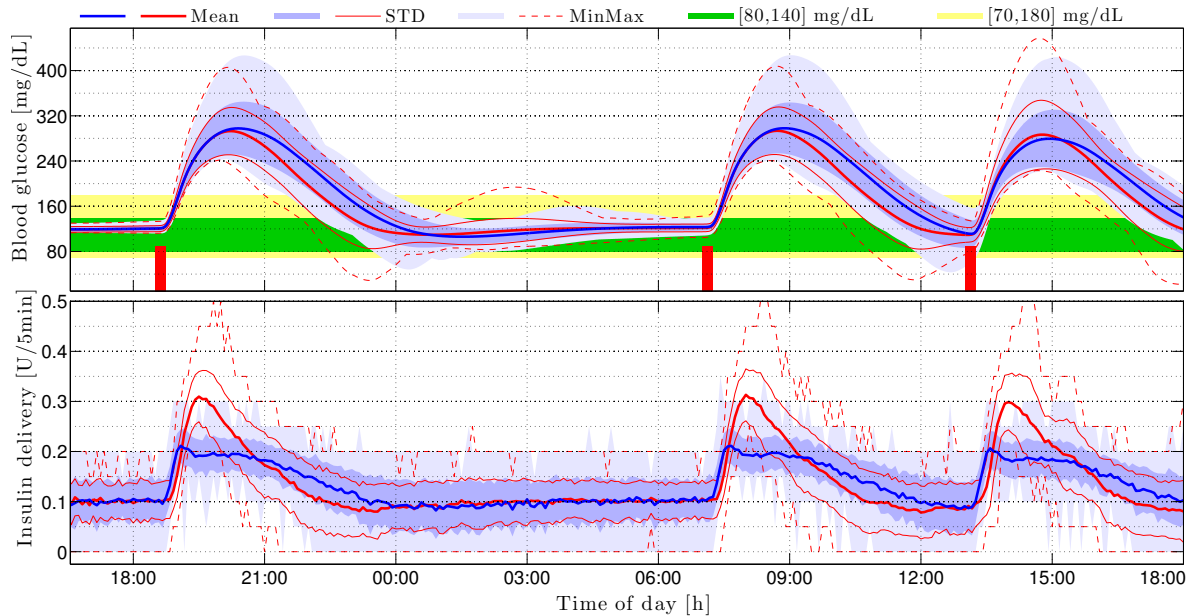
Figure 7.5: Aggregate results of all 110 simulations for the MPC/MHE approach (blue) and the MPC/LO approach (red). The top plots show the mean blood-glucose trajectory as well as its standard deviation and minimum/maximum envelope. The bottom plots show the mean insulin delivered as well as its standard deviation and minimum/maximum envelope. The red vertical bars at 18:30, 07:00, and 13:00 indicate times a 90 gCHO meal is consumed.

a rapid change in the state occurs. The advantage of this faster insulin delivery achieved with MPC/MHE is quantified in Figure 7.12 below.

These features are also seen, perhaps more clearly, in Figure 7.7, which shows results for an individual subject for both the MPC/MHE approach and the MPC/LO approach. This individual plot was chosen to highlight the differences between the approaches for a particularly challenging subject. For both the MPC/MHE and MPC/LO approaches, Figure 7.7 shows the actual BG trajectories in the top plot and the insulin delivered in the bottom plot. The MPC/MHE approach tends to keep BG levels slightly higher but effectively mitigates the extreme hypoglycemia experienced using the MPC/LO approach. The MPC/LO approach clearly suspends the pump (i.e., delivers 0 [U] of insulin) much longer than the MPC/MHE approach in the face of impending hypoglycemia. This
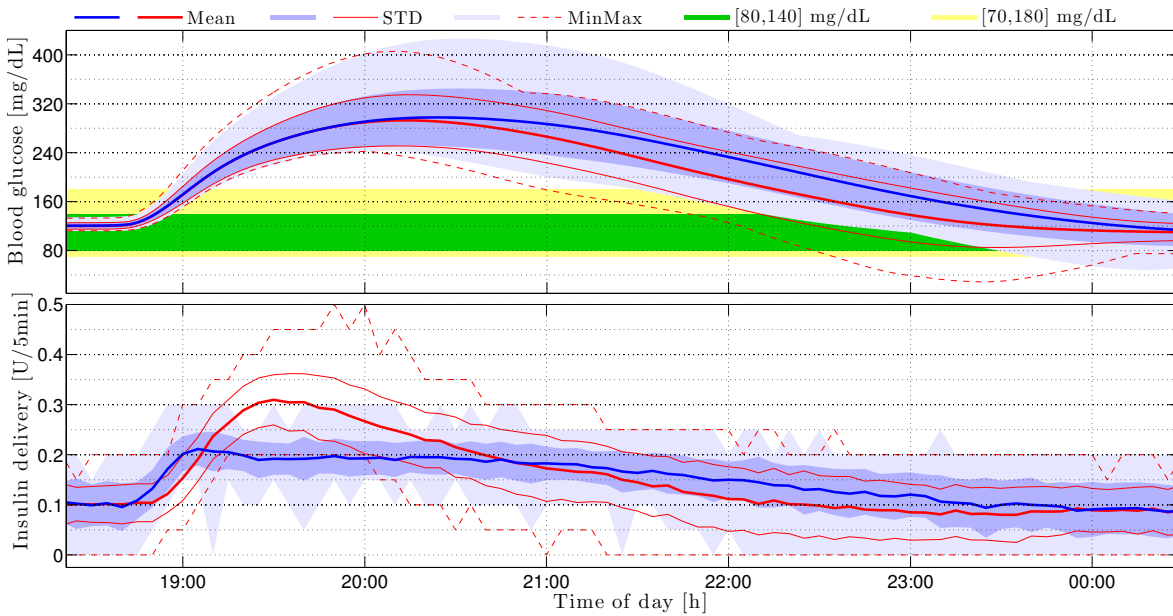
Figure 7.6: Only the first meal response of all 110 simulations for the MPC/MHE approach (blue) and the MPC/LO approach (red). The top plots show the mean blood-glucose trajectory as well as its standard deviation and minimum/maximum envelope. The bottom plots show the mean insulin delivered as well as its standard deviation and minimum/maximum envelope.

results in a larger rebound in the BG for the MPC/LO approach after 00:00 as well as a higher peak after the third meal. Finally, the bottom plot shows that the MPC/MHE approach commands insulin to be delivered above the basal rate earlier than the MPC/LO approach after meal consumption.

Another way to present aggregate performance of a controller for the AP problem is to consider Control-Variability Grid Analysis (CVGA) plots as described in [69]. Figures 7.8 and 7.9 show the CVGA plots for our MPC/MHE approach as well as the MPC/LO approach, respectively, for all 110 simulations. The black dots represent each simulation's minimum (horizontal axis) and maximum (vertical axis) BG values with the highest and lowest 2.5% removed. The large blue dot denotes the arithmetic mean of the individual black dots, and the blue circle has a radius of the standard deviation of the distances of each individual dot to the mean.

172
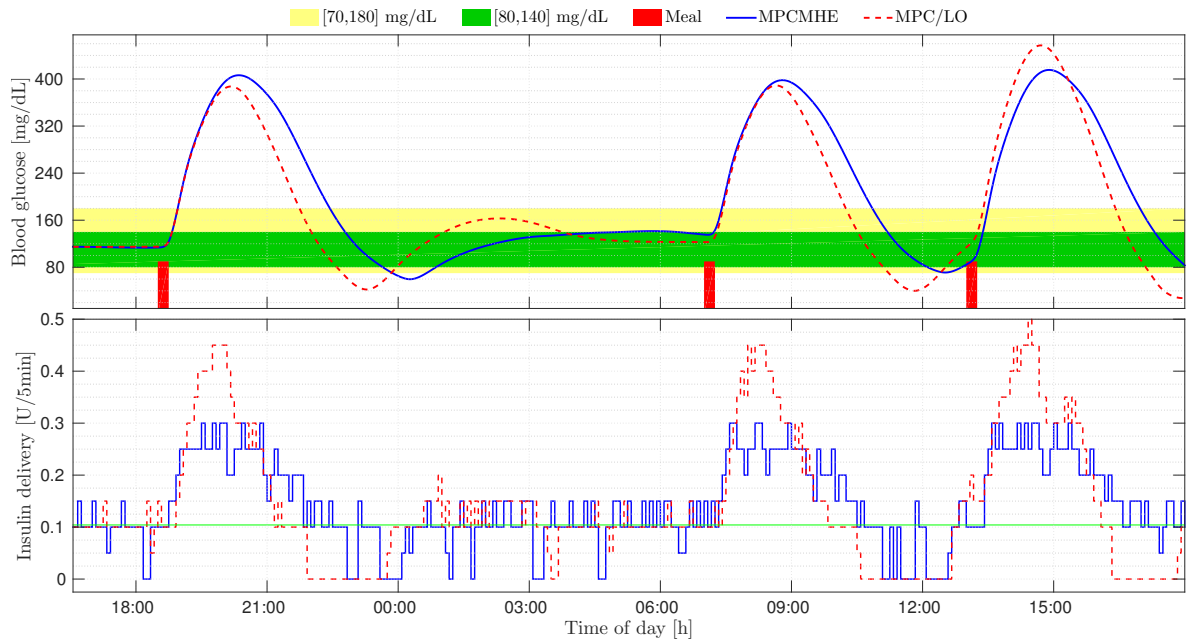
Figure 7.7: Results for Subject #7 with random additive noise seed 2. The top plot shows the BG for both the MPC/MHE approach (blue solid) and the MPC/LO approach (red dashed). Similarly, the bottom plot shows the insulin delivered for each approach as well as the basal insulin rate (green solid).

The CVGA plots in Figures 7.8 and 7.9 show that our MPC/MHE approach has a mean value that is shifted to the left and just slightly higher than the mean value of the MPC/LO approach. In addition, the standard deviation for the MPC/MHE approach is significantly smaller than the standard deviation for the MPC/LO approach. This means that our MPC/MHE approach is effective at keeping subjects from experiencing dangerously low BG levels with the caveat that they may experience slightly higher glucose levels on average. This comparison is easy to see in Figure 7.7 for a single subject. The resulting number of subjects in each of the CVGA zones, as well as the mean and standard deviation values, are given in Table 7.1 below for both approaches. These statistics are useful because they tell us things that the plots cannot; for instance, the MPC/LO approach has 11 black dots in the upper right-hand corner superimposed.

173
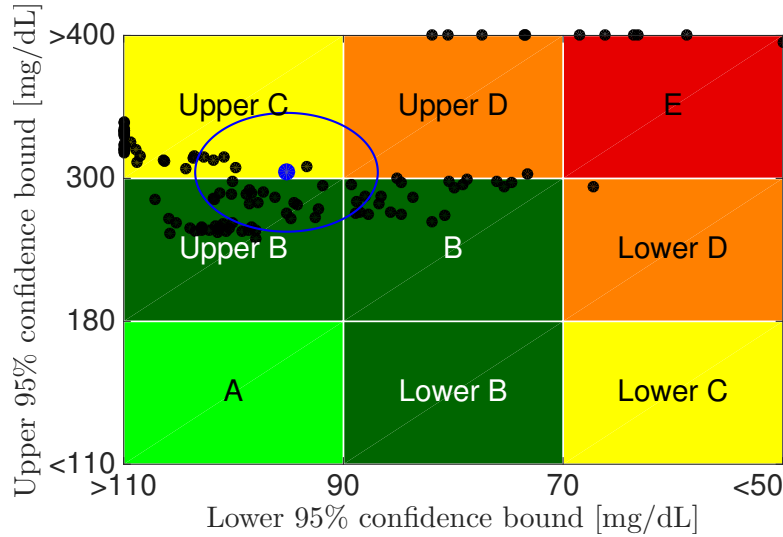
Figure 7.8: CVGA plot for MPC/MHE approach. Number of 110 simulations in each zone (A,B,C,D,E)=(0,64,33,7,6). The circle is centered on the mean with radius equal to the standard deviation. These values are given in Table 7.1.

7.4.2 Quantifying these results

In this section we quantify the advantages of using the MPC/MHE approach over the MPC/LO approach. Specifically, the MPC/MHE approach results in: 1) fewer hypoglycemic events with the same number of hyperglycemic events, 2) fewer and shorter insulin pump suspensions, resulting in smoother BG trajectories, and 3) accelerated insulin delivery in response to meal consumption. The main drawback of using the MPC/MHE approach is that the BG values are slightly higher on average.

Table 7.1 contains the aggregate results for both approaches. The first set of rows shows the time-in-range percentages of two BG intervals and several thresholds. As we noted before, the MPC/LO approach keeps the BG within the desired range a larger percentage of the time and also results in less time above hyperglycemic values. The MPC/MHE approach, on the other hand, results in significantly less time below hypoglycemic thresholds. Both approaches may be tuned to adjust these values, but in this work, we tuned the MPC/MHE approach specifically to achieve comparable hy-

174
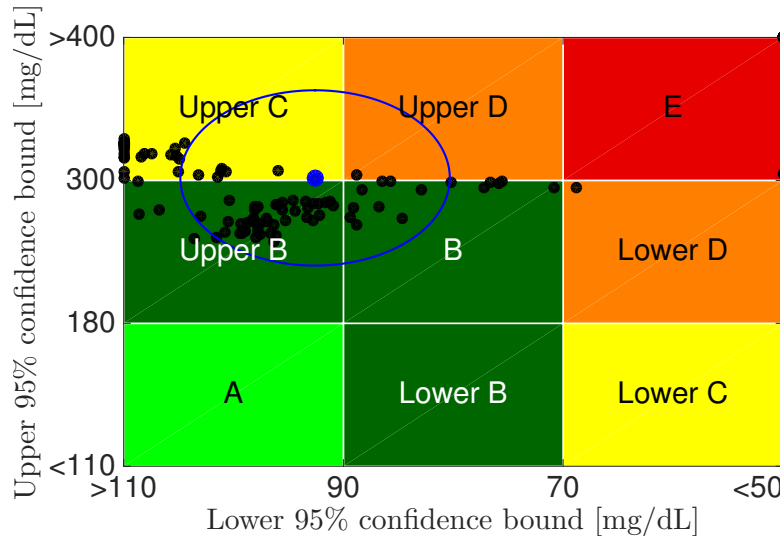
Figure 7.9: CVGA plot for MPC/LO approach. Number of 110 simulations in each zone (A,B,C,D,E)=(0,64,32,2,12). Several points are superimposed, e.g., in zone E. The circle is centered on the mean with radius equal to the standard deviation. These values are given in Table 7.1.

perglycemic peaks as the MPC/LO approach in order to contrast other features of the controllers.

The second and third set of rows in Table 7.1 list the number of subjects and events, respectively, for which BG values are above or below several thresholds. While the MPC/MHE approach is tuned to achieve the same number of hyperglycemic events as the MPC/LO approach, using the MPC/MHE approach results in half as many subjects experiencing hypoglycemia of less than 60 [mg/dL] and *far* fewer hypoglycemic events. Histograms for the number of hyperglycemic and hypoglycemic events are given in Figure 7.10.

The fourth set of rows in Table 7.1 shows the number of pump suspensions that last longer than particular lengths of time. A histogram of these results is shown in Figure 7.11. The MPC/MHE approach results in fewer suspensions of lengths greater than 15 and 30 minutes and results in no pump suspensions longer than 60 minutes. This is an advantage because, while suspending the pump is important in order to attenuate

predicted hypoglycemic BG values and safeguard from dangerous outcomes, long pump suspensions can deteriorate overall performance as less insulin is present in the body, causing BG values to rebound or peak higher after a meal is consumed. Both of these features can be seen in the aggregate results in Figure 7.5 and the individual results in Figure 7.7 where a long pump suspension, as commanded by the MPC/LO approach, causes BG values to rebound after the first meal and peak higher after the third meal.
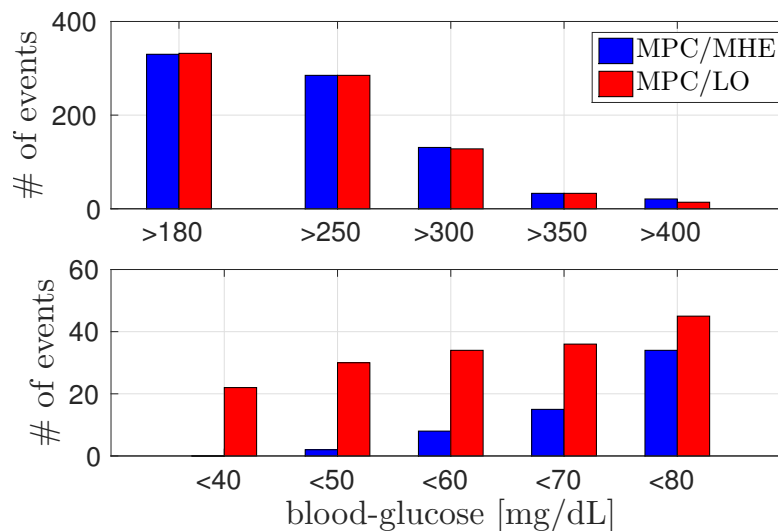
Figure 7.10: Histogram for the number of hyperglycemic and hypoglycemic events for the MPC/MHE approach (blue) and the MPC/LO approach (red).

The fifth set of rows in Table 7.1 gives the number of UCSB Health Monitoring System (HMS) alarms [45] and number of subjects who experience HMS alarms. The MPC/MHE approach causes significantly fewer alarms than the MPC/LO approach. Minimizing the number of alarms is important when implementing these controllers because if alarms go off too frequently, subjects may experience "alarm fatigue" and ignore the alarms. The sixth set of rows in Table 7.1 gives the numerical results corresponding to the CVGA plots in Figures 7.8 and 7.9. The seventh set of rows gives the mean, minimum, and maximum BG values for each approach. The MPC/MHE approach results in a slightly higher mean BG but also a higher minimum BG. Interestingly, the MPC/MHE approach
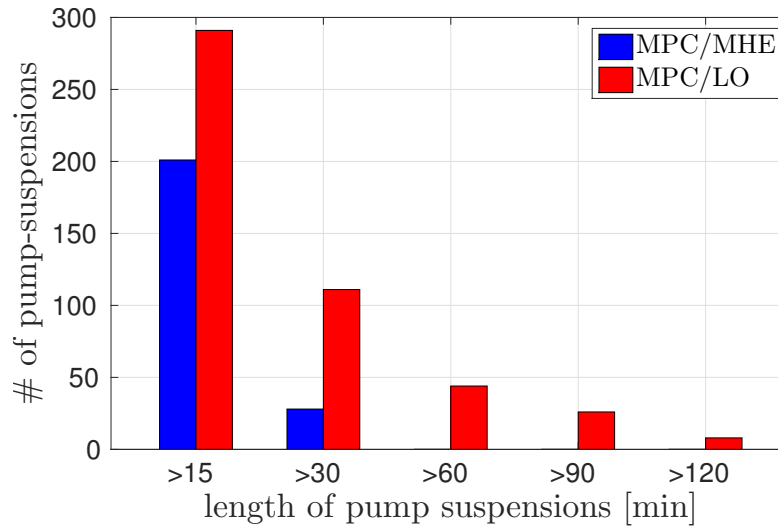
Figure 7.11: Histogram for the number of pump suspensions for the MPC/MHE approach (blue) and the MPC/LO approach (red).

produces a very slightly lower maximum BG. The eighth set of rows gives the Low Blood Glucose Index (LBGI) and High Blood Glucose Index (HGBI), as computed according to [68]. As expected, the MPC/MHE approach results in a lower LBGI but a higher HBGI. Finally, the last row gives the total amount of insulin delivered using each approach averaged over all 110 simulations as well as the standard deviation and minimums and maximums. The MPC/MHE approach delivers less insulin than the MPC/LO approach; therefore, it is not surprising that the MPC/MHE results in slightly higher mean BG.

The last advantage, and one that is not quantified in Table 7.1, is that the MPC/MHE approach delivers insulin more quickly after a meal. This was mentioned when discussing the aggregate responses after the first meal shown in Figure 7.6 and is precisely quantified in Figure 7.12, which shows the cumulative insulin delivered as well as the mean BG after a meal averaged over all three meals and all 110 simulations for both approaches. For reference, the basal insulin rate is shown and increases at a rate of 0.1026 [U/5min]. On average, BG begins to rise about 10 minutes after a meal is consumed. After 20 minutes, the MPC/MHE approach begins to deliver insulin at a rate higher than the

basal rate. The MPC/LO approach, on the other hand, does not deliver insulin at a rate higher than the basal rate until 30 minutes after a meal is consumed. Consequently, the MPC/MHE approach delivers more insulin compared to the MPC/LO approach for the first 55 minutes after a meal is consumed, after which time the MPC/LO approach continues delivering insulin at a higher rate. Moreover, the mean BG is lower for the MPC/MHE approach until about 105 minutes after meal consumption even though it starts out slightly higher than the BG for the MPC/LO approach before a meal. This faster response to increasing BG values after a meal may be due to using an MHE-like state estimator as opposed to a Luenberger Observer because the Luenberger Observer computes the estimates recursively and, therefore, may produce a lagging state estimate if the state is rapidly changing. This phenomenon is also found in [59] when specifically comparing the results of a state-feedback MPC strategy using both a Luenberger Observer and an MHE for estimating BG. The MPC/MHE approach does not, however, cumulatively deliver as much insulin as the MPC/LO approach, so BG values stay slightly higher for longer using the MPC/MHE approach.

7.5   Conclusions

We presented a new estimation and control approach for regulating BG in T1DM. This approach simultaneously performs Model Predictive Control and Moving Horizon Estimation in a single min-max optimization problem to form a feedback controller that results in elevated cost-conservatism with respect to disturbances. This combined MPC/MHE approach incorporated an asymmetric output cost penalizing "riskier" low BG values more severely than high BG values. We compared this approach to a state-feedback MPC approach that utilized a Luenberger Observer for state estimation and incorporated an asymmetric input cost in order to decouple the controller's response to low
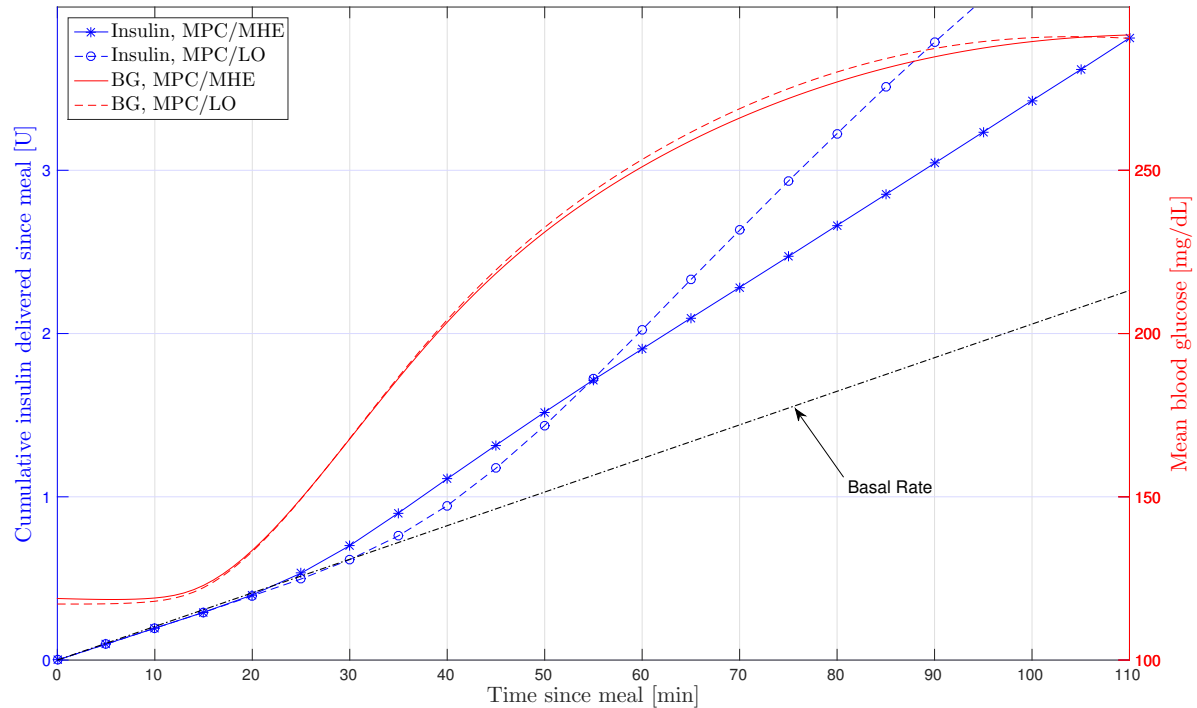
Figure 7.12: Cumulative insulin delivered (blue) and mean BG (red) since a meal consumption for both MPC/MHE and MPC/LO approaches averaged over all three meals and 110 simulations.

versus high BG values. This MPC/LO approach is a simplified version of an estimation and control approach that has been successfully tested in clinical and outpatient trials.

Both of these control and estimation approaches were evaluated by *in-silico* studies utilizing a metabolic simulator. In 110 simulations of 10 virtual subjects, we found that while the MPC/MHE approach keeps BG values slightly higher on average, it successfully reduces the number of hypoglycemic events without increasing the number of hyperglycemic events, it delivers insulin sooner in response to meal consumption, and it commands fewer and shorter insulin pump suspensions, which results in smoother BG trajectories. Therefore, this MPC/MHE approach may be advantageous for the feedback control of an AP for the treatment of T1DM.

7.6   Acknowledgments

We would like to thank Dr. Ravi Gondhalekar for collaborating on this work. We acknowledge that access to the UVA/Padova metabolic simulator was provided by an agreement with Prof. C. Cobelli (University of Padova) and Prof. B. P. Kovatchev (UVA) for research purposes, and all simulations using the metabolic simulator were performed by our collaborator, Dr. Ravi Gondhalekar, at UCSB.

Table 7.1: Aggregate results for the two comparisons considered.

		MPC/MHE	MPC/LO
	$\in [80, 140]$	45.47	49.19
	$\in [70, 180]$	56.62	61.24
	< 80	1.05	1.85
	< 70	0.39	1.30
	< 60	0.13	0.99
BG [mg/dL] % time	< 50	0.03	0.68
	< 40	0.00	0.34
	> 180	42.99	37.46
	> 250	20.84	17.53
	> 300	7.70	5.98
	> 350	1.93	1.43
	> 400	0.58	0.45
	< 80	21	20
	< 70	10	14
	< 60	6	12
	< 50	1	12
# Subjects BG	< 40	0	12
[mg/dL]	> 180	110	110
	> 250	109	108
	> 300	53	53
	> 350	11	11
	> 400	11	11
	< 80	34	45
	< 70	15	36
	< 60	8	34
	< 50	2	30
	< 40	0	22
# Events BG [mg/dL]	> 180	330	332
	> 250	285	285
	> 300	131	128
	> 350	33	33
	> 400	21	14
	$\geqslant 15$ min	201	291
	$\geqslant 30$ min	28	111
# Pump Suspensions	$\geqslant 60$ min	0	44
	$\geqslant 90$ min	0	26
	$\geqslant 120$ min	0	8
	$\geqslant 150$ min	0	0
# HMS Alarms		37	95
# Subjects with HMS Alarms		19	23
CVGA zone count: A		0	0
CVGA zone count: B		64	64
CVGA zone count: C		33	32
CVGA zone count: D		7	2
CVGA zone count: E		6	12
CVGA circle radius		8.31	12.26
BG mean: mean+-std [min,max]		180+-18.92 [152,219]	171+-14.90 [148,201]
BG min: mean+-std [min,max]		94+-16.72 [43,122]	89+-23.65 [22,121]
BG max: mean+-std [min,max]		307+-47.18 [247,427]	307+-53.62 [246,457]
LBGI: mean+-std [min,max]		0.17+-0.29 [0.00,1.97]	0.47+-1.08 [0.00,4.51]
HBGI: mean+-std [min,max]		10.87+-3.74 [5.47,19.47]	9.36+-2.95 [4.84,14.28]
Total Daily Insulin [U]: mean+-std [min,max]		41.4+-6.6 [32.7,57.3]	42.5+-7.1 [33.1,59.5]

Chapter 8

Additional examples

In this chapter we investigate two additional applications of the proposed MPC/MHE estimation and control approach: 1) output feedback control of a DC motor, and 2) distributed optimization for multi-agent consensus.

8.1 Output feedback control of a DC motor

In this section we discuss the use of the combined MPC/MHE method described in Chapter 2 for the estimation and control of a DC motor. In particular, we investigate three different formulations of the MPC/MHE approach for this problem and discuss the results.

8.1.1 Modeling a DC Motor

Differential equations describing the dynamics of a DC motor are given as

$$L_M \dot{i}_M = -R_M i_M - K_b \dot{\theta} + v_M, \qquad\qquad J_M \ddot{\theta} = -b_M \dot{\theta} + K_M i_M,$$

where i_M is the current that flows through the motor, v_M is the voltage applied to the motor, θ is the motor shaft angle, L_M, is the electric inductance, R_M is the electric resistance, K_M is the motor torque constant, K_b is the electromotive force constant, J_M is the moment of inertia of the motor's rotor, and b_M is the motor viscous friction coefficient.

State-space model

A linear time-invariant model for the control of a DC motor can be formulated and described by the following state-space model:

$$\dot{x} = Ax + Bu + Dd$$
$$y = Cx + n$$
(8.1)

where $x = [i_M \ \dot{\theta} \ \theta]' \in \mathbb{R}^3$ is the system state, $u = v_M \in \mathbb{R}^1$ is the control input, $d \in \mathbb{R}^1$ is an unmeasured disturbance, $y \in \mathbb{R}^1$ is the measured output, and $n \in \mathbb{R}^1$ is unknown measurement noise. The system matrices are given by

$$A = \begin{bmatrix} -R_M/L_M & -K_b/L_M & 0 \\ K_M/J_M & -b_M/J_M & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1/L_M \\ 0 \\ 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 1/J_M \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}.$$

Therefore, only noisy measurements of the motor shaft angle θ are available for feedback.

There are several features of a real motor that are not captured by this model. One of these features is the fact that a real motor would experience a dead zone in which the voltage applied does not cause the motor to move due to friction. We denote this dead zone by the interval $[-V_d, V_d]$. While the control input u is an applied voltage that directly affects the current flowing through the motor, the disturbance d is an external

torque that directly affects the angular velocity of the rotor $\dot{\theta}$. This is intended to capture some of the unknown plant/model mismatch, such as the unknown dead zone of the motor. The results we present later in Section 8.1.3 show the effects of including this disturbance d.

8.1.2 Output Feedback MPC with MHE

For output feedback control of a DC motor, we consider the combined MPC/MHE approach proposed in [22, 25] and presented in Chapter 2. This involves the solution of a general finite-horizon min-max optimization problem

$$\min_{u_{t:t+T-1}\in\mathcal{U}} \max_{\substack{x_{t-L}\in\mathcal{X},\\ d_{t-L:t+T-1}\in\mathcal{D}}} J_t(x_{t-L}, u_{t-L:t-1}, u_{t:t+T-1}, d_{t-L:t+T-1}, y_{t-L:t}). \tag{8.2}$$

This optimization depends on the last $L+1$ output measurements $y_{t-L:t}$, the past L inputs $u_{t-L:t-1}$, and solves for the unknown initial state $x+t-L$, past and future unknown disturbances $d_{t-L:t+T-1}$, and future control inputs $u_{t:t+T-1}$. Optimization (8.2) is subject to the system dynamics (8.1) as well as constraints on the motor voltage and the disturbance. These constraints are given by the sets $\mathcal{U} := \{u_t \in \mathbb{R}^1 | -V_{\text{supply}} \leqslant u_t \leqslant V_{\text{supply}}\}$ and $\mathcal{D} := \{d_t \in \mathbb{R}^1 | -d_{max} \leqslant d_t \leqslant d_{max}\}$, respectively. Solution of the optimization (8.2) results in a sequence of future control inputs to apply, a sequence of worst-case disturbance variables, as well as worst-case estimates of the past states and a future sequence of predicted states. In order to numerically solve the optimization (8.2), we discretize the dynamics (8.1) using a zero-order hold (ZOH) with time step T_s.

We formulate the optimization (8.2) in three different cases. Case #1 does not con-

sider disturbances d and involves the solution of the following min-max optimization

$$\min_{u_{t:t+T-1} \in \mathcal{U}} \max_{x_{t-L} \in \mathcal{X}} J_t(\cdot),$$

where

$$J_t(\cdot) := \sum_{s=t+1}^{t+T} \lambda_r \|\theta_s - r_s\|^2 + \sum_{s=t}^{t+T-1} \lambda_u \|u_s\|^2 - \sum_{s=t-L}^{t} \lambda_n \|n_s\|^2, \qquad (8.3)$$

and subject to the process dynamics (8.1) without the disturbance d. The first term in the cost function (8.3) penalizes the difference between the future predicted value of the angle θ_s and the desired reference trajectory r_s in order to facilitate reference tracking. The second term penalizes the control actions u_s, i.e. the applied voltages. The third term facilitates state estimation by penalizing unlikely noise n_s. The scalar weights λ_r, λ_u, and λ_n are used to scale the penalties on the reference tracking, control action, and measurement noise, respectively.

For Case #2, we consider the same problem as in Case #1 but with added disturbances. Therefore, we solve optimization (8.2), where

$$J_t(\cdot) := \sum_{s=t+1}^{t+T} \lambda_r \|\theta_s - r_s\|^2 + \sum_{s=t}^{t+T-1} \lambda_u \|u_s\|^2 - \sum_{s=t-L}^{t+T-1} \lambda_d \|d_s\|^2 - \sum_{s=t-L}^{t} \lambda_n \|n_s\|^2, \qquad (8.4)$$

and subject to the process dynamics (8.1). The only difference between (8.4) and (8.3) is that (8.4) for Case #2 contains a term penalizing unlikely disturbances d_s. The scalar weight λ_d is used to scale this penalty.

In Case #3, we again solve optimization (8.2) but with a cost function that specifically penalizes power consumption rather than simply the voltage applied. In particular, we

consider the cost

$$J_t(\cdot) := \sum_{s=t+1}^{t+T} \lambda_r \|\theta_s - r_s\|^2 + \sum_{s=t+1}^{t+T} \lambda_u(u_s \cdot i_{M,s}) - \sum_{s=t-L}^{t+T-1} \lambda_d \|d_s\|^2 - \sum_{s=t-L}^{t} \lambda_n \|n_s\|^2. \quad (8.5)$$

Again, the optimization is subject to the process dynamics (8.1). The second term in (8.5) penalizes the power consumed by penalizing the multiplication of the voltage applied u_s with the predicted value of the current $i_{M,s}$. The scalar weight λ_u is used to scale this penalty. Depending on how this weight is chosen, the controller may implement regenerative braking, where the applied voltage is switched quickly back and forth between a positive and negative value in order to constantly switch the direction that current flows so that it is flowing back into, and recharging, the battery powering the motor. We see this behavior in the results presented in Figures 8.3 and 8.4 below.

8.1.3   Simulations

In this section we consider the three difference cases of the optimization (8.2) described in Section 8.1.2. For all three cases, the control objective is for the angle θ to follow a reference given by $r_t = 2\pi \, \mathrm{sgn}(\sin(\pi t))$. Table 8.1 gives the values of the parameters in the model (8.1). For our prediction model, we discretize (8.1) using a zero-order hold (ZOH) with time step T_s. The time step T_s is also our sampling time for this problem.

The plant we simulate is also given by the model (8.1) but also includes a dead zone of $[-0.72, 0.72]$ V such that if the applied voltage is within this zone, it is as if 0 V is applied. In other words, the motor voltage is given by

$$V_{M,t} = \begin{cases} u_t & \text{if} \quad |u_t| > 0.72\mathrm{V} \\ 0 \text{ V} & \text{else.} \end{cases}$$

186

For simplicity and the sake of making a clear comparison between the three cases considered, the plant we simulate is not subject to any measurement noise or disturbances.

Table 8.1: Model Parameters

Parameter	Description	Value	Units
T_s	Sampling time	0.005	s
V_{supply}	Battery supply voltage	6	V
d_{max}	Constraint on size of disturbance	$6.9e - 9$	Nm
R_M	Resistance	2.62	Ω
L_M	Inductance	0.0259	H
K_b	Electromotive force constant	0.01	V/rad/s
k_M	Motor torque constant	0.01	Nm/A
J_M	Moment of inertia	$2.3e - 6$	kg m^2
b_M	Viscous friction coefficient	$1.35e - 5$	N m s
V_d	Motor dead zone	0.72	V

Remark 11 (Model uncertainty) *The DC motor model parameters in the dynamics (8.1) were found by performing system identification on a real motor. We do not discuss this procedure here but instead assume that the estimated parameter values are the true values of the model parameters. In practice, these values are uncertain or maybe even unknown. We could learn the values of these parameters online by including them as optimization variables in (8.2) (as is done in Chapter 5), but we leave this for future work.* □

Results for the three considered cases are given below in Figures 8.1, 8.2, and 8.3-8.4, respectively. The values for the horizon lengths L and T, and the scalar weights λ_r, λ_u, λ_d, and λ_n for each case are given in Table 8.2.

All of the simulation results in this section were computed using Algorithm 4.4.2 given in Chapter 4 on a desktop computer with a 3.4 GHz Intel®Core™ i7 Processor. The number of optimization variables, inequality constraints, equality constraints, and

Table 8.2: Parameter values for the three cases considered.

	Case #1	Case #2	Case #3
$\lambda_r =$	10	10	10
$\lambda_u =$	1e-5	2	1e-4
$\lambda_d =$	N/A	5	5
$\lambda_n =$	1e6	1e6	1e6
$L =$	10	10	10
$T =$	20	20	20

mean, minimum, and maximum computation times for each of the three cases are given

in Table 8.3.

Table 8.3: Numerical performance for the three cases of the DC motor problem.

	Case #1	Case #2	Case #3
# of optimization variables	113	143	143
# of inequality constraints	46	100	100
# of equality constraints	90	90	90
mean time to solve	0.344 ms	0.437 ms	0.517 ms
min. time to solve	0.168 ms	0.235 ms	0.345 ms
max. time to solve	0.764 ms	0.768 ms	1.00 ms

Case #1 considers quadratic penalty functions on the difference between the angle θ

and the reference r, the control input u, and the measurement noise n as given in the cost

function (8.3). Figure 8.1 shows results for Case #1. Due to the dead zone in the plant

that is not modeled in the prediction model, the angle θ gets close to the square-wave

reference but experiences steady-state error. This also results in the computed control

input to be nonzero to try to correct the steady state error.

Case #2 considers quadratic penalty functions on the difference between the angle θ

and the reference r, the control input u, the measurement noise n, and the unmeasured

disturbance d as given in the cost function (8.3). Figure 8.2 shows results for Case #2. In

this case, the optimizer is able to partially accommodate for the dead zone by estimating

that a disturbance (i.e. an external torque working against the controller) is present.
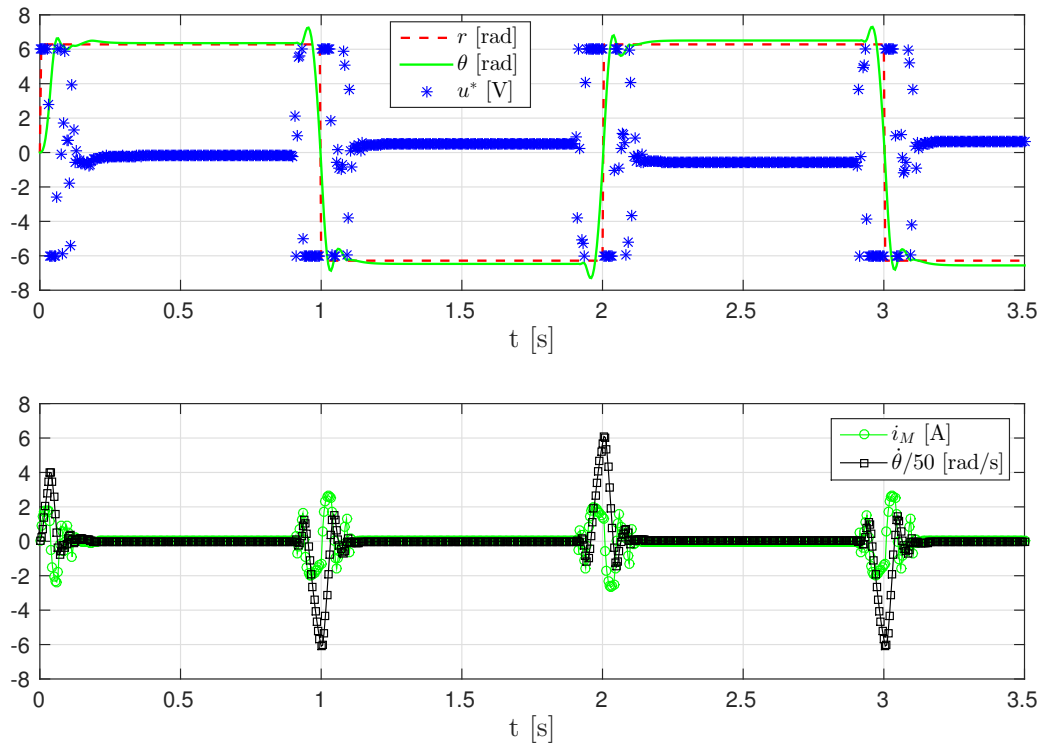
Figure 8.1: Simulation results of DC motor example Case #1. In the top plot, the reference is in red, the measured output in green, the control sequence in blue. In the bottom plot, the actual current is shown in green, and the angular velocity (divided by 50) is shown in black.

This still results in small steady state error but produces smoother trajectories for the angle θ, the control input u, the current i_M, and the angular velocity $\dot{\theta}$. It also results in essentially zero control input when θ is close to the reference.

The third and final case considers quadratic penalty functions on the difference between the angle θ and the reference r, the measurement noise n, and the unmeasured disturbance d as given in the cost function (8.3). Instead of penalizing the control input with a quadratic cost function as in Case #2, in Case #3 we instead penalize the power consumption by including a term multiplying u_s and $i_{M,s}$ in (8.5). Figure 8.3 shows results for Case #3. Figure 8.4 shows a close-up of the beginning of the results shown in Figure 8.3. In this case, the optimizer chooses a control input that quickly switches between large positive and negative voltages in order to force current back into the battery,
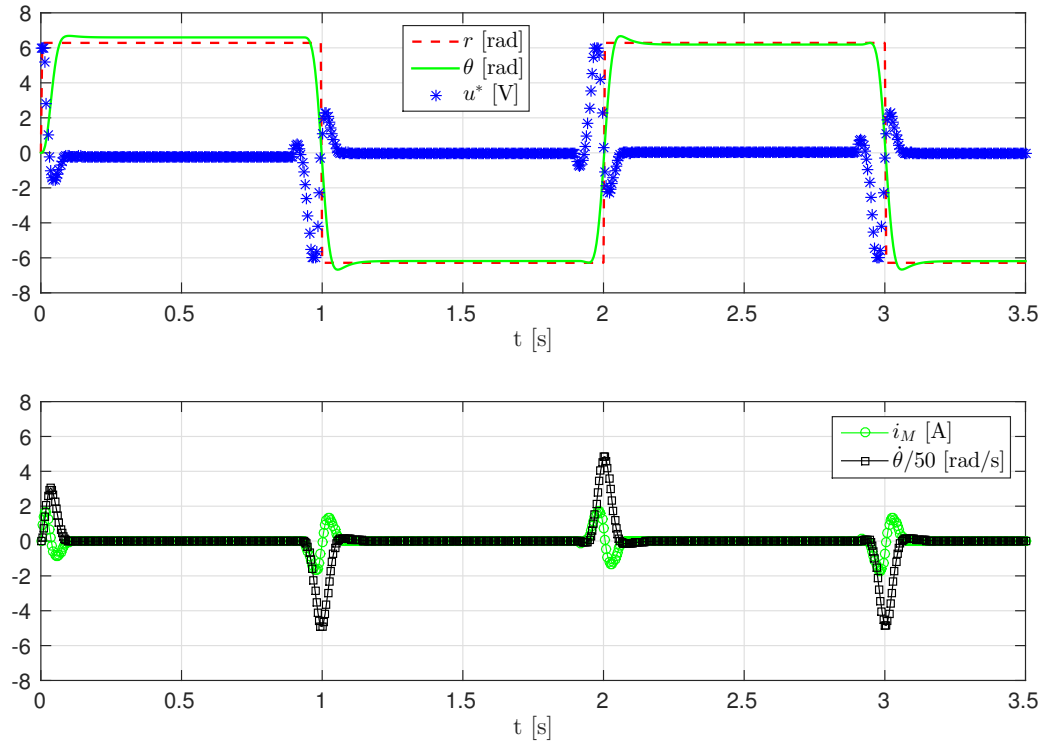
Figure 8.2: Simulation results of DC motor example Case #2. In the top plot, the reference is in red, the measured output in green, the control sequence in blue. In the bottom plot, the actual current is shown in green, and the angular velocity (divided by 50) is shown in black.

thereby recharging it. It does this while keeping the angle θ close to the reference r.

8.1.4   Conclusion

In this section, we discussed output-feedback control of a DC motor. We described a linear time-invariant state space model of the DC motor and considered three different formulations of the MPC/MHE estimation and control approach for estimating the state of the motor and regulating its shaft angle to a desired reference trajectory. These three formulations investigated the results of including a disturbance in the prediction model to account for unknown plant/model mismatch as well as a term penalizing the power consumption of the motor.
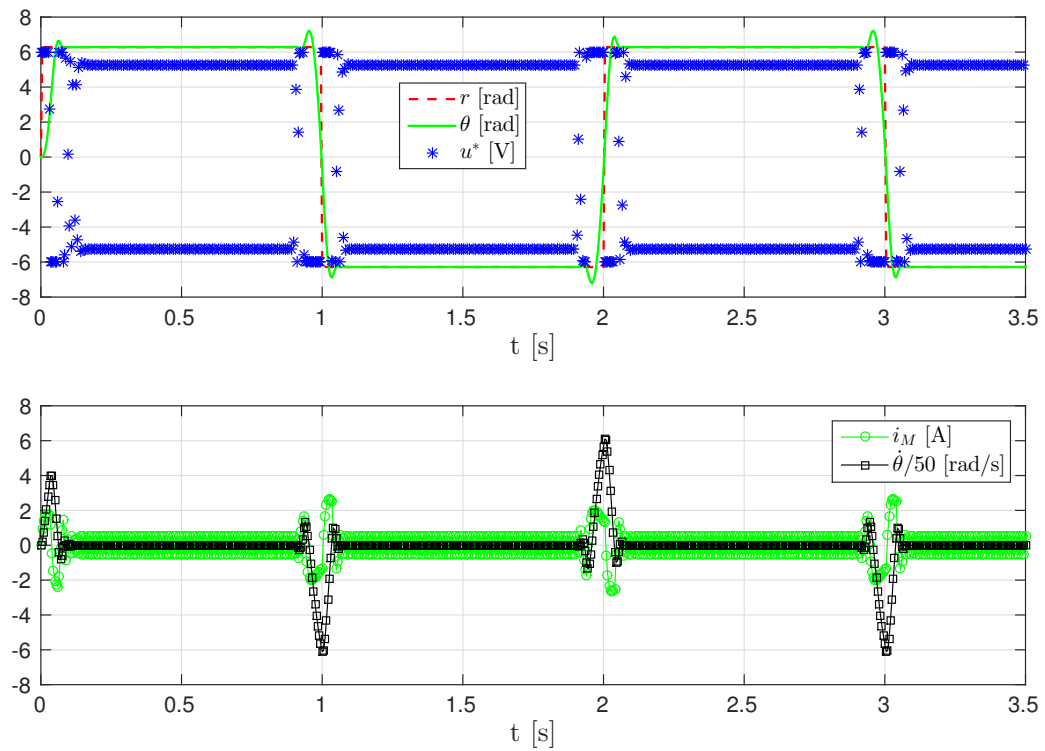
190

Figure 8.3: Simulation results of DC motor example Case #3. The top plot shows the reference in red, the measured output in green, and the control sequence in blue. The bottom plot shows the actual current in green and the angular velocity (divided by 50) in black.
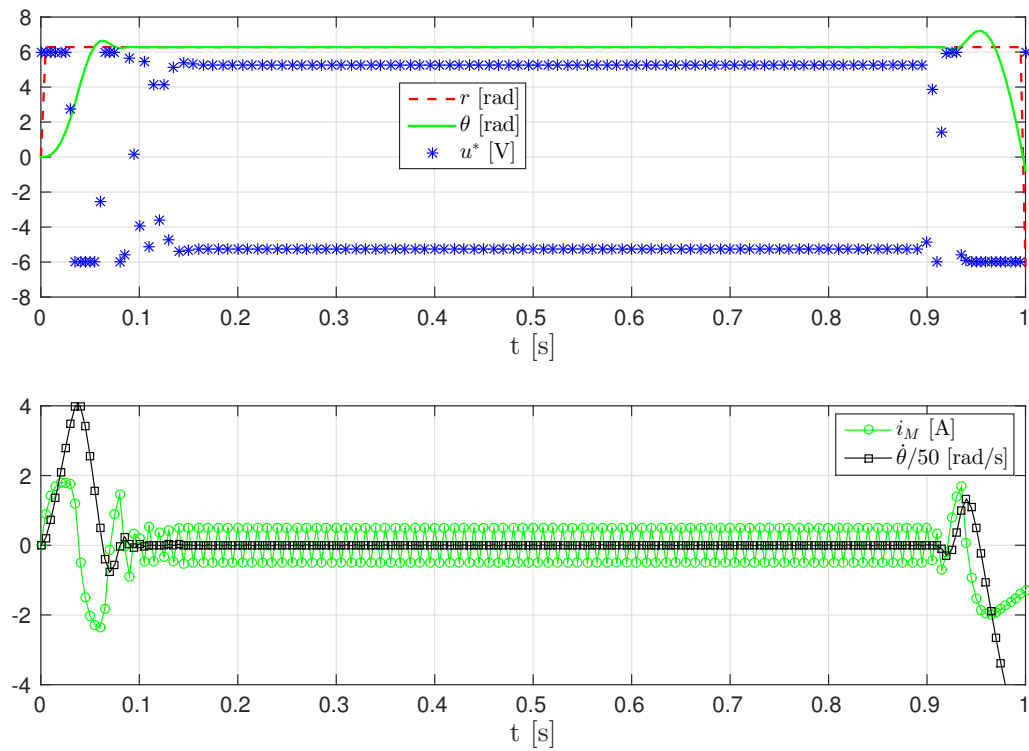
191

Figure 8.4: Close-up of the results at the beginning of the simulation of DC motor example Case #3. The top plot shows the reference in red, the measured output in green, and the control sequence in blue. The bottom plot shows the actual current in green and the angular velocity (divided by 50) in black.

8.2   Distributed optimization for multi-agent consensus

Until this point, all of the examples we have considered have been centralized, i.e. the computations have been performed serially by one processor with all information available. In this section, we consider a distributed control problem where multiple agents attempt to achieve consensus by solving independent optimization problems using only local information. In particular, each agent only has access to a distance measurement from itself to its neighbor (where neighbor is defined using some graph structure), and all of the agents try to rendezvous, or achieve consensus, with a leader agent. For background on distributed control and graph theory, we invite the reader to see, e.g., [15].

Formally, we consider the following linear discrete-time systems with state-space model for every agent $i \in \mathcal{N} := \{1, 2, 3 \cdots, N\}$ given as

$$x_{t+1}^i = A x_t^i + B_i u_t^i + D_i d_t^i, \tag{8.6}$$

where $x_t^i \in \mathbb{R}^{n_x}$ denotes the *state* of agent i at time t, u_t^i denotes the *control input* of agent i at time t and is constrained to belong to the set $\mathcal{U}_i \subset \mathbb{R}^{n_u}$, and d_t^i denotes the *unmeasured disturbance* that agent i is subjected to at time t and belongs to the set $\mathcal{D}_i \subset \mathbb{R}^{n_d}$. All of the agents have the same system matrix A, but they are heterogeneous in that they all have different input and disturbance matrices, B_i and D_i, respectively. We assume that there is a leader agent with dynamics

$$x_{t+1}^0 = A x_t^0.$$

Using only local information, our objective is to design control inputs u_t^i so that the state

of each agent achieves consensus with, or synchronizes with, the state of the leader agent x_t^0, i.e. $\lim_{t \to \infty} \|x_t^i - x_t^0\| = 0$ for all $i \in \mathcal{N}$.

8.2.1   Graphs and synchronization

Let G be a directed graph defined as the pair $G = (V, \mathcal{E})$ with a nonempty finite set of N vertices $V = \{v_1, v_2, \ldots, v_N\}$ and a set of edges $\mathcal{E} \subseteq V \times V$. The connectivity matrix E is defined such that $E = [e_{ij}]$ with $e_{ij} > 0$ if $(v_j, v_i) \in \mathcal{E}$ and $e_{ij} = 0$ otherwise. Each agent in the set of agents $i \in \mathcal{N}$ can be represented by a node v_i in the graph, and the set of neighbors of every node v_i is $N_i = \{v_j : (v_j, v_i) \in \mathcal{E}\}$.

We define the local neighborhood tracking error $\varepsilon_t^i \in \mathbb{R}^{n_x}$ for each node i as [1]

$$\varepsilon_t^i = \sum_{j \in N_i} e_{ij}(x_t^j - x_t^i) + g_i(x_t^0 - x_t^i), \tag{8.7}$$

where $g_i \geqslant 0$ is the pinning gain of node $i \in \mathcal{N}$, which is nonzero if node i is coupled to the control (leader) node v_0. The control node v_0 is assumed to only be connected to a small percentage of nodes in the graph.

8.2.2   Error dynamics

The dynamics of the local neighborhood tracking error for node $i \in \mathcal{N}$ are given as follows:

$$\varepsilon_{t+1}^i = A\varepsilon_t^i - (w_i + g_i)(B_i u_t^i + D_i d_t^i) + \sum_{j \in N_i} e_{ij}(B_j u_t^j + D_j d_t^j)$$

$$y_t^i = C_i \varepsilon_t^i + n_t^i \tag{8.8}$$

where $w_i = \sum_{j \in N_i} e_{ij}$ is the weighted in-degree of node i, and y_t^i denotes the *measured output* of agent i at time t subjected to the *measurement noise* n_t^i which belongs to the set $\mathcal{N}_i \subset \mathbb{R}^{n_n}$. Therefore, each agent only has noisy measurements of its local neighborhood tracking error for feedback. Our objective is to minimize the local neighborhood tracking errors ε_t^i, which guarantees approximate synchronization of the agents [1].

8.2.3 Estimation and Control Approach

We formulate this problem using the combined MPC/MHE estimation and control approach presented in Chapter 2. Each agent's *control objective* is to select the control signal $u_t^i \in \mathcal{U}_i$, $\forall t \in \mathbb{Z}_{\geqslant 0}$ so as to minimize a finite-horizon criterion of the form

$$J_t^i(\varepsilon_{t-L}^i, u_{t-L:t+T-1}^i, d_{t-L:t+T-1}^i, y_{t-L:t}^i) :=$$

$$\sum_{s=t}^{t+T} \|C_i \varepsilon_s^i\|^2 + \sum_{s=t}^{t+T-1} \lambda_u^i \|u_s^i\|^2 - \sum_{s=t-L}^{t} \lambda_n^i \|n_s^i\|^2 - \sum_{s=t-L}^{t+T-1} \lambda_d^i \|d_t^i\|^2 \quad (8.9)$$

where $T \in \mathbb{Z}_+$ is the forward prediction and control horizon, $L \in \mathbb{Z}_+$ is the backward estimation horizon, and λ_u^i, λ_n^i, and λ_d^i are positive scalar weights on the control input, measurement noise, and input disturbance, respectively, for agent i.

The performance criterion (8.9) for every agent $i \in \mathcal{N}$ depends on the unknown initial local neighborhood tracking error ε_{t-L}^i, the unknown disturbance input sequence $d_{t-L:t+T-1}^i$, the measured output sequence $y_{t-L:t}^i$, and the control input sequence $u_{t-L:t+T-1}^i$. The control input sequence is composed of two distinct sequences: the (known) past inputs $u_{t-L:t-1}^i$ that have already been applied, and the future inputs $u_{t:t+T-1}^i$ that still need to be selected. To select the future inputs $u_{t:t+T-1}^i$, each agent optimizes the criterion (8.9) with respect to these variables. At a given time $t \in \mathbb{Z}_{\geqslant 0}$, we do not know the value of the variables ε_{t-L}^i and $d_{t-L:t+T-1}^i$, so we optimize the criterion under worst-case

assumptions on these variables.

This motivates the solution of a finite-horizon min-max optimization problem, to be solved at each time t for each agent i, of the form

$$\min_{\hat{u}^i_{t:t+T-1|t}} \quad \max_{\substack{\hat{\varepsilon}^i_{t-L|t}, \\ \hat{d}^i_{t-L:t+T-1|t}}} \quad J^i_t(\hat{\varepsilon}^i_{t-L|t}, u^i_{t-L:t-1}, \hat{u}^i_{t:t+T-1|t}, \hat{d}^i_{t-L:t+T-1|t}, y^i_{t-L:t}) \tag{8.10}$$

with cost function $J^i_t(\cdot)$ as defined in (8.9) and subject to,

$$\varepsilon^i_{s+1} = A\hat{\varepsilon}^i_{s|t} - (w_i + g_i)(B_i u^i_s + D_i \hat{d}^i_{s|t}) + \sum_{j \in N_i} e_{ij} B_j u^j_s \qquad s = t - L \tag{8.11a}$$

$$\varepsilon^i_{s+1} = A\varepsilon^i_s - (w_i + g_i)(B_i u^i_s + D_i \hat{d}^i_{s|t}) + \sum_{j \in N_i} e_{ij} B_j u^j_s \qquad \forall s \in \mathbb{Z}^{t-1}_{t-L+1} \tag{8.11b}$$

$$\varepsilon^i_{s+1} = A\varepsilon^i_s - (w_i + g_i)(B_i \hat{u}^i_{s|t} + D_i \hat{d}^i_{s|t}) \qquad \forall s \in \mathbb{Z}^{t+T-1}_t \tag{8.11c}$$

$$y^i_s = C_i \varepsilon^i_s + n^i_s \qquad \forall s \in \mathbb{Z}^t_{t-L} \tag{8.11d}$$

$$\hat{d}^i_s \in \mathcal{D}_i \qquad \forall s \in \mathbb{Z}^{t+T-1}_{t-L} \tag{8.11e}$$

$$\hat{u}^i_s \in \mathcal{U}_i \qquad \forall s \in \mathbb{Z}^{t+T-1}_t. \tag{8.11f}$$

Equations (8.11a)-(8.11d) ensure that the local neighborhood tracking error dynamics (8.8) are satisfied. Equations (8.11e)-(8.11f) ensure that the computed variables d^i_s and u^i_s belong to their respective constraint sets.

The optimization (8.10) is repeated at each time step t, and we use as the control input for every agent i, the first element of the sequence,

$$\hat{u}^{i*}_{t:t+T-1} = \{\hat{u}^{i*}_t, \hat{u}^{i*}_{t+1}, \hat{u}^{i*}_{t+2}, \ldots, \hat{u}^{i*}_{t+T-1}\}$$

that minimizes (8.10), leading to the following control law:

$$u_t^i = \hat{u}_t^{i*}, \quad \forall t \in \mathbb{Z}_{\geqslant 0}. \tag{8.12}$$

8.2.4   Simulations

Now we consider a simple example of this distributed control problem and provide simulation results. Figure 8.5 depicts the graph that is considered in this example where there are three agents and one leader.
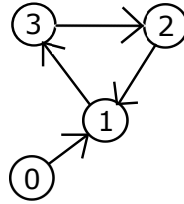

Figure 8.5: Graph topology for neighborhood of agents. Agent 1 receives information from the leader (agent 0) and agent 2, agent 2 receives information from agent 3, and agent 3 receives information from agent 1.

Each agent's dynamics and error dynamics are given as in (8.6) and (8.8), respectively, with the matrices

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \; B_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \; B_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \; B_3 = \begin{bmatrix} 2 \\ -2 \end{bmatrix},$$

$$D_1 = B_1, \; D_2 = B_2, \; D_3 = B_3,$$

$$C_1 = \begin{bmatrix} 0 & 1 \end{bmatrix}, \; C_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}, \; C_3 = \begin{bmatrix} 0 & 1 \end{bmatrix},$$

and the weights $g_1 = 2$, $g_2 = g_3 = 0$, and $w_i = 1$ for all i, and $e_{ij} = 1$ for all i and j. The true noise and disturbances are normally distributed random variables given as $n_t^i \sim \mathcal{N}(0.05, 0.1^2)$ and $d_t^i \sim \mathcal{N}(0, 0.03^2)$, respectively, for all $i \in \mathcal{N}$ and all $t \in \mathbb{Z}_{\geqslant 0}$. The other parameters and constraint sets included in cost function (8.9) and optimization

197

(8.10) are chosen as $T = 5$, $L = 5$, $\lambda_u = 1$, $\lambda_d = 100$, $\lambda_n = 1000$, $\mathcal{U}_i := \{u_t^i \in \mathbb{R}^{n_u} : -0.5 \leqslant u_t^i \leqslant 0.5\}$, $\mathcal{D}_i := \{d_t^i \in \mathbb{R}^{n_d} : -0.5 \leqslant u_t^i \leqslant 0.5\}$, and $\mathcal{N}_i := \mathbb{R}^{n_n}$ for this example.

Simulation results are given in Figures 8.6, 8.7, 8.8, and 8.9, when the initial states of the leader and agents are $x_1^0 = [1, 1]'$, $x_1^1 = [5, 1]'$, $x_1^2 = [-4, 2]'$, and $x_1^3 = [4, -3]'$. Figure 8.6 shows the values of the state for the leader and all of the agents. Even though their states begin in various locations, the states of each agent quickly converge to the states of the leader, so the agents achieve consensus. Figure 8.7 shows a phase plot of the results an again shows that the agents achieve consensus with the leader. Figure 8.8 shows the control inputs applied by each of the agents as well as the true disturbances and noise that each of the agents was subjected to. Despite the fact that the disturbances and noise are nonzero, the agents are able to compute control inputs that successfully approximately synchronize their states. Finally, Figure 8.9 shows the estimated values for the local neighborhood tracking errors ε^i for each of the agents. These errors converge to small values, so the agents approximately achieve consensus.

8.2.5   Conclusions

In this section we considered the distributed control of multiple agents with the objective of synchronizing the state of each agent with the state of a leader agent using only noisy local information. We defined a graph structure and dynamics for the local neighborhood tracking error of each agent. We formulated this problem using the combined MPC/MHE estimation and control approach described in Chapter 2 and simulated a simple example. The simulation results show that the agents were successfully able to compute control inputs that achieved approximate synchronization (consensus) even in the presence of input disturbances and measurement noise.
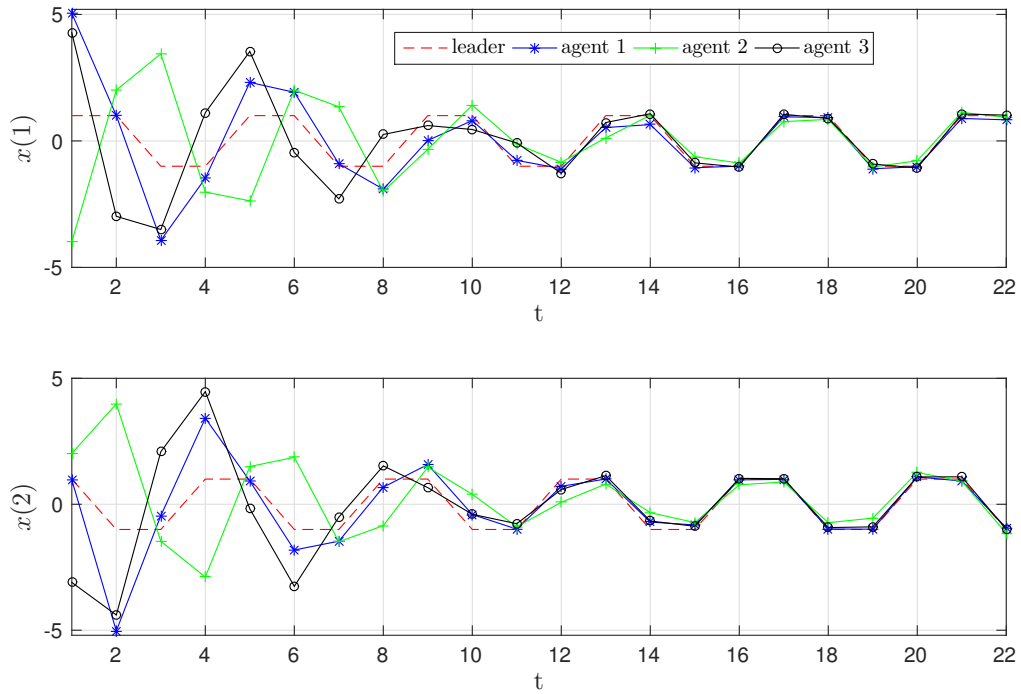
Figure 8.6: Consensus of all three agents with the leader: values for $x(1)$ (top plot) and $x(2)$ (bottom plot) converge to those of the leader.
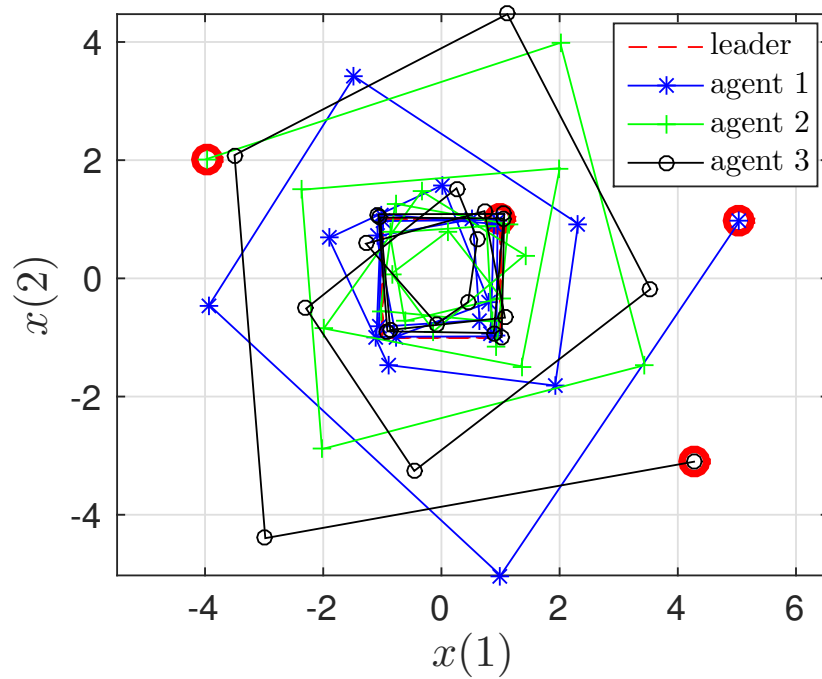

Figure 8.7: Phase plot showing consensus. The starting locations are given as red circles.

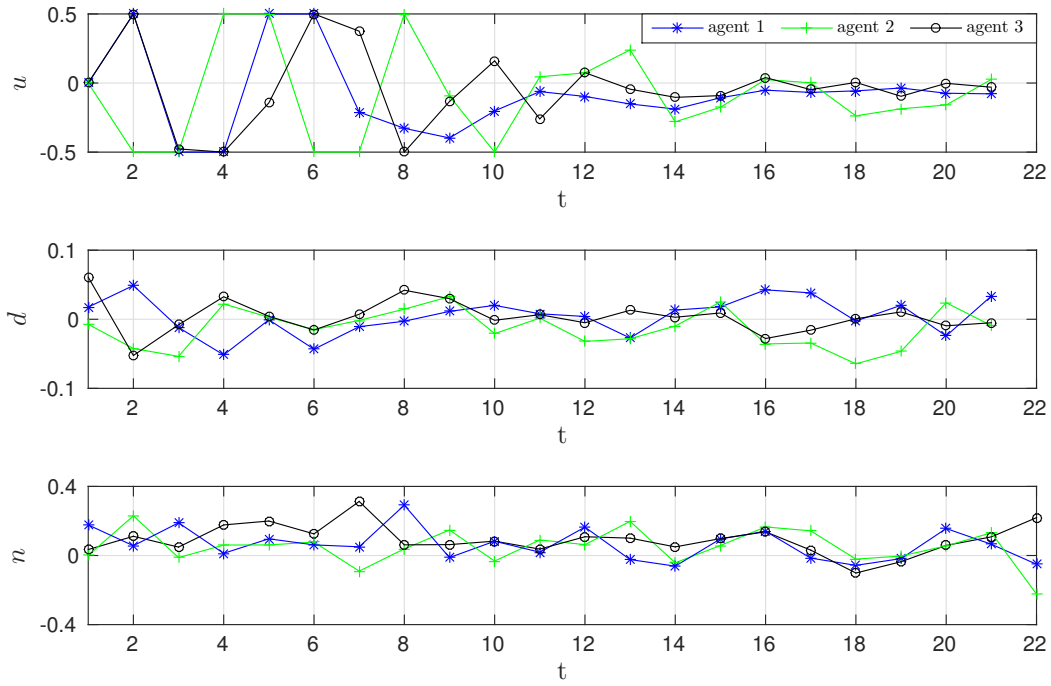Figure 8.8: u^* applied (top plot), actual d (middle plot), and actual n (bottom plot) for each agent.
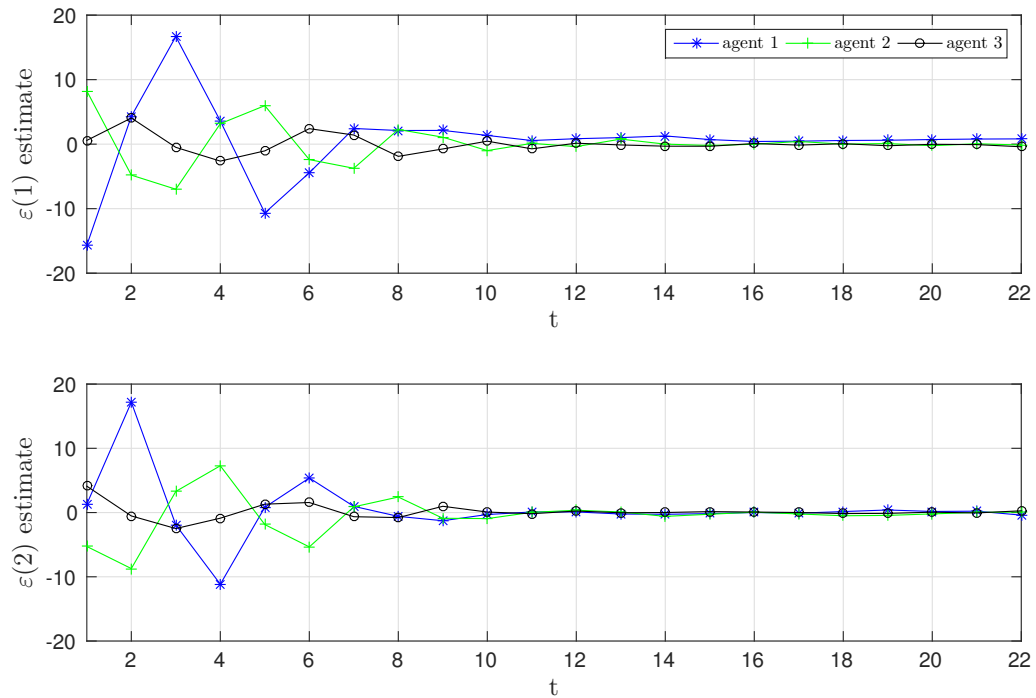

Figure 8.9: Estimates of $\varepsilon(1)$ and $\varepsilon(2)$ showing that the local neighborhood tracking errors converge to small values.

Chapter 9

Conclusion

9.1 Contributions of this thesis

As discussed in the Chapter 1, this thesis included the theory and design of a new approach to solving MHE and MPC problems simultaneously as a single min-max optimization problem [22, 25, 23] as well as the investigation of using this approach in several applications (see, e.g. [85, 21]). As a recap, each of the chapters contained the following material:

Chapter 2 provided the main theoretical contribution of this thesis addressing stability of the proposed combined MPC/MHE approach. In particular, the proposed output-feedback controller results in closed-loop trajectories along which the state of the process remains bounded, and, for tracking problems, provides explicit bounds on the tracking error. These results relied on three assumptions: the existence of saddle-point equilibria for the min-max optimization, a terminal cost that is a control ISS-Lyapunov function with respect to the disturbance input, and observability of the nonlinear process essentially requiring the backwards horizon to be sufficiently large.

Chapter 3 addressed the first assumption invoked in Chapter 2, i.e., the existence of saddle-point equilibria. We derived conditions under which a saddle-point solution to the min-max optimization is guaranteed to exist and showed that for linear processes and quadratic costs, these conditions boil down to observability of the linear process and appropriately choosing weights in the quadratic cost functions.

Chapter 4 presented two numerical algorithms that can be used to solve general min-max optimization problems, including those introduced in Chapter 2. Both algorithms involve primal-dual interior-point methods that rely on Newton's method to solve a relaxed version of the Karush-Kuhn-Tucker (KKT) conditions associated with the coupled optimizations that define a saddle-point equilibrium. The second algorithm specialized this method for formulations with common latent variables.

Chapter 5 discussed the scenario where the process model includes uncertain or unknown parameters. We showed that the uncertain parameters can be included as optimization variables in the combined MPC/MHE approach and learned online, and the results from Chapter 2 still hold. Several examples with parametric uncertainty were considered and solved effectively.

The second part of this thesis used this combined MPC/MHE approach for several applications including the coordination of unmanned aerial vehicles for vision-based target tracking of a moving ground vehicle, feedback control of an artificial pancreas system for the treatment of Type 1 Diabetes, estimation and control of a DC motor, and distributed control of multi-agents for achieving consensus.

In **Chapter 6**, we considered the coordination of multiple unmanned aerial vehicles (UAVs) for vision-based target-tracking of a moving ground vehicle. A novel cost function was used in order to achieve the best vision-based estimate of the target's location, and we showed in simulations that the UAVs are able to coordinate their motion to track the target even when the target acts evasively.

Chapter 7 explored the use of the MPC/MHE approach for the feedback control of an artificial pancreas system for the treatment of Type 1 Diabetes. We designed an asymmetric cost function to facilitate appropriate controller response to high and low blood-glucose levels and showed that the combined MPC/MHE approach is advantageous for this application when compared to another approach using state-feedback MPC and a recursive state estimator.

Finally, in **Chapter 8**, we discussed two additional applications which include output-feedback control of a DC motor and distributed optimization for multi-agent consensus. We showed that this MPC/MHE approach is an effective control approach for both of these applications.

9.2 Future work

As discussed at the end of several chapters, there are many extensions that can be made to the work in this thesis as well as other future work that may be considered.

Regarding the theoretical results given in Part I of this thesis, more specific stability results could be derived when considering certain classes of systems or cost functions. A study on recursive feasibility of this approach would also be useful. In some cases, the requirement of a saddle-point solution to the min-max optimization may be too restrictive, so relaxing this assumption to allow being ε-close to a saddle-point solution may be useful. This could be related to results for ε-Nash equilibria.

There is certainly more work to be done regarding the numerical algorithms described in Chapter 4. A convergence analysis of both Algorithms 4.3.1 and 4.4.2 should be done. Furthermore, the development of similar algorithms to solve these types of optimization problems and trade offs between methods could be investigated. For example, a Barrier interior-point algorithm could be developed which may be more robust than the primal-

dual algorithm for non-convex poorly conditioned problems.

Additional work could also be done regarding the adaptation to and learning of uncertain parameters as discussed in Chapter 5. It would be interesting to study the particular conditions for which the estimates of the uncertain parameters computed using the combined MPC/MHE approach are guaranteed to converge to their true values. This may require particular excitation signals or a sufficient number of measurements.

The simulation results for the applications in Part II show that this combined MPC/MHE approach is promising and may be advantageous compared to other estimation and control approaches. This motivates the implementation of this combined MPC/MHE approach in several real-world experiments to validate these simulation results and truly determine its potential.

Bibliography

[1] M. I. Abouheaf, F. L. Lewis, K. G. Vamvoudakis, S. Haesaert, and R. Babuska. Multi-agent discrete-time graphical games and reinforcement learning solutions. *Automatica*, 50(12):3038–3053, 2014.

[2] Veronica Adetola, Darryl DeHaan, and Martin Guay. Adaptive model predictive control for constrained nonlinear systems. *Systems & Control Letters*, 58(5):320–326, May 2009.

[3] Angelo Alessandri, Marco Baglietto, and Giorgio Battistelli. Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes. *Automatica*, 44(7):1753–1765, 2008.

[4] Frank Allgöwer, Thomas A. Badgwell, S. Joe Qin, James B. Rawlings, and Steven J. Wright. Nonlinear predictive control and moving horizon estimation-an introductory overview. In *Advances in control*, pages 391–449. Springer, 1999.

[5] Karl J. Åström and Björn Wittenmark. *Adaptive Control*. Courier Corporation, 2013.

[6] T. Başar and G. J. Olsder. *Dynamic Noncooperative Game Theory*. Academic Press, London, 1995.

[7] Tamer Başar and Pierre Bernhard. *H-infinity optimal control and related minimax design problems: a dynamic game approach*. Springer, 2008.

[8] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Min-max control of constrained uncertain discrete-time linear systems. *Automatic Control, IEEE Transactions on*, 48(9):1600–1606, 2003.

[9] Alberto Bemporad and Manfred Morari. Robust model predictive control: A survey. In *Robustness in identification and control*, pages 207–226. Springer, 1999.

[10] B. Wayne Bequette. A critical assessment of algorithms and challenges in the development of a closed-loop artificial pancreas. *Diabetes Technology and Therapeutics*, 7(1):28–47, 2005.

[11] Lorenz T Biegler. Efficient solution of dynamic optimization and nmpc problems. In *Nonlinear model predictive control*, pages 219–243. Springer, 2000.

[12] Lorenz T. Biegler and James B. Rawlings. Optimization approaches to nonlinear model predictive control. Technical report, Argonne National Lab., IL (USA), 1991.

[13] D. Boiroux. *Model predictive control algorithms for pen and pump insulin administration.* PhD thesis, Technical University of Denmark, 2012.

[14] Stephen P. Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

[15] Francesco Bullo, Jorge Cortés, and Sonia Martínez. *Distributed Control of Robotic Networks.* Applied Mathematics Series. Princeton University Press, 2009.

[16] Eduardo F. Camacho and Carlos Bordons. *Model predictive control*, volume 2. Springer London, 2004.

[17] Peter J Campo and Manfred Morari. Robust model predictive control. In *American Control Conference, 1987*, pages 1021–1026. IEEE, 1987.

[18] Vito Cerone, Davide Andreo, Mats Larsson, and Diego Regruto. Stabilization of a riderless bicycle [applications of control]. *Control Systems, IEEE*, 30(5):23–32, 2010.

[19] Hong Chen, Carsten W. Scherer, and Frank Allgöwer. A game theoretic approach to nonlinear robust receding horizon control of constrained systems. In *Proceedings of the American Control Conference, 1997.*, volume 5, pages 3073–3077. IEEE, 1997.

[20] Claudio Cobelli, Eric Renard, and Boris Kovatchev. Artificial pancreas: Past, present, future. *Diabetes*, 60:2672–2682, November 2011.

[21] David A. Copp, Ravi Gondhalekar, and João P. Hespanha. Simultaneous model predictive control and moving horizon estimation for blood glucose regulation in type 1 diabetes. *Submitted.*, 2016.

[22] David A. Copp and João P. Hespanha. Nonlinear output-feedback model predictive control with moving horizon estimation. In *53rd Conference on Decision and Control (CDC)*, pages 3511–3517, Los Angeles, CA, December 2014. IEEE.

[23] David A. Copp and João P. Hespanha. *Addressing Adaptation and Learning in the Context of MPC with MHE.* Control of Complex Systems: Theory and Applications. Elsevier, 1st edition, 2016.

[24] David A. Copp and João P. Hespanha. Conditions for saddle-point equilibria in output-feedback MPC with MHE. In *Proceedings of the American Control Conference (ACC)*, pages 13–19. IEEE, July 2016.

[25] David A. Copp and João P. Hespanha. Simultaneous nonlinear model predictive control and state estimation. *Automatica*, (in press), 2016.

[26] E. Dassau, S. A. Brown, A. Basu, J. E. Pinsker, Y. C. Kudva, R. Gondhalekar, and et al. Adjustment of open-loop settings to improve closed-loop results in type I diabetes: a multicenter randomized trial. *The Journal of Clinical Endocrinology & Metabolism*, 100:3878–3886, 2015.

[27] D. DeHaan and M. Guay. Adaptive robust mpc: A minimally-conservative approach. In *American Control Conference*, pages 3937–3942. IEEE, July 2007.

[28] Moritz Diehl and Jakob Bjornberg. Robust dynamic programming for min-max model predictive control of constrained uncertain systems. *Automatic Control, IEEE Transactions on*, 49(12):2253–2257, 2004.

[29] Moritz Diehl, Hans Joachim Ferreau, and Niels Haverbeke. Efficient numerical methods for nonlinear MPC and moving horizon estimation. In *Nonlinear Model Predictive Control*, pages 391–417. Springer, 2009.

[30] C. Ding, A. A. Morye, J. A. Farrell, and A. K. Roy-Chowdhury. Coordinated sensing and tracking for mobile camera platforms. In *American Control Conf.*, pages 5114–5119. IEEE, 2012.

[31] Francis J. Doyle III, Lauren M. Huyett, Joon Bok Lee, Howard C. Zisser, and Eyal Dassau. Closed-loop artificial pancreas systems: Engineering the algorithms. *Diabetes Care*, 37:1191–1197, May 2014.

[32] Christian Ellingsen, Eyal Dassau, Howard C. Zisser, Benyamin Grosman, Matthew W. Percival, Lois Jovanovič, and Francis J. Doyle III. Safety constraints in an artificial pancreatic β cell: An implementation of model predictive control with insulin on board. *Journal of Diabetes Science and Technology*, 3:536–544, May 2009.

[33] Francisco Facchinei, Houyuan Jiang, and Liqun Qi. A smoothing method for mathematical programs with equilibrium constraints. *Mathematical programming*, 85(1):107–134, 1999.

[34] Rolf Findeisen, Lars Imsland, Frank Allgöwer, and Bjarne A Foss. State and output feedback nonlinear model predictive control: An overview. *European journal of control*, 9(2):190–206, 2003.

[35] Eric W. Frew. Sensitivity of cooperative geolocation to orbit coordination. In *AIAA Guid., Nav., and Control Conf.*, pages 3869–3892, Hilton Head, SC, aug 2007.

[36] David Yang Gao. Canonical dual transformation method and generalized triality theory in nonsmooth global optimization. *Journal of Global Optimization*, 17(1-4):127–160, 2000.

[37] Ravi Gondhalekar, Eyal Dassau, and Francis J. Doyle III. MPC design for rapid pump-attenuation and expedited hyperglycemia response to treat T1DM with an artificial pancreas. In *American Control Conference (ACC)*, pages 4224–4230, June 2014.

[38] Ravi Gondhalekar, Eyal Dassau, and Francis J. Doyle III. Velocity-weighting to prevent controller-induced hypoglycemia in MPC of an artificial pancreas to treat T1DM. In *American Control Conference (ACC)*, pages 1635–1640, July 2015.

[39] Ravi Gondhalekar, Eyal Dassau, and Francis J. Doyle III. Periodic zone-MPC with asymmetric costs for outpatient-ready safety of an artificial pancreas to treat type 1 diabetes. *Automatica*, 71:237–246, March 2016.

[40] Ravi Gondhalekar, Eyal Dassau, Howard C. Zisser, and Francis J. Doyle III. Periodic-zone model predictive control for diurnal closed-loop operation of an artificial pancreas. *Journal of Diabetes Science and Technology*, 7:1446–1460, November 2013.

[41] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In Vincent Blondel, Stephen Boyd, and Hidenori Kimura, editors, *Recent Advances in Learning and Control*, volume 371 of *Lecture Notes in Control and Information Sciences*, pages 95–110. Springer Berlin / Heidelberg, 2008.

[42] Benyamin Grosman, Eyal Dassau, Howard C. Zisser, Lois Jovanovič, and Francis J. Doyle III. Zone model predictive control: A strategy to minimize hyper- and hypoglycemic events. *Journal of Diabetes Science and Technology*, 4(4):961–975, July 2010.

[43] Lars Grüne and Jürgen Pannek. *Nonlinear Model Predictive Control*. Springer, 2011.

[44] Ahmad Haidar. The artificial pancreas: How closed-loop control is revolutionizing diabetes. *IEEE Control Systems*, 36(5):28–47, October 2016.

[45] Rebecca A. Harvey, Eyal Dassau, Howard C. Zisser, Dale E. Seborg, Lois Jovanovič, and Francis J. Doyle III. Design of the health monitoring system for the artificial pancreas: Low glucose prediction module. *Journal of Diabetes Science and Technology*, 6:1345–1354, November 2012.

[46] Eric L. Haseltine and James B. Rawlings. Critical evaluation of extended Kalman filtering and moving-horizon estimation. *Industrial & Engineering Chemistry Research*, 44(8):2451–2460, 2005.

[47] R. Hovorka. Continuous glucose monitoring and closed-loop systems. *Diabetic Medicine*, 23:1–12, January 2005.

[48] Roman Hovorka, Valentina Canonico, Ludovic J Chassin, Ulrich Haueter, Massimo Massi-Benedetti, Marco Orsini Federici, Thomas R Pieber, Helga C Shaller, Lukas Schaupp, Thomas Vering, and Malgorzata E Wilinska. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiological Measurement*, 25:905–920, 2004.

[49] Rui Huang, Sachin C. Patwardhan, and Lorenz T. Biegler. Robust extended Kalman filter based nonlinear model predictive control formulation. In *Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, pages 8046–8051, Shanghai, P.R. China, December 2009.

[50] Lars Imsland, Rolf Findeisen, Eric Bullinger, Frank Allgöwer, and Bjarne A Foss. A note on stability, robustness and performance of output feedback nonlinear model predictive control. *Journal of Process Control*, 13(7):633–644, 2003.

[51] Petros A. Ioannou and Jing Sun. *Robust Adaptive Control.* Courier Corporation, 2012.

[52] Seungkeun Kim, Hyondong Oh, and Antonios Tsourdos. Nonlinear model predictive coordinated standoff tracking of a moving ground vehicle. *Journal of Guidance, Control, and Dynamics*, 36(2):557–566, 2013.

[53] Derek B. Kingston. *Decentralized Control of Multiple UAVs for perimeter and target surveillance.* PhD thesis, Brigham Young University, Dec. 2007.

[54] Boris P. Kovatchev, Marc Breton, Chiara Dalla Man, and Claudio Cobelli. In silico preclinical trials: A proof of concept in closed-loop control of type 1 diabetes. *Journal of Diabetes Science and Technology*, 3:44–55, January 2009.

[55] Joseph Kreimer and Reuven Y. Rubinstein. Nondifferentiable optimization via smooth approximation: General analytical approach. *Annals of Operations Research*, 39(1):97–119, 1992.

[56] Sanjay Lall and Keith Glover. A game theoretic approach to moving horizon control. In *Advances in Model-Based Predictive Control.* Oxford University Press, 1994.

[57] M. Lazar, D. Muñoz de la Peña, W. P. M. H. Heemels, and T. Alamo. On input-to-state stability of min–max nonlinear model predictive control. *Systems & Control Letters*, 57(1):39–48, 2008.

[58] Jay H. Lee and Zhenghong Yu. Worst-case formulations of model predictive control for systems with bounded parameters. *Automatica*, 33(5):763–781, 1997.

[59] Justin J. Lee, Ravi Gondhalekar, and Francis J. Doyle III. Design of an artificial pancreas using zone model predictive control with a moving horizon state estimator. In *53rd IEEE Conference on Decision and Control*, pages 6975–6980, December 2014.

[60] W. S. Levine, editor. *The control handbook*. CRC Press., Boca Raton, FL, USA, 2nd edition, 2011.

[61] Daniel Liberzon, Eduardo D. Sontag, and Yuan Wang. Universal construction of feedback laws achieving ISS and integral-ISS disturbance attenuation. *Systems & Control Letters*, 46(2):111–127, 2002.

[62] D. Limon, T. Alamo, D. M. Raimondo, D. Muñoz de la Peña, J. M. Bravo, A. Ferramosca, and E. F. Camacho. Input-to-state stability: a unifying framework for robust model predictive control. In *Nonlinear model predictive control*, pages 1–26. Springer, 2009.

[63] Jinfeng Liu. Moving horizon state estimation for nonlinear systems with bounded uncertainties. *Chemical Engineering Science*, 93:376–386, 2013.

[64] Johan Löfberg. Towards joint state estimation and control in minimax MPC. In *Proceedings of 15th IFAC World Congress, Barcelona, Spain*, 2002.

[65] Irvin J. Lustig, Roy E. Marsten, and David F. Shanno. On implementing Mehrotra's predictor-corrector interior-point method for linear programming. *Siam Journal on Optimization*, 2(3):435–449, 1992.

[66] Lili Ma and Naira Hovakimyan. Cooperative target tracking in balanced circular formation: Multiple UAVs tracking a ground vehicle. In *American Control Conf.*, pages 5386–5391. IEEE, 2013.

[67] L Magni, G De Nicolao, R Scattolini, and F Allgöwer. Robust model predictive control for nonlinear discrete-time systems. *International Journal of Robust and Nonlinear Control*, 13(3-4):229–246, 2003.

[68] L. Magni, D.M. Raimondo, C. Dalla Man, G. De Nicolao, B. Kovatchev, and C. Cobelli. Model predictive control of glucose concentration in type I diabetic patients: An in silico trial. *Biomedical Signal Processing and Control*, 4:338–346, May 2009.

[69] Lalo Magni, Davide M. Raimondo, Chiara Dalla Man, Marc Breton, Stephen Patek, Giuseppe De Nicolao, Claudio Cobelli, and Boris P. Kovatchev. Evaluating the efficacy of closed-loop glucose regulation via control-variability grid analysis. *Journal of Diabetes Science and Technology*, 2, July 2008.

[70] Lalo Magni and Riccardo Scattolini. Robustness and robust design of MPC for nonlinear discrete-time systems. In *Assessment and future directions of nonlinear model predictive control*, pages 239–254. Springer, 2007.

[71] Mahendra Mallick. Geolocation using video sensor measurements. In *IEEE International Conf. on Information Fusion*, Quebec, Canada, July 2007.

[72] David Q. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, December 2014.

[73] David Q. Mayne and H. Michalska. Adaptive receding horizon control for constrained nonlinear systems. In *Proceedings of the 32nd IEEE Conference on Decision and Control*, pages 1286–1291, December 1993.

[74] David Q. Mayne, Sasa V. Raković, Rolf Findeisen, and Frank Allgöwer. Robust output feedback model predictive control of constrained linear systems. *Automatica*, 42(7):1217–1222, 2006.

[75] David Q. Mayne, Sasa V. Raković, Rolf Findeisen, and Frank Allgöwer. Robust output feedback model predictive control of constrained linear systems: Time varying case. *Automatica*, 45(9):2082–2087, 2009.

[76] David Q. Mayne, James B. Rawlings, Christopher V. Rao, and Pierre O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[77] Hannah Michalska and David Q. Mayne. Moving horizon observers and observer-based control. *IEEE Transactions on Automatic Control*, 40(6):995–1006, June 1995.

[78] Scott A. Miller, Zachary A. Harris, and Edwin K. P. Chong. A POMDP framework for coordinated guidance of autonomous UAVs for multitarget tracking. *EURASIP J. Adv. Signal Process*, pages 1–17, 2009.

[79] Manfred Morari and Jay H Lee. Model predictive control: past, present and future. *Computers & Chemical Engineering*, 23(4):667–682, 1999.

[80] D. Munõz de la Peña, T. Alamo, D. R. Ramírez, and E. F. Camacho. Min-max model predictive control as a quadratic program. *Control Theory & Applications, IET*, 1(1):328–333, 2007.

[81] Yurii Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, May 2005.

[82] Robert S. Parker, Francis J. Doyle III, and Nicholas A. Peppas. A model-based algorithm for blood glucose control in type I diabetic patients. *IEEE Transactions on Biomedical Engineering*, 46(2):148–157, February 1999.

[83] Cameron Peterson and Derek A. Paley. Multivehicle coordination in an estimated time-varying flowfield. *Journal of guidance, control, and dynamics*, 34(1):177–191, 2011.

[84] S. Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.

[85] Steven A. P. Quintero, David A. Copp, and João P. Hespanha. Robust UAV coordination for target tracking using output-feedback model predictive control with moving horizon estimation. In *2015 American Control Conference (ACC)*, pages 3758 – 3764. IEEE, July 2015.

[86] Steven A. P. Quintero, David A. Copp, and João P. Hespanha. Robust coordination of small UAVs for vision-based target tracking using output-feedback MPC with MHE. In *Cooperative Control of Multi-agent Systems: Theory and Applications*. Wiley, 2016.

[87] Steven A. P. Quintero, Francesco Papi, Daniel J. Klein, Luigi Chisci, and João P. Hespanha. Optimal UAV coordination for target tracking using dynamic programming. In *IEEE Conf. on Decision and Control*, Atlanta, GA, December 2010.

[88] Davide Martino Raimondo, Daniel Limon, Mircea Lazar, Lalo Magni, and Eduardo Fernández Camacho. Min-max model predictive control of nonlinear systems: A unifying overview on stability. *European Journal of Control*, 15(1):5–21, 2009.

[89] Christopher V. Rao, James B. Rawlings, and Jay H. Lee. Constrained linear state estimation-a moving horizon approach. *Automatica*, 37(10):1619–1628, 2001.

[90] Christopher V. Rao, James B. Rawlings, and David Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *Automatic Control, IEEE Transactions on*, 48(2):246–258, 2003.

[91] Christopher V. Rao, Stephen J. Wright, and James B. Rawlings. Application of interior-point methods to model predictive control. *Journal of optimization theory and applications*, 99(3):723–757, 1998.

[92] James B Rawlings. Tutorial overview of model predictive control. *Control Systems, IEEE*, 20(3):38–52, 2000.

[93] James B. Rawlings, David Angeli, and Cuyler N. Bates. Fundamentals of economic model predictive control. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 3851–3861. IEEE, 2012.

[94] James B. Rawlings and Bhavik R. Bakshi. Particle filtering and moving horizon estimation. *Computers & chemical engineering*, 30(10):1529–1541, 2006.

[95] James B. Rawlings and David Q. Mayne. *Model Predictive Control: Theory and Design.* Nob Hill Publishing, 2009.

[96] Douglas G. Robertson, Jay H. Lee, and James B. Rawlings. A moving horizon-based approach for least-squares estimation. *AIChE Journal*, 42(8):2209–2224, August 1996.

[97] B.J.P. Roset, M. Lazar, H. Nijmeijer, and W.P.M.H. Heemels. Stabilizing output feedback nonlinear model predictive control: An extended observer approach. In *Proc. of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 771–781, July 2006.

[98] Eric Schmitz. *Experiments on the End-Point Position Control of a Very Flexible One-Link Manipulator.* PhD thesis, Stanford University, 1985.

[99] P. O. M. Scokaert and D. Q. Mayne. Min-max feedback model predictive control for constrained linear systems. *Automatic Control, IEEE Transactions on*, 43(8):1136–1142, 1998.

[100] Shayle R. Searle. *Matrix Algebra Useful for Statistics.* John Wiley and Sons, 1982.

[101] Eduardo D. Sontag. Control-lyapunov functions. In *Open problems in mathematical systems and control theory*, pages 211–216. Springer, 1999.

[102] Maciej Stachura, Anthony Carfang, and Eric W. Frew. Cooperative target tracking with a communication limited active sensor network. In *International Workshop on Robotic Wireless Sensor Networks*, Marina Del Ray, CA, June 2009.

[103] D. Sui, L. Feng, and M. Hovd. Robust output feedback model predictive control for linear systems via moving horizon estimation. In *American Control Conference, 2008*, pages 453–458. IEEE, 2008.

[104] Tyler H Summers, Maruthi R Akella, and Mark J Mears. Coordinated standoff tracking of moving targets: Control laws and information architectures. *AIAA Journal of Guidance, Control, and Dynamics*, 32(1):56–69, 2009.

[105] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics.* The MIT Press, 2005.

[106] Klaske van Heusden, Eyal Dassau, Howard C. Zisser, Dale E. Seborg, and Francis J. Doyle III. Control-relevant models for glucose control using a priori patient characteristics. *IEEE Transactions on Biomedical Engineering*, 59(7):1839–1849, July 2012.

[107] Lieven Vandenberghe. The CVXOPT linear and quadratic cone program solvers. Technical report, Univ. California, Los Angeles, 2010.

[108] John Walsh and Ruth Roberts. *Pumping Insulin*. Torrey Pines Press, San Diego, CA, USA, 4th edition, 2006.

[109] Yang Wang and Stephen Boyd. Fast model predictive control using online optimization. *Control Systems Technology, IEEE Transactions on*, 18(2):267–278, 2010.

[110] Stephen J. Wright. Applying new optimization algorithms to model predictive control. In *AIChE Symposium Series*, volume 93, pages 147–155, 1997.

[111] Stephen J. Wright. *Primal-dual interior-point methods*, volume 54. Siam, 1997.

[112] Jing Zhang and Jinfeng Liu. Lyapunov-based MPC with robust moving horizon estimation and its triggered implementation. *AIChE Journal*, 59(11):4273–4286, November 2013.