Learning-Based Economic Control for Repetitive Systems

by

Maxwell J. Wu

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in the University of Michigan
2023

Doctoral Committee:

Professor Kira Barton, Chair
Professor Ilya Kolmanovsky
Professor Jing Sun
Associate Professor Chris Vermillion

Maxwell J. Wu

maxwu@umich.edu

ORCID iD: 0000-0001-5631-7195

To my grandmother, Diane Wu.

ACKNOWLEDGMENTS

This dissertation is the collective product of the tremendous support that I have been fortunate to receive over the years. I would first like to thank the University of Michigan and its Department of Mechanical Engineering. It has been the opportunity of a lifetime to study here and call Ann Arbor my home. I would also like to acknowledge the National Science Foundation and the Defense Advanced Research Projects Agency for providing funding support for the various projects that I've had the pleasure of working on during my time in graduate school.

The results put forth in this work are a direct consequence of insights I gained from courses taught by two of my committee members, Prof. Ilya Kolmanovsky and Prof. Jing Sun. Thank you both for your guidance not only in preparing this dissertation and for being outstanding teachers, but also for being incredibly kind and generous with your time and knowledge.

I would also like to extend my gratitude to Prof. Chris Vermillion, who has been a tremendously positive source of guidance during my time as a grad student. Chris has significantly shaped how I think about and approach controls problems, and has provided me with lots of encouragement and opportunities to grow as a researcher. To Chris' students in the CORE Lab that I have had the fortune of working with - Mitchell Cobb, Kirti Mishra, and James Reed - thank you for sharing your controls expertise and research acumen with me. I could not have asked for better collaborators.

To my advisor Prof. Kira Barton, your mentorship and support have made my research journey incredibly fulfilling. Your continual faith and trust in me is deeply appreciated, and I know that I have become a better researcher, engineer, and communicator because of your guidance. Joining your lab after my sophomore year of college is, without a doubt, the best decision I made during my academic career.

A great deal of thanks also goes out to the members of the Barton Research Group, both past and present. I am so very lucky to have been surrounded by such incredibly talented people. I looked forward to coming into the lab every day, and it was you all who made that possible.

Finally, to my friends and family, I don't think I can express my appreciation for you adequately in words and I can't imagine where I would be without you all. In particular, I'd like to thank my parents who have, at every stage of my life, been the model of the type of person that I strive to be.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

FIGURE

LIST OF APPENDICES

LIST OF ACRONYMS

AWE Airborne Wind Energy

EMPC Economic Model Predictive Control

GT Game Theory

ILC Iterative Learning Control

MPC Model Predictive Control

REMPC Robust Economic Model Predictive Control

RAEMPC Robust Adaptive Economic Model Predictive Control

RAEILC Robust Adaptive Economic Iterative Learning Control

RC Repetitive Control

RCI Robust Control Invariant

SQP Sequential Quadratic Programming

SQP-ILC Sequential Quadratic Programming-Based Iterative Learning Control

ABSTRACT

For many engineering applications, the behavior of a system is largely repetitive as it performs a given task many times. Although control strategies have traditionally sought to enable improved behavior of these repetitive dynamic systems through enhanced reference tracking, system performance is often dictated by a non-tracking, or *economic*, performance objective such as the maximization of efficiency or safety, or minimization of energy expenditure or monetary cost. Modern control strategies often consider these economic objectives directly by leveraging tools available from the field of mathematical optimization, but their success in practice is frequently hindered by the presence of uncertainty in the system dynamics. To mitigate the harmful impacts of uncertainty on the quality of decisions made by economic controllers, learning-enhanced control has become a popular field of investigation.

The consideration of repetitive system dynamics within the field of learning-based control is the focus of iterative learning control (ILC) and repetitive control (RC) research efforts. Here, repetition facilitates performance improvements, as information about a system's behavior from previous task executions can be used to inform how to appropriately apply control in the future.

However, despite developments in the fields of ILC and RC, several limitations remain that have prevented a more widespread adoption in practice. Namely, the simultaneous presence of economic performance objectives, nonlinear plant dynamics, and system constraints has not been thoroughly considered in the controller design.

This dissertation presents various methodologies for improving the economic performance of constrained, nonlinear, repetitive systems through learning-based techniques. First, this dissertation establishes a connection between repetitive system behavior and the iterative nature of numerical optimization algorithms. Based on this insight, a controller is designed based on a sequential quadratic programming algorithm wherein sensor measurements obtained from previous trials are used to iteratively improve the system's behavior with regards to economic performance and constraint satisfaction without the requirement of a high-fidelity system model. Conditions for which a control trajectory can be identified that satisfies the constraints of the true system, and a subsequent assessment of the optimality of the resulting converged closed-loop performance are presented.

Moreover, while ILC and RC are designed to mitigate the impacts of modeling errors, closed-loop performance is nonetheless predicated upon the presence of uncertainty. Here, whereas ILC and RC have traditionally facilitated learning at the signal level through direct manipulation of the control input from trial to trial, benefits may be achieved through the additional incorporation of learning at the system level wherein historical data is leveraged to reduce the *amount* of uncertainty that exists.

Consequently, a controller is developed for application to repetitive systems commonly studied within the scope of RC, wherein uncertainty is reduced through the use of a novel adaptive control scheme based on a parametric set membership update law. Specifically, by reducing the impacts of periodic parametric uncertainties on the nominal system dynamics, improvements in economic performance are achieved. Finally, this methodology is then extended to a class of repetitive systems investigated within the ILC literature subject to state-varying parametric uncertainty. Here, the simultaneous use of signal-level and system-level learning is used to enhance economic performance. Conditions are then established for guaranteeing the robust satisfaction of hard state and input constraints. The recursive feasibility and robustly optimal closed-loop performance of these predictive controllers is additionally guaranteed and demonstrated using a set of simulation case studies.

CHAPTER 1

Introduction

1.1 Motivation

While traditional control algorithms have been developed with the aim of improving a system's ability to track a reference signal, in practice, a system's performance is frequently given by a non-tracking, or *economic*, metric. For instance, in manufacturing applications, a system's performance may be measured by its throughput production rate. Meanwhile, in racing applications, the system objective might be to minimize lap time, whereas the performance of electrical power generation systems may be dictated by their ability to maximize energy production. In these cases, the goal of accurate reference tracking does not properly encapsulate the true desired behavior of such systems. However, despite the fact that a broad class of systems aim to achieve high economic performance, reference tracking strategies remain the prevailing technique for most control applications. While careful reference design may be conducted such that these non-tracking objectives are properly addressed in proxy by a reference-tracking controller, this process can be in and of itself an arduous task [1].

Moreover, many classical control tools rely on fundamental assumptions on the system behavior that rarely hold true in practice. For instance, controller design based on frequency-domain transfer function models of a system generally require linearity and time-invariance of the system dynamics. However, in reality, these assumptions only hold, at best, in approximation, and the resulting behavior of the system may not align closely with a linearized estimate of the dynamics. In fact, while in tracking control problems the reference is commonly used as the nominal point/trajectory about which linearization of the system dynamics is performed, in reference-free economic control applications, the identification of an appropriate nominal setpoint/trajectory for linearization is often nontrivial.

Similarly, classical techniques are unable to explicitly address real-world restrictions or constraints that a system may be required to satisfy. These requirements, which may arise as actuation bounds, obstacle avoidance constraints, and velocity/acceleration limits to name a few, can significantly restrict the domain over which the system is operated. With the increasing reliance on embedded control systems for safety-critical applications such as autonomous vehicles, medical devices, and avionic systems, the ability of a controller to properly navigate these constraints is of paramount importance. Hence, the development of advanced control strategies is needed to ensure safe and successful operation of these systems.

The considerations of economic performance objectives, nonlinear system dynamics, and constraints lend themselves to the field of optimal control. Leveraging mathematical optimization tools, optimal control strategies can be employed to address each of these concerns simultaneously [2]. However, the ability of these schemes to identify an optimizing control trajectory relies on the availability of an accurate understanding of the system dynamics. In practice, the existence of disturbances and noise hinder the efficacy of optimal control techniques. Moreover, with the advent of more and more sophisticated engineered technologies, the task of generating system models that are both accurate and simple enough to be directly integrated into a numerical solver becomes increasingly difficult. These unavoidable presences of uncertainty ultimately cause suboptimality of performance or, even worse, can potentially lead to unstable system behavior. Therefore, additional attention is required in order to ensure satisfactory behavior of the system.

To counteract these uncertainties, a variety of control methods may be implemented. Feedback control has been a popular and predominant method for mitigating uncertainty, and can allow the user to achieve high performance in the absence of an accurate system model. However, feedback control is inherently *reactive* rather than *anticipatory*, meaning that undesired behavior must first be observed before a corresponding corrective action takes place. Hence, suboptimality of control derived solely through feedback is all but guaranteed. Alternatively, *robust* optimal control techniques may be employed wherein a certain degree of system performance is achieved by designing an acceptable control signal for a predefined class or set of uncertainties. While useful, these strategies tend to be overly conservative at the expense of performance; protecting the system against uncertainties that do not or are highly unlikely to exist.

Meanwhile, many systems operate *repetitively*. Although some engineered devices are designed for unique, single-use applications, systems are frequently created to complete a limited set of tasks numerous times. For instance, a manufacturing system may machine the same part many times, or a racecar may be tasked with repeatedly traversing along a closed motion path. Alternatively, in the case of electricity generation, energy systems are often operated under periodic environmental conditions or required to satisfy energy consumption demands that fluctuate cyclically. The repetitive nature of these systems offers the opportunity to leverage data and generate an improved understanding of the system behavior during its execution of a task. By learning in this manner, more effective control of the system can be achieved by overcoming the overconservative nature of robust techniques.

The goal of this dissertation is to develop learning-based control strategies for uncertain repetitive systems with economic performance metrics, as depicted in Figure 1.1. Specifically, this work seeks to address fundamental limitations of existing control strategies as it pertains to challenges in modeling, uncertainty mitigation, and robust performance. With the use of measured data, the developed tools and accompanying theory will enable improved control for repetitive systems with various modes of operation. Additionally, the consideration of system properties such as nonlinear dynamics and constraints will enable these strategies to be used in a variety of applications.

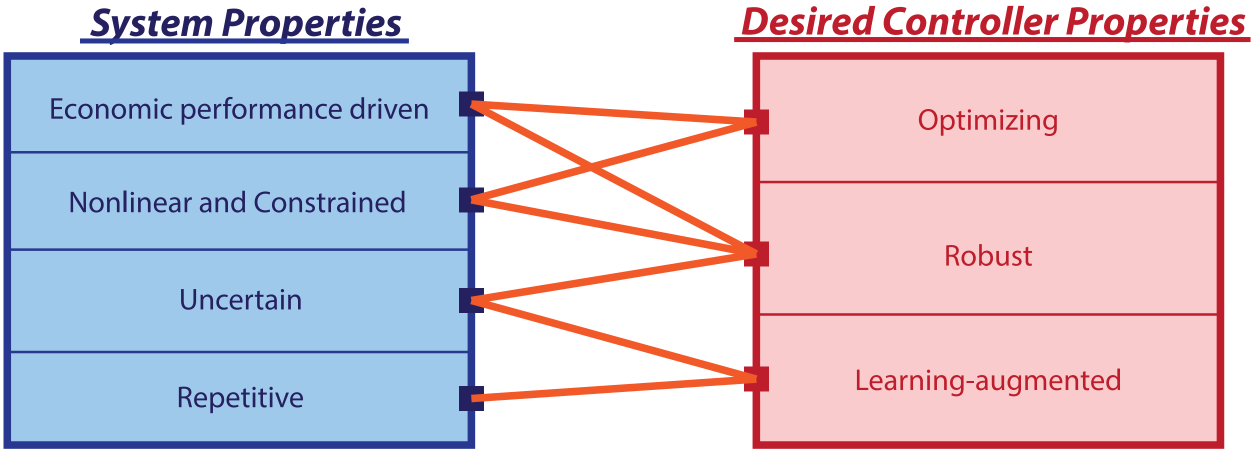
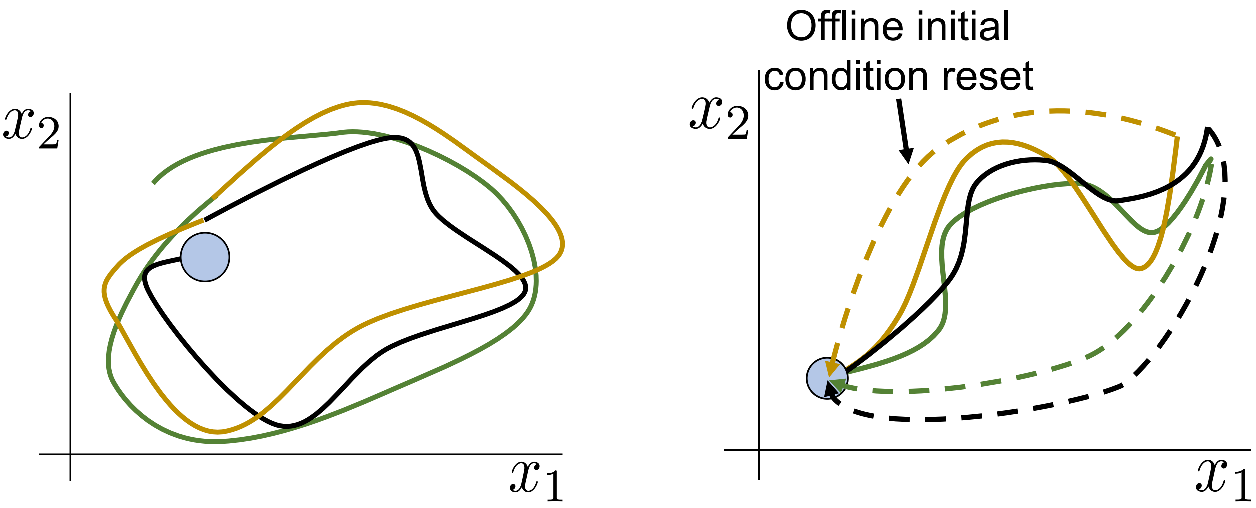

System Properties
Economic performance driven
Nonlinear and Constrained
Uncertain
Repetitive
Desired Controller Properties
Optimizing
Robust
Learning-augmented

Figure 1.1: Properties of the class of systems considered in this dissertation, and the capabilities desired from a controller in order to address or leverage these system properties. The orange lines depict how the desired controller properties arise from the properties of the system.

1.2 Learning Control for Repetitive Systems

Research on learning-based control of repetitive processes has largely focused on two system types: 1) continuously-operated systems, and 2) discontinuously-operated systems. For systems that operate continuously, once the current iteration of the repetitive task concludes, the next iteration immediately begins such that the initial condition of iteration $j + 1$ is equal to the terminal condition of iteration j. As a consequence of this behavior, control decisions made at the current iteration of continuously-operated processes have permeating effects on the system dynamics in future iterations. For example, rotary systems can be considered one type of continuously-operated repetitive system where a revolution of the system constitutes an ‘iteration’. Meanwhile, for systems that perform discontinuous processes, it is assumed that an offline phase exists between iterations of a given task such that the initial condition of the system can be reset. Often in discontinuous processes, the initial condition of the system is assumed to be the same at each iteration (termed *iteration-invariant*), or is set to a user-selected value. As an example, a material handling robot that moves from its initial location to transport a part from one point in a manufacturing line to a new position before returning to rest at its original location to wait for a new part may be classified as a discontinuous process system. The distinctions between continuously- and discontinuously-operated systems are shown in Figure 1.2.



(a) A continuously-operated repetitive system does not have an offline phase between iterations. Consequently, the state trajectory from one iteration transitions immediately to the state trajectory at the subsequent iteration.

(b) A discontinuously-operated repetitive system has an offline phase. Commonly, such systems are reset, as depicted by the dashed lines, such that the initial state condition is the same at each iteration.

Figure 1.2: From an initial state at time 0 (blue circle), the system may traverse through the state space as depicted by the solid lines, where the black, gold, and green line colors distinguish the state trajectories at three different iterations.

Within the realm of learning-based control of repetitive systems, two control strategies come to the forefront: Repetitive Control (RC) [3], and Iterative Learning Control (ILC) [4]. While RC and ILC are similar, there are slight differences in their typical domain of application. Namely, RC has historically been applied to continuously-operated systems while ILC has typically been implemented for control of batch processes [5]. Regardless of this distinction, both RC and ILC have their origins in reference tracking in the presence of repetitive uncertainties.

RC is based upon the internal model principle, which facilitates regulation through an internal representation of a periodic reference or disturbance within the controller design [6]. RC has been successfully used in a myriad of applications including the control of power supply systems [7], computer hard disk drives [8], electrical machine drives [9], and substrate carrier systems [10]. However, while RC has been successfully applied to address the problems of reference tracking and disturbance rejection, it is not directly amenable to economic objectives. Specifically, for systems subject to periodic disturbances with economic objectives, not only does a periodic reference not necessarily exist, but we may be interested in *leveraging* periodic disturbances rather than simply attempting to attenuate their effects on the system behavior. Moreover, the design of RC controllers has traditionally been performed based on frequency-domain models of the system and the reference/disturbance signal [11]. However, controller design based on transfer function analysis fundamentally assumes linearity of the system and time-invariance of uncertainties, which may not be valid for more complex systems. While advancements within the field of repetitive control have allowed for some extensions to nonlinear systems [12–14], the applicability of these methods is often limited to a small class of nonlinearities.

For discontinuous process control, ILC has become a prominent field of study to mitigate the impacts of model uncertainty and repetitive disturbances on system performance. The fundamental intuition underlying ILC is that if a process is repetitive, measurements from previous executions of the task may provide useful information that can be leveraged to improve system performance in future task iterations. Here, by designing feedforward control signals offline based on historical system data, an ILC controller can compensate for performance losses that are a consequence of repetitive uncertainties. Through this paradigm of 'iteration-domain feedback' wherein corrective control actions are defined in the offline phase between task executions, ILC can be used in conjunction with conventional real-time feedback controllers that are designed based on potentially inaccurate plant models. Previous applications wherein ILC has enabled enhanced performance through iteration-domain feedback include the control of industrial robotic manipulators [15], chemical reactors [16], crane systems [17], and satellites [18], among others. Moreover, while a significant portion of existing ILC strategies have been developed using frequency-domain system analysis and controller design, the ILC research community has more readily adopted the use of state-space system representations in comparison to its RC counterpart. Here, although linearity of the system dynamics remains a common assumption within the ILC literature, extensions to nonlinear systems are more easily facilitated. However, existing ILC theory is also primarily focused on the reference tracking problem, rather than the problem of optimizing economic performance. Hence, standard ILC schemes are only applicable to the systems of interest in this dissertation if a higher-level reference shaping controller is also developed, which is a non-trivial task.

To facilitate improvements in system performance through learning, RC and ILC leverage information available from previous iterations of a task in order to update the control signal that is applied in future task executions. However, both RC and ILC traditionally rely on the assumption that the duration of a given repetitive action is iteration-invariant: i.e. that the reference/disturbance signal is periodic with a known period length in the case of RC, or that the trial duration is the same at all iterations in ILC. This assumption can fundamentally restrict the achievable economic performance of the system, particularly in cases where the system performance is dependent upon time or time-varying signals. Moreover, neither RC nor ILC were originally developed with considerations of system constraints, which hinders their applicability to practical systems.

Due to recent advancements in the literature, many of the traditional limitations of RC and ILC schemes noted above have begun to be addressed. For instance, the consideration of a limited class of system constraints is enabled through a variant of ILC termed *point-to-point* ILC [19, 20]. Here, reference tracking is only enforced at select timesteps along a system trajectory, rather than requiring that a reference signal is tracked over the entirety of the trial duration. By relaxing the tracking requirements in this manner, additional flexibility is afforded to the controller to address hard and soft constraints imposed on the system that are linear with respect to the system input. This flexibility is further exploited in other point-to-point ILC control schemes wherein economic objectives dictate system performance. Here, reference tracking is only enforced sparsely, while the consideration of economic performance objectives is embedded directly into the controller design. For instance, sparse reference tracking requirements are defined as hard constraints in [21, 22], while economic performance objectives define the cost function to be minimized. Alternatively, sparse tracking error is used as only a single term in the system cost function in the Pareto optimal point-to-point ILC scheme proposed in [23], while additional economic objectives define the remainder of the system cost. However, while each of these strategies have enabled economic performance objectives to be addressed through iterative learning, they are only directly applicable to a limited set of system classes (e.g. systems with linear dynamics or constraints), and still require that a sparse reference trajectory is known by the user.

1.3 Contributions

The primary contributions of this dissertation are now summarized.

Contribution 1: ***A mapping between numerical optimization methods and constrained economic learning control of repetitive systems -*** Given that the class of systems considered in this work is: i) constrained, and ii) seeks to optimize an economic performance metric, the choice to leverage mathematical tools available from the field of optimization is a natural one. As noted earlier, the primary difficulty with directly applying optimization tools in practice is the existence of uncertainty, which degrades the predicted performance of the system. In spite of this, the following observation is made: the repetitive behavior of the systems of interest, in some sense, mimics the iterative behavior of many popular numerical optimization methods. Specifically, similar to how numerical optimization methods seek optima by updating decision variables based on functional behavior at previous iterates, data from a repetition of a system task may also provide valuable insight into how system control can be improved in future task executions. However, due to the existence of uncertainty, knowledge that is typically utilized within numerical optimization methods, such as gradient or Hessian information, may be unavailable or only partially known. Therefore, to overcome this limitation, the first primary contribution of this dissertation is the development of an economic iterative learning controller for repetitive systems based on the modification of an existing numerical optimization algorithm. Here, systems that operate discontinuously are investigated, and conditions for closed-loop stability and assessments of robust performance of the proposed controller are outlined. The proposed control scheme, termed Sequential Quadratic Programming-Based Iterative Learning Control (SQP-ILC), extends the types of economic performance metrics that may be addressed by ILC schemes in the literature, allowing for new control objectives such as the time-optimal control problem. Additionally, the proposed controller extends the class of constraints that are addressed in the ILC literature, enabling control for repetitive systems whose constraints may be described using smooth functions of the control sequence and time. This contribution is presented in Chapter 2.

Contribution 2: ***Robust adaptive economic control for repetitive systems with periodic uncertainties -*** Although RC has been developed to enable high-performing control of systems by counteracting the effects of unmodeled behavior, this, as with any control technique, has its limitations. Commonly, the ability of repetitive controllers to sufficiently compensate for uncertainties can be hindered if the sources of uncertainty become unwieldy. For instance, as noted in [3], for model-based RC and ILC methods, the robust closed-loop performance of a system can suffer degradation due to modeling errors. In fact, if uncertainty becomes sufficiently large, closed-loop stability may even be lost. Alternatively, if user-available system models are instead more accurate than anticipated, the learning-based controllers can become unnecessarily conservative, resulting in needless performance losses.

Given this correlation between system performance and model confidence, a natural inclination is to augment RC controllers with *adaptive control* strategies that are designed to reduce the amount of model uncertainty. While adaptive control is in many ways similar to RC, an important distinction must first be made. Namely, whereas RC directly modifies the control input, adaptive control methods implement modifications to parameters of the user's model of the plant or controller [24]. In other words, whereas RC incorporates learning by updating a *signal*, adaptive control incorporates learning by updating a *system*.

Hence, the second primary contribution of this work can then be split into three subcomponents:

1. The development of an adaptive scheme for addressing a class of uncertainties commonly addressed within the repetitive control literature. Namely, enabling the robust identification of unknown model parameters for nonlinear systems with uncertainties that appear periodically in time.

2. The integration of this adaptive scheme with a control methodology, termed Robust Adap-

tive Economic Model Predictive Control (RAEMPC), suitable for constrained, continuously operated systems with economic performance objectives.

3. The identification of sufficient conditions for which robust constraint satisfaction and robust convergence of closed-loop economic performance can be guaranteed.

In this way, the second contribution of this dissertation extends the capabilities of repetitive control schemes to a broader range of constrained systems. This contribution is presented in Chapter 3

***Contribution 3:* Robust adaptive economic control for state-dependent uncertainties -** In Contribution 2, a class of systems with unknown model parameters or disturbances that influence that system dynamics in a manner that is periodic in time is addressed. For instance, this methodology may be appropriate if the system is required to operate in an environment where rotating machinery causes unmodeled cyclic vibrations to occur [25], or if alternating current disturbances in powered electronics applications lead to periodic uncertainties in downstream current signals [26]. However, perhaps more commonly, uncertainties are better described as varying as a function of the state of the system, rather than varying periodically as a function of time. For instance, systems operating in spatially varying flow fields [27] or vehicles driving over non-uniform terrain [28] encounter state-dependent disturbances that do not necessarily appear periodically.

Additionally, while the methodology outlined in Chapter 3 is developed for a class of continuously operated systems that are commonly addressed within the RC literature, discontinuous processes offer additional opportunities to exploit the repetitive behavior of the system to improve system performance. In other words, if the system states can be reset between iterations to a known, iteration-invariant value, then this point of similarity in the system behavior between trials enables data from previous task executions to provide useful insight into how the system control can be improved in future iterations. Consequently, the third primary contribution of this dissertation builds directly off of the results of the second contribution by extending these results to systems with: 1) state-dependent parametric uncertainty, and 2) discontinuous process behavior. More specifically, the third primary contribution of this work can be split into the following subcomponents:

1. The development of a control scheme, termed Robust Adaptive Economic Iterative Learning Control (RAEILC), for nonlinear, discontinuously operated systems wherein economic objectives are considered and input/state constraints are robustly enforced when state-varying unknown model parameters and noise exist. Moreover, methods for integrating an outlined class of adaptive methods that are able to robustly identify the state-varying model parameters are developed.

2. Through the direct use of input and state data available from previous task executions, an additional methodology for learning is developed and integrated into the aforementioned

control algorithm that does not rely on the explicit use of a potentially inaccurate system model. Consequently, conservativism commonly exhibited by similar robust strategies is further mitigated.

3. The development of conditions for which recursive feasibility and robust monotonic reductions in the system cost can be guaranteed.

4. The creation of a specific adaptive scheme to robustly identify unknown state-varying model parameters. This adaptive scheme updates user-known uncertainty sets wherein the nominal model parameter estimates, as well as an upper bound on the parameter estimate error are updated. A strategy for integrating this adaptive scheme with the RAEILC algorithm is also developed.

To summarize, the third contribution of this dissertation enables two forms of learning to be applied to the discontinuously operated systems traditionally addressed within the ILC literature. Specifically, through the improved estimation of state-varying model parameters and the direct manipulation of the control signal based on historical data, the economic performance of the system can by improved without violating state and input constraints. This contribution is presented in Chapter 4.

1.3.1 Organization

The remainder of the dissertation is organized as follows. Chapter 2 outlines the numerical optimization-inspired controller described in Contribution 1. For Contribution 2, the adaption algorithm and corresponding controller applicable to systems with periodic parametric uncertainties are presented in Chapter 3 with additional theory presented in Appendix A. As per Contribution 3, the extension of this work to discontinuously operated systems with state-varying model parameters is presented in Chapter 4 with supplementary theoretical results given in Appendix B. A numerical case study application of each of the developed controls schemes is presented in the corresponding chapters. Conclusions drawn from this work are presented in Chapter 5, including an assessment of potential application spaces for the contributions outlined in Chapters 2-4, and future avenues of research to address limitations of this work. A preliminary approach for overcoming some of these limitations is presented in Appendix C, wherein a control strategy based on concepts originating within the field of game theory is outlined.

To prevent against verbose notation and to remain consistent with existing literature, several variables are represented using identical notation across Chapters 2-4 to represent different values or signals. For instance, the use of x in Chapter 2 is used to denote the decision variable of an optimization problem to remain consistent with standard notation in the numerical optimization literature, whereas it is used to denote a system's states in Chapters 3 and 4 to match the standard convention found in the control theory literature. Therefore, the variable notation in each of these chapters should be considered distinct from the notation used in other chapters. To serve as a reference for the reader, a variable notation guide for each chapter is provided in Appendix D.

CHAPTER 2

Economic Iterative Learning Control: Numerical Optimization-Inspired Controller Design

2.1 Background and Motivation

For systems that perform tasks *repetitively*, learning-based controllers have proven to be a particularly valuable tool for improving system performance. In particular, the field of iterative learning control (ILC) has developed powerful strategies for achieving high performance when accurate system model information is limited. While traditionally, ILC has aimed to counteract uncertainties for the purpose of improved reference tracking [4], more recent developments have examined the case of economic performance objectives. In [21–23] for example, ILC strategies are developed for systems where accurate reference tracking is only enforced at a few locations. By leveraging the control freedom afforded through this relaxation of the tracking requirements, system performance with respect to economic performance objectives is more readily addressed. However, these techniques are limited to linear systems and only consider cost functions that are described as functions of the control inputs and outputs. In [29], this idea is expanded upon to address economic objectives that are given as functions of time, as in the time-optimal control problem. However, the strategy outlined in [29] remains limited to linear system applications, and while closed-loop convergence results allow for a monotonic improvement in cost, convergence of the control trajectory relative to an optimizing solution is not demonstrated.

Similar ideas are explored in [30] wherein a path variable parametrization is leveraged to convert a traditional trajectory tracking problem to a path tracking problem. By relaxing the system time requirements in this manner, the control algorithm is able to address a broad class of performance metrics given as functions of the system inputs and outputs. However, this formulation requires the reference path to be defined as a piecewise linear function, and, similar to [21–23], does not allow for the system performance metric to be given as a function of time. Alternative methods, such as those proposed in [31–33], use a learning-based Model Predictive Control (MPC) strategy to address economic performance metrics for uncertain iterative systems. Here, by leveraging measured experimental data, these strategies rely on the construction of control-invariant safe-sets to ensure robust constraint satisfaction. An alternative ILC strategy inspired by economic MPC is developed in [34] wherein data from previous iterations is explicitly incorporated into the constraints of the optimal control problem that is solved to identify the control sequence. However, while non-strict monotonic improvement in performance is demonstrated in [31–34], optimality of the converged control solution is not guaranteed, even in the case where uncertainty does not exist.

Due to the fact that the systems of interest considered here are repetitive, an intriguing strategy for controller design is to mimic mathematical optimization methods that minimize a cost function through the use of iterative updates to a selection of design or decision variables. This idea has been explored, for instance, in [35] where a learning-based control scheme is proposed based on the successive projection algorithm. Alternatively, [36] develops a controller based on the forward-backward splitting algorithm, while a strategy inspired by barrier methods is employed in [37]. However, in [35–37], only tracking objectives and linear systems are considered. An optimization-based ILC strategy is proposed in [38] for nonlinear systems with potentially economic performance objectives. However, robust convergence of the control signal relies on the ability to compute a sufficiently accurate Jacobian of the system dynamics with respect to the input sequence. Consequently, if the user model of the plant is poor, the domain of attraction corresponding to an equilibrium control sequence may be prohibitively small, resulting in a loss of stability.

To address these issues, an ILC strategy is proposed for uncertain, potentially nonlinear, iterative systems with economic performance objectives. This work extends the scheme outlined in [29] by addressing performance metrics given as functions of the input sequence and time, while also providing applicability to systems with soft output constraints. The proposed algorithm, which is inspired by a numerical optimization strategy described in [39], utilizes a filter-based Sequential Quadratic Programming (SQP) scheme wherein the optimal control problem is successively approximated at each iteration as a quadratic program. Here, additional trust-region constraints are imposed as a measure to ensure robustness against uncertainty caused by modelling approximations and errors. The 'success' of a given trial is then measured using a filter that evaluates the trade-off between low economic cost and large constraint violations based on information available from experimental data in comparison to previously completed iterations. By learning in this manner, convergence to a low-cost control trajectory that satisfies system constraints is facilitated.

The contributions of this work are:

1. The development of a learning-based control scheme for iterative systems with input constraints and soft output constraints that is capable of addressing economic performance metrics given as functions of a system's input sequence, as well as time.

2. Analysis of the closed-loop convergence of the input and output trajectory demonstrating convergence to a first-order critical point of the optimal control problem in the case of perfect model knowledge.

3. Guarantees for convergence to feasible input and output trajectories in the presence of model uncertainty, regardless of the initial selection of the control signal. Additionally, the economic performance of the system at convergence is shown to depend upon the accuracy of the user-estimated gradient of the constraint functions.

4. A simulation demonstration of the proposed algorithm where the time-optimal waypoint tracking problem is considered for an actuation constrained two-mass, spring, damper system.

To serve as a reference for the reader, a notation guide for the variables used in this chapter is provided in Appendix D.2. The contents of this chapter have been submitted for publication to IEEE Transactions on Automatic Control as [40].

2.2 System Description

In this work, systems given by the continuous-time dynamics of the form

$$\begin{aligned} \dot{z} &= f_z^c(z, u), \\ y &= f_y(z) \end{aligned} \tag{2.1}$$

are considered where $z \in \mathbb{R}^{n_z}$, $u \in \mathbb{R}^{n_u}$, $y \in \mathbb{R}^{n_y}$ denote the system states, inputs, and outputs respectively. Using a zero-order hold on the inputs, system (2.1) is discretized as

$$\begin{aligned} z_{k+1} &= f_z(z_k, u_k, \tau_k), \\ y_k &= f_y(z_k) \end{aligned} \tag{2.2}$$

where k denotes a timestep index, and τ_k denotes the sample period length between timesteps k and $k+1$. Consequently, in this work the sample period is allowed to vary over time.

Suppose that the economic performance of the system over a given iteration of operation is captured by the cost function

$$\boldsymbol{J} = \sum_{k=0}^{n_\tau - 1} J(u_k, \tau_k)$$

where a given iteration of a task is defined by the system behavior over n_τ timesteps. Additionally, denote the set of feasible inputs and outputs as U and Y. The set of feasible sample periods is similarly denoted as T which is given as

$$T = \{\tau : 0 \leq \tau \leq \tau_{max}\} \tag{2.3}$$

where the zero lower bound ensures that the dynamics given in (2.2) evolve over positive time, and τ_{max} is a user-defined constant used to restrict the total trial duration to be no larger than $\tau_{max} n_\tau$. We assume resettability of the system such that the initial condition of the system is the same at each iteration and is given by the constant z^0.

The optimal control problem we seek to solve is then expressed as

$$\begin{aligned} \underset{\substack{y_{k+1}, u_k, \tau_k, \\ k=0,1,\ldots,n_\tau-1}}{\text{minimize}} \quad & \sum_{k=0}^{n_\tau-1} J(u_k, \tau_k), \\ \text{subject to} \quad & z_{k+1} = f_z(z_k, u_k, \tau_k), \\ & y_k = f_y(z_k), \\ & z_0 = z^0, \\ & u_k \in U, y_k \in Y, \tau_k \in T. \end{aligned} \tag{2.4}$$

To simplify notation, we can equivalently express system (2.2) in the 'lifted' iteration domain through concatenation of the inputs, outputs, and sample periods over an iteration. We define the lifted input, output, and sample period vectors as

$$\begin{aligned} \boldsymbol{u} &= \begin{bmatrix} u_0^\top & u_1^\top & \ldots & u_{n_\tau-1}^\top \end{bmatrix}^\top, \\ \boldsymbol{y} &= \begin{bmatrix} y_1^\top & y_2^\top & \ldots & y_{n_\tau}^\top \end{bmatrix}^\top, \\ \boldsymbol{\tau} &= \begin{bmatrix} \tau_0 & \tau_1 & \ldots & t_{n_\tau-1} \end{bmatrix}^\top. \end{aligned}$$

Here, bold notation is used to distinguish the lifted-domain vectors from their time-domain counterparts. The dynamic system given by (2.2) can then be equivalently expressed as

$$y_k = p^{(k)}(\boldsymbol{u}, \boldsymbol{\tau}, z_0) = f_y\bigg(f_z^{(k)}(\boldsymbol{u}, \boldsymbol{\tau}, z_0)\bigg)$$

where the parenthetical superscript notation $^{(k)}$ is used to denote k recursive calls of f_z as in

$$f_z^{(k)}(\boldsymbol{u}, \boldsymbol{\tau}, z_0) = f_z(f_z(\ldots, u_{k-2}, \tau_{k-2}), u_{k-1}, \tau_{k-1}).$$

The lifted output vector $\boldsymbol{y}$ is then given as a function of $\boldsymbol{u}$ and $\boldsymbol{\tau}$ according to

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{p}(\boldsymbol{u}, \boldsymbol{\tau}, z_0), \\ &= \begin{bmatrix} p^{(1)}(\boldsymbol{u}, \boldsymbol{\tau}, z_0)^\top & \dots & p^{(n_\tau)}(\boldsymbol{u}, \boldsymbol{\tau}, z_0)^\top \end{bmatrix}^\top . \end{aligned}$$

Using the lifted notation, problem (2.4) can be more compactly written as

$$\begin{aligned} \underset{\boldsymbol{y},\boldsymbol{u},\boldsymbol{\tau}}{\text{minimize}} \quad & \boldsymbol{J}(\boldsymbol{u}, \boldsymbol{\tau}), \\ \text{subject to} \quad & \boldsymbol{y} = \boldsymbol{p}(\boldsymbol{u}, \boldsymbol{\tau}, z_0), \\ & z_0 = z^0, \\ & \boldsymbol{u} \in \mathcal{U}, \boldsymbol{y} \in \mathcal{Y}, \boldsymbol{\tau} \in \mathcal{T}. \end{aligned} \tag{2.5}$$

Let $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{u}^\top & \boldsymbol{\tau}^\top \end{bmatrix}^\top \in \mathbb{R}^{n_x}$. $\boldsymbol{x}$ then contains all the information required to construct a given control trajectory, as it contains information about the input sequence as well as the time over which the inputs will be applied. Since $\boldsymbol{y}$ is given by $\boldsymbol{u}$, $\boldsymbol{\tau}$, and the constant z^0, we can then further simplify the optimal control problem given by (2.5) as

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \boldsymbol{J}(\boldsymbol{x}), \\ \text{subject to} \quad & \boldsymbol{c}_E^R(\boldsymbol{x}) = 0, \\ & \boldsymbol{c}_I^R(\boldsymbol{x}) \leq 0 \end{aligned} \tag{2.6}$$

where $\boldsymbol{c}_E^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^R(\boldsymbol{x})$ are vector-valued functions that describe the feasible set of (2.5). However, due to model uncertainty, the constraint functions $\boldsymbol{c}_E^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^R(\boldsymbol{x})$ of the real system may not be entirely known. Rather, estimates, $\boldsymbol{c}_E^M(\boldsymbol{x})$ and $\boldsymbol{c}_I^M(\boldsymbol{x})$, of these functions based on a model of the system dynamics are instead available to the user. The following assumptions are then made.

Assumption 1. *The functions $\boldsymbol{J}(\boldsymbol{x})$, $\boldsymbol{c}_E^R$, $\boldsymbol{c}_I^R$, $\boldsymbol{c}_E^M$, and $\boldsymbol{c}_I^M$ are all twice continuously differentiable.*

Assumption 2. *Given a bounded domain $\mathcal{X}$, there exist constants M^{R_1}, M^{R_2}, M^{M_1}, and M^{M_2} such that the derivatives of the real and user-model based constraint functions $\boldsymbol{c}_I^R$ and $\boldsymbol{c}_I^M$ are bounded according to*

$$\|\nabla_x \boldsymbol{c}_i^R(\boldsymbol{x})\| \leq M^{R_1}, \quad \|\nabla_x \boldsymbol{c}_i^M(\boldsymbol{x})\| \leq M^{M_1}, \quad \|\nabla_x^2 \boldsymbol{c}_i^R(\boldsymbol{x})\| \ \leq M^{R_2}, \quad \|\nabla_x^2 \boldsymbol{c}_i^M(\boldsymbol{x})\| \leq M^{M_2}$$

for all $\boldsymbol{x} \in \mathcal{X}$ and all $i \in \{I, E\}$.

Assumption 2 states that if the control signal is restricted to lie within some bounded domain $\mathcal{X}$, then the size of the first and second derivatives of the constraint functions with respect to the control signals are correspondingly bounded. This assumption may be satisfied if, for instance, the input and output constraint sets U and Y are described by functions whose first and second derivatives are locally Lipschitz continuous over $\mathcal{X}$, and if the first and second derivatives of the dynamics $\boldsymbol{p}$ are also locally Lipschitz continuous over $\mathcal{X}$.

The primary difficulties in solving (2.6) come from the fact that the problem is, in general, non-convex, as well as the lack of absolute knowledge about the true system dynamics that dictate the behavior of the system with regards to economic performance and constraint satisfaction. We therefore seek to derive a control scheme that addresses these issues by leveraging the fact that the system operates iteratively. Namely, by utilizing information about the system behavior from previous iterations of a task, an improved understanding of how to optimally control the system may be achieved.

2.3 Methodology

To address optimal control problem (2.6), a control scheme based on SQP optimization methods is employed. Specifically, a strategy similar to the trust-region SQP-filter algorithm described in [39] is used. The use of this strategy is facilitated by the assumption on the resettability of the initial condition at the beginning of each iteration of the system task, which allows for direct comparison of the system performance and constraint behavior across multiple system trials. However, whereas in [39] it is assumed that $\boldsymbol{c}_E^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^R(\boldsymbol{x})$ are known over the entirety of $\mathbb{R}^{n_x}$, that is no longer the case here due to the existence of model uncertainty. Rather, we assume that the values of $\boldsymbol{c}_E^R(\boldsymbol{x}_j)$ and $\boldsymbol{c}_I^R(\boldsymbol{x}_j)$ are only known based on measurable data observed after conducting experiments using a particular selection of the input and sample period sequences as denoted by $\boldsymbol{x}_j$. In other words, evaluations of the constraint functions cannot be determined over the entirety of $\mathbb{R}^{n_x}$ as $\boldsymbol{c}_E^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^R(\boldsymbol{x})$ are not known in closed form, but are instead limited to the locations within $\mathbb{R}^{n_x}$ for which an experiment has been conducted such that corresponding data is available.

Principally, the algorithm described in [39] solves the non-convex optimization problem of (2.6) by iteratively solving a sequence of simpler convex problems of the form

$$\begin{aligned} \underset{\boldsymbol{s}}{\text{minimize}} \quad & m_j(\boldsymbol{x}_j + \boldsymbol{s}), && \text{(2.7a)} \\ \text{subject to} \quad & \boldsymbol{c}_E^R(\boldsymbol{x}_j) + \boldsymbol{A}_E^R(\boldsymbol{x}_j)\boldsymbol{s} = 0, && \text{(2.7b)} \\ & \boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_I^R(\boldsymbol{x}_j)\boldsymbol{s} \leq 0. && \text{(2.7c)} \end{aligned}$$

Here, at each iteration, j, we aim to identify a candidate step, $\boldsymbol{s}_j$, corresponding to an incremental update to the control signal, as the solution to problem (2.7) from which to move from the current iterate $\boldsymbol{x}_j$. m_j denotes a quadratic approximation of the cost function $\boldsymbol{J}(\boldsymbol{x})$ at $\boldsymbol{x}_j$ given as

$$m_j(\boldsymbol{x}_j + \boldsymbol{s}) = \boldsymbol{J}_j + \langle \boldsymbol{g}_j, \boldsymbol{s} \rangle + \frac{1}{2}\langle \boldsymbol{s}, \boldsymbol{H}_j \boldsymbol{s} \rangle$$

where $\boldsymbol{J}_j = \boldsymbol{J}(\boldsymbol{x}_j)$, $\boldsymbol{g}_j = \nabla_x \boldsymbol{J}(\boldsymbol{x}_j)$, and $\boldsymbol{H}_j = \nabla_x^2 \boldsymbol{J}(\boldsymbol{x}_j)$. $\boldsymbol{A}_E^R(\boldsymbol{x}_j)$ and $\boldsymbol{A}_I^R(\boldsymbol{x}_j)$ in (2.7b) and (2.7c) are the Jacobians of $\boldsymbol{c}_E^R$ and $\boldsymbol{c}_I^R$ evaluated at $\boldsymbol{x}_j$. However, due to the existence of uncertainty, $\boldsymbol{A}_E^R$ and $\boldsymbol{A}_I^R$ are unknown, and thus (2.7) is not able to be solved by the user. Hence, problem (2.7) as given in [39] is modified to an alternative subproblem, denoted as QP($\boldsymbol{x}_j$), which is given as

$$\begin{aligned}
\big(QP\,(\boldsymbol{x}_j)\big): \quad & \underset{\boldsymbol{s}}{\text{minimize}} \quad m_j(\boldsymbol{x}_j + \boldsymbol{s}), && (2.8a)\\
& \text{subject to} \quad \boldsymbol{c}_E^R(\boldsymbol{x}_j) + \boldsymbol{A}_E^M(\boldsymbol{x}_j)\boldsymbol{s} = 0, && (2.8b)\\
& \qquad\qquad\quad \boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_I^M(\boldsymbol{x}_j)\boldsymbol{s} \le 0. && (2.8c)
\end{aligned}$$

QP($\boldsymbol{x}_j$) and (2.7) are similar, however the local constraint approximation of (2.6) is now dependent upon $\boldsymbol{A}_E^M(\boldsymbol{x}_j)$ and $\boldsymbol{A}_I^M(\boldsymbol{x}_j)$, which denote the Jacobians of $\boldsymbol{c}_E^M$ and $\boldsymbol{c}_I^M$ evaluated at $\boldsymbol{x}_j$. Since $\boldsymbol{A}_E^M(\boldsymbol{x}_j)$ and $\boldsymbol{A}_I^M(\boldsymbol{x}_j)$ are obtained from the user-accessible constraint function model, and because $\boldsymbol{c}_E^R(\boldsymbol{x}_j)$ and $\boldsymbol{c}_I^R(\boldsymbol{x}_j)$ are known after conducting an experiment at $\boldsymbol{x}_j$, QP($\boldsymbol{x}_j$) is therefore able to be solved by the user.

However, since (2.8a)-(2.8c) can only be expected to accurately approximate the cost function and constraints of (2.6) in a neighborhood around $\boldsymbol{x}_j$, we wish to restrict the size of the step found by solving (2.8). Hence, (2.8) is reformulated to an alternative problem, TRQP($\boldsymbol{x}_j, \Delta_j$), which is given as

$$\begin{aligned}
\big(TRQP(\boldsymbol{x}_j, \Delta_j)\big): \quad & \underset{\boldsymbol{s}}{\text{minimize}} \quad m_j(\boldsymbol{x}_j + \boldsymbol{s}), && (2.9a)\\
& \text{subject to} \quad \boldsymbol{c}_E^R(\boldsymbol{x}_j) + \boldsymbol{A}_E^M(\boldsymbol{x}_j)\boldsymbol{s} = 0, && (2.9b)\\
& \qquad\qquad\quad \boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_I^M(\boldsymbol{x}_j)\boldsymbol{s} \le 0, && (2.9c)\\
& \qquad\qquad\quad \|s\| \le \Delta_j. && (2.9d)
\end{aligned}$$

Here, (2.9d) is included as an additional constraint that enforces $\boldsymbol{x}_j + \boldsymbol{s}_j$ to lie within a *trust-region* centered at $\boldsymbol{x}_j$ with radius Δ_j. Intuitively, this trust-region constraint ensures that the control signal update will be sufficiently conservative such that overaggressive control actions made as a result of inaccurate cost and constraint approximations are prevented.

Importantly, simply iteratively solving TRQP($\boldsymbol{x}_j, \Delta_j$) and setting $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{s}_j$ before incrementing iteration index j does not ensure convergence of the algorithm. This loss of convergence may occur if Δ_j is chosen to be too large and (2.9a)-(2.9c) do not sufficiently approximate the cost and constraints of (2.6). Instead, careful selection of Δ_j and selective updates to $\boldsymbol{x}_j$ are required.

2.3.1 Decomposition of $\boldsymbol{s}_j$

The solution, $\boldsymbol{s}_j$, to TRQP($\boldsymbol{x}_j, \Delta_j$) may be decomposed as

$$\boldsymbol{s}_j = \boldsymbol{n}_j + \boldsymbol{t}_j. \tag{2.10}$$

Here $\boldsymbol{n}_j$ corresponds to a 'normal step' wherein $\boldsymbol{n}_j$ satisfies the constraints of TRQP($\boldsymbol{x}_j, \Delta_j$) such that

$$\boldsymbol{c}_E^R(\boldsymbol{x}_j) + \boldsymbol{A}_E^M(\boldsymbol{x}_j)\boldsymbol{n}_j = 0, \tag{2.11a}$$

$$\boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_I^M(\boldsymbol{x}_j)\boldsymbol{n}_j \leq 0, \tag{2.11b}$$

$$\|\boldsymbol{n}_j\| \leq \Delta_j. \tag{2.11c}$$

The existence of $\boldsymbol{n}_j$ relies on an assumption that TRQP($\boldsymbol{x}_j, \Delta_j$) has a non-empty feasible set. The case for which this assumption does not hold will be addressed in Section 2.3.2. If the assumption does hold, then $\boldsymbol{n}_j$ can be found according to

$$\boldsymbol{n}_j = P_j(\boldsymbol{x}_j) - \boldsymbol{x}_j$$

where $P_j(\boldsymbol{x}_j)$ is an operator that identifies the orthogonal projection of $\boldsymbol{x}_j$ onto the feasible set of QP($\boldsymbol{x}_j$), and subsequently define

$$\boldsymbol{x}_j^N = \boldsymbol{x}_j + \boldsymbol{n}_j.$$

We then make the assumption that when the maximum violation of the constraints at $\boldsymbol{x}_j$, denoted by $\theta_j = \theta(\boldsymbol{x}_j)$ where

$$\theta(\boldsymbol{x}_j) = \max\left[0, \max_{i \in E} |c_i^R(\boldsymbol{x}_j)|, \max_{i \in I} c_i^R(\boldsymbol{x}_j)\right],$$

is small, that $\boldsymbol{n}_j$ exists and that the size of $\boldsymbol{n}_j$ is also proportionally small. This is formalized in the following assumption.

Assumption 3. *If $\theta_j \leq \delta_n$, then $\boldsymbol{n}_j$ exists and*

$$\|\boldsymbol{n}_j\| \leq \kappa_{usc}\theta_j$$

for some positive constants δ_n and κ_{usc}.

In other words, if the distance between the current iterate and the feasible region of (2.6) is small, as determined by the proxy measurement θ_j, we assume that the linearized feasible set of TRQP($\boldsymbol{x}_j, \Delta_j$) exists and the distance between this set and $\boldsymbol{x}_j$ approximately scales with θ_j. This assumption has previously been leveraged in [41] and [42].

Meanwhile, in (2.10), $\boldsymbol{t}_j$ corresponds to a 'tangent step' that seeks to reduce the value of $m_j(\boldsymbol{x}_j^N + \boldsymbol{t}_j)$ while also maintaining the constraint satisfaction of TRQP($\boldsymbol{x}_j, \Delta_j$) achieved by $\boldsymbol{n}_j$. In order to achieve a significant reduction of the cost function, $\boldsymbol{t}_j$ should not be limited to be too small. In other words, $\boldsymbol{x}_j^N$ should be sufficiently far away from the trust-region boundary such that we have enough design freedom to meaningfully reduce $m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) - m_j(\boldsymbol{x}_j^N)$. To accomplish this, we modify (2.11c) to the more restrictive condition

$$\|\boldsymbol{n}_j\| \leq \kappa_\Delta \Delta_j \text{min}\left[1, \kappa_\mu \Delta_j^{\mu-1}\right] \tag{2.12}$$

where $\kappa_\Delta \in (0,1]$, $\kappa_\mu > 0$, and $\mu \in (0,1)$ are user-defined parameters. If (2.11c) is true, but (2.12) does not hold, the size of $\boldsymbol{t}_j$ would be restricted to be very small in order to ensure that $\boldsymbol{s}_j$ remains feasible for TRQP($\boldsymbol{x}_j, \Delta_j$). Consequently, it is unlikely that a significant decrease of m_j would be able to be achieved. In this case, we do not compute $\boldsymbol{t}_j$ and instead treat TRQP($\boldsymbol{x}_j, \Delta_j$) as if it were an infeasible problem. However, if condition (2.12) is met, then we say that TRQP($\boldsymbol{x}_j, \Delta_j$) is *compatible* and proceed by computing $\boldsymbol{t}_j$, which solves (or approximately solves)

$$\begin{aligned}
\underset{\boldsymbol{t}}{\text{minimize}} \quad & \langle \boldsymbol{g}_j + \boldsymbol{H}_j \boldsymbol{n}_j, \boldsymbol{t} \rangle + \frac{1}{2}\langle \boldsymbol{H}_j \boldsymbol{t}, \boldsymbol{t} \rangle, \\
\text{subject to} \quad & \boldsymbol{A}_E^M(\boldsymbol{x}_j)\boldsymbol{t} = 0, \\
& \boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_E^M(\boldsymbol{x}_j)(\boldsymbol{n}_j + \boldsymbol{t}) \leq 0, \\
& \|\boldsymbol{n}_j + \boldsymbol{t}\| \leq \Delta_j.
\end{aligned} \tag{2.13}$$

Additionally, the metric

$$\chi_j = \left| \begin{array}{rl}
\underset{\boldsymbol{t}}{\min} & \langle \boldsymbol{g}_j + \boldsymbol{H}\boldsymbol{n}_j, \boldsymbol{t} \rangle, \\
\text{s.t.} & \boldsymbol{A}_E^M(\boldsymbol{x}_j)\boldsymbol{t} = 0, \\
& \boldsymbol{c}_I^R(\boldsymbol{x}_j) + \boldsymbol{A}_I^M(\boldsymbol{x}_j)(\boldsymbol{n}_j + \boldsymbol{t}) \leq 0, \\
& \|\boldsymbol{t}_j\| \leq 1.
\end{array} \right| \tag{2.14}$$

is defined as a measure of determining first-order criticality. Importantly, note that if $\boldsymbol{t} = 0$ is a first-order critical point of problem (2.13), then $\chi_j = 0$.

Specifically, we seek to identify a $\boldsymbol{t}_j$ such that the sufficient decrease condition

$$m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j^N + \boldsymbol{t}_j) \geq \kappa_{tmd}\chi_j \text{min}\left[\frac{\chi_j}{\beta_j}, \Delta_j\right] \tag{2.15}$$

is met for some $\kappa_{tmd} > 0$ with $\beta_j = 1 + \|\boldsymbol{H}_j\|$. Numerical strategies for computing a $\boldsymbol{t}_j$ that satisfies (2.15) are discussed in [43].

The case when TRQP($\boldsymbol{x}_j, \Delta_j$) is compatible is shown graphically for the simplified case when $n_x = 2$ with decision variables x^1 and x^2 in Fig. 2.1. Here we note that the feasible region of (2.6) is, in general, both non-convex and unknown. However, by linearizing the constraints using the available user-model of the system according to (2.8b) and (2.8c), the resulting feasible region is known and convex, which greatly simplifies the calculation of a unique normal step $\boldsymbol{n}_j$, and the subsequent tangent step $\boldsymbol{t}_j$.

2.3.2 The Restoration Procedure

In Section 2.3.1, we assumed that TRQP($\boldsymbol{x}_j, \Delta_j$) was compatible. However, this is not always the case, as the feasible set of TRQP($\boldsymbol{x}_j, \Delta_j$) may be empty, or condition (2.12) may not hold. To remedy this situation, we then seek to reduce the value of θ_j such that the condition of Assumption 3 and (2.12) hold.

Specifically, we aim to identify a step $\boldsymbol{r}_j$ such that TRQP($\boldsymbol{x}_j + \boldsymbol{r}_j, \Delta_{j+1}$) is compatible. This process, termed the 'restoration procedure', is performed by approximately solving the problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^{n_x}} \quad \theta(\boldsymbol{x}) \tag{2.16}$$

by identifying a sequence of iterates that converge to $\theta(\boldsymbol{x}) = 0$.

Following an attempted implementation of the restoration procedure, we consider two outcomes. Either: 1) a sequence of iterates is identified such that $\theta(\boldsymbol{x}_j + \boldsymbol{r}_j)$ can be made to be arbitrarily small, or 2) the sequence of iterates does not converge to a point of feasibility for problem (2.6). Both cases will be considered in Section 2.4.

Note: Problem (2.16) can effectively be treated as an ILC reference tracking problem, where the goal is to identify an input $\boldsymbol{u} \in \mathcal{U}$ such that a reference sequence that lies within the set $\mathcal{Y}$ with corresponding sample period sequence $\boldsymbol{\tau} \in \mathcal{T}$ is tracked. A variety of existing strategies, such as the one described in [37], exist to solve such a problem.

Alternatively, if a point $\bar{\boldsymbol{x}}$ is known such that $\bar{\boldsymbol{x}}$ lies in the interior of $\mathcal{U} \times \mathcal{T}$ and $\boldsymbol{y} = \boldsymbol{p}(\bar{\boldsymbol{x}}, z^0) \in \mathcal{Y}$, then performing a line search along the segment connecting $\boldsymbol{x}_j$ and $\bar{\boldsymbol{x}}$ will enable the user to identify a step $\boldsymbol{r}_j$ that results in arbitrarily small values of $\theta(\boldsymbol{x}_j + \boldsymbol{r}_j)$ due to the assumed continuity of $\boldsymbol{c}_E^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^R(\boldsymbol{x})$.

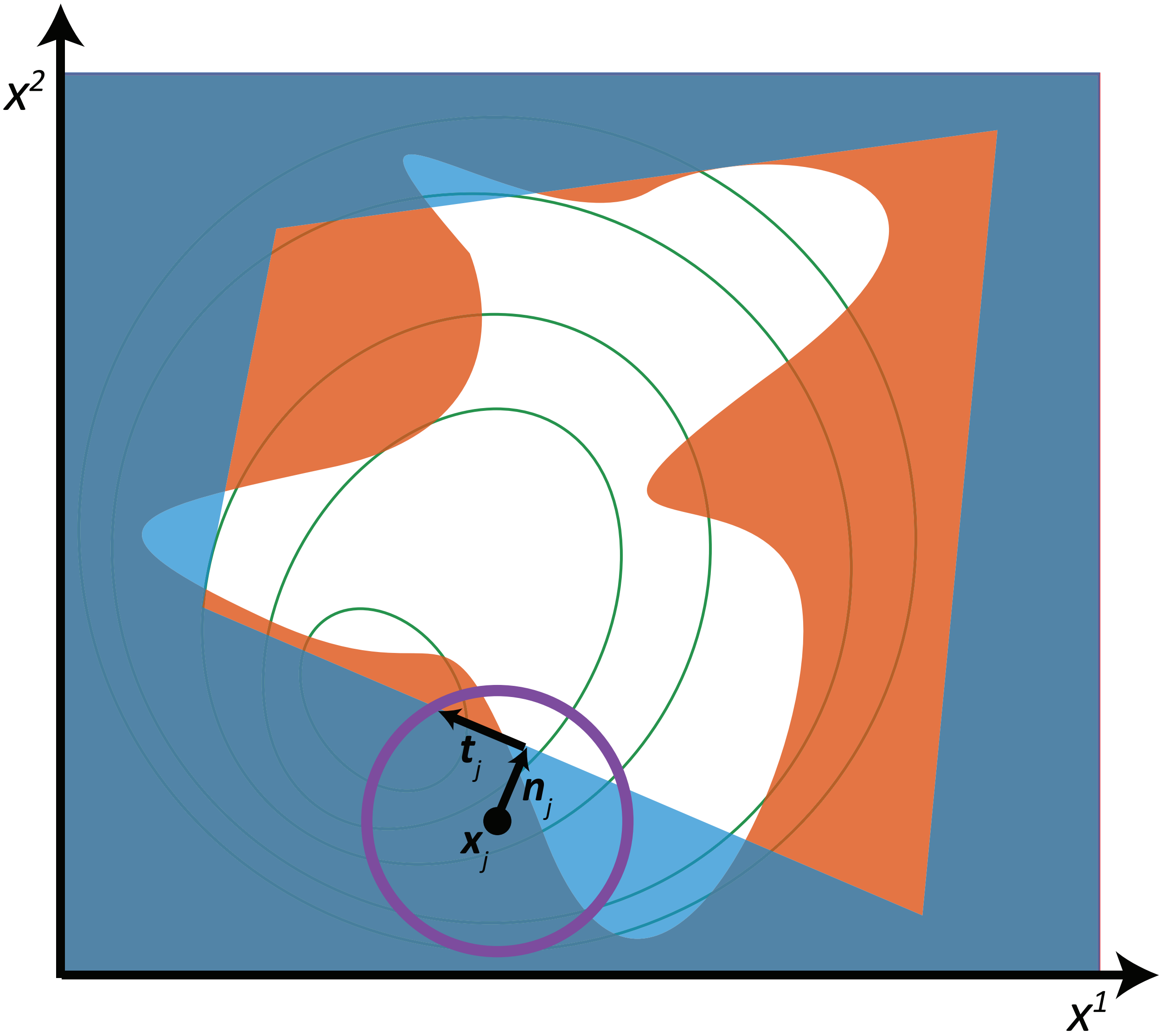


Figure 2.1: A graphical representation of the normal step, $\boldsymbol{n}_j$, and tangent step, $\boldsymbol{t}_j$, when TRQP($\boldsymbol{x}_j, \Delta_j$) is compatible. The green curves denote level sets of $\boldsymbol{J}(\boldsymbol{x})$ for different selections of the decision variables (x^1, x^2), while the orange region represents the infeasible region of the true optimal control problem, (2.6). This true infeasible region is approximated by the blue region, which depicts the infeasible region of QP($\boldsymbol{x}_j$). If $\boldsymbol{x}_j$ is infeasible for QP($\boldsymbol{x}_j$), $\boldsymbol{n}_j$ is found by projection of $\boldsymbol{x}_j$ onto the feasible set of QP($\boldsymbol{x}_j$). $\boldsymbol{t}_j$ then aims to reduce estimated cost function $m_j(\boldsymbol{x})$ while remaining within the trust-region boundary depicted by the purple circle.

2.3.3 The SQP Filter

Once a step $\boldsymbol{s}_j$ or $\boldsymbol{r}_j$ has been identified as described in Sections 2.3.1 and 2.3.2, we seek to identify whether the point $\boldsymbol{x}_j + \boldsymbol{s}_j$ or $\boldsymbol{x}_j + \boldsymbol{r}_j$ is successful. Namely, we wish to evaluate the behavior of this prospective iterate with respect to two potentially competing objectives: 1) low-cost performance and 2) small constraint violation. To accomplish this, we first say that if for two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$,

$$\boldsymbol{J}_i \leq \boldsymbol{J}_k \text{ and } \theta_i \leq \theta_k,$$

$\boldsymbol{x}_i$ *dominates* $\boldsymbol{x}_k$. In this case, input trajectory $\boldsymbol{x}_k$ is of little use from a control perspective, as it is outperformed by $\boldsymbol{x}_i$ in terms of both cost and constraint violation. Hence, we are only interested in iterates in the sequence $\{\boldsymbol{x}_j\}$ that are not dominated by other iterates.

We then introduce the idea of a 'filter', $\mathcal{F}$, which is a list of observed $(\theta_i, \boldsymbol{J}_i)$ corresponding to points $\boldsymbol{x}_i$ such that $\boldsymbol{x}_i$ is not dominated by any other observed iterate. In other words, either

$$\boldsymbol{J}_i < \boldsymbol{J}_k \text{ or } \theta_i < \theta_k \tag{2.17}$$

for each $k \neq i$.

In the event that the pair $(\theta(\boldsymbol{x}_j), \boldsymbol{J}(\boldsymbol{x}_j))$ is being added to the filter, we will frequently refer to this action as *adding* $\boldsymbol{x}_j$ *to the filter*. This choice is made for the sake of brevity. The process of populating the filter is equivalent to the identification of the Pareto frontier, wherein we seek to identify points that provide some optimal trade-off between minimizing cost and reducing infeasibility. This concept is shown graphically in Figure 2.2 wherein the elements of the filter, as depicted by black circles, generate a corresponding filter 'boundary' shown by the solid black line. As new elements are added to the filter that satisfy condition (2.17), the boundary develops and approaches the unknown Pareto frontier shown as the blue line.

To ensure that points added to the filter sufficiently reduce the distance between the filter boundary and the Pareto frontier, we refine requirement (2.17) to a stricter condition. Namely, we state that for a point, $\boldsymbol{x}_i$ to be *acceptable for the filter*, it must satisfy

$$\theta(\boldsymbol{x}_i) \leq (1 - \gamma_\theta)\theta_k \text{ or } \boldsymbol{J}(\boldsymbol{x}_i) \leq \boldsymbol{J}_k - \gamma_\theta \theta_k \tag{2.18}$$

for each $(\theta_k, \boldsymbol{J}_k) \in \mathcal{F}$ where $\gamma_\theta \in (0, 1)$ is a user-selected parameter. Further, we say that $\boldsymbol{x}_i$ is *acceptable for the filter and* $\boldsymbol{x}_j$ if (2.18) holds for all $(\theta_k, \boldsymbol{J}_k) \in \mathcal{F} \cup (\theta_j, \boldsymbol{J}_j)$. Then, the step $\boldsymbol{s}_j$ is only taken if $\boldsymbol{x}_j + \boldsymbol{s}_j$ is observed to be acceptable for the filter and $\boldsymbol{x}_j$. Otherwise, $(\theta(\boldsymbol{x}_j + \boldsymbol{s}_j), \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j))$ would not sufficiently develop the filter boundary towards the Pareto frontier. Hence, (2.18) prevents new points from being added to the filter that are arbitrarily close to the existing filter boundary. Condition (2.18) is depicted in Figure 2.2 wherein any new iterates added to the filter are required to lie below and to the left of the dotted brown line.

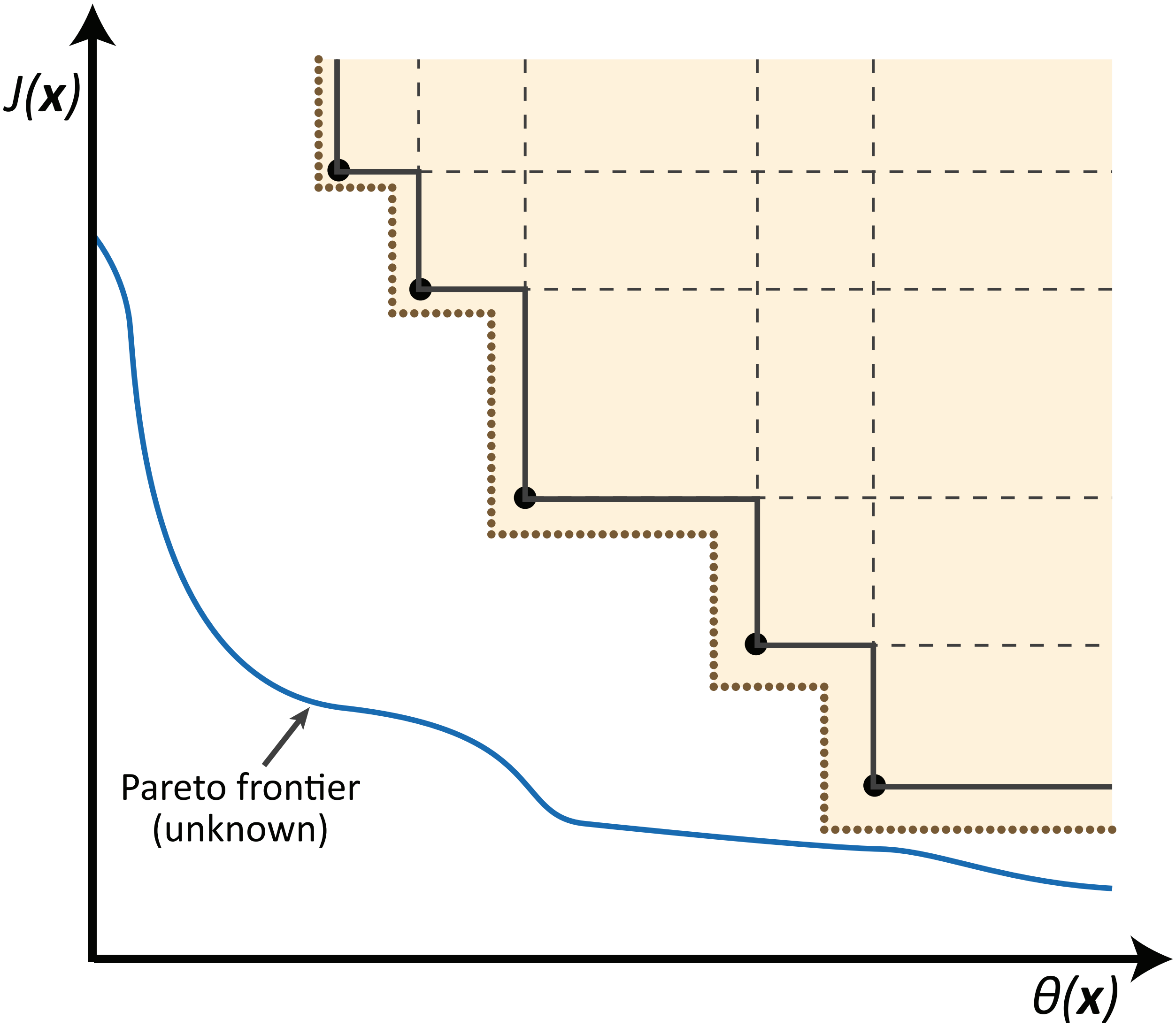


Figure 2.2: The SQP filter with five entries indicated as black circles. The dashed black lines emanating from a given filter entry denote the boundary of the region of the $(\theta, \boldsymbol{J})$ space that is dominated by that iterate. In other words, the iterate dominates all input trajectories corresponding to $(\theta, \boldsymbol{J})$ values above and to the right of the dashed black lines. The solid black line is the filter boundary that is cumulatively established by all of the current entries in the filter. Candidate entries must lie below and to the left of the filter boundary margin indicated by the dotted brown line in order to be acceptable for the filter.

2.3.4 The SQP-ILC Algorithm

The proposed framework, termed the *SQP-ILC algorithm* is now summarized below as Algorithm 1.

Algorithm 1 The SQP-ILC algorithm

Step 0: Initialization. Define an initial point $\boldsymbol{x}_0 = \begin{bmatrix}\boldsymbol{u}_0^\top & \boldsymbol{\tau}_0^\top\end{bmatrix}^\top$, initial trust-region radius $\Delta_0 > 0$, and constants $0 < \gamma_0 < \gamma_1 \leq 1 \leq \gamma_2$, $0 < \eta_1 \leq \eta_2 < 1$, $\gamma_\theta \in (0,1)$, $\kappa_\theta \in (0,1)$, $\kappa_\Delta \in (0,1]$, $\kappa_\mu > 0$, $\mu \in (0,1)$, $\psi > 1/(1+\mu)$, and $\kappa_{tmd} \in (0,1]$. Compute $\boldsymbol{J}_0$ and θ_0. Set $\mathcal{F} = \emptyset$ and $j = 0$.
Step 1: Test for optimality. If $\theta_j = \chi_j = 0$, stop.
Step 2: Ensure compatibility. Attempt to compute normal step $\boldsymbol{n}_j$. If TRQP($\boldsymbol{x}_j, \Delta_j$) is compatible, go to Step 3. Otherwise, include $\boldsymbol{x}_j$ in the filter and compute a restoration step $\boldsymbol{r}_j$ for which TRQP($\boldsymbol{x}_j + \boldsymbol{r}_j, \Delta_{j+1}$) is compatible for some $\Delta_{j+1} > 0$, and $\boldsymbol{x}_j + \boldsymbol{r}_j$ is acceptable for the filter. If this is impossible, stop. Otherwise, define $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{r}_j$ and go to Step 7.
Step 3: Determine a trial step. Compute the tangent step $\boldsymbol{t}_j$ and set $\boldsymbol{s}_j = \boldsymbol{n}_j + \boldsymbol{t}_j$.
Step 4: Test to accept the trial step.

- Run an experiment and evaluate $\theta(\boldsymbol{x}_j + \boldsymbol{s}_j)$ and $\boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j)$.
- If $\boldsymbol{x}_j + \boldsymbol{s}_j$ is not acceptable for the filter and $\boldsymbol{x}_j$, set $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j$, set $\boldsymbol{n}_{j+1} = \boldsymbol{n}_j$, and choose $\Delta_{j+1} \in [\gamma_0\Delta_j, \gamma_1\Delta_j]$. If $\Delta_{j+1} < \left(\frac{1}{\kappa_\mu}\right)^{\frac{1}{\mu-1}}$, set $\Delta_{j+1} = \left(\frac{1}{\kappa_\mu}\right)^{\frac{1}{\mu-1}}$. Increment j by 1, and go to Step 2.
- If
$$m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \geq \kappa_\theta \theta_j^\psi \tag{2.19}$$
and
$$\rho_j = \frac{\boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j)}{m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j)} < \eta_1, \tag{2.20}$$
set $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j$, $\Delta_{j+1} \in [\gamma_0\Delta_j, \gamma_1\Delta_j]$, $\boldsymbol{n}_{j+1} = \boldsymbol{n}_j$, increment j and go to Step 2.

Step 5: Test to include the current iterate in the filter. If
$$\boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) < \kappa_\theta \theta_j^\psi \tag{2.21}$$
include $\boldsymbol{x}_j$ in the filter $\mathcal{F}$.
Step 6: Move to the new iterate. Set $x_{j+1} = x_j + s_j$ and set Δ_{j+1} such that

$$\Delta_{j+1} \in [\Delta_j, \gamma_2\Delta_j] \text{ if (2.21) is false}$$

Step 7: Increment j by one and go to Step 1.

We note the following properties of Algorithm 1:

1. Every iterate $\boldsymbol{x}_j$ is acceptable for the filter at the beginning of each iteration, regardless of whether or not it eventually gets added to the filter. This is because we either: 1) enforce $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{r}_j$ to be 'acceptable for the filter and $\boldsymbol{x}_j$' when identifying $\boldsymbol{r}_j$ during the restoration procedure, or 2) only move to $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{s}_j$ if $\boldsymbol{x}_j + \boldsymbol{s}_j$ is acceptable for the filter.

2. The restoration step is never equal to zero. In other words, we cannot simply increase the size of the trust region from iteration j and iteration $j+1$ in order to make TRQP($\boldsymbol{x}_j + \boldsymbol{r}_j, \Delta_{j+1}$) compatible. This is because we add $\boldsymbol{x}_j$ to the filter before computing $\boldsymbol{r}_j$, and therefore $\boldsymbol{x}_j + \boldsymbol{r}_j$ is not acceptable for the filter if $\boldsymbol{r}_j = 0$.

3. The restoration step is never run in consecutive iterations. This is because we require TRQP($\boldsymbol{x}_j + \boldsymbol{r}_j, \Delta_{j+1}$) to be compatible.

4. No feasible iterate is ever added to the filter. Note that iterates are only added to the filter in Step 2 or Step 5. In Step 2, $\boldsymbol{x}_j$ is added to the filter when a restoration step is required. Restoration is only required when $\boldsymbol{x}_j$ lies too far in the infeasible region, so $\boldsymbol{x}_j$ is not a feasible point in this case. Meanwhile, to proceed to Step 5 from Step 4, we observe that at least one of (2.19) or (2.20) must not hold. Note that if $\boldsymbol{x}_j$ is feasible, that $\boldsymbol{n}_j = 0$, and therefore $\boldsymbol{x}_j + \boldsymbol{s}_j = \boldsymbol{x}_j + \boldsymbol{t}_j$. Consequently, given the sufficient decrease condition in (2.15), we have that $m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \geq 0$, and it follows that (2.19) holds since $\theta_j = 0$ for a feasible $\boldsymbol{x}_j$. Therefore, to continue to Step 5, (2.20) must not hold, which implies that $\boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) > 0$. Proceeding to Step 5, we see again, that if $\boldsymbol{x}_j$ is feasible, then $\theta_j = 0$. Hence, (2.21) does not hold and any feasible $\boldsymbol{x}_j$ is not added to the filter.

2.4 Convergence Analysis

The convergence behavior of Algorithm 1 is now studied. Although the SQP-ILC algorithm has been modified from the strategy in [39] to allow for model uncertainty, we demonstrate that the algorithm remains *globally convergent*. By this, we do not mean that there exists a unique limit point, $\boldsymbol{x}^*$, such that all iterate sequences converge to $\boldsymbol{x}^*$. Rather, for global convergence, we require that for any selection of the starting point $\boldsymbol{x}_0$ within the 'global' design space of $\mathbb{R}^{n_x}$, Algorithm 1 will yield a sequence of iterates that converge to some, potentially non-unique, point $\boldsymbol{x}^*$.

We first define the following sets:

$$\mathcal{S} = \{j : \boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{s}_j\}$$

which represents the indices corresponding to successful iterations wherein $\boldsymbol{x}_j$ is updated based on the calculated step $\boldsymbol{s}_j$, and

$$\mathcal{R} = \{j : \boldsymbol{n}_j \text{ does not satisfy Assumption 3 or } \|\boldsymbol{n}_j\| > \kappa_\Delta \Delta_j \text{min}[1, \kappa_\mu \Delta_j^{\mu-1}]\}$$

denotes the set of indices corresponding to iterations when the restoration procedure is run. We then make the following assumptions.

Assumption 4. *There exists some constant $\kappa_{umh} > 1$ such that $\|\boldsymbol{H}_j\| \leq \kappa_{umh} - 1$ for all j.*

Assumption 5. *The set of iterates $\{\boldsymbol{x}_j\}$ remain in a closed, bounded domain $\mathcal{X} \subset \mathbb{R}^{n_x}$.*

Note that if $\mathcal{U}$ is restricted to be a bounded polytope, T is defined as in (2.3), with $\boldsymbol{u}_0 \in \mathcal{U}$ and $\boldsymbol{t}_0 \in \mathcal{T}$, selection of a restoration scheme that enforces $\boldsymbol{x}_j + \boldsymbol{r}_j$ to lie within $\mathcal{U} \times \mathcal{T}$ will allow Assumption 5 to be given by Assumption 1. This is because $\mathcal{U}$ and $\mathcal{T}$ are known, bounded affine sets that are captured entirely by constraints (2.9b) and (2.9c). Hence, both $\boldsymbol{x}_j + \boldsymbol{s}_j$ and $\boldsymbol{x}_j + \boldsymbol{r}_j$ will lie within $\mathcal{U} \times \mathcal{T}$ for any j, and therefore the sequence $\{\boldsymbol{x}_j\}$ will always remain within $\mathcal{U} \times \mathcal{T}$.

Since a continuous function over a closed, bounded set is bounded, it is immediately observed from Assumptions 1 and 5 that there exist values $\boldsymbol{J}^{min}$ and $\theta^{max} > 0$ such that

$$\boldsymbol{J}^{min} \leq \boldsymbol{J}(\boldsymbol{x}_j) \text{ and } 0 \leq \theta_j \leq \theta^{max}$$

for all j. Therefore, the portion of the $(\theta, \boldsymbol{J})$ space that contains elements of the filter can be restricted to the rectangle

$$\mathcal{A}_0 = [0, \theta^{max}] \times [\boldsymbol{J}^{min}, \infty].$$

We will now present a series of 12 lemmas used to show convergence of the SQP-ILC algorithm. These lemmas and supporting proofs are largely derived from the results of [39]. However, several modifications must be made in order to allow for uncertainty in the constraint functions.

Lemma 1. *Suppose Assumptions 3 and 5 hold and that $\theta_j \leq \delta_n$. Then there exists $\kappa_{lsc} > 0$ such that*

$$\kappa_{lsc}\theta_j \leq \|\boldsymbol{n}_j\| \tag{2.22}$$

Proof. Define

$$\mathcal{V}_j = \{i \in E : \theta_j = |c_i^R(\boldsymbol{x}_j)|\} \bigcup \{i \in I : \theta_j = c_i^R(\boldsymbol{x}_j)\}$$

as the set of constraint subfunction indices corresponding to the most violated constraints of the true system at iteration j. Using this definition of $\mathcal{V}_j$, the proof then follows from [39, Lemma 3.1]. □

Hence, if the amount of constraint violation corresponding to a given iterate is sufficiently small, the length of the normal step can be correspondingly lower bounded based on the value of θ_j.

Lemma 2. *Suppose that finite termination does not occur, i.e., that the restoration procedure is always successful and the condition of Step 1 does not hold. Further, given that Assumptions 1, 3, and 5 hold, and that there exists a subsequence* $\{j_i\} : j_i \notin \mathcal{R}$ *and*

$$\lim_{i \to \infty} \chi_{j_i} = 0 \textit{ and } \lim_{i \to \infty} \theta_{j_i} = 0.$$

Then any arbitrary limit point, $\boldsymbol{x}^*$*, of the subsequence* $\{\boldsymbol{x}_{j_i}\}$ *satisfies*

$$\left| \begin{array}{ll} \underset{\boldsymbol{t}}{\mathit{min}} & \langle \boldsymbol{g}^*, \boldsymbol{t} \rangle, \\ \mathit{s.t.} & \boldsymbol{A}_E^M(\boldsymbol{x}^*)\boldsymbol{t} = 0, \\ & \boldsymbol{c}_I^R(\boldsymbol{x}^*) + \boldsymbol{A}_I^M(\boldsymbol{x}^*)\boldsymbol{t} \leq 0, \\ & \|\boldsymbol{t}\| \leq 1. \end{array} \right| = 0. \tag{2.23}$$

Proof. The proof is given by [39, Lemma 3.2] □

Lemma 2 states that if we can take an infinite subsequence of the iterates such that, at every limit point of this subsequence, the metric χ_{j_i} is equal to zero because either the gradient of the objective function is equal to zero or the tangent step is equal to 0, and that the maximum constraint violation of the limit points is equal to zero, then each of the limit points satisfies (2.23). In other words, if the limit points are stationary points, or a useful tangent step cannot be taken without violating constraints and the maximum observed constraint violation at these limit points is zero, then the limit points satisfy (2.23).

For the following lemma, we first define the set

$$\mathcal{Z} = \{j : x_j \text{ is added to the filter}\}$$

as the set of iterations for which $\boldsymbol{x}_j$ is added to the filter.

Lemma 3. *Suppose that finite termination does not occur. Given Assumptions 1 and 5, and that* $|\mathcal{Z}| = \infty$*, then*

$$\lim_{j \to \infty, j \in \mathcal{Z}} \theta_j = 0.$$

Proof. The proof is given by [39, Lemma 3.3]. □

Lemma 3 states that as long as the restoration procedure is always successful, the maximum constraint violation of the iterates that are added to the filter converges to zero. This is based on the idea that, since θ_j is bounded from Assumptions 1 and 5, and because the condition (2.18) requires that new filter iterates cannot be arbitrarily close to the filter boundary, iterates added to the filter must progress towards the condition $\theta_j = 0$.

While Lemmas 1-3 are effectively equivalent to the statements developed in [39], the remaining lemmas, with the exception of Lemmas 6, 10, and 11, present a departure from the analysis in [39]. These differences are a consequence of the use of $\boldsymbol{A}_E^M$ and $\boldsymbol{A}_I^M$ in place of $\boldsymbol{A}_E^R$ and $\boldsymbol{A}_I^R$ within the optimization problems, the use of the modified compatibility condition (2.12) within Algorithm 1, and the existence of model uncertainty.

Lemma 4. *Suppose finite termination does not occur. Additionally, suppose Assumptions 1 and 2 hold, that $j \notin \mathcal{R}$, and that $\boldsymbol{n}_j$ satisfies (2.22) such that*

$$\kappa_{lsc}\theta_j \leq \|\boldsymbol{n}_j\|.$$

Then

$$\theta_j \leq \kappa_{ubt}\Delta_j^{\mu} \tag{2.24}$$

and

$$\theta(\boldsymbol{x}_j + \boldsymbol{s}_j) \leq \kappa_{ubt}\Delta_j^2 + M^{P_1}\Delta_j \tag{2.25}$$

for some constants $\kappa_{ubt} \geq 0$ and $M^{P_1} \geq 0$.

Proof. The proof is given in Appendix A.1. □

Hence, if the restoration procedure always succeeds, and the length of $\boldsymbol{n}_j$ is lower bounded by θ_j by satisfying Lemma 1, then for iterates where the restoration step is not calculated, θ_j and $\theta(\boldsymbol{x}_j + \boldsymbol{s}_j)$ can be upper bounded based on the size of the trust region radius Δ_j.

We now consider conditions for which a lower bound on the estimated cost reduction caused by taking the step $\boldsymbol{s}_j$ can be placed.

Lemma 5. *Suppose that finite termination does not occur. Given Assumptions 1, 4, and 5,* (2.12), (2.15), *that $j \notin \mathcal{R}$, that*

$$\chi_j \geq \epsilon \tag{2.26}$$

for some $\epsilon > 0$, that

$$\Delta_j \leq \delta_m = \mathit{min} \left[\frac{\epsilon}{\kappa_{umh}}, \left(\frac{2\kappa_{ubg}}{\kappa_{umh}\kappa_\Delta\kappa_\mu} \right)^{\frac{1}{\mu}}, \left(\frac{\kappa_{tmd}\epsilon}{4\kappa_{ubg}\kappa_\Delta\kappa_\mu} \right)^{\frac{1}{\mu-1}} \right] \tag{2.27}$$

where $\kappa_{ubg} = \max_{x \in X} \|\nabla \boldsymbol{J}(x)\|$. Then

$$m_j(x_j) - m_j(x_j + s_j) \geq \frac{1}{2}\kappa_{tmd}\epsilon\Delta_j.$$

Proof. Combining (2.15) and Assumption 4 we have that

$$m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \geq \kappa_{tmd}\chi_j \text{min} \left[\frac{\chi_j}{\kappa_{umh}}, \Delta_j \right].$$

From (2.26) and (2.27) this gives

$$m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \geq \kappa_{tmd}\epsilon\Delta_j.$$

Recall that since $m(\boldsymbol{x})$ is quadratic, $m(\boldsymbol{x}_j^N)$ can be expressed as

$$m_j(\boldsymbol{x}_j^N) = m_j(\boldsymbol{x}_j) + \langle \boldsymbol{g}_j, \boldsymbol{n}_j \rangle + \frac{1}{2}\langle \boldsymbol{n}_j, \boldsymbol{H}_j \boldsymbol{n}_j \rangle.$$

The Cauchy-Schwarz inequality and definitions of κ_{ubg} and κ_{umh} give

$$|m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j^N)| \leq \kappa_{ubg}\|\boldsymbol{n}_j\| + \frac{1}{2}\kappa_{umh}\|\boldsymbol{n}_j\|^2. \tag{2.28}$$

Compatibility condition (2.12), Assumption 4, and (2.27) then give

$$|m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j^N)| \leq \frac{1}{2}\kappa_{tmd}\epsilon\Delta_j.$$

□

Lemma 6. *Suppose that finite termination does not occur. Given Assumptions 1, 4, and 5,* (2.15), (2.26)*, that $j \notin \mathcal{R}$, and that*

$$\Delta_j \leq \delta_\rho = \mathit{min} \left[\delta_m, \frac{(1-\eta_2)\kappa_{tmd}\epsilon}{2\kappa_{ubh}} \right]$$

where $\kappa_{ubh} > 1$ is a constant such that for all j,

$$|\boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j)| \leq \kappa_{ubh}\Delta_j^2,$$

which is known to exist as demonstrated in [43]. Then

$$\rho_j \geq \eta_2.$$

Proof. The proof is given in [39, Lemma 3.6]. □

We now establish conditions for which a lower bound on the estimated improvement in performance can be made based on the value of θ_j.

Lemma 7. *Suppose finite termination does not occur. Given Assumptions 1, 4, and 5,* (2.12), (2.15), (2.22), (2.26), *$j \notin \mathcal{R}$, and that*

$$\Delta_j \leq \delta_f = min\left[\delta_m, \left(\frac{\kappa_{tmd}\epsilon}{2\kappa_\theta \kappa_{ubt}^{\psi}}\right)^{\frac{1}{\psi\mu - 1}}\right], \tag{2.29}$$

then

$$m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \geq \kappa_\theta \theta_j^{\psi}.$$

Proof. From Lemma 4 and (2.29) we have

$$\kappa_\theta \theta_j^{\psi} \leq \kappa_\theta \kappa_{ubt}^{\psi} \Delta_j^{\psi\mu} \leq \frac{1}{2}\kappa_{tmd}\epsilon\Delta_j.$$

Using Lemma 5 gives us the desired conclusion

$$\kappa_\theta \theta_j^{\psi} \leq m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j).$$

□

Lemma 8. *Suppose that finite termination does not occur. Suppose also that Assumptions 1, 2, 4, and 5, (2.15), (2.26), and* (2.27) *hold, that $j \notin \mathcal{R}$, that n_j satisfies (2.22), and that*

$$\theta_j \leq \delta_\theta = \kappa_{ubt}^{-\frac{1}{\mu-1}}\left(\frac{\eta_2 \kappa_{tmd}\epsilon}{2\gamma_\theta}\right)^{\frac{\mu}{\mu-1}}. \tag{2.30}$$

Then

$$\boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) \le \boldsymbol{J}(\boldsymbol{x}_j) - \gamma_\theta \theta_j.$$

Proof. From Lemmas 4-6 and (2.30), we have that

$$\begin{aligned}
\boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) &\ge \eta_2 \left[m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j) \right], \\
&\ge \frac{1}{2} \eta_2 \kappa_{tmd} \epsilon \Delta_j, \\
&\ge \frac{\eta_2 \kappa_{tmd} \epsilon}{2} \left(\frac{\theta_j}{\kappa_{ubt}} \right)^{\frac{1}{\mu}}, \\
&\ge \gamma_\theta \theta_j.
\end{aligned}$$

□

Lemma 8 demonstrates under what conditions it can be assured that $\boldsymbol{x}_j + \boldsymbol{s}_j$ is not dominated by the point $\boldsymbol{x}_j$. Note that this does not necessarily mean that $\boldsymbol{x}_j + \boldsymbol{s}_j$ is acceptable for the filter, as there may be another point within $\mathcal{F}$ for which (2.18) does not hold.

We now establish under what conditions we can ensure that TRQP($\boldsymbol{x}_j, \Delta_j$) is compatible such that a restoration step will not need to be applied at iterate j. Before proceeding, we introduce the following assumption.

Assumption 6. *κ_μ, μ, and κ_Δ are selected such that*

$$\left(\frac{1}{\kappa_\mu} \right)^{\frac{1}{\mu - 1}} < \delta_{\mathcal{R}}$$

where $\delta_{\mathcal{R}}$ is the smallest number greater than zero in the set

$$\boldsymbol{\delta_{\mathcal{R}}} = \{ \delta_{\mathcal{R}} : \delta_{\mathcal{R}} = \gamma_0 \delta_m \textit{ or } \frac{\kappa_{usc}}{1 - \gamma_\theta} \left(\kappa_{ubt} \frac{\delta_{\mathcal{R}}^2}{\gamma_0^2} + M^{P_1} \frac{\delta_{\mathcal{R}}}{\gamma_0} \right) - \kappa_\Delta \kappa_\mu \delta_{\mathcal{R}}^\mu = 0 \}.$$

Lemma 9. *Suppose that finite termination does not occur. Additionally, suppose that Assumptions 1-6, and* (2.26) *hold, that* (2.15) *holds for $j \notin \mathcal{R}$, and that*

$$\Delta_j = \left(\frac{1}{\kappa_\mu} \right)^{\frac{1}{\mu - 1}}. \tag{2.31}$$

Suppose further that $j > 0$ and that

$$\theta_j \le \mathit{min} \left[\delta_\theta, \delta_n \right].$$

Then $j \notin \mathcal{R}$.

Proof. The proof is given in Appendix A.2 □

Lemma 10. *Suppose finite termination does not occur. Suppose also that Assumptions 1-6 hold, and* (2.15) *holds for $j \notin \mathcal{R}$. Additionally, suppose that $|\mathcal{Z}| = \infty$. Then there exists a subsequence $\{j_i\} \subseteq \mathcal{Z}$ such that*

$$\lim_{i\to\infty} \theta_{j_i} = 0$$

and

$$\lim_{i\to\infty} \chi_{j_i} = 0.$$

Proof. The proof is given by [39, Lemma 3.10]. □

Hence, if an infinite sequence of iterates are added to the filter, then a subsequence of these iterates converges to a point where the maximum constraint violation is zero and the metric χ_{j_i} is equal to zero because either the gradient of the objective function or the tangent step is equal to 0 at the point of convergence.

Now that we have considered the case when $|\mathcal{Z}| = \infty$, we will now examine the alternative when a finite number of elements are added to the filter such that $|\mathcal{Z}| < \infty$. If this is the case, let j_0 denote the iteration such that x_{j_0-1} is the last iterate added to the filter.

Lemma 11. *Suppose finite termination does not occur and that $|\mathcal{Z}| < \infty$. Suppose that Assumptions 1-6 and* (2.15) *hold for $j \notin \mathcal{R}$. Then*

$$\lim_{j\to\infty} \theta_j = 0.$$

Additionally, n_j satisfies (2.22) for all $j \geq j_0$ sufficiently large.

Proof. The proof is given by [39, Lemma 3.11] and reproduced here for reference.

Consider an successful iterate with $j \geq j_0$ such that

$$\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{s}_j.$$

Since $j \geq j_0$, this means that $\boldsymbol{x}_j$ is not in the filter. Hence $j \notin \mathcal{R}$ and therefore $\rho_j \geq \eta_1$ such that

$$\begin{aligned} \boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_j + \boldsymbol{s}_j) &\geq \eta_1[m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j)], \\ &\geq \eta_1 \kappa_\theta \theta_j^\psi \geq 0. \end{aligned} \tag{2.32}$$

Thus, the objective function is decreasing for all successful iterations with $j \geq j_0$. But since Assumptions 1 and 5 imply that $\boldsymbol{J}(\boldsymbol{x}_j) \geq \boldsymbol{J}^{min}$ for all j, this means that

$$\lim_{\substack{j \in \mathcal{S} \\ j \to \infty}} [\boldsymbol{J}(\boldsymbol{x}_j) - \boldsymbol{J}(\boldsymbol{x}_{j+1})] = 0. \tag{2.33}$$

Combining (2.32) with (2.33) gives the first claim of the proof and Assumption 3 and Lemma 1 give the second claim of the proof. □

Lemma 12. *Suppose that finite termination does not occur, and that $|\mathcal{Z}| < \infty$. Additionally, suppose that Assumptions 1-6 hold, and that (2.15) holds for $j \notin \mathcal{R}$. Then*

$$\lim_{j \to \infty} \chi_j = 0.$$

Proof. Lemma 11 along with Assumption 3 gives for j sufficiently large that

$$\|n_j\| \leq \kappa_{usc} \theta_j$$

must hold. Additionally, from Lemma 11, we have that (2.32) holds and therefore that (2.33) holds for all $j \in \mathcal{S}, j \geq j_0$.

For the purpose of obtaining a contradiction, suppose that (2.26) holds.

Combining (2.28), Lemma 11, and Assumption 3 together give that

$$\lim_{j \to \infty} [m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j^N)] = 0. \tag{2.34}$$

Recall that we can decompose the estimated change in cost between iterations as

$$m_j(\boldsymbol{x}_j) - m(\boldsymbol{x}_j + \boldsymbol{s}_j) = m_j(\boldsymbol{x}_j) - m_j(\boldsymbol{x}_j^N) + m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j).$$

Along with (2.32), (2.33), and (2.34), we then have

$$\lim_{\substack{j \in \mathcal{S} \\ j \to \infty}} [m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j + \boldsymbol{s}_j)] = 0.$$

Eq. (2.15) gives

$$m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j^N + t_j) \geq \kappa_{tmd} \chi_j \min \left[\frac{\chi_j}{\beta_j}, \Delta_j \right].$$

Combining with the fact that, by nature of the algorithm, $\Delta_j \geq \left(\frac{1}{\kappa_\mu}\right)^{\frac{1}{\mu - 1}} > 0$ for all j, Assumption

4, and the assumption that (2.26) holds such that $\chi_j \geq \epsilon$ for all j gives

$$m_j(\boldsymbol{x}_j^N) - m_j(\boldsymbol{x}_j^N + \boldsymbol{t}_j) \geq \kappa_{tmd}\epsilon \min\left[\frac{\epsilon}{\kappa_{umh}}, \left(\frac{1}{\kappa_\mu}\right)^{\frac{1}{\mu-1}}\right].$$

However, this contradicts (2.34). Hence, the assumption that (2.26) holds must not be true. Therefore

$$\lim_{j\to\infty} \chi_j = 0.$$

□

Hence, for the case when the algorithm adds only a finite number of iterates to the filter, the metric χ_j converges to zero. We now show the primary claim of this chapter.

Theorem 1. *Suppose that finite termination does not occur. Suppose that Assumptions 1-6 and that* (2.15) *holds for $j \notin \mathcal{R}$. Let $\{x_{j_i}\}$ be the sequence of iterates produced by the algorithm. Then either the restoration procedure terminates unsuccessfully, or there is a subsequence $\{j_i\}$ for which*

$$\lim_{i\to\infty} \boldsymbol{x}_{j_i} = \boldsymbol{x}^*$$

and $\boldsymbol{x}^*$ satisfies

$$\theta(\boldsymbol{x}^*) = 0 \text{ and } \left| \begin{array}{ll} \underset{\boldsymbol{t}}{\textit{min}} & \langle \boldsymbol{g}^*, \boldsymbol{t} \rangle, \\ \textit{s.t.} & \boldsymbol{A}_E^M(\boldsymbol{x}^*)\boldsymbol{t} = 0, \\ & \boldsymbol{c}_I^R(\boldsymbol{x}^*) + \boldsymbol{A}_I^M(\boldsymbol{x}^*)\boldsymbol{t} \leq 0, \\ & \|\boldsymbol{t}\| \leq 1. \end{array} \right| = 0. \tag{2.35}$$

Proof. The claim of this theorem comes from combining Assumption 5 with Lemmas 10, 11, and 12 to show that as long as the restoration procedure is always successful

$$\lim_{j\to\infty} \theta_j = 0 \text{ and } \lim_{j\to\infty} \chi_j = 0.$$

The proof's claim is then given by applying Lemma 2. □

Theorem 1 accepts the possibility that the restoration procedure terminates unsuccessfully. The selection of a universal algorithm to implement the restoration procedure that eliminates this possibility remains an open question. As discussed in Section 2.3.2, constrained ILC reference-tracking strategies are good candidates for the restoration procedure due to their ability to compensate for uncertainties, particularly if a reference signal within $\mathcal{Y}$ with a corresponding input within $\mathcal{U}$ is known. However, the exact ILC controller type will likely vary on a case-by-case basis.

If the restoration procedure always terminates successfully, Theorem 1 states that any of the limit points of the sequence $\{\boldsymbol{x}_j\}$ will satisfy (2.23). This claim says that at the converged points, the system performance may not be improved without, according to the system model, violating constraints. If the model is accurate, this would imply that the algorithm is able to locate a Karush–Kuhn–Tucker point of problem (2.6). Further, if the model is inaccurate, convergence to the feasible region of (2.6) is still achieved. This is contrary to the result that would be obtained by using a non-learning optimal control strategy, where the theorem would only be able to claim (2.35) if $c_i^R(\boldsymbol{x}^*)$ was replaced with $c_i^M(\boldsymbol{x}^*)$. Then, if the model is inaccurate and therefore $c_i^R(\boldsymbol{x}^*)$ and $c_i^M(\boldsymbol{x}^*)$ do not agree, convergence to a feasible point of (2.6) cannot be guaranteed. Additionally, in the event that $c_i^M(\boldsymbol{x})$ is more restrictive than $c_i^R(\boldsymbol{x})$, as is the case for non-learning robust control strategies, a degradation in performance is likely to be observed. Hence, the SQP-ILC algorithm is able to provide the benefit of robust constraint handling, while also providing improved performance guarantees.

2.5 Simulation Example

The SQP-ILC framework is simulated on the multiple mass-spring-damper system shown in Fig. 2.3. The input is a force applied to one of the masses, and the output is the position of that mass.

The continuous time dynamics are given by

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t), \\ y(t) &= Cz(t). \end{aligned}$$

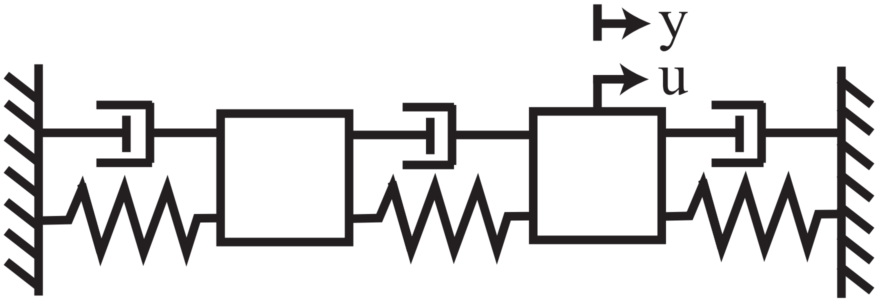


Figure 2.3: Mass-spring-damper system used for simulation.

where

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1.25 & -1.25 & 0.25 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 4 & -2 & -8 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2 \end{bmatrix}, \\
C &= \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}.
\end{aligned} \tag{2.36}
$$

However, suppose that the user only has access to an inaccurate model of the system with A and B given by

$$
A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & -2 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & -2 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{2.37}
$$

Consequently, the discrete time dynamics are given by

$$
\begin{aligned}
f_z^d(z_k, u_k, \tau_k) &= e^{A\tau_k} z_k + \int_0^{\tau_k} e^{A\sigma} d\sigma B u_k, \\
f_y(z_k) &= C z_k.
\end{aligned} \tag{2.38}
$$

Here, while the system is linear with respect to the system states and input, it is nonlinear with respect to the sample period selection.

For this case study, the time-optimal waypoint tracking problem is investigated wherein the objective is to move one of the masses from an initial position to two subsequent locations within a predefined tolerance in as little time as possible.

More formally, the performance objective is given as $\boldsymbol{J}(\boldsymbol{u}, \boldsymbol{\tau}) = \boldsymbol{\tau}^\top \boldsymbol{\tau}$ where an iteration consists of $n_\tau = 80$ timesteps. As the system dynamics and performance objective are twice continuously differentiable with respect to (u_k, τ_k), Assumption 1 holds. Moreover, from this definition of the performace objective, Assumption 4 holds for a value of $\kappa_{umh} = 3$.

The waypoint tracking constraints are defined as

$$
|y_{41} - 1| \leq 0.2, \quad |y_{81} - 0| \leq 0.2 \tag{2.39}
$$

which is to say that the output should be equal to 1 and 0 at the 41^{st} and 81^{st} samples respectively within a tracking tolerance of 0.2.

Table 2.1: Parameters for the SQP-ILC simulation

Parameter	Value
γ_0	0.1
γ_1	0.5
γ_2	2
η_1	0.01
η_2	0.9
γ_θ	10^{-4}
κ_Δ	0.7
κ_μ	5
μ	0.01
κ_θ	10^{-4}
κ_{tmd}	0.01
ψ	2
Δ_0	6

The sets U and T are additionally given as

$$U = \{u_k : -40 \leq u_k \leq 40, \forall k \in 0, \ldots, 80\}, \quad T = \{\tau_k : 0 \leq \tau_k \leq 1, \forall k \in 0, \ldots, 80\}. \quad (2.40)$$

Since the sets U and T are compact polytopes, then, as noted in Section 2.4, the iterates $\boldsymbol{x}_j$ remain within a compact set $\mathcal{X}$ such that Assumption 5 holds. Moreover, the first and second derivatives of the discrete time dynamics given by (2.38) are Lipschitz continuous over $\mathcal{X}$, which gives Assumption 2.

Constraints (2.39) and (2.40) are used to construct the functions $\boldsymbol{c}_I^R(\boldsymbol{x})$ and $\boldsymbol{c}_I^M(\boldsymbol{x})$ from the relation $y_{k+1} - f_y(f_z^d(z_k, u_k, \tau_k))$ using the state space representations given by (2.36) and (2.37) respectively. Recall that $\boldsymbol{c}_I^R(\boldsymbol{x})$ is unknown to the user, but is measurable during experiments for a specific selection of $\boldsymbol{x}$.

The algorithm was run for 100 iterations. For the restoration procedure, a line search strategy was used by conducting a series of experiments along the segment connecting $\boldsymbol{x}_j$ and $\bar{\boldsymbol{x}}$ where $\bar{\boldsymbol{x}}$ is a point in the design space that lies in the interior of the feasible set $\mathcal{U} \times \mathcal{T}$ such that $\boldsymbol{y} = \boldsymbol{p}(\bar{\boldsymbol{x}}, z^0) \in \mathcal{Y}$. $\boldsymbol{r}$ is then found by identifying a point along this segment for which $\theta(\boldsymbol{x}_j + \boldsymbol{r}_j)$ is satisfactorily small. Optimization problems were solved using CVX, a MATLAB package for convex optimization [44, 45]. The algorithm parameters for the simulation are given in Table 2.1. The initial input sequence is $\boldsymbol{u}_0 = 0$ and the initial sample period vector is $\boldsymbol{\tau} = \begin{bmatrix} \frac{1}{80} & \cdots & \frac{1}{80} \end{bmatrix}^\top$. Each trial has an initial condition of $z^0 = 0$.

The evolution of the trial duration from iteration to iteration is shown in Fig. 2.4. Here, the color of the circles denote whether the corresponding iterate $\boldsymbol{x}_j$ satisfied the constraints within ξ-*precision* such that $\theta(\boldsymbol{x}_j) < \xi$ with $\xi = 0.01$. From an initial value of $1s$, we see that the trial duration is quickly reduced to a value of $0.15s$ by the 20^{th} iteration. However, at this iterate, the waypoint tracking constraints are not satisfied. Consequently, a restoration procedure was conducted to identify a point with reduced infeasibility. By allowing the algorithm to continue to run, the trial duration converges to a value of $0.36s$ while also satisfying the constraints within ξ-precision. Hence, the economic cost of trial duration is able to be significantly reduced from the nominal value of $1s$ while also ensuring that constraints are satisfied.

The converged output trajectory is shown in Fig. 2.5. Here the output is still able to track the waypoints within the specified tolerances in spite of the reduction in trial duration. By allowing the sample periods to vary by treating $\boldsymbol{\tau}$ as a design variable, the time at which the waypoints are required to be tracked as enforced by constraint (2.39) is flexible. Hence, the algorithm is able to identify an updated sample period sequence that reduces the economic cost of trial duration.

Similarly, the converged input trajectory is shown in Fig. 2.6 by the dashed yellow line. Here, the converged input remains within the saturation limits at all times. We further observe that the converged control trajectory is very similar to the unknown optimal control signal, $\boldsymbol{u}^{opt}$, shown by the solid blue line. Not only does $\boldsymbol{u}_{100}$ nearly mimic the bang-bang signal of $\boldsymbol{u}^{opt}$, but the durations of the two signals are different by only 1.6%. In other words, the SQP-ILC algorithm is able to very nearly identify the time-optimal control signal, in spite of the fact that an inaccurate system model is used for the control signal identification. Hence, by incorporating learning, the algorithm is largely able to overcome the existence of uncertainty and uncover a nearly globally minimizing control solution.

As observed in Fig. 2.4, large constraint violations occurred at several iterations prior to convergence. While this behavior is permissible for the simulation example described in this section, it may be inadmissible for practical systems with safety-critical constraints. A potential strategy to mitigate this phenomenon is to dynamically adjust the trust region radius Δ_j to shrink as θ_j increases. In this manner, large updates to the control signal (and therefore potentially large increases to the constraint violation) can be prevented when the system constraints are already being violated. This idea is successfully leveraged in [46] to enforce robust obstacle avoidance of a robotic manipulator, but the implications of using this strategy on the converged performance of the system remains a point of further investigation.

2.6 Conclusions

This work presents an ILC strategy for constrained systems with time and input-based economic performance metrics. Namely, by treating the system timing as a decision variable, greater flexi-

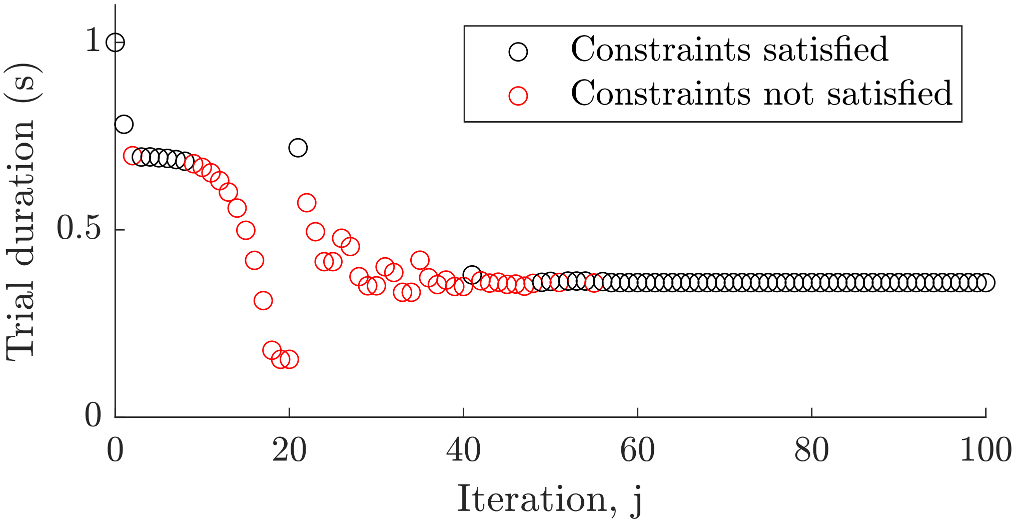


Figure 2.4: Trial duration corresponding to each iterate, $\boldsymbol{x}_j$. Black circles denote iterations where the constraints are satisfied within ξ-precision, while red circles correspond to iterations wherein constraints were violated beyond ξ-precision with $\xi = 0.01$.

bility is afforded in how the control signal is designed, to allow for performance metrics beyond reference tracking to be addressed. This framework extends the capabilities of prior work given in [29] by allowing for a much broader class of constraints. It is shown, in the case that the model is accurate, that the algorithm identifies, at convergence, a first-order critical point of the optimal control problem. Additionally, in the presence of model uncertainty, conditions are established for which the closed-loop control trajectory converges to a solution such that the constraints are satisfied, while also outlining properties of the converged control trajectory with regards to economic performance. These properties are demonstrated in simulation where the time-optimal point-to-point control problem is solved for a constrained multiple mass-sping-damper system, reducing the trial duration by 64% while maintaining accurate waypoint tracking.

Future work includes extensions to iteration-varying systems, as well as an exploration of systems with uncertain economic performance objectives and strict output constraints. Moreover, the identification of the classes of system objectives and constraints that satisfy Assumption 3, and the creation of methods for selecting the controller parameters given in Table 2.1 such that Assumption 6 holds remain points of further investigation.

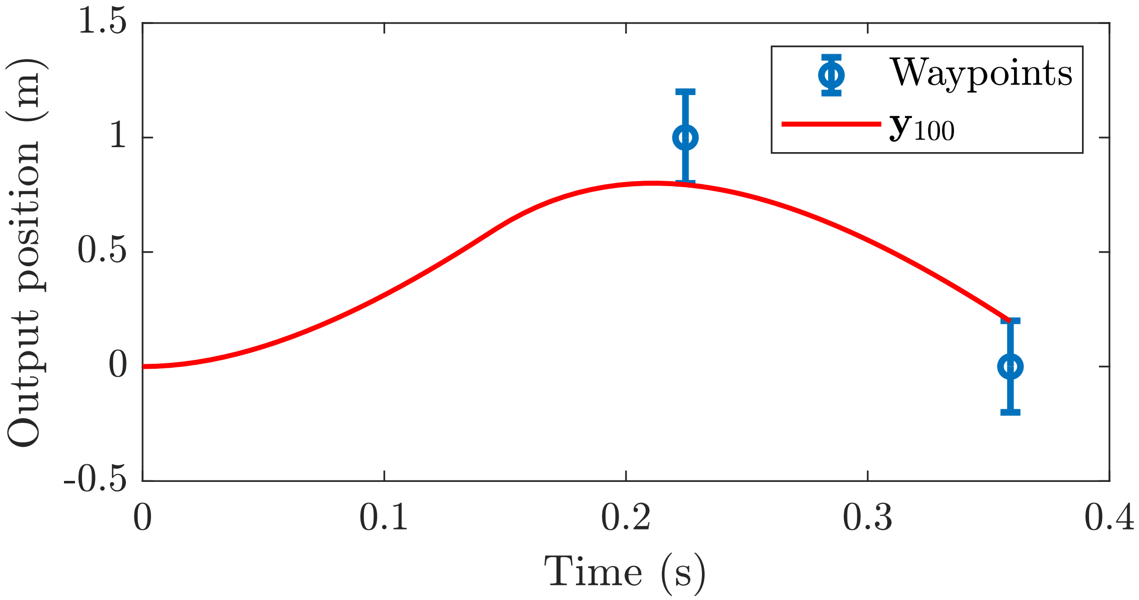


Figure 2.5: Output trajectory at iteration 100. The blue markers indicate the waypoint positions with their corresponding tracking tolerance at timesteps 41 and 81 as designated in tracking constraint (2.39).

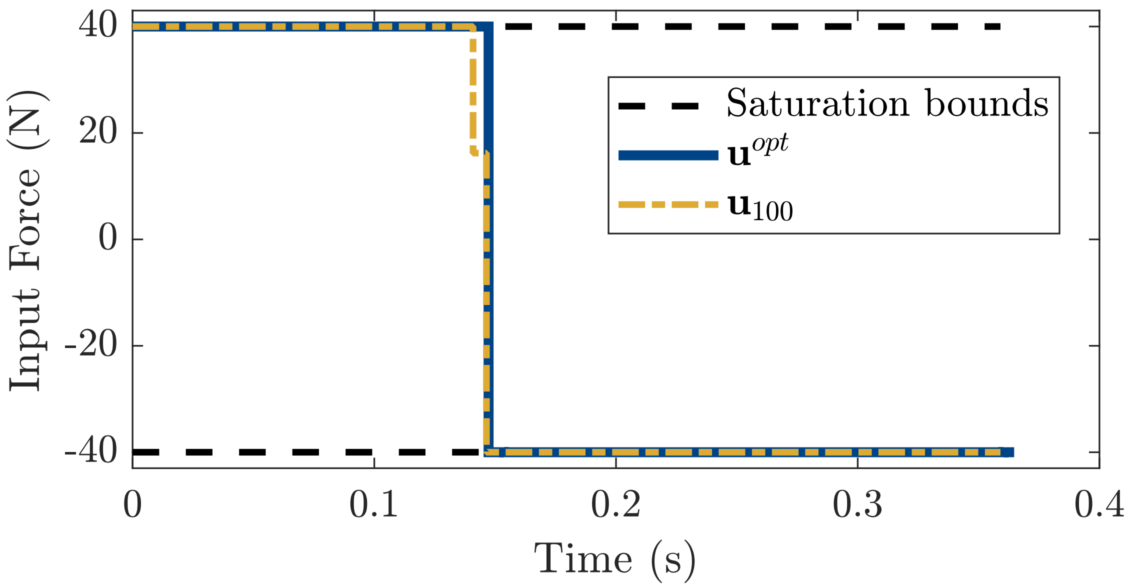


Figure 2.6: Input trajectory at iteration 100. The input remains within the designated saturation limits enforced by (2.40).

CHAPTER 3

Robust Adaptive Economic Model Predictive Control

3.1 Preliminaries: Tube-Based Model Predictive Control

In Chapter 2 a method was described for solving control problems for uncertain repetitive systems that operate discontinuously. Namely, the control problem was framed as a mathematical optimization problem wherein the control signal served as the decision variable. Here, the SQP-ILC algorithm presented in Chapter 2 relied on the resettability of the system state between iterations. While this control strategy borrowed much of its structure from numerical optimization methods, forming practical analogous methods for continuously operated systems is not straightforward. Rather, methods for control of economically driven, continuously operated systems has frequently been considered within the field of optimal control wherein the control problem is similarly expressed as an optimization problem to identify an optimal input signal or control policy [2]. While optimal control is a mature field of research that has enjoyed success for unconstrained linear systems with quadratic costs, the development of practical extensions to nonlinear systems, constrained systems, and systems with more general performance objectives has remained a persistent challenge. Namely, when considering these classes of systems, analytical solutions for the optimal control signal rarely exist. Consequently identification of the control signal instead relies on the use of numerical methods. However, the lack of an offline phase for continuously operated systems necessitates that these optimization problems are solved in real time, which is often impossible despite recent advancements in computing and processing speed.

To mitigate this issue, significant research effort has been directed towards the use of approximate solutions to optimal control problems. In particular, interest has been centered within the field of model predictive control (MPC) [47], which simplifies the optimal control problem to reduce the computational demand. Specifically, whereas optimal control problems require a control trajectory to be derived over a long (potentially infinite) time interval, MPC lessens the numerical burden by limiting the length of time over which input and state predictions are performed. The standard method of operation within MPC algorithms is to leverage a measurement of the system state and a user-available model of the system dynamics to optimize the predicted inputs and system states over a short time interval termed the *prediction horizon*. This procedure is repeated at each timestep, wherein the end of the prediction horizon is incremented in real time. This so-called 'receding horizon' strategy is depicted in Fig.3.1.

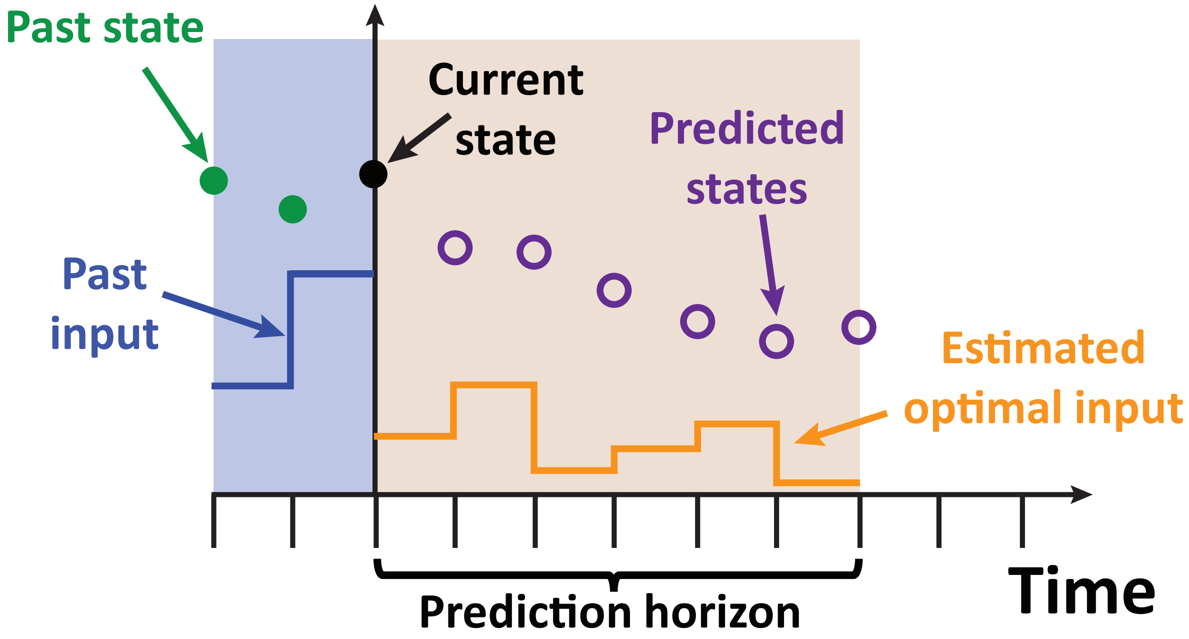
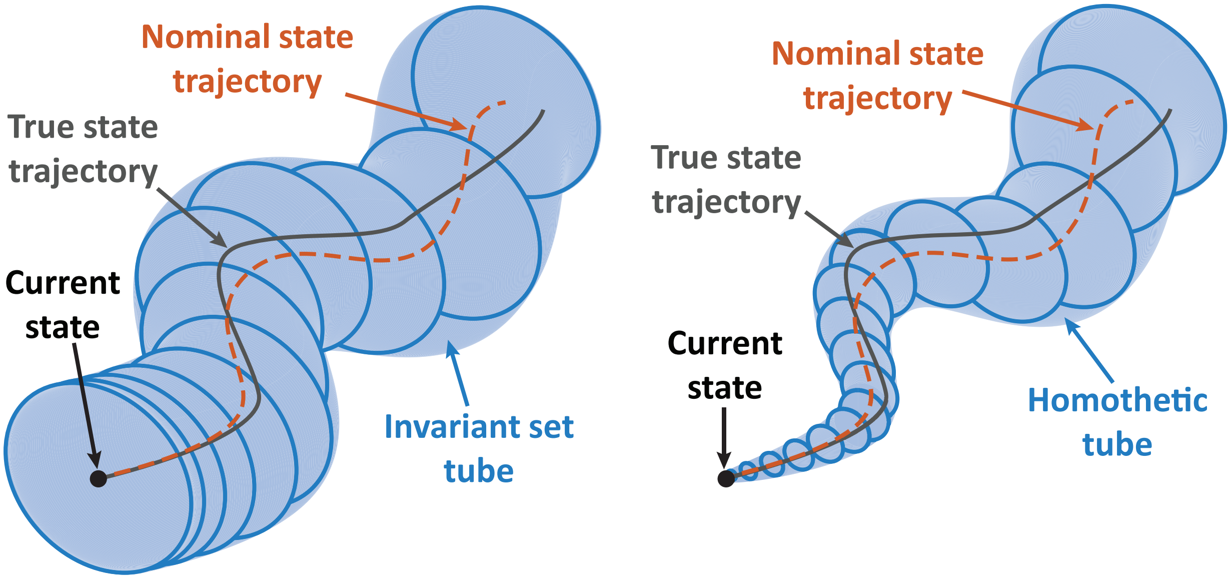


Figure 3.1: Rather than deriving the optimal input signal at all future times, MPC reduces the computational load of optimal control methods by limiting the state and input predictions to a truncated prediction horizon.

Classical MPC theory assumes that the underlying model used to propagate the plant dynamics over the prediction horizon is an accurate reflection of the true system. Extensions of these methods to address the existence of uncertainty is the focus of *robust* MPC [48]. A variety of robust MPC techniques, such as min-max MPC [49] and stochastic MPC [50], have been developed. Of particular interest is tube-based MPC [51]. By leveraging known bounds on the error in the predicted state trajectory, tube-based approaches constrict the feasible space of the nominal system in order to guarantee robust constraint satisfaction. Due to the computational tractability of this approach, tube-based MPC has garnered significant popularity.

Commonly, as in [52, 53], tube-based MPC leverages known bounds on the uncertainty in the system to define an invariant set [54] for the true system states, or an invariant set containing the state estimation error given by the difference between the true system states and a prediction of the states based on a nominal, undisturbed system model. Such an approach then enables guarantees to be made regarding the robust performance and constraint satisfaction of the system by ensuring that the true states of the system remain in this invariant set or *tube* around the nominal state trajectory. The computation of these invariant sets can be performed offline, which enables these methods to be implemented in a way that is amenable for real-time operation.

Alternatively, the construction of the tube can be based on a forward propagation in time of bounds placed on the state estimation error. Here, whereas invariant set methods rely on the calculation of a static set that the state estimate error is known to lie in, the tube geometry evolves over the state prediction horizon based on knowledge of the underlying plant model. Consequently, by using such a dynamic approach, the tube geometry is dependent upon the nominal system trajectory. These *homothetic* tube methods wherein the tube size is dependent upon the system behavior can enable reduced conservatism in comparison to their invariant set counterparts at the expense of additional computational complexity [55, 56]. A graphical comparison of tube-based MPC based on rigid invariant sets versus homothetic tubes is shown in Fig. 3.2.



(a) The size of the tube is fixed over the prediction horizon in invariant set tube-based MPC

(b) The size of the tube varies over the prediction horizon in homothetic tube-based MPC.

Figure 3.2: Comparison between invariant set tubes and homothetic tubes (light blue area) for identical nominal state trajectories (dashed orange line). Homothetic tubes can enable more aggressive control in comparison to invariant set tubes, but incur greater online computational demand as the size of the tube over the prediction horizon serves as an additional decision variable in the control input design. In both cases, starting at the current state (black circle), the true state trajectory (grey line) stays within the tube centered around the nominal state trajectory. The blue ellipses denote the tube geometry at each sample in the state prediction horizon.

However, while sufficient conditions for robust recursive feasibility and closed-loop stability can be shown for tube-based MPC, the robust performance of these algorithms is still largely dependent upon the accuracy of the underlying system model used, and the user's estimate of the amount of uncertainty in the system. In the event that the user's model of the system is highly inaccurate, performance suffers due to inaccurate forecasts of future system behavior. Meanwhile, if the available model of the system accurately reflects the true behavior of the system, but the user's estimate of the amount of uncertainty in the system is larger than what truly exists, then robust methods are frequently overconservative, where unrealized performance losses or unnecessary costs are incurred to ensure robust constraint satisfaction.

3.2 Background and Motivation

In Section 3.1, the concept of tube-based MPC was introduced. Traditionally, this type of control strategy has been developed for the case where the control objective is to drive the system to a predefined equilibrium setpoint. However, in many applications, system performance is not predicated solely upon setpoint tracking, but is instead dependent upon the optimization of an *economic* performance metric. As a consequence of this, the performance stage cost may not be positive definite with respect to a given equilibrium point of the system, which prevents traditional regulation or tracking based robust MPC stability criteria from being directly applied to these systems.

To address this issue, the field of Economic Model Predictive Control (EMPC) has enabled the closed-loop stability and performance of systems with nontraditional performance objectives to be studied [57]. As a consequence of the economic stage cost or the natural dynamics of the system, the optimal behavior for a system controlled using EMPC is often repetitive [58]. Ensuring stability in such cases is generally predicated upon satisfying a *periodic dissipativity* condition, wherein the system trajectories do not necessarily converge to an equilibrium setpoint, but rather a periodic orbit or limit cycle. The propensity of systems with economic objectives to have an optimal behavior that is periodic has thus made repetitive systems a point of interest within the field of EMPC [59, 60]. Additionally, robust EMPC strategies capable of explicitly addressing uncertainties have been developed in [61–63], with considerations towards robust optimal periodic trajectories given in [64, 65].

However, like many other robust control strategies, robust MPC/EMPC is often overly conservative. In particular, if the estimated bounds on the uncertainty impacting the system are too large, the robust MPC/EMPC controller will unnecessarily protect the system against nonexistent uncertainties. This behavior may result in less aggressive control, and therefore hindered performance. To combat this, an alternative or supplementary strategy for mitigating uncertainty is learning-based control. Here, the fundamental idea is that as new data is collected while the system undergoes operation, additional information about the impact that uncertainties have on system behavior or performance may be ascertained. This improved understanding about the uncertainties can then be leveraged to improve the manner in which a control signal is applied.

In particular, the strategies of ILC and RC have played a crucial role towards developing the use of learning-based control for repetitive systems. ILC has predominantly been applied to discontinuously operated systems, while RC literature has focused on continuously operated systems. The lack of an offline phase required to satisfy the continuous operation condition has been the predominant assumption for systems studied within the MPC literature as well, with a few exceptions as in [66]. This distinction has led to variations in the state of the art between the ILC and RC fields. Recent efforts have enabled ILC strategies to address economic performance objectives [22, 29, 30, 67], including systems with constraints as in [31, 34]. However, the need for resettability of the initial condition limits the number of systems to which these strategies may be appropriately applied. In contrast, the field of RC has had limited development towards addressing systems with economic performance objectives. In [68], an economic cost is minimized through the use of a hierarchical control structure wherein an outer-loop identifies an optimal periodic reference trajectory, while an inner-loop drives the system to track this reference trajectory. Meanwhile, in [69, 70], the problem of sparse periodic reference tracking is addressed, wherein a relaxation of system tracking requirements is exploited to address economic performance objectives. However, [68–70] are not applicable to systems with general state and input constraints.

While RC and ILC directly update the control input based on historical system data, learning-based schemes utilizing concepts from indirect adaptive control have enabled uncertainty to be addressed by performing online updates to model parameter estimates. In particular, a common strategy used within robust adaptive MPC approaches leverages data to update uncertainty sets, wherein new data is used to shrink the domain over which uncertainties are known to exist. This idea is utilized in [71–75] for application to linear systems, and in [76–78] for nonlinear systems. However, [71–78], are only suitable for tracking or regulation problems, and are therefore not directly applicable to systems with economic performance objectives, nor can they address systems with repetitive uncertainties. Indeed, developing adaptive MPC strategies that are capable of addressing economic objectives in the presence of uncertainty poses a unique challenge. In this case, not only is the optimal behavior of the true system unknown due to the existence of uncertainty and a non-trivial minimizer of the economic objective, but estimates of the economically optimal behavior of the nominal system vary over time as a consequence of the adaptive model update. Initial efforts to address economic objectives using adaptive MPC schemes include [79] and [80], but both of these approaches are designed for specific applications and recursive feasibility and stability results are not proven. Meanwhile, the scheme offered in [81] establishes conditions for which the proposed robust adaptive MPC framework drives the states to converge to a target set containing an economically optimizing point that is unknown *a priori*. However, this method does not consider the case when the uncertainties or optimal mode of operation is periodic, which limits the types of systems and performance objectives that may be addressed. Additionally, the approach in [81] relies on the ability to, at each timestep, identify an optimizing feedback policy that solves a min-max optimization problem. While this strategy enables conditions to be identified such that recursive feasibility and stability can be provably guaranteed, such a formulation is computationally intractable in practice, and does not investigate the cases where uncertainties arise periodically or where the optimal mode of operation is periodic.

Given these shortcomings, we seek to identify a control strategy applicable to nonlinear systems that are simultaneously: (i) continuously operated and subject to both periodic parametric uncertainty and additive disturbances, (ii) constrained in the inputs and states, and (iii) evaluated based on their ability to minimize an economic performance cost. Therefore, this chapter presents the following contributions:

1. A model adaptation method that leverages historical state and input data to enhance the user's knowledge of the influence of parametric uncertainty on system behavior. Specifically, rather than seeking to identify the value of unknown model parameters that are constant in time, an update law is developed to modify estimates of unknown model parameters that are allowed to vary periodically. An additional update law is proposed that modifies bounds on the set over which the uncertain model parameters are known to exist. Conditions are then established for which convergence of the estimated model parameters to the true model parameters may be achieved, and for which the recursive contraction of the domain over which the true model parameters are known to lie is guaranteed.

2. An RAEMPC algorithm that pairs the adaptive scheme with an optimization step to identify a control signal that minimizes an economic stage cost while robustly enforcing input and state constraints. Through the use of calculated invariant sets, the min-max formulation of [81] can be avoided in favor of a computationally tractable rigid tube-based MPC optimization problem. Further, by exploiting improved knowledge about the parametric uncertainty obtained through the adaptive scheme, the developed controller can be more aggressive in its decision making, thus reducing the conservatism of standard robust EMPC approaches while maintaining robust constraint satisfaction under the influence of both periodic parametric uncertainty and bounded additive disturbances.

3. Sufficient conditions for which recursive feasibility of the MPC optimization problem and convergence of the true system states to a compact set is guaranteed. Specifically, it is demonstrated that the user's estimate of an economically optimizing repetitive state trajectory converges, and that the true system states converge to some bounded neighborhood around this trajectory. Additional insight outlining conditions for which the nominal model

parameters used to define the dynamic constraints within the optimization problem can be guaranteed to be updated at each system repetition under sufficiently small state estimation error are also provided.

4. A simulation study wherein the performance of the robust EMPC framework outlined in [65] is compared to the proposed RAEMPC framework when applied to a nonlinear model of a mechanical braking system.

To summarize, using a tube EMPC scheme, this work enables uncertain repetitive systems with a wide variety of performance metrics to be robustly controlled. Additionally, conservatism of existing tube EMPC strategies is mitigated through the use of a learning strategy to enable more aggressive control, and therefore further improvements in performance in a computationally feasible manner.

To serve as a reference for the reader, a notation guide for the variables used in this chapter is provided in Appendix D.3. The contents of this chapter have been submitted for publication to IEEE Transactions on Automatic Control as [82].

3.3 System description

We consider systems of the form

$$x_{k+1} = x_k + F(x_k, u_k) + G(x_k, u_k)\theta_k + v_k \tag{3.1}$$

where $k \in \mathbb{I}_{\geq 0}$ denotes a timestep index, and $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, and $v \in \mathbb{R}^{n_x}$ denote the system states, inputs, and noise respectively. $F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ and $G : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x \times n_\theta}$ denote known, potentially nonlinear, continuous functions. $\theta_k \in \mathbb{R}^{n_\theta}$ denotes a vector of unknown model parameters. The states and inputs are restricted to lie within the compact, convex sets $\mathcal{X}$ and $\mathcal{U}$ with $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{U}$. We assume that the system is n_c*-periodic* with cycle length n_c such that $\theta_{k+n_c} = \theta_k = \theta^i$ where $i \in \mathbb{I}_{[0,n_c-1]}$ denotes an 'intracycle step index' with $i = k \bmod n_c$. Such may be the case if the system is impacted by repetitive disturbances, or if there exists uncertainty in model parameters that vary over time in a periodic fashion. The specific instance wherein the model parameters θ_k are constant is captured by the case $n_c = 1$. Let j denote the 'intercycle step index' given by $\lfloor \frac{k}{n_c} \rfloor$ which corresponds to the number of completed cycles. The timestep dynamics of (3.1) given in terms of k can be equivalently expressed in terms of i and j as

$$\begin{aligned} x_j^{i+1} &= x_j^i + F(x_j^i, u_j^i) + G(x_j^i, u_j^i)\theta_j^i + v_j^i, \\ &= x_j^i + F(x_j^i, u_j^i) + G(x_j^i, u_j^i)\theta^i + v_j^i \end{aligned} \tag{3.2}$$

where the second equality arises since θ^i_j has no dependency on j due to the periodicity of θ_k with

$$x^0_{j+1} = x^{n_c}_j.$$

While the true values of parameters θ^i are unknown, suppose that at cycle j, a user-estimate, $\hat{\theta}^i_j$, of the parameters is known. The set of estimated parameters over a cycle is denoted using the shorthand $\hat{\theta}_j \triangleq \{\hat{\theta}^0_j, \ldots, \hat{\theta}^{n_c-1}_j\}$.

Assumption 7. *At cycle number j, the model parameters θ^i lie within some known parameter uncertainty set $\Theta^i_j = B(\hat{\theta}^i_j, z^{\Theta^i_j})$ where $z^{\Theta^i_j}$ denotes the uncertainty set radius and $z^{\Theta j}_{max} \triangleq \max_{i \in \mathbb{I}_{[0,n_c-1]}} z^{\Theta^i_j}$ is well defined.*

Satisfying Assumption 7, the parameter estimate error is then given as $\tilde{\theta}^i_j \triangleq \theta^i - \hat{\theta}^i_j \in \tilde{\Theta}^i_j \triangleq B(0, z^{\Theta^i_j})$. A collection of parameter estimate errors in a cycle is denoted as $\tilde{\theta}_j \triangleq \{\tilde{\theta}^0_j, \ldots, \tilde{\theta}^{n_c-1}_j\}$ and $\tilde{\Theta}_j \triangleq \tilde{\Theta}^0_j \times \ldots \times \tilde{\Theta}^{n_c-1}_j$.

Assumption 8. *The noise is bounded such that $\|v_k\| \in \mathcal{V} \triangleq \{v \in \mathbb{R}^{n_x} : \|v\| \leq v^{max}\}$ where v^{max} is a known constant.*

Let $H(\tilde{\Theta}^i_j) \triangleq \{G(x,u)\tilde{\theta} : x \in \mathcal{X}, u \in \mathcal{U}, \tilde{\theta} \in \tilde{\Theta}^i_j\}$ denote the set of state disturbances potentially caused by parameter estimate error $\tilde{\theta}^i_j$. Since $\tilde{\Theta}^i_j$ is known as a consequence of Assumption 7, then $H(\tilde{\Theta}^i_j)$ is also known. Consequently, by defining the uncertainty signal $d^i_j \triangleq G(x^i_j, u^i_j)\tilde{\theta}^i_j + v^i_j \in \mathcal{D}^i_j$ where $\mathcal{D}^i_j \triangleq H(\tilde{\Theta}^i_j) \oplus \mathcal{V}$, system (3.2) is alternatively expressed as

$$x^{i+1}_j = f_{\hat{\theta}^i_j}(x^i_j, u^i_j, d^i_j) \triangleq x^i_j + F(x^i_j, u^i_j) + G(x^i_j, u^i_j)\hat{\theta}^i_j + d^i_j. \tag{3.3}$$

The shorthand notation $f_{\hat{\theta}_j} \triangleq \{f_{\hat{\theta}^0_j}, \ldots, f_{\hat{\theta}^{n_c-1}_j}\}$ will be used to denote the governing dynamics of system (3.3) where appropriate. For given input and disturbance sequences $\boldsymbol{u} = \{u(0), \ldots, u(T-1)\} \in (\mathcal{U})^T$ and $\boldsymbol{d} = \{d(0), \ldots, d(T-1)\} \in (\mathcal{D}_j)^T$ where $\mathcal{D}_j \triangleq \mathcal{D}^0_j \cup \mathcal{D}^1_j \cup \ldots \cup \mathcal{D}^{n_c-1}_j$, the resulting state sequence generated by system model $f_{\hat{\theta}_j}$ is denoted as $\{x^{\boldsymbol{u}}_{\hat{\theta}_j}(0,x), \ldots, x^{\boldsymbol{u}}_{\hat{\theta}_j}(T,x)\}$ with initial condition $x^{\boldsymbol{u}}_{\hat{\theta}_j}(0,x) = x$. Let $(\mathcal{U}_{\hat{\theta}_j,\mathcal{D}_j})^T(x) \subseteq (\mathcal{U})^T$ denote the set of control sequences, $\boldsymbol{u}$, of length T such that $\boldsymbol{u} \in (\mathcal{U})^T$ and $x^{\boldsymbol{u}}_{\hat{\theta}_j}(t,x) \in \mathcal{X}$ for all $\boldsymbol{d} \in (\mathcal{D}_j)^T$ and $t \in \mathbb{I}_{[0,T]}$. In other words, $(\mathcal{U}_{\hat{\theta}_j,\mathcal{D}_j})^T(x)$ contains the feasible control sequences that are guaranteed to drive the system to remain within the feasible set of states over the next T timesteps despite the existence of parametric and additive uncertainties.

Given the existence of model parameter estimate error $\tilde{\theta}^i_j$ and noise v^i_j, the *true* system dynamic model given by (3.3) is unknown to the user. Rather, a nominal system model that *is* available to the user based on the current estimate of the unknown model parameters can be made according to

$$\bar{x}_j^{i+1} = f_{\hat{\theta}_j^i}(\bar{x}_j^i, \bar{u}_j^i, 0) \tag{3.4}$$

with corresponding state estimation error $e_j^i = x_j^i - \bar{x}_j^i$. Although a variety of approaches may be used to derive controllers based on nominal system model (3.4) while remaining robust to disturbances d_j^i, the ability of these strategies to achieve a high level of system performance is inherently intertwined with the structure of $\mathcal{D}_j$. Namely, the expected performance of the system tends to degrade if $\mathcal{D}_j$ is large.

Of particular interest are systems whose performance is given by an *economic cost function* $\ell(x, u) : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$. In this work, $\ell(x, u)$ is assumed to be a continuous function. As noted in [57], standard tracking or regulation MPC approaches typically consider $\ell(x, u)$ to satisfy the condition

$$0 = \ell(x_s, u_s) \leq \ell(x, u) \text{ for all } (x, u) \in \mathcal{Z} \tag{3.5}$$

where (x_s, u_s) denote a known setpoint state and input pair corresponding to an equilibrium of the system. However, for the case of systems with economic performance objectives, condition (3.5) does not necessarily hold. Such may be the case if (x_s, u_s) is unknown or does not exist, or if the definition of the economic cost function is structured such that there exists an $(x, u) \in \mathcal{Z}$ with $\ell(x, u) < \ell(x_s, u_s)$. Hence, economic cost functions encapsulate a broader class of systems whose optimal mode of operation is unknown *a priori* or does not occur at a fixed equilibrium point.

3.4 Uncertainty Set Adaptation

In order to more effectively address the system's robust economic performance, an adaptive scheme, modified from the one developed in [77], is proposed. The goal of this scheme is to leverage historical data to reduce the size of $\mathcal{D}_j$ through systematic updates to the parameter uncertainty set Θ_j^i between cycles while maintaining that $\theta^i \in \Theta_j^i$. In doing so, a robust controller, which will be discussed in Section 3.5, can be more aggressive when designing the control signal to enable improved system performance. Here, adaptation of Θ_j^i is performed in two ways: through updates to the parameter estimate $\hat{\theta}_j^i$, which defines the center of Θ_j^i, as well as reductions of the uncertainty set radius $z^{\Theta_j^i}$.

3.4.1 Parameter Estimate Adaptation

We now detail the method for which the model parameter estimates are updated. Based on a strategy developed in [77], this scheme relies on a sequence of auxiliary variables, η_k, upon which the parameter estimate update law is based. However, whereas the update law in [77] is performed in the time-domain, such an approach is not appropriate here due to the fact that θ_k is not a constant, but rather a periodically time-varying parameter. This property necessitates modifications to the frequency that updates to θ^i_j are performed, as well as the signals upon which the θ^i_j update law is based.

First, the variable ω^i_j is introduced, which is given as a filtered form of the regressor matrix $G(x^i_j, u^i_j)$ according to

$$\omega^i_{j+1} = \omega^i_j + G(x^i_j, u^i_j) - K_\omega \omega^i_j, \quad \omega^i_0 = 0 \tag{3.6}$$

for each $i \in \mathbb{I}_{[0,n_c-1]}$ with $K_\omega \in (0,1)$ chosen by the user. Here, the evolution of ω occurs over the $j-$domain such that updates to ω^i_j only occur once every cycle.

Given (3.6), at intercycle step $j+1$ and intracycle step i, a filtered state variable, $\hat{x}^i_{j+1}$, is now constructed according to

$$\hat{x}^i_{j+1} = \hat{x}^i_j + F(x^i_j, u^i_j) + G(x^i_j, u^i_j)\hat{\theta}^i_{j+1} + K_\omega \tilde{x}^i_j - (1-K_\omega)\omega^i_j(\hat{\theta}^i_j - \hat{\theta}^i_{j+1}) + x^i_{j+1} - x^{i+1}_j \tag{3.7}$$

where x^i_{j+1} is given by the measurement of the current state and $\tilde{x}^i_j \triangleq x^i_j - \hat{x}^i_j$. This filtered state variable definition necessarily differs from the one used in [77], due to the fact that θ^i only directly appears in (3.2) periodically. Auxiliary variable η^i_j is subsequently defined as

$$\eta^i_j \triangleq \tilde{x}^i_j - \omega^i_j \tilde{\theta}^i_j. \tag{3.8}$$

Substituting (3.2) for x^{i+1}_j in (3.7) gives that the evolution of $\tilde{x}^i_j$ in the $j-$domain can be expressed according to

$$\tilde{x}^i_{j+1} = \tilde{x}^i_j + G(x^i_j, u^i_j)\tilde{\theta}^i_{j+1} - K_\omega \tilde{x}^i_j + (1-K_\omega)\omega^i_j(\hat{\theta}^i_j - \hat{\theta}^i_{j+1}) + v^i_j. \tag{3.9}$$

Then, combining (3.8) with the dynamics of the filtered regressor matrix in (3.6) and the $\tilde{x}$ dynamics in (3.9), we have that

$$\eta^i_{j+1} = \eta^i_j - K_\omega \eta^i_j + v^i_j, \quad \eta^i_0 = \tilde{x}^i_0$$

for each $i \in \mathbb{I}_{[0,n_c-1]}$. As v^i_j is an unknown signal, η^i_j is therefore unknown as well. Therefore, an estimated auxiliary variable, $\hat{\eta}_j^i$, is instead generated and is given by

$$\hat{\eta}_{j+1}^i = \hat{\eta}_j^i - K_\omega \hat{\eta}_j^i.$$

The resulting auxiliary variable estimation error, defined as

$$\tilde{\eta}_j^i \triangleq \eta_j^i - \hat{\eta}_j^i, \tag{3.10}$$

then evolves according to

$$\tilde{\eta}_{j+1}^i = \tilde{\eta}_j^i - K_\omega \tilde{\eta}_j^i + v_j^i. \tag{3.11}$$

Let $\Sigma_j^i \in \mathbb{R}^{n_\theta \times n_\theta}$ denote the matrix obtained from

$$\Sigma_{j+1}^i = \Sigma_j^i + (\omega_j^i)^\top \omega_j^i, \quad \Sigma_0^i = \beta I \tag{3.12}$$

for some user-defined constant $\beta > 0$. We observe that, because ω_j^i is a real valued matrix with $\beta > 0$, Σ_j^i is a symmetric, positive definite matrix. The dynamics of $(\Sigma_j^i)^{-1}$ are consequently given by

$$(\Sigma_{j+1}^i)^{-1} = (\Sigma_j^i)^{-1} - (\Sigma_j^i)^{-1}(\omega_j^i)^\top (I + \omega_j^i (\Sigma_j^i)^{-1} (\omega_j^i)^\top)^{-1} \omega_j^i (\Sigma_j^i)^{-1}, \quad (\Sigma_0^i)^{-1} = \frac{1}{\beta} I. \tag{3.13}$$

As Σ_j^i is positive definite, this gives that $(\Sigma_j^i)^{-1}$ is also positive definite with $\lambda_{max}\left((\Sigma_j^i)^{-1}\right) \leq \frac{1}{\beta}$. Further, as ω_j^i is real valued and bounded as a consequence of the continuity of $G(x, u)$, compactness of $\mathcal{Z}$, and definition of K_ω, this then implies that there exists a constant $\gamma^i > 0$ such that

$$\gamma^i \leq \|(I + \omega_j^i (\Sigma_j^i)^{-1} (\omega_j^i)^\top)^{-1}\| < 1 \text{ for all } j.$$

Through the construction of these user-generated variables, a parameter estimate update law can then be designed according to

$$\hat{\theta}_{j+1}^i = \hat{\theta}_j^i + (\Sigma_j^i)^{-1}(\omega_j^i)^\top \left(I + \omega_j^i (\Sigma_j^i)^{-1} (\omega_j^i)^\top\right)^{-1} (\tilde{x}_j^i - \hat{\eta}_j^i), \tag{3.14}$$

$$= \hat{\theta}_j^i + (\Sigma_j^i)^{-1}(\omega_j^i)^\top \left(I + \omega_j^i (\Sigma_j^i)^{-1} (\omega_j^i)^\top\right)^{-1} (\omega_j^i \tilde{\theta}_j^i + \tilde{\eta}_j^i) \tag{3.15}$$

where the first equality denotes the user-implementable update law, while the second equality arises through the definitions of η_j^i and $\tilde{\eta}_j^i$ in (3.8) and (3.10). From (3.15), we see that through the use of the auxiliary variable, the parameter update law is dependent upon parameter estimation error $\tilde{\theta}_j^i$, despite the fact that this variable is unknown. Hence, under sufficient excitation of ω_j^i, the existence of model parameter error can trigger an update to the model parameter estimates. This will enable conditions to be established for which the parameter estimation error can be driven towards zero.

As θ^i is assumed to exist within the uncertainty set Θ_j^i from Assumption 7, it is undesirable for update law (3.14) to update $\hat{\theta}_j^i$ such that $\hat{\theta}_{j+1}^i \notin \Theta_j^i$. Therefore, (3.14) is refined to be of the form

$$\bar{\hat{\theta}}_{j+1}^i = \text{Proj}\Big\{\hat{\theta}_j^i + (\Sigma_j^i)^{-1}(\omega_j^i)^\top \left(I + \omega_j^i(\Sigma_j^i)^{-1}(\omega_j^i)^\top\right)^{-1}(\tilde{x}_j^i - \hat{\eta}_j^i), \Theta_j^i\Big\} \tag{3.16}$$

such that $\bar{\hat{\theta}}_{j+1}^i \in \Theta_j^i$. Consequently, because Θ_j^i is a convex set with $\theta^i \in \Theta_j^i$,

$$(\bar{\tilde{\theta}}_{j+1}^i)^\top \Sigma_{j+1}^i \bar{\tilde{\theta}}_{j+1}^i \leq (\tilde{\theta}_{j+1}^i)^\top \Sigma_{j+1}^i \tilde{\theta}_{j+1}^i \tag{3.17}$$

where $\bar{\tilde{\theta}}_j^i \triangleq \theta^i - \bar{\hat{\theta}}_j^i$.

We now outline sufficient conditions to guarantee that the parameter estimation error converges to zero.

Lemma 13. *Define* $V_{\tilde{\theta}_j^i} = (\tilde{\theta}_j^i)^\top \Sigma_j^i \tilde{\theta}_j^i$. *If*

$$\sum_{j=0}^{\infty} \left(\|\tilde{\eta}_j^i\|^2 - \gamma^i \|\tilde{x}_j^i - \hat{\eta}_j^i\|^2\right) < \infty \textit{ and } \lim_{j\to\infty} \lambda_{min}(\Sigma_j^i) = \infty, \tag{3.18}$$

then the application of parameter estimate update law (3.16) *yields*

$$V_{\tilde{\theta}_{j+1}^i} - V_{\tilde{\theta}_j^i} = -(\tilde{x}_j^i - \hat{\eta}_j^i)^\top \left(I + \omega_j^i(\Sigma_j^i)^{-1}(\omega_j^i)^\top\right)^{-1}(\tilde{x}_j^i - \hat{\eta}_j^i) + (\tilde{\eta}_j^i)^\top \tilde{\eta}_j^i \tag{3.19}$$

and

$$\lim_{j\to\infty} \tilde{\theta}_j^i = 0. \tag{3.20}$$

Proof. The proof is similar to [77, Lemma 2] with some minor differences and is included in Appendix B.1. However, whereas [77, Lemma 2] establishes conditions for which the parameter error converges to zero in the case of *constant* unknown model parameters, the modified construction of the filtered state variable $\hat{x}_j^i$ according to (3.7) yields conditions for which property (3.20) holds such that parameter error converges to zero in the case that the model parameters are *periodic*. □

Given the dynamics of Σ_j^i in (3.12), the second condition of (3.18) effectively places a benchmark on the required excitation of ω_j^i over time needed to ensure parameter estimate convergence to the true parameter values. From the relationship between ω_j^i and $G(x_j^i, u_j^i)$ in (3.6), this therefore implies that sufficient excitation of $G(x^i_j, u^i_j)$ is needed to ensure parameter convergence. Given the dependence of $\tilde{x}^i_j$ on $G(x^i_j, u^i_j)$ in (3.9), satisfaction of the first condition of (3.18) is also dependent upon excitation of $G(x^i_j, u^i_j)$ such that parameter error $\tilde{\theta}^i_j$ sufficiently impacts $\tilde{x}^i_j$. Yet, while Lemma 13 outlines conditions for which the parameter estimates converge to the true parameter values, even if these conditions are not met, the use of the projected update law in (3.16) ensures that the parameter estimate error is, in the worst case, bounded by the initial parameter error uncertainty set $\tilde{\Theta}^i_0$.

3.4.2 Uncertainty Set Radius Adaptation

Following the update to model parameter estimates $\hat{\theta}^i_j$, an update to the uncertainty set radius $z^{\Theta^i_j}$ is then performed. The combination of these two updates allows for the center and radius of Θ^i_j to be updated between cycles. The update to $z^{\Theta^i_j}$ mimics the strategy in [77] but, similar to the update to $\hat{\theta}^i_j$, is performed periodically rather than at each timestep.

From Lemma 13, an expression is derived for the change in $V_{\tilde{\theta}^i_j}$ between parameter updates, and therefore for the cycle-to-cycle change in $\tilde{\theta}^i_j$. But, as $\tilde{\theta}^i_j$ is unknown to the user, the change in $\tilde{\theta}^i_j$ is subsequently unknown. However, from the $\tilde{\eta}^i_j$ dynamics given in (3.11) along with Assumption 8, we have that $(\tilde{\eta}^i_j)^\top \tilde{\eta}^i_j \leq \left(\frac{v^{max}}{K_\omega}\right)^2$. This knowledge will enable us to place a known bound on the cycle-to-cycle change in $\tilde{\theta}^i_j$, which can be leveraged to construct an update law for $z^{\Theta^i_j}$. First, let $\bar{z}_{\Theta^i_j}$ denote a candidate uncertainty set radius at cycle j corresponding to intracycle step i which is given as follows:

$$\begin{aligned} \bar{z}_{\Theta^i_j} =& \sqrt{\frac{V_{z^{\Theta^i_j}}}{\lambda_{min}(\Sigma^i_j)}}, \\ V_{z^{\Theta^i_{j+1}}} =& V_{z^{\Theta^i_j}} - (\tilde{x}^i_j - \hat{\eta}^i_j)^\top (I + \omega^i_j (\Sigma^i_j)^{-1} (\omega^i_j)^\top)^{-1} (\tilde{x}^i_j - \hat{\eta}^i_j) + \left(\frac{v^{max}}{K_\omega}\right)^2, \quad V_{z^{\Theta^i_0}} = \beta (z^{\Theta^i_0})^2. \end{aligned} \tag{3.21}$$

Then, the uncertainty set $\Theta^i_j = B(\hat{\theta}^i_j, z^{\Theta^i_j})$ is updated according to Algorithm 2.

Algorithm 2 Adaptive update to Θ^i_j

```
**Initialize:** $i = j = k = 0$.
**while** $k \geq 0$ **do**
    Calculate $\bar{\hat{\theta}}^i_{j+1}$ from (3.16) and $\bar{z}^i_{\Theta j+1}$ from (3.21).
    **if** $\bar{z}^i_{\Theta j+1} \leq z^{\Theta^i_j} - \|\bar{\hat{\theta}}^i_{j+1} - \hat{\theta}^i_j\|$ **then**
        $(\hat{\theta}^i_{j+1}, z^{\Theta^i_{j+1}}) = (\bar{\hat{\theta}}^i_{j+1}, \bar{z}^i_{\Theta j+1})$.
    **else**
        $(\hat{\theta}^i_{j+1}, z^{\Theta^i_{j+1}}) = (\hat{\theta}^i_j, z^{\Theta^i_j})$.
    **end if**
    **if** $k = K_1 n_c - 1, K_1 \in \mathbb{I}_{\geq 0}$ **then**
        $j \to j + 1, i \to 0, k \to k + 1$.
    **else**
        $i \to i + 1, k \to k + 1$.
    **end if**
**end while**
```

Lemma 14. *If $\theta^i \in \Theta^i_0$ and Assumption 8 holds, then the application of Algorithm 2 ensures that*

$$\Theta^i_{j+1} \subseteq \Theta^i_j \tag{3.22}$$

and

$$\theta^i \in \Theta^i_j \textit{ for all } j. \tag{3.23}$$

Proof. The proof follows closely to [77, Lemma 3] and is given in Appendix B.2. However, whereas [77, Lemma 3] establishes conditions for which the parameter uncertainty set contracts in the case of *constant* unknown model parameters, updating the uncertainty set center and radius according to (3.16) and (3.21) such that the uncertainty set is only updated intermittently produces property (3.22) which guarantees contraction of the parameter uncertainty sets Θ^i_j in the event that the unknown model parameters are *periodic*. □

Here we see that if the conditions of Lemma 14 are met, that result (3.23) gives that Assumption 7 holds. Through Lemmas 13 and 14 we have outlined conditions for which the model parameter identification problem is solved, and the parameter uncertainty set contracts over time while maintaining inclusion of the true model parameters as in Figure 3.3. Further, as a consequence of (3.22), we have that Θ^i_j converges to some Θ^i_∞. Since $\hat{\theta}^i_j$ denotes the center of Θ^i_j, this implies that $\hat{\theta}^i_j$ also converges to some $\hat{\theta}^i_\infty$, with $\hat{\theta}^i_\infty = \theta^i$ if (3.18) holds.

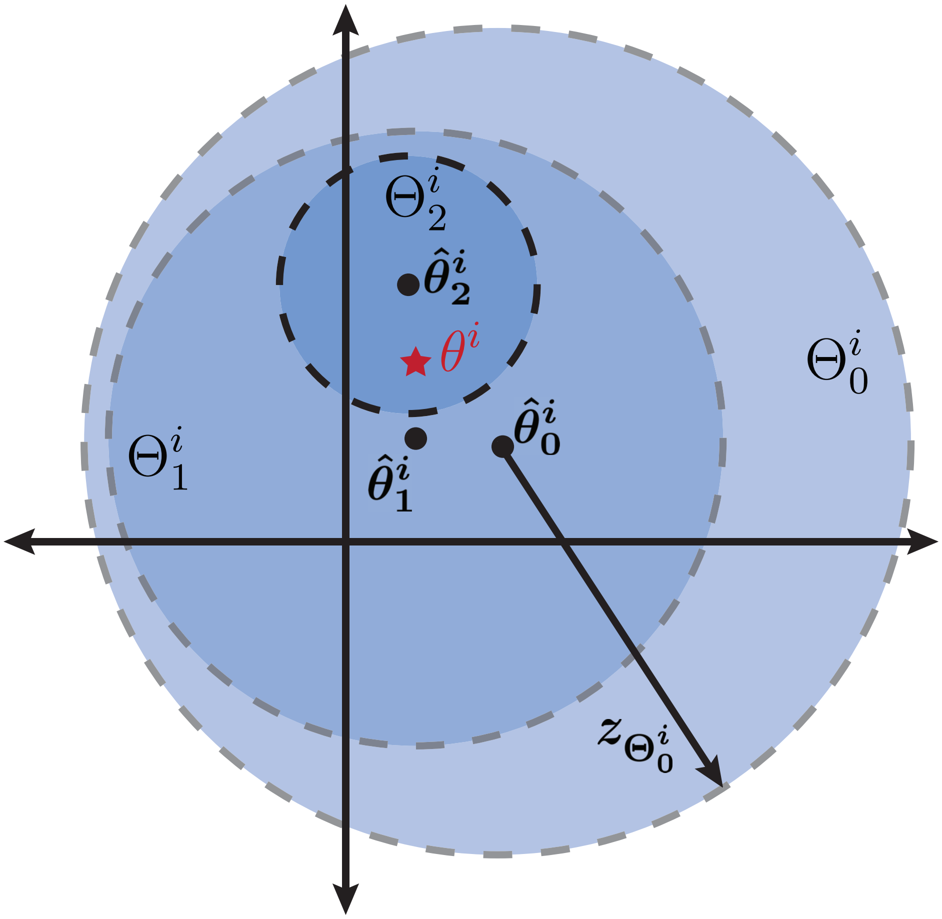


Figure 3.3: The evolution of the parameter uncertainty set over $j = 0, 1, 2$. The parameter uncertainty sets (blue circles) have centers at $\hat{\theta}_j^i$ and radii $z^{\Theta_j^i}$. After each update, the uncertainty sets contract, but continue to contain the unknown true model parameters, θ^i (red star).

Importantly, the properties of the adaptive scheme given by Lemmas 13 and 14 are agnostic to the control law, which affords us freedom when designing the controller. In effect, as $\tilde{\theta}_j^i$ shrinks between cycles, the set $\mathcal{D}_j^i$ does as well. This will allow more aggressive control to be employed over time.

3.5 Proposed RAEMPC Framework

Now that the adaptive scheme has been developed, we can leverage the improved parameter estimation in the control design. In Section 3.5.1, assumptions regarding the existence of invariant sets are outlined and periodicity properties of the nominal system are explored. In Section 3.5.2, the RAEMPC algorithm is outlined. Recursive feasibility and stability properties are investigated in Sections 3.5.3 and 3.5.4.

3.5.1 Error Invariance and Periodicity Properties

Suppose that the control input applied to the true system (3.3) is given according to the feedback law

$$u_j^i = \phi(\bar{u}_j^i, x_j^i, \bar{x}_j^i). \tag{3.24}$$

Then the state estimation error dynamics are given by

$$e_j^{i+1} = f_{\hat{\theta}_j^i}(x_j^i, \phi(\bar{u}_j^i, x_j^i, \bar{x}_j^i), d_j^i) - f_{\hat{\theta}_j^i}(\bar{x}_j^i, \bar{u}_j^i, 0). \tag{3.25}$$

Definition 1. *From [62], a set $\Omega_j \subseteq \mathbb{R}^{n_x}$ is Robust Control Invariant (RCI) if there exists a feedback law* (3.24) *such that for all $x_k, \bar{x}_k \in \mathcal{X}$ and $\bar{u}_k \in \mathcal{U}$ with $e_k \in \Omega_j$, $\phi(\bar{u}_k, x_k, \bar{x}_k) \in \mathcal{U}$ and $e_{k+1} \in \Omega_j$ for all $d_k \in \mathcal{D}_j$.*

Assumption 9. *There exists a continuous feedback law $\phi : \mathcal{U} \times \mathcal{X} \times \mathcal{X} \to \mathcal{U}$ such that an RCI set, Ω_j, exists for error dynamics* (3.25) *and $\lim_{\bar{x} \to x} \phi(\bar{u}, x, \bar{x}) = \bar{u}$.*

As in [53, 62], to ensure that constraints are robustly satisfied in the presence of uncertainties, we restrict the nominal system (3.4) to lie within a tightened constraint set given as

$$\bar{\mathcal{Z}}_j \triangleq \{(\bar{x}, \bar{u}) \in \mathcal{Z} : (x, \phi(\bar{u}, x, \bar{x})) \in \mathcal{Z}, \quad \forall x \in \{\bar{x}\} + \Omega_j\}. \tag{3.26}$$

The projections of $\bar{\mathcal{Z}}_j$ onto $\mathcal{X}$ and $\mathcal{U}$ are denoted as $\bar{\mathcal{X}}_j$ and $\bar{\mathcal{U}}_j$ respectively. Define $\Delta\hat{\theta}_{j+1}^i \triangleq \hat{\theta}_{j+1}^i - \hat{\theta}_j^i$ as the change in the model parameter estimates between consecutive cycles. Analogously, define $\Delta\Theta_{j+1}^i \triangleq \Theta_j^i \ominus \Theta_{j+1}^i$ and $\Delta\tilde{\Theta}_{j+1} \triangleq \tilde{\Theta}_j \ominus \tilde{\Theta}_{j+1}$. Consequently, at cycle $j+1$, system (3.4) can be rewritten as

$$\begin{aligned} \bar{x}_{j+1}^{i+1} =& \bar{x}_{j+1}^i + F(\bar{x}_{j+1}^i, \bar{u}_{j+1}^i) + G(\bar{x}_{j+1}^i, \bar{u}_{j+1}^i)\hat{\theta}_{j+1}^i, \\ =& f_{\hat{\theta}_j^i}(\bar{x}_{j+1}^i, \bar{u}_{j+1}^i, 0) + G(\bar{x}_{j+1}^i, \bar{u}_{j+1}^i)\Delta\hat{\theta}_{j+1}^i. \end{aligned} \tag{3.27}$$

Define the sets $R^i(\Delta\hat{\theta}_{j+1}^i) \triangleq \{r \in \mathbb{R}^{n_x} : r = G(x,u)\Delta\hat{\theta}_{j+1}^i, (x,u) \in \mathcal{Z}\}$ and $R(\Delta\hat{\theta}_{j+1}) \triangleq R^0(\Delta\hat{\theta}_{j+1}^0) \cup \ldots \cup R^{n_c-1}(\Delta\hat{\theta}_{j+1}^{n_c-1})$. Here, $R(\Delta\hat{\theta}_{j+1})$ denotes the set of all possible differences in the nominal state estimate as a consequence of updating the model parameters between iterations j and $j+1$. This leads to the following lemma.

Lemma 15. *If $\theta^i \in \Theta_0^i$ for $i \in \mathbb{I}_{[0,n_c-1]}$, and Assumptions 8 and 9 are satisfied, then*

1. *$\Omega_{j+1} \oplus R(\Delta\hat{\theta}_{j+1}) \subseteq \Omega_j$,*
2. *$\bar{\mathcal{X}}_{j+1} \supseteq \bar{\mathcal{X}}_j \oplus R(\Delta\hat{\theta}_{j+1})$, and $\bar{\mathcal{U}}_{j+1} \supseteq \bar{\mathcal{U}}_j$.*

Proof. To demonstrate the first claim, assume for the sake of identifying a contradiction that $\Omega_{j+1} \oplus R(\Delta\hat{\theta}_{j+1}) \nsubseteq \Omega_j$. Then there exists an $e^{i+1}_{j+1} \in \Omega_{j+1}$, $(\bar{x}, \bar{u}) \in \bar{\mathcal{Z}}_{j+1}$, and $i \in \mathbb{I}_{[0,n_c-1]}$ such that

$$e^{i+1}_{j+1} + G(\bar{x}, \bar{u})\Delta\hat{\theta}^i_{j+1} \notin \Omega_j. \tag{3.28}$$

Substitution of error dynamics (3.25) and the definition of $\Delta\hat{\theta}^i_{j+1}$ gives, for some $x \in \mathcal{X}$ such that $(x - \bar{x}) \in \Omega_j$, $u = \phi(\bar{u}, x, \bar{x}) \in \mathcal{U}$, and $v \in \mathcal{V}$, that

$$\begin{aligned}
&e^{i+1}_{j+1} + G(\bar{x}, \bar{u})\Delta\hat{\theta}^i_{j+1} \\
&\quad = e^i_{j+1} + F(x, u) + G(x, u)\hat{\theta}^i_{j+1} + G(x, u)\tilde{\theta}^i_{j+1} + v - F(\bar{x}, \bar{u}) - G(\bar{x}, \bar{u})\hat{\theta}^i_j, \\
&\quad = e^i_{j+1} + F(x, u) + G(x, u)\hat{\theta}^i_j + G(x, u)\tilde{\theta}^i_j + v - F(\bar{x}, \bar{u}) - G(\bar{x}, \bar{u})\hat{\theta}^i_j, \\
&\quad = f_{\hat{\theta}^i_j}(x, u, d) - f_{\hat{\theta}^i_j}(\bar{x}, \bar{u}, 0)
\end{aligned}$$

where the second equality is derived from the definition of $\tilde{\theta}^i_{j+1}$. However, from Assumption 9, $f_{\hat{\theta}^i_j}(x, u, d) - f_{\hat{\theta}^i_j}(\bar{x}, \bar{u}, 0) \in \Omega_j$. This contradicts (3.28), and therefore $\Omega_{j+1} \oplus R(\Delta\hat{\theta}_{j+1}) \subseteq \Omega_j$.

The second claim follows from the definition of $\bar{\mathcal{Z}}_j$ in (3.26). □

Lemma 16. *If $\theta^i \in \Theta^i_0$ and Assumption 8 are satisfied, and Ω_0 exists, then the following properties hold:*

1. *$\mathcal{D}_{j+1} \subseteq \mathcal{D}_j$,*

2. *$\Omega_{j+1} \subseteq \Omega_j$.*

Proof. Since Lemma 14 ensures that $z^{\Theta^i_{j+1}} \leq z^{\Theta^i_j}$, then $\tilde{\Theta}^i_{j+1} \subseteq \tilde{\Theta}^i_j$ and hence $H(\tilde{\Theta}^i_{j+1}) \subseteq H(\tilde{\Theta}^i_j)$. Consequently $\mathcal{D}^i_{j+1} \subseteq \mathcal{D}^i_j$ and therefore $\mathcal{D}_{j+1} \subseteq \mathcal{D}_j$ giving the first claim. Similarly, from Definition 1, since $\mathcal{D}_{j+1} \subseteq \mathcal{D}_j$, this implies that there exists an $\Omega_{j+1} \subseteq \Omega_j$ giving the second claim. □

Lemmas 15 and 16 establish criteria for convergence of the RCI set and feasible set sequences $\{\Omega_j\}$ and $\{\bar{\mathcal{Z}}\}$, and will enable the development of conditions that guarantee recursive feasibility in Section 3.5.3.

Given the periodicity of the system model parameters, we are naturally interested in studying system trajectories that are periodic or repetitive with period length $P \in \mathcal{I}_{\geq n_c}$. We distinguish the difference between the terminology of a 'cycle' and an 'iteration' used within the remainder of the chapter, where a cycle denotes a sequence of n_c timesteps, while an iteration refers to a sequence of P timesteps. It is assumed that $P = K_P n_c$ for some $K_P \in \mathcal{I}_{\geq 1}$ such that multiple cycles constitute an iteration. To simplify further analysis, system (3.3) is lifted to the P-periodic orbit domain as follows.

Definition 2. *Let*

$$\begin{aligned}\mathring{x} &= (x^0, \dots, x^{P-1}) \in (\mathcal{X})^P,\\ \mathring{u} &= (u^0, \dots, u^{P-1}) \in (\mathcal{U})^P,\\ \mathring{d} &= (d^0, \dots, d^{P-1}) \in (\mathcal{D}_j)^P.\end{aligned}$$

The analogous P-step system of system (3.3) *is defined as $\mathring{x}(k+P) \triangleq f^P_{\hat{\theta}_j}(\mathring{x}(k), \mathring{u}(k), \mathring{d}(k))$ where*

$$f^P_{\hat{\theta}_j}(\mathring{x}, \mathring{u}, \mathring{d}) = \big(f_{\hat{\theta}^0_j}(x^{P-1}, u^0, d^0), f_{\hat{\theta}^1_j}(f_{\hat{\theta}^0_j}(x^{P-1}, u^0, d^0), u^1, d^1), \dots\big)$$

and $x^{P-1}(0) = x \in \mathcal{X}$. For initial state x, control sequence $\boldsymbol{u} \in (\mathcal{U})^{PK}$, and disturbance sequence $\boldsymbol{d} \in (\mathcal{D}_j)^{PK}$ with $K \in \mathbb{I}_{\geq 1}$, the resulting state of the P-step system based on dynamic model $f_{\hat{\theta}_j}$ is denoted as

$$\mathring{x}^{\boldsymbol{u}}_{\hat{\theta}_j}(k, x) = (x^{\boldsymbol{u}}_{\hat{\theta}_j}(k-P+1, x), \dots, x^{\boldsymbol{u}}_{\hat{\theta}_j}(k, x))$$

such that $\mathring{x}^{\boldsymbol{u}}_{\hat{\theta}_j}(k+P, x) = f^P_{\hat{\theta}_j}(\mathring{x}^{\boldsymbol{u}}_{\hat{\theta}_j}(k, x), \mathring{u}(k), \mathring{d}(k))$. The closed-loop $P-$step system states and inputs are denoted as $\mathring{x}_k \triangleq (x_{k-P+1}, \dots, x_k)$ and $\mathring{u}_k \triangleq (u_k, \dots, u_{k+P-1})$.

Definition 3. *A sequence of state/input pairs $\pi_j = \{(\bar{x}^p_{j,0}, \bar{u}^p_{j,0}), \dots, (\bar{x}^p_{j,P-1}, \bar{u}^p_{j,P-1})\}$ is termed a nominal feasible P-periodic orbit of system* (3.4) *if $(\bar{x}^p_{j,t}, \bar{u}^p_{j,t}) \in \bar{\mathcal{Z}}_j$ and for $t \in \mathbb{I}_{[0,P-2]}$, $\bar{x}^p_{j,t+1} = f_{\hat{\theta}^i_j}(\bar{x}^p_{j,t}, \bar{u}^p_{j,t}, 0)$ with $i = t \bmod n_c$ and $\bar{x}^p_{j,0} = f_{\hat{\theta}^{n_c-1}_j}(\bar{x}^p_{j,P-1}, \bar{u}^p_{j,P-1}, 0)$. The sequence is termed a minimal P-periodic orbit if $\bar{x}^p_{j,t_1} \neq \bar{x}^p_{j,t_2}$ for $t_1 \neq t_2$, $0 \leq t_1, t_2 \leq P-1$. Let $\mathring{\bar{x}}^p_j = \{\bar{x}^p_{j,0}, \dots, \bar{x}^p_{j,P-1}\}$ and $\mathring{\bar{u}}^p_j = \{\bar{u}^p_{j,0}, \dots, \bar{u}^p_{j,P-1}\}$ denote the projections of a minimal π_j onto $\bar{\mathcal{X}}_j$ and $\bar{\mathcal{U}}_j$ respectively. The set of all nominal feasible P-periodic orbits of system* (3.4) *is denoted as Π_j.*

The benefits of introducing the P-step system are two-fold. First, although system model (3.4) is time-varying, by leveraging the periodicity of parameters $\hat{\theta}^i_j$, the P-step system is iteration-invariant due to the assumption that P is a multiple of n_c. Second, a P-periodic orbit, π_j, of system (3.4) is equivalent to an equilibrium point, $(\mathring{\bar{x}}^p_j, \mathring{\bar{u}}^p_j)$, of the analogous nominal P-step system which enables simpler steady-state analysis to be used when investigating stability.

For the nominal system, the cycle-to-cycle variation in $f_{\hat{\theta}^i_j}(\bar{x}, \bar{u}, 0)$ is solely dependent on the change in the parameter estimates, $\hat{\theta}_j$, and therefore on the parameter estimate error, $\tilde{\theta}_j$. Consequently, the sequence π_j is dictated by the current parameter estimate error. This dependency is expressed using the notation $\pi(\tilde{\theta}_j) = \pi_j$. Let

$$\mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j} \triangleq \{(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in (\bar{\mathcal{X}})^P \times (\bar{\mathcal{U}})^P : \tilde{\theta} \in \tilde{\Theta}_j, (\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \pi(\tilde{\theta})\}$$

denote the set of all possible minimal periodic orbits of the nominal system (3.4) for parameter error uncertainty set $\tilde{\Theta}_j$.

Inspired by [62], the integrated stage cost $\ell^{\text{int}}(\bar{x}, \bar{u}, \Omega)$ is now introduced as

$$\ell^{\text{int}}(\bar{x}, \bar{u}, \Omega) = \int_{x \in \bar{x} \oplus \Omega} \ell(x, \phi(\bar{u}, x, \bar{x}))dx. \tag{3.29}$$

The intuition behind integrated stage cost (3.29) is that, given the uncertainty in the accuracy of the nominal system state $\bar{x}$, simply evaluating the stage cost at $(\bar{x}, \bar{u})$ may not accurately reflect the cost of the true system. Alternatively, by integrating the stage cost over the state estimation errors given by RCI set Ω_j, (3.29) produces an average of the costs associated with any possible states/inputs of the true system that may appear due to uncertainties. Without loss of generality, we assume that $\ell(x, u)$, and therefore $\ell^{\text{int}}(\bar{x}, \bar{u}, \Omega_j)$, is nonnegative for all $(x, u) \in \mathcal{Z}$. As a consequence of Lemma 16, we then have that since $\Omega_{j+1} \subseteq \Omega_j$ and $\ell(x, u) \geq 0$, that $\ell^{\text{int}}(\bar{x}, \bar{u}, \Omega_{j+1}) \leq \ell^{\text{int}}(\bar{x}, \bar{u}, \Omega_j)$. The integrated stage cost can also be expressed such that it is compatible with the P-step system model as

$$\mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\bar{u}}, \Omega) = \sum_{t=0}^{P-1} \ell^{\text{int}}(\bar{x}^{\mathring{\bar{u}}}(t, \bar{x}), \bar{u}_k, \Omega)$$

where $\bar{x} = (\mathring{\bar{x}})_{P-1}$.

Definition 4. *The robust optimal periodic orbit π_j^* with optimal period length P^* of system* (3.4) *is given by*

$$(P^*, \pi_j^*) = \underset{\substack{P \in \mathbb{I}_{\geq n_c} \\ \pi_j \in \Pi_j}}{\text{argmin}} \quad \sum_{t=0}^{P-1} \int_{x \in \{\bar{x}^p_{j,t}\} + \Omega_j} \ell(x, \phi(\bar{u}^p_{j,t}, x, \bar{x}^p_{j,t}))dx \tag{3.30}$$

with $(\bar{x}^p_{j,t}, \bar{u}^p_{j,t}) \in \pi_j$.

From (3.30), the robust optimal periodic orbit can be interpreted as the nominal periodic trajectory that results in the smallest integrated cost over the course of the orbit.

Lemma 17. *If Assumptions 8 and 9 are met, and $\theta^i \in \Theta^i_0$ for all $i \in \mathbb{I}_{[0,n_c-1]}$, then $\{\pi_j^*\}$ converges to some orbit π^*_∞.*

Proof. As a consequence of Lemma 14, we have that $\hat{\theta}^i_j \to \hat{\theta}^i_\infty$. Hence, by continuity of (3.3) with respect to $\hat{\theta}^i_j$, we have that $f_{\hat{\theta}^i_j}(x, u, d)$ converges to $f_{\hat{\theta}^i_\infty}(x, u, d)$ for all $i \in \mathbb{I}_{[0,n_c-1]}$, and therefore $\Pi_j \to \Pi_\infty$. Additionally, from Lemma 16 it has been shown that Ω_j converges to some set Ω_∞.

Hence the parameters Π_j and Ω_j of (3.30) converge. Since ϕ and ℓ are continuous, this implies that $\{\pi_j^*\}$ converges to π_∞^*. $\square$

If, in addition, we have that (3.18) holds, then π_∞^* corresponds to the robust optimal periodic orbit of the true system. For use in subsequent analysis, the operator $(\cdot)_{P-1}$ is introduced which extracts the last element of its input argument. Additionally, for simplicity of notation, π_j is used to denote the robust optimal periodic orbit of system (3.4), π_j^*, in the remainder of the chapter, rather than an arbitrary element of Π_j.

3.5.2 The Robust Adaptive Economic MPC problem

The open-loop robust MPC optimization problem is now introduced, which is based on the control strategy proposed in [65]. However, due to the fact that the adaptive scheme leads to modifications of Θ_j^i, new considerations must be made to understand requirements for recursive feasibility and stability of the algorithm, which will be addressed in Sections 3.5.3 and 3.5.4. Whereas a traditional MPC controller would solve an optimization problem at each timestep, it has been shown in [60] that, for periodic systems, stability when using this approach is difficult to guarantee. Rather, the optimization problem is solved every P timesteps starting at time $k = 0$. Let $n_h = K_h P$ with $K_h \in \mathbb{I}_{\geq 1}$ denote the prediction horizon length and

$$\begin{aligned} J_\Omega^{\text{MPC}}(\bar{x}, \bar{\boldsymbol{u}}) &= \sum_{t=0}^{n_h-1} \ell^{\text{int}}(\bar{x}^{\bar{\boldsymbol{u}}}(t, \bar{x}), \bar{u}(t), \Omega) + \bar{V}^f(\bar{x}^{\bar{\boldsymbol{u}}}(n_h, \bar{x})), \\ &= \sum_{t=0}^{K_h-1} \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}(tP, \bar{x}), \mathring{\bar{u}}(tP), \Omega) + \bar{V}^f(\bar{x}^{\bar{\boldsymbol{u}}}(n_h, \bar{x})) \end{aligned}$$

with $\left(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}(0, \bar{x})\right)_{P-1} = \bar{x}$ denote the MPC cost function where $\bar{V}^f : \mathcal{X} \to \mathbb{R}$ is a terminal cost function. For some cost, J, the MPC optimization problem, $P_{\hat{\theta}}^{\text{RAEMPC}}(J, \bar{x})$, is given as

$$\underset{\bar{\boldsymbol{u}} \in (\mathcal{U})^{n_h}}{\text{minimize}} \quad J, \tag{3.31a}$$

$$\text{subject to} \quad \forall t \in \mathbb{I}_{[0,n_h-1]}, i = t \bmod n_c, \tag{3.31b}$$

$$\bar{x}^{\bar{\boldsymbol{u}}}(t+1, \bar{x}) = f_{\hat{\theta}^i}(\bar{x}^{\bar{\boldsymbol{u}}}(t, \bar{x}), \bar{u}(t), 0), \tag{3.31c}$$

$$(\bar{x}^{\bar{\boldsymbol{u}}}(t, \bar{x}), \bar{u}(t)) \in \bar{\mathcal{X}} \times \bar{\mathcal{U}} = \bar{\mathcal{Z}}, \tag{3.31d}$$

$$\bar{x}^{\bar{\boldsymbol{u}}}(n_h, \bar{x}) \in \bar{\mathcal{X}}_j^f, \bar{x}^{\bar{\boldsymbol{u}}}(0, \bar{x}) = \bar{x}, \tag{3.31e}$$

where $\bar{\mathcal{X}}_j^f$ denotes a terminal constraint set.

The notation of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_\Omega^{\text{MPC}}, \bar{x})$ is used to denote the use of system model $f_{\hat{\theta}}$ to set dynamic constraint (3.31c), J_Ω^{MPC} as the cost to be minimized, and $\bar{x}$ as the initial condition of the MPC state prediction. The resulting solution of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ at cycle j is denoted as $\bar{\boldsymbol{u}}_j^*(\bar{x}) = \left(\bar{u}_{0|j}^*(\bar{x}), \ldots, \bar{u}_{n_h-1|j}^*(\bar{x})\right)$ which can be decomposed into P-step inputs $\mathring{\bar{u}}_{tP|j}^*(\bar{x}) = (\bar{u}_{tP|j}^*(\bar{x}), \ldots, \bar{u}_{tP+P-1|j}^*(\bar{x}))$ for $t \in \mathbb{I}_{[0,K_h-1]}$. The corresponding estimated state sequence is denoted as $\left(\bar{x}_{0|j}^*, \ldots, \bar{x}_{n_h|j}^*\right) \triangleq \left(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x})}(0, \bar{x}), \ldots, \bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x})}(n_h, \bar{x})\right)$.

Let $\bar{\boldsymbol{x}}^{cand} = \left(\bar{x}_0^{cand}, \ldots, \bar{x}_{n_h-P}^{cand}\right)$ with

$$\bar{x}_{\tau+1}^{cand} \triangleq f_{\hat{\theta}_{j+1}^i}(\bar{x}_\tau^{cand}, \bar{u}_\tau^{cand}, 0), \quad \bar{x}_0^{cand} = \bar{x}_{j+1}^0$$

denote a candidate state sequence for the first $n_h - P + 1$ timesteps of $P_{\hat{\theta}_{j+1}}^{\text{RAEMPC}}(J_{\Omega_{j+1}}^{\text{MPC}}, \bar{x}_{j+1}^0)$ where $\bar{\boldsymbol{u}}^{cand} = \left(\bar{u}_0^{cand}, \ldots, \bar{u}_{n_h-P-1}^{cand}\right)$ with $\bar{u}_\tau^{cand} \triangleq \phi(\bar{u}_{\tau+P|j}^*, \bar{x}_\tau^{cand}, \bar{x}_{\tau+P|j}^*)$ denotes the elements of the corresponding candidate input sequence. The control algorithm is then given by Algorithm 3.

Algorithm 3 Robust Adaptive EMPC for periodic systems

```
**Initialize:** Parameter uncertainty sets $\Theta_0^a$, $a = \mathbb{I}_{[0,n_c-1]}$, $\beta > 0$, $K_\omega \in (0,1)$, $i = j = k = 0$.
    Feedback law $\phi(\bar{u}, x, \bar{x})$ with associated initial RCI set $\Omega = \Omega_0$ for estimated model parameters
    $\hat{\theta} = \hat{\theta}_0$. $\bar{x}_0^0 = x_0^0 = x_0$.
**while** $k \geq 0$ **do**
    **if** $k = K_1 P, K_1 \in \mathbb{I}_{\geq 0}$ **then**
        Set $\bar{x} = \bar{x}_j^0$ and $t \to 0$
        **if** $e_j^0 \in \Omega_j$ and $(\bar{\boldsymbol{x}}^{cand}, \bar{\boldsymbol{u}}^{cand})$ is feasible for the first $n_h - P$ timesteps of
    $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ with $\bar{x}_{n_h-P}^{cand} \in \bar{\mathcal{X}}_j^f$ **then**
            Set $\hat{\theta} = \hat{\theta}_j$, $\Omega = \Omega_j$, $\bar{\mathcal{Z}} = \bar{\mathcal{Z}}_j$, $\bar{\mathcal{X}}^f = \bar{\mathcal{X}}_j^f$.
        **end if**
        Solve $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$
    **end if**
    Apply $\phi(\bar{u}_{t|j}^*(\bar{x}), x_j^i, \bar{x}_j^i)$ to the true system $x_j^{i+1} = f_{\hat{\theta}}(x_j^i, \phi(\bar{u}_{t|j}^*(\bar{x}), x_j^i, \bar{x}_j^i), d_j^i)$, and simu-
    late the nominal state dynamics as $\bar{x}_j^{i+1} = f_{\hat{\theta}}(\bar{x}_j^i \bar{u}_{t|j}^*(\bar{x}), 0)$.
    Update $\Theta_j^i \to \Theta_{j+1}^i$ according to Algorithm 2.
    $k \to k+1, i \to i+1, t \to t+1$.
    **if** $k = K_2 n_c, K_2 \in \mathbb{I}_{\geq 0}$ **then**
        Identify $\Omega_{j+1}$ and corresponding $\bar{\mathcal{Z}}_{j+1}$, $\bar{\mathcal{X}}_{j+1}^f$.
        $j \to j+1, i \to 0$
    **end if**
**end while**
```

Remark 1. *As noted in [83], a consequence of only reupdating the control input by solving*

$P_{\hat{\theta}}^{\textit{RAEMPC}}(J_{\Omega}^{\textit{MPC}}, \bar{x})$ every P timesteps as in Algorithm 3 is that the system's closed-loop robustness to disturbances is potentially diminished. An alternative strategy is to solve $P_{\hat{\theta}}^{\textit{RAEMPC}}(J_{\Omega}^{\textit{MPC}}, \bar{x})$ at every timestep, but with a periodically-varying horizon length. Specifically, at timestep k, the prediction horizon can be set to $n_h - (k \bmod P)$ and have only the input corresponding to the first timestep of $P_{\hat{\theta}}^{\textit{RAEMPC}}(J_{\Omega}^{\textit{MPC}}, \bar{x})$ applied to the system. From [84], by Bellman's principle of optimality, implementing control in this way does not impact the resulting analysis, but can allow for improved robustness to disturbances. However, for the sake of simplifying notation, subsequent feasibility and stability analysis is performed under the assumption that $P_{\hat{\theta}}^{\textit{RAEMPC}}(J_{\Omega}^{\textit{MPC}}, \bar{x})$ is only solved every P timesteps using a constant prediction horizon length.

Let $V_\delta(x, \bar{x}) \triangleq \|x - \bar{x}\|$ and $z^{\Omega_j} \triangleq r(\Omega_j)$. We then make the following assumption.

Assumption 10. *At cycle j, there exists a compact set $\bar{\mathcal{X}}_j^f \subseteq \bar{\mathcal{X}}_j$ such that $\mathring{\bar{x}}_j^p \subseteq \bar{\mathcal{X}}_j^f$, a feedback law $\mathring{\kappa}_f : \bar{\mathcal{X}}_j^f \to (\bar{\mathcal{U}}_j)^P$, and a continuous terminal cost function $\bar{V}^f : \mathcal{X} \to \mathbb{R}$ such that for all $\mathring{\bar{x}}$ with $(\mathring{\bar{x}})_{P-1} \in \bar{\mathcal{X}}_j^f$ the following conditions hold:*

1. *$\mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}) \in (\bar{\mathcal{U}}_j)^P$,*
2. *$f_{\hat{\theta}}^P(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), 0) \in (\bar{\mathcal{X}}_j^f)^P$.*

Additionally, for $\mathring{\bar{x}}^+ \in \{\mathring{\bar{x}}^+ \in \mathbb{R}^{n_x P} : V_\delta((\mathring{\bar{x}}^+)_{P-1}, (f_{\hat{\theta}_j}^P(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), 0))_{P-1}) \le z^{\Omega_j}\}$ a class $\mathcal{K}_\infty$ function, α_1, exists such that

$$\bar{V}^f\left((\mathring{\bar{x}}^+)_{P-1}\right) - \bar{V}^f((\mathring{\bar{x}})_{P-1}) \le - \sup_{\substack{h \in \mathbb{I}_{\ge j} \\ (\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}}} \left(\mathring{\ell}^{int}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), \Omega_h) - \mathring{\ell}^{int}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j)\right)$$
$$+ \alpha_1\left(|(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}\right).$$

Without loss of generality, assume that $\bar{V}^f(\bar{x}) \ge 0$ for all $\bar{x} \in \bar{\mathcal{X}}_j^f$.

3.5.3 Recursive Feasibility

In this section we outline sufficient conditions that guarantee recursive feasibility of Algorithm 3. As a consequence of updating the estimated model parameters between cycles, the feasibility of $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ is not immediately guaranteed. In other words, solving the MPC problem using a system model based on the current system parameter estimates and RCI error set is not necessarily possible. Rather, it will be demonstrated in Section 3.5.3.1 that the problem $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ is recursively feasible, wherein $\hat{\theta}$ and Ω correspond to some known model parameter estimates and corresponding RCI error set. Then, in Section 3.5.3.2, sufficient conditions are outlined such that feasibility of $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ is guaranteed for the specific parameter estimates $\hat{\theta}_j$ and RCI error set Ω_j.

3.5.3.1 Feasibility of $P_{\hat{\theta}}^{\textbf{RAEMPC}}(J_{\Omega}^{\textbf{MPC}}, \bar{x})$

Before proceeding, we first assume that the MPC problem is initially feasible.

Assumption 11. *Problem $P_{\hat{\theta}_0}^{RAEMPC}(J_{\Omega_0}^{MPC}, \bar{x}(0))$ is feasible for $\bar{x}(0) = \bar{x}_0^0 = x_0^0$.*

Theorem 2. *If $\theta^i \in \Theta_0^i$ for $i \in \mathbb{I}_{[0,n_c-1]}$ and Assumptions 8, 9, 10, and 11 hold with $\bar{\mathcal{X}}_{j+1}^f \supseteq \bar{\mathcal{X}}_j^f$, then Algorithm 3 is recursively feasible.*

Proof. To demonstrate recursive feasibility of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$, we first observe that at a given iteration, two potential outcomes exist:

1. The candidate state and input sequences $\bar{\boldsymbol{x}}^{cand}$ and $\bar{\boldsymbol{u}}^{cand}$ are feasible for the first $n_h - P$ timesteps of $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ with $\bar{x}_{n_h-P}^{cand} \in \bar{\mathcal{X}}_j^f$ and $e_j^0 \in \Omega_j$. In this case, as $\hat{\theta} = \hat{\theta}_j$ and $\Omega = \Omega_j$ according to Algorithm 3, feasibility of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ is maintained as a result of the first and second conditions of Assumption 10 despite the fact that the estimated model parameters have been updated in system dynamics constraint (3.31c).

2. $\bar{\boldsymbol{x}}^{cand}$ and $\bar{\boldsymbol{u}}^{cand}$ are not compatible for $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$, $\bar{x}_{n_h-P}^{cand} \notin \bar{\mathcal{X}}^f$, or $e_j^0 \notin \Omega_j$. Here, per Algorithm 3, $\hat{\theta}$ and Ω are not updated to $\hat{\theta}_j$ and Ω_j and instead maintain their values from the previous iteration. The remainder of the proof will demonstrate that feasibility of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ is still maintained in this case.

For simplicity of notation, assume, without loss of generality, that $P = n_c$ such that if cycle j begins at timestep k, then cycle $j + 1$ begins at timestep $k + P$. Let

$$\bar{\boldsymbol{w}}^k(P) \triangleq \left(\mathring{\bar{w}}^k(P), \mathring{\bar{w}}^k(2P), \ldots, \mathring{\bar{w}}^k(n_h)\right)$$

where, for $\tau \in \{0, P, \ldots, n_h\}$,

$$\mathring{\bar{w}}^k(\tau) \triangleq \begin{cases} (\bar{u}_{\tau|j}^*(\bar{x}_j^0), \ldots, \bar{u}_{\tau+P-1|j}^*(\bar{x}_j^0)), & \tau \leq n_h - P \\ \mathring{\kappa}_f(\bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_j^0)}(n_h, \bar{x}_j^0)), & \tau = n_h \end{cases}$$

denotes a candidate input sequence to be applied starting at timestep $k + P$ based on the solution, $\bar{\boldsymbol{u}}_j^*(\bar{x}_j^0)$, to $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_j^0)$ at timestep k. It is now demonstrated that $\bar{\boldsymbol{w}}^k(P)$ is feasible for $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_{j+1}^0)$ at timestep $k + P$.

The initial condition of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_{j+1}^0)$ at timestep $k + P$ is given as $\bar{x}_{j+1}^0 = \bar{x}_{k+P} = \bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{w}}^k(0)}(P, \bar{x}_j^0)$. Then, at time $t \in \mathbb{I}_{[0,n_h-P]}$ within the prediction horizon, the nominal state resulting from applying input sequence $\bar{\boldsymbol{w}}^k(P)$ to system $f_{\hat{\theta}}$ is given as $\bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{w}}^k(P)}(t, \bar{x}_{j+1}^0)$. Since $\bar{\boldsymbol{w}}^k(0) \in (\bar{\mathcal{U}})^{n_h}$ is feasible for $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_j^0)$ at time k, then $\bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{w}}^k(P)}(t, \bar{x}_{j+1}^0) \in \bar{\mathcal{X}}$.

Since $\bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{u}}^*(\bar{x}_j^0)}(n_h, \bar{x}_j^0) = \bar{x}_{\hat{\theta}}^{\bar{\boldsymbol{w}}^k(P)}(n_h - P, \bar{x}_{j+1}^0) \in \bar{\mathcal{X}}_j^f$ by design with $\bar{\mathcal{X}}_{j+1}^f \supseteq \bar{\mathcal{X}}_j^f$, then $\bar{x}_{j+1}^{\bar{\boldsymbol{w}}^k(P)}(n_h - P, \bar{x}_{j+1}^0) \in \bar{\mathcal{X}}_{j+1}^f$. Assumption 10 then gives that $\bar{x}_{j+1}^{\bar{\boldsymbol{w}}^k(P)}(t, \bar{x}_{j+1}^0) \in \bar{\mathcal{X}}_{j+1}^f$ for $t \in \mathbb{I}_{[n_h-P+1, n_h]}$ such that constraint (3.31e) is satisfied with $\mathring{\bar{w}}^k(n_h) \in (\bar{\mathcal{U}}_{j+1})^P$.

Therefore constraints (3.31c)-(3.31e) are satisfied by input sequence $\bar{\boldsymbol{w}}^k(P)$. Consequently, $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_{j+1}^0)$ is feasible when $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x}_j^0)$ is feasible. Since $P_{\hat{\theta}_0}^{\text{RAEMPC}}(J_{\Omega_0}^{\text{MPC}}, \bar{x}_0^0)$ is feasible from Assumption 11, $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ is recursively feasible by induction. □

Similar to [65], Theorem 2 establishes conditions for recursive feasibility of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ but now allows for, potentially intermittent, updates to the estimated system model to improve state prediction and enable more aggressive control as new knowledge of system behavior is obtained. However, Theorem 2 does not establish the frequency with which updates to $\hat{\theta}$ are performed for use within $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$.

3.5.3.2 Feasibility of $\bar{\boldsymbol{u}}^{cand}$ for $P_{\hat{\theta}_j}^{\textbf{RAEMPC}}(J_{\Omega_j}^{\textbf{MPC}}, \bar{x})$

While conditions for guaranteed recursive feasibility of $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ have been established in Section 3.5.3.1, Theorem 2 does not assess the frequency with which the most recent model parameter estimates can be used for the dynamic model within the MPC optimization problem. This hesitancy to leverage updated system models ensures safe system operation, but comes with the drawback of potentially overconservative control. To address this issue, we seek to identify conditions for which $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ is guaranteed to be feasible such that more aggressive control actions can be safely made.

Assumption 12. *There exists some $\rho_{\bar{\theta}_j} > 0$ such that for all $(\bar{x}^i, \bar{u}) \in \mathcal{Z}$, $(x^i, \phi(\bar{u}, x^i, \bar{x}^i)) \in \mathcal{Z}$, and $\bar{\theta}_j \in \Theta_j$ with $V_\delta(x^i, \bar{x}^i) < z^{\Omega j}$,*

$$V_\delta(x^{i+1}, \bar{x}^{i+1}) \leq \rho_{\bar{\theta}_j} V_\delta(x^i, \bar{x}^i), \tag{3.32}$$

where $x^{i+1} = f_{\bar{\theta}_j}(x^i, \phi(\bar{u}, x^i, \bar{x}^i), 0)$, $\bar{x}^{i+1} = f_{\bar{\theta}_j}(\bar{x}^i, \bar{u}, 0)$.

If there is a $\rho_{\bar{\theta}_j} < 1$ that satisfies condition (3.32), Assumption 12 gives that if the state estimation error is sufficiently small at timestep k, that application of feedback law ϕ will result in a contraction of the state estimation error at timestep $k+1$ for the uncertaintyless system ($d = 0$).

A stronger condition on the selected feedback law is also established.

Assumption 13. *Feedback ϕ is designed such that for all $x_1, x_2 \in \bar{x} \oplus \Omega_j$, $\phi(\bar{u}, x_1, \bar{x}) \in \mathcal{U}$ implies that $\phi(\phi(\bar{u}, x_2, \bar{x}), x_1, x_2) \in \mathcal{U}$.*

Assumption 13 is satisfied if, for instance, the feedback $\phi(\bar{u}, x, \bar{x})$ is of the form $\phi = u + b(x - \bar{x})$.

Define

$$\bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{A}}(\bar{x},\bar{u}) \triangleq \max_{\substack{i\in\mathbb{I}_{[0,n_c-1]}\\ \tilde{\theta}\in B(0,1)\\ a\in\mathcal{A}}} \left(z^{\Theta^i_j}\|G(\bar{x},\bar{u})\tilde{\theta}\| + \|a\| \right). \tag{3.33}$$

For a given state/input pair $(\bar{x},\bar{u})$, $\bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u})$ represents an upper bound on the size of a corresponding disturbance caused by parametric uncertainty and noise. We now show that an upper bound can be placed on all possible disturbances that the system may encounter in a neighborhood around $(\bar{x},\bar{u})$. Define

$$K_{G,\Omega_j} \triangleq \sup_{(x,\bar{x},\bar{u})\in\Psi_{\Omega_j}} \frac{\|G(x,\phi(\bar{u},x,\bar{x})) - G(\bar{x},\bar{u})\|}{\|x-\bar{x}\|} \tag{3.34}$$

where $\Psi_{\Omega_j} \triangleq \{(x,\bar{x},\bar{u}) \in \mathcal{X}\times\mathcal{Z} : (x,\phi(\bar{u},x,\bar{x})) \in \mathcal{Z}, (x-\bar{x}) \in \Omega_j\}$. K_{G,Ω_j} bounds the error between the predicted regressor matrix, $G(\bar{x},\bar{u})$, and the true regressor matrix, $G(x,\phi(\bar{u},x,\bar{x}))$, for any potential true system state within the tube Ω_j centered at $\bar{x}$, and normalized by the size of the state estimation error.

Note: K_{G,Ω_j} is guaranteed to exist if Assumptions 9 and 12 hold as a consequence of the continuity of $G(x,u)$ and compactness of $\mathcal{Z}$. If, in addition, the conditions of Lemma 16 hold, then $K_{G,\Omega_{j+1}} \le K_{G,\Omega_j}$ since $\Omega_{j+1} \subseteq \Omega_j$ and therefore $\Psi_{\Omega_{j+1}} \subseteq \Psi_{\Omega_j}$.

Lemma 18. *Let $\theta^i \in \Theta^i_0$ for all $i \in \mathbb{I}_{[0,n_c-1]}$,* (3.18) *hold, and Assumptions 8, 9, and 12 hold. Define*

$$d^{max}_{\tilde{\Theta}_j,\mathcal{A}}(\bar{x},\bar{u},c) \triangleq \bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{A}}(\bar{x},\bar{u}) + z^{\Theta j}_{max}K_{G,\Omega_j}c. \tag{3.35}$$

Then, for any $(x,\bar{x},\bar{u}) \in \Psi_{\Omega_j}$, $\delta_c \le z^{\Omega j}$ with $V_\delta(x,\bar{x}) \le \delta_c$,

$$d^{max}_{\tilde{\Theta}_j,\mathcal{V}}(x,\phi(\bar{u},x,\bar{x}),c) \le d^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u},c+\delta_c),$$

and

$$d^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u},c) \ge d^{max}_{\tilde{\Theta}_{j+1},\mathcal{V}}(\bar{x},\bar{u},c) + d^{max}_{\tilde{\Theta}_j,\{0\}}(\bar{x},\bar{u},c).$$

Proof. We first demonstrate that $z^{\Theta j}_{max}K_{G,\Omega_j}V_\delta(x,\bar{x}) \ge \bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(x,\phi(\bar{u},x,\bar{x})) - \bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u})$. By invoking the definitions of K_{G,Ω_j} in (3.34), $z^{\Theta j}_{max}$, and $V_\delta(x,\bar{x})$, we have that

$$\begin{aligned} z^{\Theta j}_{max} K_{G,\Omega_j} V_\delta(x,\bar{x}) \geq & \sup_{\substack{i\in\mathbb{I}_{[0,n_c-1]} \\ \tilde{\theta}\in\tilde{\Theta}^i_j}} \|G(x,\phi(\bar{u},x,\bar{x})) - G(\bar{x},\bar{u})\|\|\tilde{\theta}\|, \\ \geq & \sup_{\substack{i\in\mathbb{I}_{[0,n_c-1]} \\ \tilde{\theta}\in\tilde{\Theta}^i_j}} \|G(x,\phi(\bar{u},x,\bar{x}))\tilde{\theta}\| - \sup_{\substack{i\in\mathbb{I}_{[0,n_c-1]} \\ \tilde{\theta}\in\tilde{\Theta}^i_j}} \|G(\bar{x},\bar{u})\tilde{\theta}\|, \\ = & \bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(x,\phi(\bar{u},x,\bar{x})) - \bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u}). \end{aligned} \tag{3.36}$$

Additionally, as $\bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u})$ is affine with respect to $z^{\Theta^i_j}$ in (3.33), we have that

$$\bar{d}^{max}_{\tilde{\Theta}_j,\mathcal{V}}(\bar{x},\bar{u}) = \bar{d}^{max}_{\tilde{\Theta}_{j+1},\mathcal{V}}(\bar{x},\bar{u}) + \bar{d}^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x},\bar{u}). \tag{3.37}$$

Since, as noted earlier, $K_{G,\Omega_{j+1}} \leq K_{G,\Omega_j}$, we also have that

$$z^{\Theta j}_{max} K_{G,\Omega_j} \geq \max_{i\in\mathbb{I}_{[0,n_c-1]}} z^{\Theta^i_{j+1}} K_{G,\Omega_{j+1}} + z^{\Delta\Theta^i_{j+1}} K_{G,\Omega_{j+1}}. \tag{3.38}$$

Equations (3.36), (3.37), and (3.38) mimic conditions in [78, Eqs. (16b)-(16d)], with modifications made due to consideration of the adaptive scheme. The remainder of the proof then follows from [78, Proposition 2]. □

Before outlining conditions for guaranteed feasibility of $P^{\text{RAEMPC}}_{\hat{\theta}_j}(J^{\text{MPC}}_{\Omega_j},\bar{x})$, without loss of generality, we assume as in Theorem 2 that $P = n_c$. From Algorithm 3, we have that $\bar{x}^0_{j+1} = \bar{x}^{\bar{\boldsymbol{u}}^*(\bar{x}^0_j)}_{\hat{\theta}_j}(P,\bar{x}^0_j)$. For $t\in\mathbb{I}_{[0,n_h]}$, let

$$\bar{x}^{nom}_{t+1} \triangleq f_{\hat{\theta}^i_{j+1}}(\bar{x}^{nom}_t,\bar{u}^{nom}_t,0), \quad \bar{x}^{nom}_0 = \bar{x}^0_{j+1} \tag{3.39}$$

denote a candidate state sequence for $P^{\text{RAEMPC}}_{\hat{\theta}_{j+1}}(J^{\text{MPC}}_{\Omega_{j+1}},\bar{x})$ where $\bar{\boldsymbol{u}}^{nom} \triangleq \left(\bar{u}^{nom}_0,\ldots,\bar{u}^{nom}_{n_h-1}\right) = \left(\bar{\boldsymbol{u}}^{cand},\mathring{\kappa}_f(\bar{x}^{nom}_{n_h-P})\right)$ denotes the corresponding candidate input sequence.

Theorem 3. *Let $\theta^i\in\Theta^i_0$ for $i\in\mathbb{I}_{[0,n_c-1]}$. Additionally, let Assumptions 8, 9, 10, 12, and 13 hold. Define*

$$c_{t+1} = \rho_{\hat{\theta}_j} c_t + d^{max}_{\Delta\tilde{\Theta}_j,\{0\}}(\bar{x}^*_{t+P|j},\bar{u}^*_{t+P|j},c_t), \quad c_0 = 0 \tag{3.40}$$

for $t\in\mathbb{I}_{[0,n_h-P]}$. Then, if

$$\Omega_{j+1}\oplus B(0,c_\tau)\subseteq\Omega_j \tag{3.41}$$

for all $\tau \in \mathbb{I}_{[0,n_h-P-1]}$, and $\bar{\mathcal{X}}^f_{j+1}$ is chosen such that

$$\bar{\mathcal{X}}^f_j \oplus B(0, c_{n_h-P}) \subseteq \bar{\mathcal{X}}^f_{j+1} \tag{3.42}$$

is satisfied, then $\bar{\mathbf{u}}^{nom}$ is feasible for $P^{RAEMPC}_{\hat{\theta}_{j+1}}(J^{MPC}_{\Omega_{j+1}}, \bar{x})$.

Proof. We first demonstrate that the distance between $\bar{x}^{nom}_\tau$ and $\bar{x}^*_{\tau+P|j}$ is upper bounded by c_τ for $\tau \in \mathbb{I}_{[0,n_h-P-1]}$ with $\bar{u}^{nom}_\tau \in \bar{\mathcal{U}}_{j+1}$. To show that this property holds at $\tau + 1$, first assume that it holds at step τ according to

$$V_\delta(\bar{x}^{nom}_\tau, \bar{x}^*_{\tau+P|j}) \leq c_\tau. \tag{3.43}$$

We then have that

$$\begin{aligned}
&V_\delta(\bar{x}^{nom}_{\tau+1}, \bar{x}^*_{\tau+P+1|j}) \\
&\quad = V_\delta(f_{\hat{\theta}^i_{j+1}}(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau, 0), f_{\hat{\theta}^i_j}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}, 0)), && \text{(3.44a)}\\
&\quad = \| f_{\hat{\theta}^i_j}(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau, 0) + G(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau)\Delta\hat{\theta}^i_{j+1} - f_{\hat{\theta}^i_j}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}, 0)\|, && \text{(3.44b)}\\
&\quad \leq \| f_{\hat{\theta}^i_j}(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau, 0) - f_{\hat{\theta}^i_j}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}, 0)\| + \|G(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau)\Delta\hat{\theta}^i_{j+1}\|, && \text{(3.44c)}\\
&\quad \leq \rho_{\hat{\theta}_j}\|\bar{x}^{nom}_\tau - \bar{x}^*_{\tau+P|j}\| + \bar{d}^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x}^{nom}_\tau, \bar{u}^{nom}_\tau), && \text{(3.44d)}\\
&\quad \leq (\rho_{\hat{\theta}_j} + z^{\Theta j+1}_{max} K_{G,\Omega_{j+1}})\|\bar{x}^{nom}_\tau - \bar{x}^*_{\tau+P|j}\| + \bar{d}^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}), && \text{(3.44e)}\\
&\quad \leq (\rho_{\hat{\theta}_j} + z^{\Theta j+1}_{max} K_{G,\Omega_{j+1}})c_\tau + \bar{d}^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}), && \text{(3.44f)}\\
&\quad = \rho_{\hat{\theta}_j} c_\tau + d^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}, c_t) = c_{\tau+1}. && \text{(3.44g)}
\end{aligned}$$

Equation (3.44a) arises by invoking (3.31c) and (3.39). Equation (3.44b) arises from the definition of V_δ and (3.27). Equation (3.44c) results from the triangle inequality, while (3.44d) follows from (3.32) and (3.33). Equation (3.44e) results from (3.36) and the definition of $\bar{u}^{nom}_t$. Equation (3.44f) is a result of (3.43), and (3.44g) is given by invoking the definition of $d^{max}_{\Delta\tilde{\Theta}_{j+1},\{0\}}(\bar{x}^*_{\tau+P|j}, \bar{u}^*_{\tau+P|j}, c_\tau)$ in (3.35) and the c_τ dynamics given by (3.40). At $t = 0$, since $\bar{x}^{nom}_0 = \bar{x}^*_{0+P|j}$, we have that $V_\delta(\bar{x}^{nom}_0, \bar{x}^*_{0+P|j}) = c_0 = 0$. Hence, by induction, (3.43) holds for all $\tau \in \mathbb{I}_{[0,n_h-P-1]}$. In other words, $\bar{x}^{nom}_\tau \in \bar{x}^*_{\tau+P|j} \oplus B(0, c_\tau)$. Invoking (3.41) and (3.31d) gives that

$$\begin{aligned}
\bar{x}^{nom}_\tau \oplus \Omega_{j+1} &\subseteq \bar{x}^*_{\tau+P|j} \oplus \Omega_{j+1} \oplus B(0, c_\tau), \\
&\subseteq \bar{x}^*_{\tau+P|j} \oplus \Omega_j \subseteq \mathcal{X}.
\end{aligned} \tag{3.45}$$

Consequently, from the definition of $\bar{\mathcal{X}}_{j+1}$, we have that $\bar{x}^{nom}_\tau \in \bar{\mathcal{X}}_{j+1}$. Additionally, following from (3.45), Assumption 13, and the definitions of $\bar{u}^{nom}_\tau$ and $\bar{\mathcal{U}}_j$, we have that $\phi(\bar{u}^{nom}_\tau, x, \bar{x}^{nom}_\tau) \in \mathcal{U}$

for all $x \in \bar{x}_{\tau}^{nom} \oplus \Omega_{j+1}$. Therefore, $\bar{u}_{\tau}^{nom} \in \bar{\mathcal{U}}_{j+1}$ for all $\tau \in \mathbb{I}_{[0,n_h-P-1]}$.

We now demonstrate that $(\bar{x}_{\tau}^{nom}, \bar{u}_{\tau}^{nom}) \in \bar{\mathcal{Z}}_{j+1}$ for $\tau \in \mathbb{I}_{[n_h-P,n_h-1]}$ and $\bar{x}_{n_h}^{nom} \in \bar{\mathcal{X}}_{j+1}^{f}$. As indicated by (3.44g), we have that $V_\delta(\bar{x}_{n_h-P}^{nom}, \bar{x}_{n_h|j}^{*}) \leq c_{n_h-P}$. Equation (3.31e), gives that $\bar{x}_{n_h|j}^{*} \in \bar{\mathcal{X}}_{j}^{f}$, and therefore $\bar{x}_{n_h-P}^{nom} \in \bar{\mathcal{X}}_{j+1}^{f}$ follows from (3.42). Assumption 10 then gives that $(\bar{u}_{n_h-P}^{nom}, \ldots, \bar{u}_{n_h-1}^{nom}) = \mathring{\kappa}_f(\bar{x}_{n_h-P}^{nom}) \in (\bar{\mathcal{U}}_{j+1})^P$ with $\bar{x}_\tau \in \bar{\mathcal{X}}_{j+1}^{f} \subseteq \bar{\mathcal{X}}_{j+1}$ for $\tau \in \mathbb{I}_{[n_h-P,n_h]}$. Therefore, $(\bar{x}_{\tau}^{nom}, \bar{u}_{\tau}^{nom}) \in \bar{\mathcal{Z}}_{j+1}$ for all $\tau \in \mathbb{I}_{[0,n_h-1]}$ and $\bar{x}_{n_h}^{nom} \in \bar{\mathcal{X}}_{j+1}^{f}$ such that $\bar{\boldsymbol{u}}^{nom}$ is a feasible solution to $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$. □

Note: While meeting conditions (3.41) and (3.42) ensures that $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ is feasible, Theorem 3 does not guarantee that these conditions are satisfied at each iteration. If problem $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ is infeasible, Algorithm 3 dictates that older parameter estimates must be used in the MPC optimization problem, which may result in degraded performance. To mitigate this issue, additional constraints may be introduced in a similar fashion to the strategy given in [78] such that (3.41) and (3.42) are guaranteed to be satisfied at the beginning of each iteration. This technique can enable more aggressive control at the cost of additional controller complexity.

3.5.4 Stability Analysis

Following demonstration of recursive feasibility, conditions for ensuring closed-loop convergence of the system trajectories are now investigated. As a consequence of transforming the nominal system (3.3) to an equivalent P-step system, dissipativity-based approaches leveraged within the economic MPC literature for steady-state convergence analysis are used.

Definition 5. *The distance between the P-step state and input pair $(\mathring{x}, \mathring{u})$ and orbit π_j is given as*

$$|(\mathring{x}, \mathring{u})|_{\pi_j} = \sum_{h=0}^{P-1} |(x^{\mathring{u}}(h, (\mathring{x})_{P-1}), u_h)|_{\pi_j}. \tag{3.46}$$

Assumption 14. *At cycle j, for $t \in \mathbb{I}_{[0,n_h]}$, let*

$$x_{\phi,t+1} \triangleq f_{\hat{\theta}_{j+1}}(x_{\phi,t}, u_{\phi,t}, 0), \quad x_{\phi,0} = x$$

denote a state sequence generated by the feedback input $u_{\phi,\tau} = \phi(\breve{u}_\tau, x_{\phi,\tau}, \breve{x}_\tau)$ for $\tau \in \mathbb{I}_{[0,n_h-P-1]}$ and $(u_{\phi,n_h-P}, \ldots, u_{\phi,n_h-1}) = \mathring{\kappa}_f(\breve{x}_{n_h-P})$ where $\breve{u}_\tau \in \mathcal{U}$ and $\breve{x}_\tau \in \mathcal{X}$ with $\breve{x}_0 = x$ and $\breve{x}_{n_h-P} \in \bar{\mathcal{X}}_{j}^{f}$. In other words, $(x_{\phi,t}, u_{\phi,t})$ is a generalization of $(\bar{x}_{t}^{nom}, \bar{u}_{t}^{nom})$. The corresponding $P-$step system states and inputs are denoted as $\mathring{x}_{\phi,t} \triangleq (\mathring{x}_{\phi,t-P+1}, \ldots, \mathring{x}_{\phi,t})$ and $\mathring{u}_{\phi,t} = (\mathring{u}_{\phi,t}, \ldots, \mathring{u}_{\phi,t+P-1})$. Assume there exists a continuous storage function $\mathring{\lambda} : \mathbb{R}^{n_x P} \to \mathbb{R}$ and $\mathcal{K}_\infty$ function α such that the nominal P-step system at cycle j, $f_{\hat{\theta}_j}^{P}$, is strictly dissipative with respect to associated periodic *orbit* π_j *and integrated stage cost function* $\mathring{\ell}^{int}$*. Specifically, for* $(\mathring{\bar{x}}, \mathring{\bar{u}}) \in (\mathcal{Z})^P$*,*

$$\mathring{\lambda}(f^P_{\hat{\theta}_j}(\mathring{\bar{x}}, \mathring{\bar{u}}, 0)) - \mathring{\lambda}(\mathring{\bar{x}}) \leq s_j(\mathring{\bar{x}}, \mathring{\bar{u}}) - \alpha(|(\mathring{\bar{x}}, \mathring{\bar{u}})|_{\pi_j}), \tag{3.47}$$

where $s_j(\mathring{\bar{x}}, \mathring{\bar{u}})$ *denotes a supply rate given as*

$$s_j(\mathring{\bar{x}}, \mathring{\bar{u}}) = \sup_{\substack{h \in \mathbb{I}_{\geq j} \\ (\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}}} \left(\mathring{\ell}^{int}(\mathring{\bar{x}}, \mathring{\bar{u}}, \Omega_h) - \mathring{\ell}^{int}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j) \right),$$

and that

$$\begin{aligned} s_{j+1}(\mathring{x}_{\phi,\tau P}, \mathring{u}_{\phi,\tau P}) + \mathring{\lambda}(\mathring{x}_{\phi,\tau P}) - \lambda(f^P_{\hat{\theta}_{j+1}}(\mathring{x}_{\phi,\tau P}, \mathring{u}_{\phi,\tau P}, 0)) \\ \leq s_j(\mathring{\breve{x}}_{\tau P}, \mathring{\breve{u}}_{\tau P}) + \mathring{\lambda}(\mathring{\breve{x}}_{\tau P}) - \mathring{\lambda}(f_{\hat{\theta}_j}(\mathring{\breve{x}}_{\tau P}, \mathring{\breve{u}}_{\tau P}, 0)). \end{aligned}$$

Additionally, for $\mathring{\bar{x}}^+ \in \{\mathring{\bar{x}}^+ \in \mathbb{R}^{n_x} : V_\delta((\mathring{\bar{x}}^+)_{P-1}, (f^P_{\hat{\theta}_j}(\mathring{\bar{x}}, \mathring{\kappa}_f(\breve{x}_{n_h-P}), 0))_{P-1}) \leq z^{\Omega_j}\}$*, there exists a* $\mathcal{K}_\infty$ *function* α_2 *such that*

$$\mathring{\lambda}(\mathring{\bar{x}}^+) - \mathring{\lambda}(f^P_{\hat{\theta}_j}(\mathring{\bar{x}}, \mathring{\kappa}_f(\breve{x}_{n_h-P}), 0)) \leq \alpha_2(|(\mathring{\bar{x}}, \mathring{\kappa}_f(\breve{x}_{n_h-P}))|_{\pi_j}), \tag{3.48}$$

and

$$\alpha(a) - \alpha_1(a) - \alpha_2(a) > 0, a > 0, \tag{3.49}$$

$$\alpha_1(|(\mathring{\breve{x}}_{n_h-P}, \mathring{\kappa}_f(\mathring{\breve{x}}_{n_h-P}))|_{\pi_j}) + \alpha_2(|(\mathring{\breve{x}}_{n_h-P}, \mathring{\kappa}_f(\mathring{\breve{x}}_{n_h-P}))|_{\pi_j}) < \alpha(|(\mathring{\bar{x}}, \mathring{\bar{u}})|_{\pi_j}), \quad (\mathring{\bar{x}})_{P-1} \notin \bar{\mathcal{X}}^f_j, \tag{3.50}$$

$$\alpha(|(f_{\hat{\theta}_j}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), 0), \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}) \leq \alpha(|(\mathring{\bar{x}}, \mathring{\bar{u}})|_{\pi_j}), \quad (\mathring{\bar{x}})_{P-1} \in \bar{\mathcal{X}}^f_j \tag{3.51}$$

are satisfied where the inequality in (3.51) *is strict for all* $\mathring{\bar{x}} \neq \mathring{\bar{x}}^p_j$*.*

Given the assumption that $\ell(x, u) \geq 0$, and if Lemma 16 holds such that $\Omega_{j+1} \subseteq \Omega_j$, then $s_j(\mathring{\bar{x}}, \mathring{\bar{u}})$ can be written as

$$\begin{aligned} s_j(\mathring{\bar{x}}, \mathring{\bar{u}}) &= \sup_{(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}} \left(\mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\bar{u}}, \Omega_j) - \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j) \right), \\ &= \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\bar{u}}, \Omega_j) - \inf_{(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}} \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j). \end{aligned} \tag{3.52}$$

To aid in analysis, define the *rotated stage cost* $\mathring{\bar{L}}_j(\mathring{\bar{x}}, \mathring{\bar{u}}) \triangleq s_j(\mathring{\bar{x}}, \mathring{\bar{u}}) + \mathring{\lambda}(\mathring{\bar{x}}) - \mathring{\lambda}(f^P_{\hat{\theta}}(\mathring{\bar{x}}, \mathring{\bar{u}}, 0))$, the *rotated* P*-step terminal cost* $\mathring{\bar{V}}^f(\mathring{\bar{x}}) \triangleq \bar{V}^f((\mathring{\bar{x}})_{P-1}) + \mathring{\lambda}(\mathring{\bar{x}})$, and the *auxiliary objective* $J^{\text{aux}}_j(\bar{x}, \bar{\boldsymbol{u}}) =$

$\sum_{t=0}^{K_h-1} \mathring{\bar{L}}_j(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP,\bar{x}),\mathring{\bar{u}}(tP)) + \mathring{\bar{V}}^f(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}(n_h,\bar{x}))$.

To establish stability of the closed-loop system, we develop a series of lemmas that establish convergence criterion of the nominal system (3.4), and subsequently bound the deviation of the true system (3.3) from the nominal system. This is done by demonstrating that $J_j^{\text{aux}}(\bar{x},\bar{\boldsymbol{u}})$ can be used as a Lyapunov function for the nominal system. While $J_j^{\text{aux}}(\bar{x},\bar{\boldsymbol{u}})$ is not necessarily known by the user and therefore cannot be used within the MPC optimization stage, we first demonstrate that the known function $J_{\Omega_j}^{\text{MPC}}(\bar{x},\bar{\boldsymbol{u}})$ is a sufficient proxy.

Lemma 19. *If $\theta^i \in \Theta_0^i$, (3.18) holds, and Assumptions 8 and 9 are satisfied, then the solutions to $P_{\hat{\theta}}^{RAEMPC}(J_{\Omega_j}^{MPC},\bar{x})$ and $P_{\hat{\theta}}^{RAEMPC}(J_j^{aux},\bar{x})$ are equivalent.*

Proof. We first observe that the constraints of these two optimization problems are identical. Hence, if $J_{\Omega_j}^{\text{MPC}}$ and J_j^{aux} are minimized within the same location over this feasible set, then $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}},\bar{x})$ and $P_{\hat{\theta}}^{\text{RAEMPC}}(J_j^{\text{aux}},\bar{x})$ are equivalent problems.

Expanding $J_j^{\text{aux}}(\bar{x},\bar{\boldsymbol{u}})$ using the definitions of $\mathring{\bar{L}}_j$, $\mathring{\bar{V}}^f$, and, since Lemma 16 holds, the modified definition of s_j in (3.52) gives

$$\begin{aligned}
J_j^{\text{aux}}(\bar{x},\bar{\boldsymbol{u}}) = \sum_{t=0}^{K_h-1} \Big(& \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP,\bar{x}),\mathring{\bar{u}}(tP),\Omega_j) - \inf_{(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p)\in\mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}} \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p,\Omega_j) \\
& + \mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP,\bar{x})) - \mathring{\lambda}(f_{\hat{\theta}}^P(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP,\bar{x}),\mathring{\bar{u}}(tP),0))\Big) \\
& + \bar{V}^f\big((\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(n_h,\bar{x}))_{P-1}\big) + \mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(n_h,\bar{x})), \\
= & J_{\Omega_j}^{\text{MPC}}(\bar{x},\bar{\boldsymbol{u}}) + \mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(0,\bar{x})) - \sum_{t=0}^{K_h-1} \inf_{(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p)\in\mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}} \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p,\Omega_j).
\end{aligned}$$

The second equality holds since $\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP+P,\bar{x}) = f_{\hat{\theta}}^P(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(tP,\bar{x}),\mathring{\bar{u}}(tP),0)$. Because $\mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(0,\bar{x})) - \sum_{t=0}^{K_h-1} \inf_{(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p)\in\mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}} \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p,\mathring{\bar{u}}^p,\Omega_j)$ is not impacted by the choice of $\bar{\boldsymbol{u}}$, the cost functions $J_j^{\text{aux}}(\bar{x},\bar{\boldsymbol{u}})$ and $J_{\Omega_j}^{\text{MPC}}(\bar{x},\bar{\boldsymbol{u}})$ differ only by some constant value. Consequently, problems $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}},\bar{x})$ and $P_{\hat{\theta}}^{\text{RAEMPC}}(J_j^{\text{aux}},\bar{x})$ are equivalent. □

We now demonstrate how the third condition of Assumption 10 can be analogously mapped to the rotated stage cost and rotated P-step terminal cost.

Lemma 20. *Given Assumptions 10 and 14, for all $\mathring{\bar{x}} \in \{\mathring{\bar{x}} \in (\bar{\mathcal{X}}_j)^P : (\mathring{\bar{x}})_{P-1} \in \bar{\mathcal{X}}_j^f\}$ and $\mathring{\bar{x}}^+ \in \{\mathring{\bar{x}}^+ \in \mathbb{R}^{n_x} : V_\delta((\mathring{\bar{x}}^+)_{P-1},(f_{\hat{\theta}_j}^P(\mathring{\bar{x}},\mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}),0))_{P-1}) \leq z^{\Omega_j}\}$, it holds that*

$$\mathring{\bar{V}}^f(\mathring{\bar{x}}^+) - \mathring{\bar{V}}^f(\mathring{\bar{x}}) \leq -\mathring{\bar{L}}_j(\mathring{\bar{x}},\mathring{\kappa}_f((\mathring{\bar{x}})_{P-1})) + \alpha_1(|(\mathring{\bar{x}},\mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}) + \alpha_2(|(\mathring{\bar{x}},\mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}). \tag{3.53}$$

Proof. From Assumption 10 it holds that

$$\begin{aligned}
&\bar{V}^f\big((\mathring{\bar{x}}^+)_{P-1}\big) - \bar{V}^f((\mathring{\bar{x}})_{P-1}) \\
&\quad \leq - \sup_{\substack{h \in \mathbb{I}_{\geq j} \\ (\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}}} \Big(\mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), \Omega_h) - \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j) \Big) + \alpha_1(|(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}).
\end{aligned}$$

Adding $\mathring{\lambda}(f^P_{\hat{\theta}}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), 0)) - \mathring{\lambda}(\mathring{\bar{x}})$ to both sides yields

$$\begin{aligned}
\bar{V}^f\big((\mathring{\bar{x}}^+)_{P-1}\big) &+ \mathring{\lambda}(f^P_{\hat{\theta}}(\mathring{\bar{x}}, \mathring{\kappa}_f(\bar{x}_{P-1}), 0)) - \mathring{\bar{V}}^f(\mathring{\bar{x}}) \\
&\leq - \sup_{\substack{h \in \mathbb{I}_{\geq j} \\ (\mathring{\bar{x}}^p, \mathring{\bar{u}}^p) \in \mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}}} \Big(\mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), \Omega_h) - \mathring{\ell}^{\text{int}}(\mathring{\bar{x}}^p, \mathring{\bar{u}}^p, \Omega_j) \Big) - \mathring{\lambda}(\mathring{\bar{x}}) \\
&\quad + \alpha_1(|(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}))|_{\pi_j}) + \mathring{\lambda}(f^P_{\hat{\theta}}(\mathring{\bar{x}}, \mathring{\kappa}_f((\mathring{\bar{x}})_{P-1}), 0)).
\end{aligned}$$

Applying (3.48) and the definition of $\mathring{\bar{L}}_j$ gives (3.53). □

Let $\bar{\mathcal{X}}_{j,n_h} = \{\bar{x} \in \bar{\mathcal{X}}_j : \exists \bar{\boldsymbol{u}} \in \bar{\mathcal{U}}^{n_h}(\bar{x}) \text{ such that } \bar{x}^{\bar{\boldsymbol{u}}}_{\hat{\theta}}(n_h, \bar{x}) \in \bar{\mathcal{X}}^f_j\}$ denote the set of initial conditions for which $P^{\text{RAEMPC}}_{\hat{\theta}}(J^{\text{MPC}}_{\Omega_j}, \bar{x})$ is feasible. We now show that an input exists such that auxiliary cost J^{aux}_j decreases between iterations.

Lemma 21. *Assume that $\theta^i \in \Theta^i_0$, and Assumptions 8, 9, 10, and 14 hold. Then, for all $\bar{x} \in \bar{\mathcal{X}}_{j,n_h}$ and $j \in \mathbb{I}_{\geq 0}$, it holds at iteration j that*

$$J^{aux}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J^{aux}_j(\bar{x}_{jP}, \bar{\boldsymbol{u}}^*_j(\bar{x}_{jP})) \leq 0$$

where the inequality is strict for any $\alpha(|\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}|_{\pi_j}) \neq 0$.

Proof. The proof is given in Appendix B.3. □

It is now demonstrated that the closed-loop state trajectories of true system (3.3) converge to a tube around periodic orbit $\mathring{\bar{x}}^p_\infty$.

Theorem 4. *Let Assumptions 8, 9, 10, 11, and 14 hold with initial state $x_0 = \bar{x}_0 = x(0)$. Then the true closed-loop state sequence converges asymptotically to the neighborhood $\mathring{\bar{x}}^p_\infty \oplus \Omega_\infty$ about the robust optimal periodic orbit $\mathring{\bar{x}}^p_\infty$.*

Proof. From Lemma 21 we have that

$$J^{\text{aux}}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J^{\text{aux}}_j(\bar{x}_{jP}, \bar{\boldsymbol{u}}^*_j(\bar{x}_{jP})) \leq 0$$

where the inequality is strict for any $\mathring{\bar{x}}_{jP} \neq \mathring{\bar{x}}^p_j$. Let the solution to $P^{\text{RAEMPC}}(J^{\text{aux}}_{j+1}, \bar{x}_{jP+P})$ be denoted as $\bar{\boldsymbol{u}}^*(\bar{x}_{jP+P})$. From Theorem 2 we have that $\bar{\boldsymbol{u}}^{nom}$ is feasible, but potentially sub-optimal, for $P^{\text{RAEMPC}}(J^{\text{aux}}_{j+1}, \bar{x}_{jP+P})$. Then,

$$\begin{aligned} J^{\text{aux}}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^*(\bar{x}_{jP+P})) &\leq J^{\text{aux}}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}), \\ &\leq J^{\text{aux}}_{j}(\bar{x}_{jP}, \bar{\boldsymbol{u}}^*(\bar{x}_{jP})). \end{aligned}$$

Since the rotated stage cost $\mathring{\bar{L}}_j$ and rotated terminal cost $\mathring{\bar{V}}^f_j$ are assumed continuous and $\bar{\mathcal{Z}}$ is compact, the auxiliary cost sequence converges. This implies that $\lim\limits_{j\to\infty} \alpha(|\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}|_{\pi_j}) = 0$. Therefore, from (3.46), this implies that $(\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}) \to \pi_\infty$.

As $x_k = \bar{x}_k + e_k$, Lemma 16 then gives that $x_k \to \mathring{\bar{x}}^p_\infty \oplus \Omega_\infty$. □

3.6 Simulation Example

The RAEMPC algorithm outlined by Algorithm 3 is now applied in simulation to a bilinear system whose dynamics are given by

$$x_{k+1} = -0.15x_k + -0.01x_k u_{1,k} + 0.05u_{2,k} + 0.30\theta^i + v_k \tag{3.54}$$

where $u_k \triangleq \begin{bmatrix} u_{1,k} & u_{2,k} \end{bmatrix}^\top$. The cycle and iteration lengths are $n_c = P = 5$ timesteps with the set of unknown model parameters given by $\theta = \{1.0, 3.9, 2.8, -0.8, -1.9\}$, and v_k is bounded with $v^{max} = 0.1$. This type of system model may be used to represent a mechanical braking system as described in [85] where x_k corresponds to the translational velocity of a wheel, $u_{1,k}$ represents the normal force applied to the brake, and $u_{2,k}$ denotes the engine force. The initial parameter uncertainty sets are given by $\Theta^i_0 = B(0.9, 3.5)$ for $i \in \mathbb{I}_{[0,n_c-1]}$. The state and input constraint sets are given as $\mathcal{X} = \{x : -10 \leq x \leq 10\}$ and $\mathcal{U} = \{u_k : 0 \leq u_{1,k} \leq 30, 0 \leq u_{2,k} \leq 10, u_{1,k}u_{2,k} = 0\}$. The user-defined constants used for the adaptive scheme are chosen as $K_\omega = 0.99$ and $\beta = 1$.

The economic stage cost is given as

$$\ell(x_k, u_k) = \|x_k - x^i_{ref}\|^2 + 0.1\|u_{k,2}\|^2 + 1.5\|x_k - x_{k-1}\|^2$$

where the first term penalizes tracking error with respect to some cyclic signal x_{ref}, the second term establishes a cost on the applied engine force as a proxy penalty on fuel consumption, and the third term penalizes translational acceleration of the wheel. The feedback law is given as $\phi(\bar{u}, x, \bar{x}) = \bar{u}$. Consequently, from [86], the RCI set Ω_j can be given as

$$\Omega_j = \{e : -2z^{\Theta j}_{max} - 0.67 \leq e \leq 2z^{\Theta j}_{max} + 0.67\}.$$

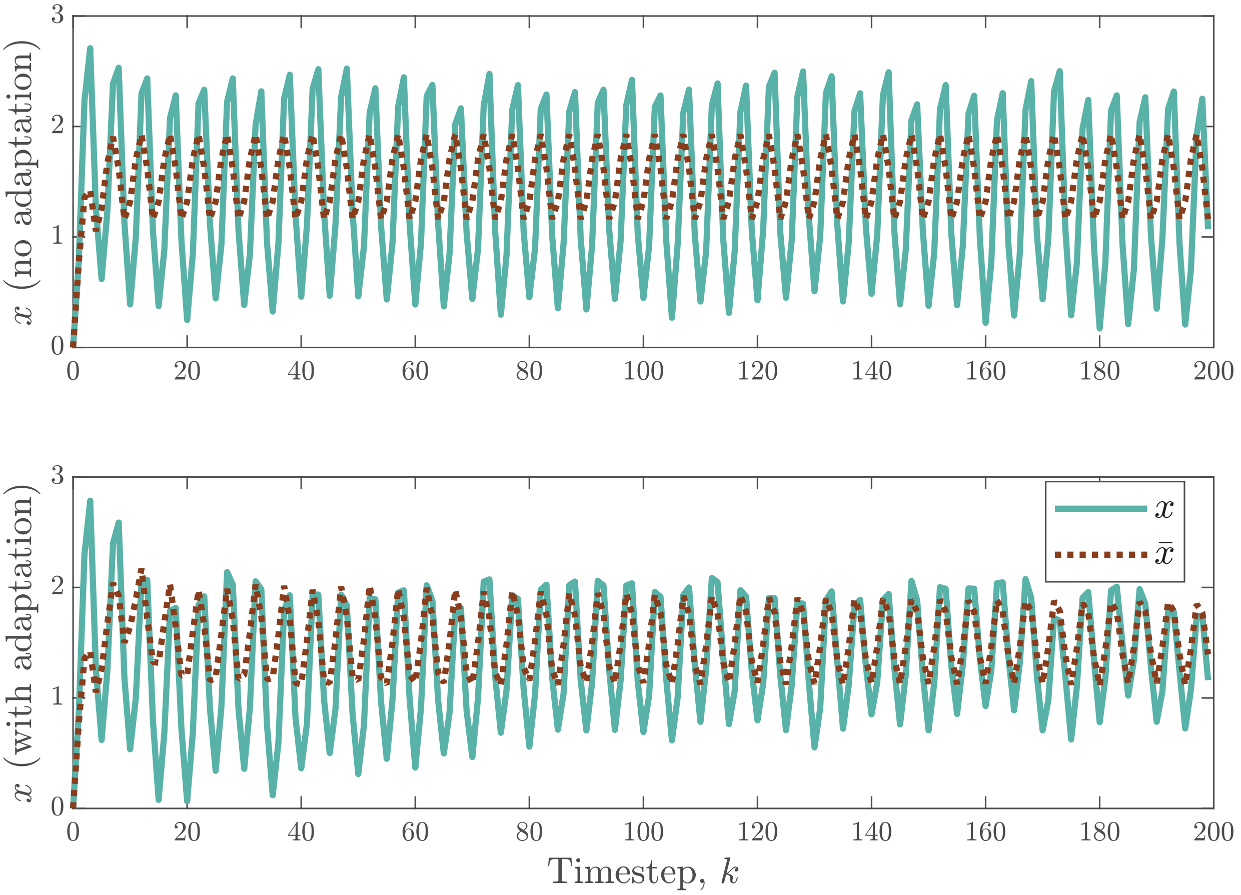


Figure 3.4: State trajectories resulting from application of the REMPC algorithm (top) and the RAEMPC algorithm (bottom). The solid lines denote the true states, while the dotted lines depict the estimated states.

The prediction horizon is set to $n_h = 15$ timesteps and the simulation is conducted over 200 timesteps, or $\frac{200}{P} = 40$ iterations. The Robust Economic Model Predictive Control (REMPC) strategy proposed in [65], which does not incorporate an adaptive model update, is compared to the RAEMPC strategy described by Algorithm 3. The state trajectories resulting from applying the two control strategies are shown in Figure 3.4. Here, by supplementing the robust controller with an adaptive update of the parameter estimates, the average squared state estimation error is reduced from 0.44 in the REMPC case, to 0.16 in the RAEMPC simulation. The performance of the two schemes, as measured by the P-step system cost, is shown in Figure 3.5. Although the two controllers result in similar performance over the first two iterations, the use of an adaptive scheme within the RAEMPC framework enables the system to rapidly improve its performance with an average P-step stage cost of 40.3 over the course of the simulation, in comparison to an average P-step cost of 72.1 when using the REMPC scheme.

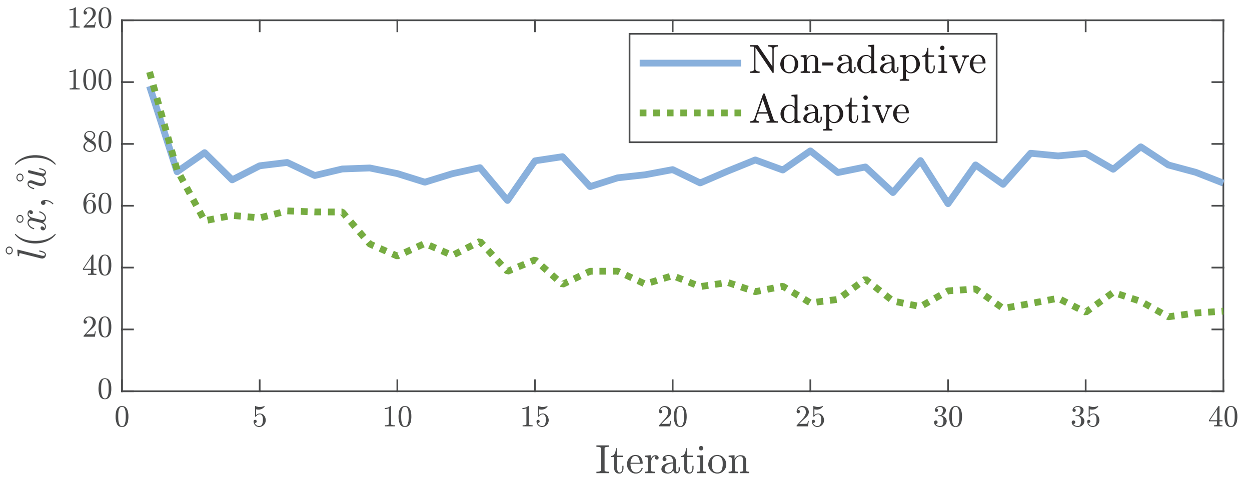


Figure 3.5: P-step stage cost corresponding to the true state-input pairs $(\mathring{x}, \mathring{u})$ for the REMPC scheme (solid blue line) and RAEMPC scheme (dotted green line).

The immediate result of incorporating an adaptive scheme into the controller is observed in Figure 3.6, wherein, by the end of the simulation, the normed parameter estimation error converges and is reduced from its initial value for each intracycle step index. Despite the fact that this convergence is non-monotonic, selective updates to $z^{\Theta_j^i}$ as described by Algorithm 2 ensure that $z^{\Theta_j^i}$ is reduced monotonically while remaining an upper bound on $\|\tilde{\theta}_j^i\|$ as described by Lemma 14.

3.7 Conclusions

In this chapter, an RAEMPC framework is proposed for application to nonlinear systems subject to periodic parametric uncertainty and additive uncertainty. Sufficient conditions for ensuring convergence of the model parameter estimates to their true values, as well as recursive contraction of the uncertainty sets are outlined. Additional conditions that guarantee robust constraint satisfaction, recursive feasibility, and convergence of the true states to a neighborhood around the economically optimal robust periodic orbit are also developed. By combining an adaptive scheme with a robust MPC scheme, the typical overconservatism of robust MPC approaches can be mitigated, as an improved understanding regarding the nature of unknown model parameters facilitates more aggressive and accurate control decisions to be made. This property is demonstrated through a simulation study performed on a model of a mechanical braking system.

The identification of methods to evaluate satisfaction of Assumption 14 in practice remains a point of further investigation.

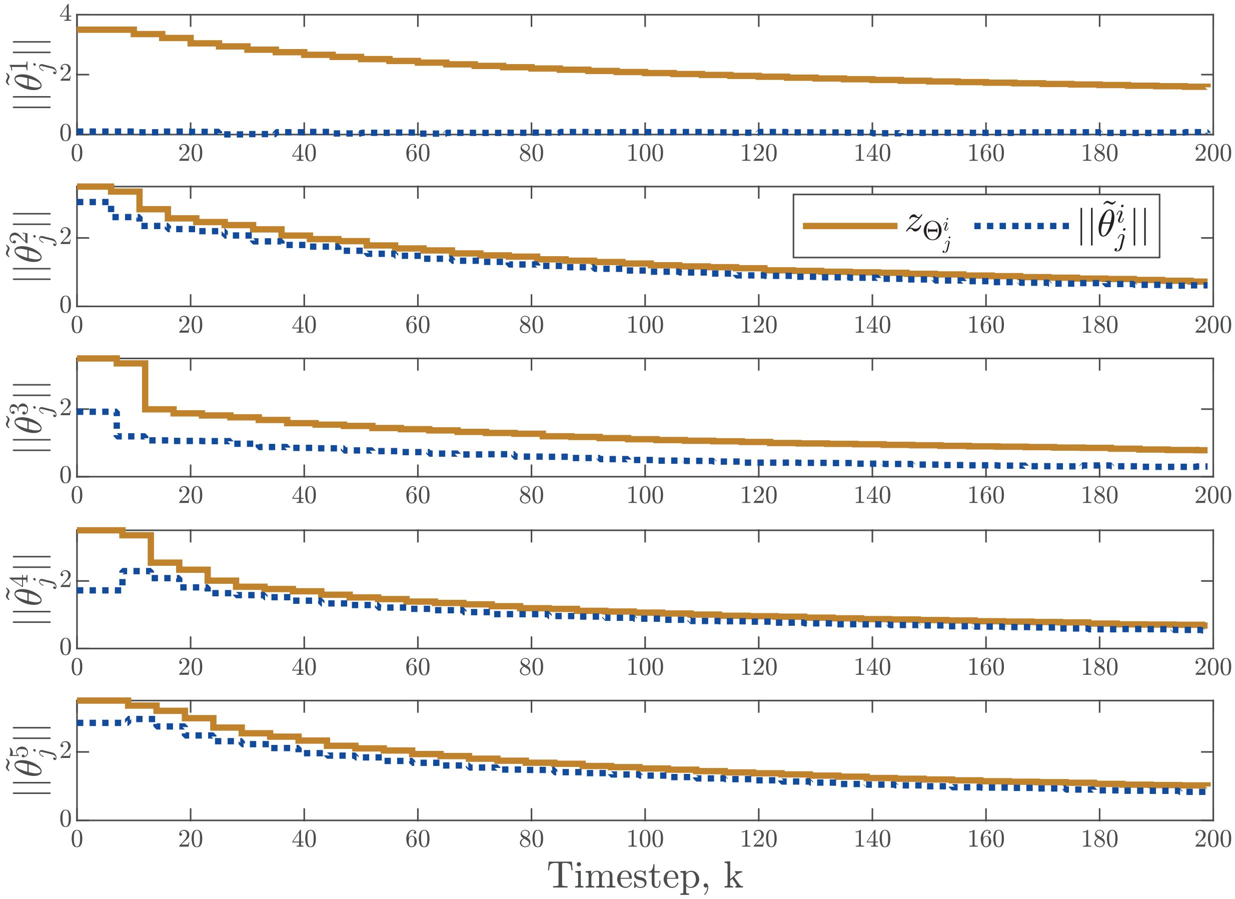


Figure 3.6: Parameter estimate error evolution under the RAEMPC scheme corresponding to each of the unknown model parameters θ^i. Here, the normed estimation error $\|\tilde{\theta}_j^i\|$ is upper bounded by the corresponding uncertainty set radius $z^{\Theta_j^i}$.

CHAPTER 4

Robust Adaptive Economic Iterative Learning Control

4.1 Background and Motivation

As discussed in Chapter 3, mitigating the impacts of inaccurate models or overly conservative control in robust MPC has motivated significant interest in learning-enhanced MPC strategies [87]. In the specific case where state prediction errors manifest as a consequence of parametric uncertainties, augmenting robust MPC strategies with adaptive control schemes has become a popular methodology [56, 66, 71, 76–78, 88]. These methods seek to improve performance through the successive contraction of *parameter uncertainty sets* that the unknown model parameters are known to lie in. Here, by reducing the size of the parameter space that needs to be considered to ensure robust constraint satisfaction, increased confidence in user-estimates of the unknown model parameters can be achieved. Consequently, the overly conservative nature of standard robust MPC methods can be alleviated. Although extensions of these works have enabled time-varying model parameters [73] and economic performance objectives [81] to be addressed, the case where the unknown model parameters vary as a function of the system state has not been explicitly investigated. While the above methodologies can be adjusted to handle this class of uncertainty by treating the effects of state-dependent variations in the model parameters as noise, doing so requires additional conservatism in the control, which may compromise system performance. Moreover, of [56, 66, 71, 73, 76–78, 81, 88], only [66] has examined the case of iterative systems and investigated how the repetitiveness of the system behavior can be leveraged to further encourage learning.

As an alternative to MPC, ILC has become a popular strategy to enable learning for discontinuously operated repetitive systems. Here, whereas adaptive MPC methods have typically sought to improve control through the identification of unknown model parameters, ILC instead seeks to update the control signal directly by exploiting information available from previous trial data

[4]. Although early efforts in ILC were focused on unconstrained systems with tracking objectives [89], recent advancements in ILC have sought to address the existence of system constraints [36, 37], while a select few others have also examined the case of constrained economic control [22, 30, 34, 35]. However, theoretical guarantees on the robust performance of [22, 30, 34, 35] have not been established. Consequently, the performance of these techniques may be compromised when large parametric uncertainties exist.

Improving the robust performance of ILC methods by integrating adaptive methods into the controller design has also been explored in the literature [90–97]. However, each of these works is limited to at least one of the following restrictive system classes:

1. linear systems or input-affine systems,

2. unconstrained systems,

3. noiseless systems,

4. systems with constant unknown parameters, or where the change in the model parameters from iteration to iteration is known *a priori*.

Consequently, while the robust performance of economic ILC methods for constrained systems can suffer due to the existence of parametric uncertainties, existing strategies that utilize adaptation to mitigate these effects have limited applicability.

In this work we propose a strategy that synthesizes tools available from the fields of robust economic MPC and economic ILC, while also enabling adaptation for constrained, discontinuously operated, repetitive systems with both unknown state-dependent model parameters and additive disturbances. The following contributions are then presented in this work:

1. The development of a robust, adaptive, economic ILC (RAEILC) algorithm for nonlinear discontinuously operated systems wherein economic objectives are considered and input/state constraints are robustly enforced through the construction of homothetic tubes. Moreover, in addition to enabling the integration of adaptive parameter uncertainty set updates, learning is further incorporated through the use of historical input and state data from previous task executions. Here, by leveraging knowledge about Lipschitz continuity of the system dynamics, conservativism commonly exhibited by similar robust strategies is further mitigated. Conditions for which recursive feasibility and robust monotonic reductions in the system cost can be guaranteed are also presented.

2. The development of an adaptive method applicable to uncertain nonlinear systems. This uncertainty arises due to the existence of unknown state-varying model parameters that are Lipschitz with respect to the system states, as well as the presence of additive disturbances.

This adaptive scheme updates user-known uncertainty sets wherein the nominal model parameter estimates, as well as an upper bound on the parameter estimate error, are updated. A methodology for integrating this adaptive scheme with the proposed RAEILC algorithm is described.

3. A comparative example of the proposed algorithm against a state-of-the-art adaptive MPC algorithm applied to a simulated pendulum-cart system. An evaluation of the closed-loop system performance and ability of the different adaptive schemes to identify the unknown state-varying model parameters is presented.

To serve as a reference for the reader, a notation guide for the variables used in this chapter is provided in Appendix D.4. The contents of this chapter are in preparation for submission to the International Journal of Robust and Nonlinear Control as [98].

4.2 System description

In this section, the category of systems addressed in this chapter are described. Additionally, the types of uncertainties, constraints, and objectives that these systems are subject to are also presented.

The systems of interest constitute a subclass of batch-process repetitive systems commonly investigated within the ILC literature [24]. Here, the systems iteratively execute a task wherein a sequence of n_t timesteps constitutes an iteration. Moreover, it is assumed that an offline phase exists such that after the conclusion of an iteration, the state of the system is able to be reset to some known value x_0 that serves as the initial condition of the next iteration. The dynamics of the systems considered in this chapter are given as

$$x_{t+1} = f(x_t, u_t) + G(x_t, u_t)\theta(x_t) + d_t \tag{4.1}$$

where $t \in \mathbb{I}_{[0,n_t-1]}$ denotes a timestep index, and $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, and $d \in \mathbb{R}^{n_x}$ represent the system states, inputs, and noise. $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ and $G : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x \times n_\theta}$ denote known, potentially nonlinear, continuous functions. $\theta : \mathbb{R}^{n_x} \to \mathbb{R}^{n_\theta}$ denotes the unknown state-dependent parameters. The states are required to lie within the compact feasible set $\mathcal{X} = \{x : h_x(x) \leq 0\}$ and the inputs are required to remain within the compact set $\mathcal{U} = \{u : h_u(u) \leq 0\}$ where the constraint functions $h = \{h_x \ h_u\} = \{h_1 \dots h_{n_h}\}$ are assumed continuous. Define $\mathcal{Z} = \mathcal{X} \times \mathcal{U}$. The following assumptions are placed on the system dynamics.

Assumption 15. *The functions f, G, and θ are locally Lipschitz continuous over $\mathcal{Z}$ such that for* *any x^a, $x^b \in \mathcal{X}$ and u^a, $u^b \in \mathcal{U}$,*

$$\begin{aligned}
\|f(x^a, u^a) - f(x^b, u^b)\|_P &\leq L_f^x \|x^a - x^b\|_P + L_f^u \|u^a - u^b\|, \\
\|G(x^a, u^a) - G(x^b, u^b)\|_P &\leq L_G^x \|x^a - x^b\|_P + L_G^u \|u^a - u^b\|, \\
\|\theta(x^a) - \theta(x^b)\| &\leq L_\theta^x \|x^a - x^b\|_P
\end{aligned}$$

where L_f^x, L_f^u, L_G^x, L_G^u, L_θ^x and the weighting matrix P are known.

Assumption 16. *The noise is bounded such that $\|d_t\| \in \mathcal{D} = \{d \in \mathbb{R}^{n_x} : \|d\| \leq M_d\}$ where M_d is a known constant.*

Although $\theta(x)$ is an unknown function, we make the following assumption.

Assumption 17. *At each iteration, the user has knowledge of a set-valued function $\c{\Theta}^j(x) : \mathbb{R}^{n_x} \to 2^{\mathbb{R}^{n_\theta}}$ satisfying*

$$\theta(x) \in \c{\Theta}^j(x) = B(\c{\theta}^j(x), z^{\Theta^j}(x)),\ \forall x \in \mathcal{X}, j \in \mathbb{I}_{\geq 0} \tag{4.2}$$

where j denotes an iteration index. Here, the center $\c{\theta}^j(x)$ and radius $z^{\Theta^j}(x)$ of the parametric uncertainty set $\c{\Theta}^j(x)$ are iteration- and state-varying. Moreover, it is assumed that $\c{\theta}^j(x)$ and $z^{\Theta^j}(x)$ are Lipschitz continuous over $\mathcal{X}$.

$\c{\theta}^j(x)$ corresponds to a user estimate of the true model parameters $\theta(x)$ at iteration j. As a consequence of the compactness of $\mathcal{X}$, the condition of Lipschitz continuity in Assumption 17 implies that $\c{\Theta}(x)$ is a compact set for all $x \in \mathcal{X}$. From Assumption 17, the error in this parameter estimate is bounded by the known set $\tilde{\c{\Theta}}^j(x)$ such that

$$\theta(x) - \c{\theta}^j(x) \in \tilde{\c{\Theta}}^j(x) = B(0, z^{\Theta^j}(x)).$$

The dynamic model in (4.1) can be reexpressed in terms of $\c{\theta}^j(x)$ as

$$\begin{aligned}
x_{t+1} &= f(x_t, u_t) + G(x_t, u_t)\c{\theta}^j(x) + d_w(x_t, u_t, d_t), \\
&= f_{\c{\theta}^j}(x_t, u_t) + d_w(x_t, u_t, d_t)
\end{aligned} \tag{4.3}$$

where

$$d_w(x, u, d) \in \mathcal{W}_{\tilde{\c{\Theta}}^j, \mathcal{D}}(x, u) = \{d_w = w + d : w = G(x, u)\tilde{\c{\theta}}, \tilde{\c{\theta}} \in \tilde{\c{\Theta}}^j(x), d \in \mathcal{D}\}. \tag{4.4}$$

Here, $f_{\c{\theta}^j}$ denotes the nominal system model based on the parameter estimate $\c{\theta}^j(x)$ and d_w is a value that combines the noise and the effects of parametric uncertainty at iteration j. While various control methods exist to robustly handle the uncertainty introduced by d_w, the ability of these approaches to maintain a satisfactory level of system performance is closely linked to the characteristics of the set $\mathcal{W}_{\tilde{\Theta}^j,\mathcal{D}}(x,u)$. Specifically, if $\mathcal{W}_{\tilde{\Theta}^j,\mathcal{D}}(x,u)$ is large, the performance of the system is likely to suffer.

This tradeoff between robustness and performance is further accentuated when the system performance is defined by an economic metric rather than a reference tracking objective. Namely, in pure tracking applications the optimal behavior corresponds to system trajectories that track a predefined reference signal regardless of the existence of uncertainty. However, the optimal system behavior when performance is measured by an economic metric may be *a priori* unknown in the presence of uncertainty. In this chapter, performance is given by the economic cost accrued by the system over an iteration. Specifically, the system seeks to minimize costs of the form

$$J = V_f(x_{n_t-1}) + \sum_{t=0}^{n_t-2} \ell(x_t, u_t). \tag{4.5}$$

Assumption 18. *The stage cost $\ell(x,u)$ and terminal cost $V_f(x)$ are continuous over $\mathcal{Z}$. Without loss of generality, we additionally assume that the $\ell(x,u)$ and $V_f(x)$ are positive over $\mathcal{Z}$.*

The combination of economic costs, constraints, and state-dependent uncertainty makes the task of robust performance optimization nontrivial. However, in the following section a control strategy is presented that employs adaptive techniques and cycle-to-cycle learning. In this manner, the proposed strategy aims to eliminate unnecessary compromises in performance that are made to ensure constraint satisfaction.

4.3 Robust Adaptive Economic ILC

The proposed robust adaptive economic ILC framework is now presented. Section 4.3.1 outlines various assumptions that must be met in order for the controller to be successfully applied. The proposed control framework is presented in Section 4.3.2. Theoretical analysis of the RAEILC algorithm is presented in Section 4.3.3, including an assessment of recursive feasibility and robust performance.

4.3.1 Requirements

In this subsection, necessary conditions placed on the system, uncertainty description, and adaptive update to the parametric uncertainty set $\Theta^j(x)$ are outlined.

The construction of a tube that bounds the error in the state trajectory predicted by the nominal model $f_{\utilde{\theta}^j}$ is required in order to robustly satisfy the constraints. While, as in [76, 77, 81], knowledge of the Lipschitz continuity of the system dynamics can be used to design this robust tube, the size of the tube grows exponentially along the prediction horizon, which results in very conservative control for large prediction horizons or for systems with large uncertainties. Consequently, a strategy utilized in [78, 99] is also leveraged here for the construction of the robust tube wherein an assumption on the existence of a stabilizing lower level feedback controller is made. Specifically, it is assumed that system (4.1) is incrementally stabilizable such that, for a known control law, κ, the following assumption holds.

Assumption 19. *A Lyapunov function of the form $V_\delta(x, \bar{x}) = \|x - \bar{x}\|_P$, a control law $\kappa : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}^{n_u}$, and constants δ_{loc}, $\kappa_{max} > 0$, and $\rho_{\utilde{\theta}} \in (0, 1)$ exist such that for all $(\bar{x}, \bar{u}) \in \mathcal{Z}$, $(x, \kappa(x, \bar{x}, \bar{u})) \in \mathcal{Z}$, $V_\delta(x, \bar{x}) \le \delta_{loc}$, and all $\utilde{\theta}(x) \in \utilde{\Theta}^0(x)$:*

$$\|\kappa(x, \bar{x}, \bar{u}) - \bar{u}\| \le \kappa_{max} V_\delta(x, \bar{x}) \tag{4.6}$$

and

$$V_\delta(x^{j+1}, \bar{x}^{j+1}) \le \rho_{\utilde{\theta}} V_\delta(x, \bar{x}) \tag{4.7}$$

where $x^{j+1} = f_{\utilde{\theta}}(x, \kappa(x, \bar{x}, \bar{u}))$ and $\bar{x}^{j+1} = f_{\utilde{\theta}}(\bar{x}, \bar{u})$. We further assume that κ is locally Lipschitz in its arguments over $\mathcal{X} \times \mathcal{Z}$ in that there exist constants L_κ^x, $L_\kappa^{\bar{x}}$, and $L_\kappa^{\bar{u}}$ such that

$$\|\kappa(x^a, \bar{x}^a, \bar{u}^a) - \kappa(x^b, \bar{x}^b, \bar{u}^b)\| \le L_\kappa^x \|x^a - x^b\| + L_\kappa^{\bar{x}} \|\bar{x}^a - \bar{x}^b\| + L_\kappa^{\bar{u}} \|\bar{u}^a - \bar{u}^b\|$$

holds for all x^a, $\bar{x}^a$, x^b, $\bar{x}^b \in \mathcal{X}$ and $\bar{u}^a$, $\bar{u}^b \in \mathcal{U}$.

We now show that [78, Proposition 1] can be extended to the case that the uncertainty set $\utilde{\Theta}(x)$ is state dependent. Here it is demonstrated that any change in the contraction factor $\rho_{\utilde{\theta}^j}$ incurred as a consequence of updating the model parameter estimates can be upper bounded.

Lemma 22. *Suppose that Assumptions 15, 17, and 19 hold with some $\utilde{\theta}^j$, $\Delta\utilde{\Theta}$ satisfying $\utilde{\theta}^j(x) \oplus \Delta\utilde{\Theta}(x) \subseteq \utilde{\Theta}^0(x)$ for all $x \in \mathcal{X}$. Then, for any $\utilde{\theta}^{j+1}(x)$ where $\utilde{\theta}^{j+1}(x) \in \utilde{\theta}^j(x) \oplus \Delta\utilde{\Theta}(x)$ for all $x \in \mathcal{X}$, there exists a constant $L_{\rho,\utilde{\theta},\Delta\utilde{\Theta}} \ge 0$ such that*

$$\rho_{\utilde{\theta}^{j+1}} \le \rho_{\utilde{\theta}^j} + L_{\rho,\utilde{\theta},\Delta\utilde{\Theta}}. \tag{4.8}$$

Proof. Let $\Delta\c{\theta}(x) = \c{\theta}^{j+1}(x) - \c{\theta}^{j}(x)$. Then,

$$
\begin{aligned}
&V_\delta(f_{\c{\theta}^{j+1}}(x,\kappa(x,z,v)), f_{\c{\theta}^{j+1}}(z,v)) \\
&\quad = \|f_{\c{\theta}^j}(x,\kappa(x,z,v)) + G(x,\kappa(x,z,v))\Delta\theta(x) - f_{\theta^j}(z,v) - G(z,v)\Delta\c{\theta}(z)\|_P, \\
&\quad \leq \|f_{\c{\theta}^j}(x,\kappa(x,z,v)) - f_{\c{\theta}^j}(z,v)\|_P + \|G(x,\kappa(x,z,v))\Delta\c{\theta}(x) - G(z,v)\Delta\c{\theta}(z)\|_P, \\
&\quad \overset{(4.7)}{\leq} \rho_{\c{\theta}^j} V_\delta(x,z) + \|G(x,\kappa(x,z,v))\Delta\c{\theta}(x) - G(z,v)\Delta\c{\theta}(z)\|_P, \\
&\quad \leq \rho_{\c{\theta}^j} V_\delta(x,z) + \|G(x,\kappa(x,z,v))\Delta\c{\theta}(x) - G(z,v)\Delta\c{\theta}(x)\|_P \\
&\qquad + \|G(z,v)\Delta\c{\theta}(x) - G(z,v)\Delta\c{\theta}(z)\|_P, \\
&\quad \leq \rho_{\c{\theta}^j} V_\delta(x,z) + \epsilon_{\Delta\c{\Theta}}\|G(x,\kappa(x,z,v)) - G(z,v)\|_P + G^P_{max}\|\Delta\c{\theta}(x) - \Delta\c{\theta}(z)\|
\end{aligned}
$$

where $\epsilon_{\Delta\c{\Theta}} = \max\limits_{\Delta\c{\theta}(x)\in\Delta\c{\Theta}(x), x\in\mathcal{X}} \|\Delta\c{\theta}(x)\|$ and $G^P_{max} = \max\limits_{(z,v)\in\mathcal{Z}} \|G(z,v)\|_P$.

Due to the assumptions on Lipschitz continuity of $G(z,v)$, $\c{\theta}^j$, $\c{\theta}^{j+1}$, and κ, we have that

$$V_\delta(f_{\c{\theta}^{j+1}}(x,\kappa(x,z,v)), f_{\c{\theta}^{j+1}}(z,v)) \leq \rho_{\c{\theta}^j} V_\delta(x,z) + \epsilon_{\Delta\c{\Theta}} L_{G,\kappa} V_\delta(x,z) + G^P_{max} L_{\Delta\c{\theta}} V_\delta(x,z)$$

for some $L_{G,\kappa}$ and $L_{\Delta\c{\theta}}$. Dividing by $V_\delta(x,z)$ on both sides gives the desired result:

$$\rho_{\c{\theta}^{j+1}} \leq \rho_{\c{\theta}^j} + L_{\rho,\c{\theta},\Delta\c{\Theta}}.$$

with $L_{\rho,\c{\theta},\Delta\c{\Theta}} = \epsilon_{\Delta\c{\Theta}} L_{G,\kappa} + G^P_{max} L_{\Delta\c{\theta}}$. □

Let

$$\Psi = \{(x,\bar{x},\bar{u}) \in \mathbb{R}^{n_x} \times \mathcal{Z} | (x,\kappa(x,\bar{x},\bar{u})) \in \mathcal{Z}, V_\delta(x,\bar{x}) \leq \delta_{loc}\}$$

denote the set of true states, nominal states, and nominal inputs wherein the true state and feedback are feasible and the distance between the true and nominal states is sufficiently small such that (4.7) holds.

A scalar function $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(x,u)$ is now introduced which serves as an upper bound on the size of the disturbance signal $d_w(\bar{x},\bar{u},d)$ at the point $(\bar{x},\bar{u}) \in \mathcal{Z}$. We assume that $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}$ meets the following conditions which closely mimic [78, Assumption 6].

Assumption 20. *Suppose there exist set-valued functions $\tilde{\c{\Theta}}^{j+1}$, $\Delta\tilde{\c{\Theta}}$, $\tilde{\c{\Theta}}^j$ satisfying $\tilde{\c{\Theta}}^{j+1}(x) \oplus \Delta\tilde{\c{\Theta}}(x) \subseteq \tilde{\c{\Theta}}^j(x)$, and parameters $\c{\theta}^j$ satisfying $\c{\theta}^j(x) \oplus \tilde{\c{\Theta}}^j(x) \subseteq \Theta^0(x)$ for all $x \in \mathcal{X}$. Then, for some set-valued function $\tilde{\c{\Theta}}(x)$, there exists a function $\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}} : \mathcal{Z} \to \mathbb{R}_{\geq 0}$ and scalar $L_{\tilde{\c{\Theta}}} \geq 0$ such that for all $\breve{x} \in \mathcal{X}$, $(x,\bar{x},\bar{u}) \in \Psi$ and disturbances $d_w \in \mathcal{W}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x},\bar{u})$ the following properties hold:*

$$V_\delta(\breve{x} + d_w, \breve{x}) \le \tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u}), \tag{4.9a}$$

$$\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(x, \kappa(x, \bar{x}, \bar{u})) - \tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u}) \le L_{\tilde{\c{\Theta}}^j} V_\delta(x, \bar{x}), \tag{4.9b}$$

$$\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u}) \ge \tilde{w}_{\tilde{\c{\Theta}}^{j+1},\mathcal{D}}(\bar{x}, \bar{u}) + \tilde{w}_{\Delta\tilde{\c{\Theta}},\{0\}}(\bar{x}, \bar{u}), \tag{4.9c}$$

$$L_{\tilde{\c{\Theta}}^j} \ge L_{\tilde{\c{\Theta}}^{j+1}} + L_{\Delta\tilde{\c{\Theta}}}, \tag{4.9d}$$

$$L_{\rho,\c{\theta},\Delta\c{\Theta}} \le L_{\Delta\tilde{\c{\Theta}}}. \tag{4.9e}$$

Condition (4.9a) requires that the size of the disturbance d_w is upper bounded by $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u})$ where V_δ serves as the distance metric. Condition (4.9b) requires that $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}$ is Lipschitz continuous with respect to the system states. Monotonic reductions in the value of $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u})$ and the Lipschitz constant $L_{\tilde{\c{\Theta}}^j}$ as a result of updating the parameter uncertainty sets are enforced by (4.9c) and (4.9d). Within the context of Lemma 22, Condition (4.9e) states that any increase to the contraction factor $\rho_{\c{\theta}^j}$ resulting from an update to the model parameter estimates is smaller than the corresponding decrease to $L_{\tilde{\c{\Theta}}^j}$. An example definition of $\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}$ such that conditions (4.9a)-(4.9e) hold will be presented in Section 4.4.3.

While $\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u})$ places a bound on the size of the disturbance at $(\bar{x}, \bar{u})$ given parameter uncertainty set $\tilde{\c{\Theta}}$, constraints cannot be robustly enforced using this information alone. Specifically, if the nominal value of the system state does not coincide with the true system state, then the size of d_w may be larger than what is predicted at point $(\bar{x}, \bar{u})$, which may result in a loss of constraint satisfaction. Hence, an augmented version of $\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u})$, denoted by $\tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}, s)$, is introduced as

$$\tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}, s) = \tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}) + L_{\tilde{\c{\Theta}}} s. \tag{4.10}$$

Given Assumptions 19 and 20, two important properties of $\tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}, s)$ are provided by [78, Proposition 2]. First, if $V_\delta(x, \bar{x}) \le \Delta s$ for some $\Delta s \ge 0$, then

$$\tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(x, \kappa(x, \bar{x}, \bar{u}), s) \le \tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}, s + \Delta s). \tag{4.11}$$

for any $(x, \bar{x}, \bar{u}) \in \Psi$. From (4.10), not only does $\tilde{w}_{\delta,\tilde{\c{\Theta}}^j,\mathcal{D}}(\bar{x}, \bar{u}, s)$ provide an upper bound on the size of d_w at $(\bar{x}, \bar{u})$, but by exploiting the Lipschitz continuity assumption placed upon $\tilde{w}_{\tilde{\c{\Theta}}^j,\mathcal{D}}$ in (4.9c), this disturbance bound also holds at all points $(x, \kappa(x, \bar{x}, \bar{u}))$ within a neighborhood of $(\bar{x}, \bar{u})$ that satisfy $V_\delta(x, \bar{x}) \le s$.

Second, suppose there exists some $\widetilde{\c{\Theta}}^j$, $\widetilde{\c{\Theta}}^{j+1}$, $\Delta\tilde{\c{\Theta}}$ satisfying $\widetilde{\c{\Theta}}^{j+1}(x) \oplus \Delta\tilde{\c{\Theta}}(x) \subseteq \tilde{\c{\Theta}}(x) \subseteq$

$\tilde{\Theta}^0(x)$ for all $x \in \mathcal{X}$. Then

$$\tilde{w}_{\delta,\tilde{\Theta}^j,\mathcal{D}}(\bar{x},\bar{u},s) \geq \tilde{w}_{\delta,\tilde{\Theta}^{j+1},\mathcal{D}}(\bar{x},\bar{u},s) + \tilde{w}_{\delta,\Delta\tilde{\Theta},\{0\}}(\bar{x},\bar{u},s). \tag{4.12}$$

Equation (4.12) states that the bound $\tilde{w}_{\delta,\tilde{\Theta}^j,\mathcal{D}}(\bar{x},\bar{u},s)$ is reduced monotonically in response to iterative updates to the parameter uncertainty set.

4.3.2 RAEILC Algorithm

The RAEILC algorithm is now described. The algorithm consists of two stages: 1) an online control signal identification stage, and 2) an offline model parameter uncertainty set update. In this section, the adaptive scheme for updating the parameter uncertainty set is addressed in a general manner to facilitate the utilization of diverse parameter uncertainty methods. A specific method for implementing the parameter estimation updates will be presented in Section 4.4.

Prior to running the control algorithm, constants $c_i \geq 0$, $i \in \{1,\dots,n_h\}$ are computed for each constraint function h_i such that

$$h_i(x,\kappa(x,\bar{x},\bar{u})) - h_i(\bar{x},\bar{u}) \leq c_i V_\delta(x,\bar{x}) \tag{4.13}$$

holds for all $(x,\bar{x},\bar{u}) \in \Psi$. Additionally, it is assumed that the user has knowledge of a feasible input and state trajectory of the true system before the algorithm is initiated such that the following condition holds.

Assumption 21. *At iteration* 0*, there exists a known input sequence* $\overset{\therefore}{\boldsymbol{u}}^0 = \{\overset{\therefore}{u}{}^0_0,\dots,\overset{\therefore}{u}{}^0_{n_t-2}\} \in (\mathcal{U})^{n_t-1}$*, and corresponding state sequence* $\overset{\therefore}{\boldsymbol{x}}^0 = \{\overset{\therefore}{x}{}^0_0,\dots,\overset{\therefore}{x}{}^0_{n_t-1}\} \in (\mathcal{X})^{n_t}$ *such that* $\overset{\therefore}{x}{}^0_{t+1} = f_\theta(\overset{\therefore}{x}{}^0_t,\overset{\therefore}{u}{}^0_t) + d^0_t$ *for* $t \in \{0,\dots,n_t-2\}$ *and some* $\{d^0_0,\dots,d^0_{n_t-2}\} \in (\mathcal{D})^{n_t-1}$ *with* $\overset{\therefore}{x}{}^0_0 = x_0$*.*

Assumption 21 may be satisfied if historical data has been available where a feasible input sequence has been applied to the system, and the resulting state sequence remains in $\mathcal{X}$ for the entirety of an iteration. $\overset{\therefore}{\boldsymbol{u}}^j$ and $\overset{\therefore}{\boldsymbol{x}}^j$ are respectively termed the *benchmark input and state sequences* at iteration j.

Given compactness of the uncertainty set $\Theta^j(x)$ and continuity of $G(x,u)$, define $\theta^j_{max} = \max\limits_{\theta\in\Theta^j(x),x\in\mathcal{X}} \|\theta(x)\|$ and $G_{max} = \max\limits_{(x,u)\in\mathcal{Z}} \|G(x,u)\|$. Define $\boldsymbol{x}^j = \{x^j_0,\dots,x^j_{n_t-1}\}$ and $\boldsymbol{u}^j = \{u^j_0,\dots,u^j_{n_t-2}\}$ where x^j_t and u^j_t denote the measured state and input at timestep t of iteration j. Moreover, for some signal a, let $\delta a^j_t = a^j_t - \overset{\therefore}{a}{}^j_t$. In the following, we will demonstrate that an upper bound can be placed on $\|\delta x^j_t\|_P$.

Lemma 23. *Suppose that Assumptions 15, 16, 17, and 19 hold. For some* $(x_t, \bar{x}_t, \bar{u}_t) \in \Psi$*, let* $u_t = \kappa(x_t, \bar{x}_t, \bar{u}_t)$ *and define the coefficients* $\rho_{q,\Theta}$*,* $\rho_{s,\Theta}$*, and* $\rho_{\bar{u},\overset{\therefore}{u},\Theta}$ *as*

$$\begin{aligned} \rho_{q,\Theta} &= L_f^x + L_G^x \theta_{max} + L_\theta^x G_{max}, \\ \rho_{s,\Theta} &= \kappa_{max}(L_f^u + L_G^u \theta_{max}), \\ \rho_{\bar{u},\overset{\therefore}{u},\Theta} &= L_f^u + L_G^u \theta_{max}. \end{aligned} \tag{4.14}$$

Then, for some $(\overset{\therefore}{x}_t, \overset{\therefore}{u}_t) \in \mathcal{Z}$ *and* d_t*,* $\overset{\therefore}{d}_t \in \mathcal{D}$*,*

$$\|\delta x_{t+1}\|_P \leq \rho_{q,\Theta} \|\delta x_t\|_P + \rho_{s,\Theta} V_\delta(x_t, \bar{x}_t) + \rho_{\bar{u},\overset{\therefore}{u},\Theta} \|\bar{u}_t - \overset{\therefore}{u}_t\| + 2\sigma^{max}(P) M_d. \tag{4.15}$$

Proof. From (4.1) we have that

$$\begin{aligned} \|\delta x_{t+1}\|_P =& \|f(x_t, u_t) + g(x_t, u_t)\theta(x_t) + d_t - f(\overset{\therefore}{x}_t, \overset{\therefore}{u}_t) - g(\overset{\therefore}{x}_t, \overset{\therefore}{u}_t)\theta(\overset{\therefore}{x}_t) - \overset{\therefore}{d}_t\|_P, \\ \leq& \|f(x_t, u_t) - f(\overset{\therefore}{x}_t, \overset{\therefore}{u}_t)\|_P + \|g(x_t, u_t)\theta(x_t) - g(\overset{\therefore}{x}_t, \overset{\therefore}{u}_t)\theta(\overset{\therefore}{x}_t)\|_P + \|d_t - \overset{\therefore}{d}_t\|_P \end{aligned}$$

Applying Assumption 15 and substituting in the feedback law for u_t yields

$$\begin{aligned} \|\delta x_{t+1}\|_P \leq& L_f^x \|x_t - \overset{\therefore}{x}_t\|_P + L_f^u \|u_t - \overset{\therefore}{u}_t\| + L_G^x \|x_t - \overset{\therefore}{x}_t\|_P \theta_{max} + L_G^u \|u_t - \overset{\therefore}{u}_t\| \theta_{max} \\ &+ L_\theta^x G_{max} \|x_t - \overset{\therefore}{x}_t\|_P + 2\sigma^{max}(P) M_d, \\ =& (L_f^x + L_G^x \theta_{max} + L_\theta^x G_{max}) \|\delta x_t\|_P + (L_f^u + L_G^u \theta_{max}) \|\kappa(x_t, \bar{x}_t, \bar{u}_t) - \overset{\therefore}{u}_t\| \\ &+ 2\sigma^{max}(P) M_d. \end{aligned}$$

Invoking the triangle inequality and Assumption 19 gives

$$\begin{aligned} \|\delta x_{t+1}\|_P \leq& (L_f^x + L_G^x \theta_{max} + L_\theta^x G_{max}) \|\delta x_t\|_P \\ &+ (L_f^u + L_G^u \theta_{max}) \Big(\|\kappa(x_t, \bar{x}_t, \bar{u}_t) - \bar{u}_t\| + \|\bar{u}_t - \overset{\therefore}{u}_t\| \Big) + 2\sigma^{max}(P) M_d, \\ \overset{(4.6)}{\leq}& (L_f^x + L_G^x \theta_{max} + L_\theta^x G_{max}) \|\delta x_t\|_P \\ &+ (L_f^u + L_G^u \theta_{max}) \Big(\kappa_{max} V_\delta(x_t, \bar{x}_t) + \|\bar{u}_t - \overset{\therefore}{u}_t\| \Big) + 2\sigma^{max}(P) M_d \end{aligned}$$

which yields the desired result. □

Definition 6. *For some* $(\bar{x}, \bar{u}) \in \mathcal{Z}$ *and* $s \geq 0$*, let the s-tube* $\Omega_{\mathcal{X}}(\bar{x}, s)$ *and* $\Omega_{\mathcal{U}}(\bar{u}, s)$ *be given as*

$$\Omega_{\mathcal{X}}(\bar{x}, s) = \{x \in \mathcal{X} : V_{\delta}(x, \bar{x}) \leq s\}, \tag{4.16}$$

$$\Omega_{\mathcal{U}}(\bar{u}, s) = \{u \in \mathcal{U} : \|u - \bar{u}\| \leq \kappa_{max} s\}. \tag{4.17}$$

The $s-$tube $\Omega_{\mathcal{X}}(\bar{x}, s)$ therefore denotes the set of states within a neighborhood of radius s around the point $\bar{x}$, while, following from Assumption 4.6, $\Omega_{\mathcal{U}}(\bar{u}, s)$ corresponds to the set of all potential inputs that may be applied by the feedback law $\kappa(x, \bar{x}, \bar{u})$ for some $x \in \Omega_{\mathcal{X}}(\bar{x}, s)$.

Given the uncertainty in the system model and the existence of noise, the nominal model of the system may not provide an accurate prediction of the cost accrued by the true system. Consequently, seeking to minimize the nominal cost directly can result in unexpectedly poor performance. Inspired by [62], we instead minimize an *integrated cost*, J^{int}, accumulated between timesteps t and $n_t - 1$ which is given as

$$J^{int}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{s}, t) = V_f^{int}(x_{n_t}, s_{n_t}) + \sum_{k=0}^{n_t-1-t} \ell^{int}(x_{t+k}, u_{t+k}, s_{t+k})$$

where

$$V_f^{int}(\bar{x}, s) = \int_{\Omega_{\mathcal{X}}(\bar{x}, s)} V_f(\bar{x}) d\bar{x}, \qquad \ell^{int}(\bar{x}, \bar{u}, s) = \int_{\Omega_{\mathcal{X}}(\bar{x}, s)} \int_{\Omega_{\mathcal{U}}(\bar{u}, s)} \ell(\bar{x}, \bar{u}) d\bar{u}\, d\bar{x}.$$

Here, the integrated terminal cost V_f^{int} and integrated stage cost ℓ^{int} produce an average of any potential costs incurred by the true system.

The control signal is calculated by solving a shrinking horizon optimization problem, denoted as $R(\Theta^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$, at each timestep in an iteration where $R(\Theta, x, \dot{\boldsymbol{x}}, \dot{\boldsymbol{u}}, t)$ is given according to

$R(\c{\Theta}, x, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}, t)$:

$$\underset{\bar{u}_{\cdot|t}, a_{s,\cdot|t}, a_{q,\cdot|t}}{\text{minimize}} \quad V_f^{int}(\bar{x}_{n_t-t|t}, s_{n_t-t|t}) + \sum_{k=0}^{n_t-1-t} \ell^{int}(\bar{x}_{k|t}, \bar{u}_{k|t}, s_{k|t}) \tag{4.18a}$$

$$\text{subject to} \quad \bar{x}_{k+1|t} = f_{\c{\theta}}(\bar{x}_{k|t}, \bar{u}_{k|t}),\ \bar{x}_{0|t} = x, \tag{4.18b}$$

$$w_{k|t} = \tilde{w}_{\delta, \tilde{\c{\Theta}}, \mathcal{D}}(\bar{x}_{k|t}, \bar{u}_{k|t}, s_{k|t}), \tag{4.18c}$$

$$s_{k+1|t} = \rho_{\c{\theta}} s_{k|t} + w_{k|t},\ s_{0|t} = 0, \tag{4.18d}$$

$$s_{k|t} \leq \delta_{loc}, \tag{4.18e}$$

$$\begin{gathered} q_{k+1|t} = \rho_{q,\c{\Theta}} q_{k|t} + \rho_{s,\c{\Theta}} s_{k|t} + \rho_{\bar{u}, \overset{\therefore}{u}, \c{\Theta}} \|\bar{u}_{k|t} - \overset{\therefore}{u}_{t+k}\| + 2\sigma^{max}(P) M_d, \\ q_{0|t} = \|\overset{\therefore}{x}_t - x\|_P, \end{gathered} \tag{4.18f}$$

$$a_{s,k|t}(h_i(\bar{x}_{k|t}, \bar{u}_{k|t}) + c_i s_{k|t}) \leq 0, \tag{4.18g}$$

$$a_{q,k|t}(h_i(\overset{\therefore}{x}_{t+k}, \overset{\therefore}{u}_{t+k}) + c_i q_{k|t}) \leq 0, \tag{4.18h}$$

$$a_{s,k|t} + a_{q,k|t} \geq 1,\ a_{s,k|t} \in [0,1],\ a_{q,k|t} \in [0,1], \tag{4.18i}$$

$$k = 0, \ldots, n_t - t,\ \ i = 1, \ldots, n_h. \tag{4.18j}$$

The solution of $R(\c{\Theta}^j, x_t^j, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$ is denoted as $(\bar{u}_{\cdot|t}^{j*}, a_{s,\cdot|t}^{j*}, a_{q,\cdot|t}^{j*})$. We correspondingly denote $\bar{x}_{\cdot|t}^{j*}$, $w_{\cdot|t}^{j*}$, $s_{\cdot|t}^{j*}$, and $q_{\cdot|t}^{j*}$ from (4.18b), (4.18c), (4.18d), and (4.18f). Let $\boldsymbol{J}_{\c{\Theta}, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}}^{int*}$ denote the optimal cost of $R(\c{\Theta}, x_0, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}, 0)$. The input is then set as $u_t^j = \bar{u}_t^j = \bar{u}_{0|t}^{j*}$. Note that $u_t^j = \bar{u}_t^j$ since in (4.18b) we have that $\bar{x}_{0|t} = x_t^j$ and $\kappa(x, x, \bar{u}) = \bar{u}$ according to (4.6). The system dynamics and initial condition of the optimization problem are enforced by constraint (4.18b) using the parameter estimate $\c{\theta}(x)$. The evolution of the $s-$tube is defined by constraints (4.18c) and (4.18d) based on the amount of parametric uncertainty given by $\tilde{\c{\Theta}}$, and is restricted by constraint (4.18e) to be sufficiently small. Leveraging the results of Lemma 23, constraint (4.18f) is used to construct an additional tube, termed the $q-$tube, that places a bound on the true system states over the prediction horizon based on the benchmark system behavior. Robust satisfaction of the state and input constraints given by $h(x, u)$ is ensured by constraints (4.18g)-(4.18i). The decision variables, $a_{s,\cdot|t}$ and $a_{q,\cdot|t}$ serve as coefficients to the h_i functions and (4.18i) ensures that, for each timestep in the prediction horizon, at least one of $a_{s,k|t}$ or $a_{q,k|t}$ is positive. The rationale here is that both the $s-$tube, which is centered around the nominal state trajectory, and the $q-$tube, which is centered around the benchmark state trajectory, serve as acceptable bounds on the true system trajectory. In fact, the true system states lie within the intersection of the $s-$tube and $q-$tube. Consequently, if either the $s-$tube or $q-$tube can ensure that $h(x, u) \leq 0$, then robust feasibility can be ensured.

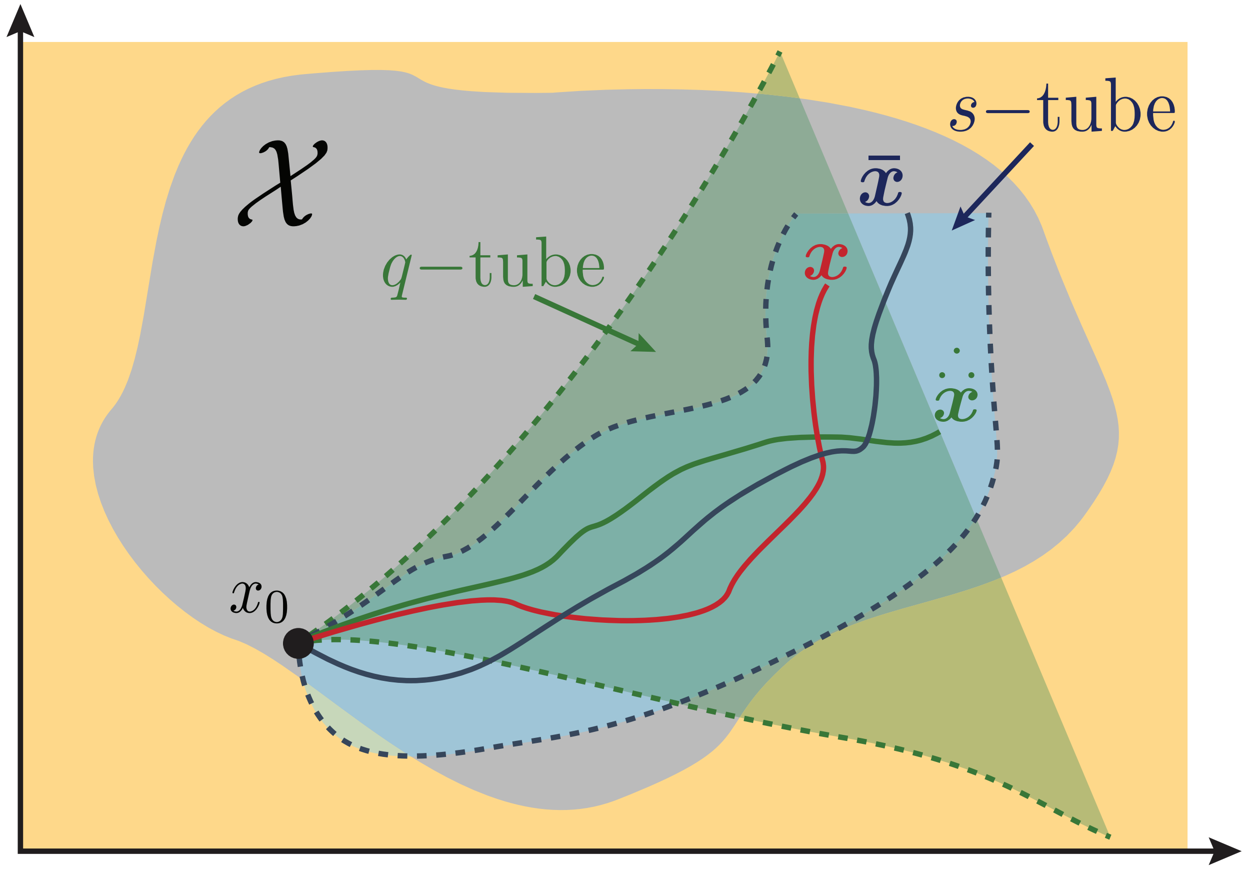


Figure 4.1: Starting from the initial state x_0, the true system state trajectory $\boldsymbol{x}$ (red line) lies within the intersection of the $s-$tube (light blue area), which is centered around the nominal state trajectory $\boldsymbol{x}$ (blue line), and the $q-$tube (green area), which is centered around the benchmark state trajectory $\overset{\therefore}{\boldsymbol{x}}$ (green line). Robust constraint satisfaction is ensured by enforcing that at least one of the $s-$tube or $q-$tube lies within the feasible region $\mathcal{X}$ (gray area) at each timestep.

This feature of $R(\text{Θ̧}^j, x_t^j, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$ allows the RAEILC algorithm to exploit the repetitive behavior of the system and enables more aggressive control while maintaining system constraint satisfaction. A graphical depiction of the $s-$ and $q-$tubes is shown in Fig. 4.1.

The RAEILC framework is given in Algorithm 4

4.3.3 Theoretical Analysis

An analysis of recursive feasibility of $R(\text{Θ̧}^j, x_t^j, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$, convergence of the benchmark input $\overset{\therefore}{\boldsymbol{u}}^j$, and of the robust optimal cost $\boldsymbol{J}^{int*}_{\text{Θ̧}^{j+1}, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j}$ is now presented.

Algorithm 4 Robust Adaptive Economic ILC

```
Given: Parameter uncertainty set $\underset{\smile}{\Theta}^0(x)$, feedback law $\kappa(x, \bar{x}, \bar{u})$, benchmark state and input sequences $\dot{\boldsymbol{x}}^0$ and $\dot{\boldsymbol{u}}^0$, and user-defined parameter $\gamma_{\boldsymbol{u}^j, \dot{\boldsymbol{u}}^j} > 0$. Initialize $j = t = 0$. Set the initial state $x_0^0 = x_0$.
while True do
    Calculate $\underset{\smile}{\theta}^j_{max}$ and the contraction factor $\rho_{\underset{\smile}{\theta}^j}$.
    for $t \leq n_t - 2$ do
        Solve $R(\underset{\smile}{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$.
        Apply the input $u_t^j = \bar{u}_{0|t}^{j*}$ to system (4.1).
        $t \to t + 1$.
    end for
    Run the adaptive update to identify $\underset{\smile}{\Theta}^{j+1}(x)$.
    if $R(\underset{\smile}{\Theta}^{j+1}, x_0, 0, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, 0)$ is feasible and

        $J^{int*}_{\underset{\smile}{\Theta}^{j+1}, \boldsymbol{x}^j, \boldsymbol{u}^j} + \gamma_{\boldsymbol{u}^j, \dot{\boldsymbol{u}}^j} \|\boldsymbol{u}^j - \dot{\boldsymbol{u}}^j\| \leq J^{int*}_{\underset{\smile}{\Theta}^{j+1}, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j}$     (4.19)

     then
        Set $(\dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}) = (\boldsymbol{x}^j, \boldsymbol{u}^j)$.
    else
        Set $(\dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}) = (\dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j)$.
    end if
    $j \to j + 1, t \to 0$
end while
```

4.3.3.1 Recursive Feasibility

Recursive feasibility of $R(\underset{\smile}{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$ will be examined in two parts. First, time domain recursive feasibility is addressed where it is shown that feasibility of $R(\underset{\smile}{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$ implies feasibility of $R(\underset{\smile}{\Theta}^j, x_{t+1}^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t+1)$. Second, iteration domain recursive feasibility is demonstrated wherein feasibility of $R(\underset{\smile}{\Theta}^j, x_0, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, 0)$ implies feasibility of $R(\underset{\smile}{\Theta}^{j+1}, x_0, \dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}, 0)$ The synthesis of these two results therefore yields feasibility of $R(\underset{\smile}{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$ at all times and at each iteration.

Lemma 24. *Let Assumptions 15, 16, 17, 19, 20, and 21 hold and suppose* $R(\underset{\smile}{\Theta}^j, x_0, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, 0)$ *is feasible. Then* $R(\underset{\smile}{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$ *is feasible and* $(x_t^j, u_t^j) \in \mathcal{Z}$ *for all* $t \in \{0, \ldots, n_t - 1\}$ *at iteration* j.

Proof. We break the proof into parts as follows.

Part 1: Candidate solution To produce a candidate solution at time $t + 1$, the feedback law κ is applied to the optimal solution obtained at timestep t according to

$$\bar{u}^{j}_{k|t+1} = \kappa(\bar{x}^{j}_{k|t+1}, \bar{x}^{j*}_{k+1|t}, \bar{u}^{j*}_{k+1|t}). \tag{4.20}$$

Assumption 20 gives that

$$V_\delta(x^j_{t+1}, \bar{x}^{j*}_{1|t}) \overset{(4.3)}{=} V_\delta(\bar{x}^{j*}_{1|t} + d_w, \bar{x}^{j*}_{1|t}) \overset{(4.9a),(4.10)}{\leq} \tilde{w}_{\delta,\tilde{\underset{\varsigma}{\Theta}}^j,\mathcal{D}}(\bar{x}_{0|t}, \bar{u}_{0|t}, 0) \overset{(4.18c),(4.18d)}{=} w^{j*}_{0|t} \overset{(4.18e)}{\leq} \delta_{loc}.$$

The difference between the state solution at timestep t and the candidate state trajectory can be bounded as

$$V_\delta(\bar{x}^{j}_{k|t+1}, \bar{x}^{j*}_{k+1|t}) \overset{(4.7)}{\leq} \rho^k_{\underset{\varsigma}{\theta}^j} w^{j*}_{0|t} \leq \delta_{loc},\ k = 0, \dots, n_t - t. \tag{4.21}$$

Part 2: Tube dynamics We show that the following inequalities hold for $k = 0, \dots, n_t - t$:

$$s^j_{k|t+1} \leq s^{j*}_{k+1|t} - \rho^k_{\underset{\varsigma}{\theta}^j} w^{j*}_{0|t}, \tag{4.22}$$

$$w^j_{k|t+1} \leq w^{j*}_{k+1|t}, \tag{4.23}$$

$$q^j_{k|t+1} \leq q^{j*}_{k+1|t}. \tag{4.24}$$

1) For $k = 0$: Inequality (4.22) is satisfied with

$$s_{0|t+1} \overset{(4.18d)}{=} 0 \overset{(4.18d)}{=} s^{j*}_{1|t} - w^{j*}_{0|t}.$$

To demonstrate (4.23) at $k = 0$, consider $s_{0|t+1} \overset{(4.18d)}{=} 0 \leq s^{j*}_{1|t} \leq \delta_{loc}$, and

$$V_\delta(\bar{x}^j_{0|t+1}, \bar{x}^{j*}_{1|t}) \leq w^{j*}_{0|t} \overset{(4.18d)}{=} s^{j*}_{1|t} - s_{0|t+1}.$$

Eq. (4.11) then gives that

$$w^j_{0|t+1} \overset{(4.18c)}{=} \tilde{w}_{\delta,\tilde{\underset{\varsigma}{\Theta}}^j,\mathcal{D}}(\bar{x}^j_{0|t+1}, \bar{u}^j_{0|t+1}, s_{0|t+1}) \overset{(4.11)}{\leq} \tilde{w}_{\delta,\tilde{\underset{\varsigma}{\Theta}}^j,\mathcal{D}}(\bar{x}^{j*}_{1|t}, \bar{u}^{j*}_{1|t}, s^{j*}_{1|t}) \overset{(4.18c)}{=} w^{j*}_{1|t}.$$

Meanwhile, (4.24) is satisfied at $k = 0$ as a result of the following

$$
\begin{aligned}
q^j_{0|t+1} &\overset{(4.18\text{f})}{=} \|\delta x^j_{t+1}\|_P, \\
&\overset{(4.15)}{\leq} \rho_{q,\underaccent{\check}{\Theta}^j}\|\delta x^j_t\|_P + \rho_{s,\underaccent{\check}{\Theta}^j} V_\delta(x^j_t, \bar{x}^{j*}_{0|t}) + \rho_{\bar{u},\dot{u},\underaccent{\check}{\Theta}^j}\|\bar{u}^{j*}_{0|t} - \dot{u}^j_t\| + 2\sigma^{max}(P)M_d, \\
&\overset{(4.18\text{b})}{=} \rho^j_q\|\delta x^j_t\|_P + \rho^j_{\bar{u},u}\|\bar{u}^{j*}_{0|t} - \dot{u}^j_t\| + 2\sigma^{max}(P)M_d, \\
&\overset{(4.18\text{f})}{=} q^{j*}_{1|t}.
\end{aligned}
$$

2) For $k \geq 0$: From (4.18d), the tube size s satisfies

$$
s^j_{k+1|t+1} \overset{(4.18\text{d})}{=} \rho_{\underaccent{\check}{\theta}^j} s^j_{k|t+1} + w^j_{k|t+1} \overset{(4.22),(4.23)}{\leq} \rho_{\underaccent{\check}{\theta}^j} s^{j*}_{k+1|t} - \rho^{k+1}_{\underaccent{\check}{\theta}^j} w^{j*}_{0|t} + w^{j*}_{k+1|t} \overset{(4.18\text{d})}{=} s^{j*}_{k+2|t} - \rho^{k+1}_{\underaccent{\check}{\Theta}^j} w^{j*}_{0|t}.
$$

Consider $s^j_{k+1|t+1} \overset{(4.22)}{\leq} s^{j*}_{k+2|t} \overset{(4.18\text{e})}{\leq} \delta_{loc}$. Then, from (4.11), we have that $w^j_{k+1|t+1} \leq w^{j*}_{k+2|t}$. For the $q-$tube, we achieve the desired property through the following steps.

$$
\begin{aligned}
q^j_{k+1|t+1} &\overset{(4.18\text{f}),(4.20)}{=} \rho_{q,\underaccent{\check}{\Theta}^j} q^j_{k|t+1} + \rho_{s,\underaccent{\check}{\Theta}^j} s^j_{k|t+1} + \rho_{\bar{u},\dot{u},\underaccent{\check}{\Theta}^j}\|\kappa(\bar{x}^j_{k|t+1}, \bar{x}^{j*}_{k+1|t}, \bar{u}^{j*}_{k+1|t}) - \dot{u}^j_{t+k+1}\| \\
&\qquad + 2\sigma^{max}(P)M_d, \\
&\overset{(4.6)}{\leq} \rho_{q,\underaccent{\check}{\Theta}^j} q^j_{k|t+1} + \rho_{s,\underaccent{\check}{\Theta}^j} s^j_{k|t+1} + \rho_{\bar{u},\dot{u},\underaccent{\check}{\Theta}^j}\left(\kappa_{max} V_\delta(\bar{x}^j_{k|t+1}, \bar{x}^{j*}_{k+1|t}) + \|\bar{u}^{j*}_{k+1|t} - \dot{u}^j_{t+k+1}\|\right) \\
&\qquad + 2\sigma^{max}(P)M_d, \\
&\overset{(4.14)}{\leq} \rho_{q,\underaccent{\check}{\Theta}^j} q^j_{k|t+1} + \rho_{s,\underaccent{\check}{\Theta}^j}\left(s^j_{k|t+1} + V_\delta(\bar{x}^j_{k|t+1}, \bar{x}^{j*}_{k+1|t})\right) + \rho_{\bar{u},\dot{u},\underaccent{\check}{\Theta}^j}\|\bar{u}^{j*}_{k+1|t} - \dot{u}^j_{t+k+1}\| \\
&\qquad + 2\sigma^{max}(P)M_d, \\
&\overset{(4.21),(4.22),(4.24)}{\leq} \rho_{q,\underaccent{\check}{\Theta}^j} q^{j*}_{k+1|t} + \rho_{s,\underaccent{\check}{\Theta}^j}\left(s^j_{k+1|t} - \rho^k_{\underaccent{\check}{\theta}^j} w^{j*}_{0|t} + \rho^k_{\underaccent{\check}{\theta}^j} w^{j*}_{0|t}\right) + \rho_{\bar{u},\dot{u},\underaccent{\check}{\Theta}^j}\|\bar{u}^{j*}_{k+1|t} - \dot{u}^j_{t+k+1}\| \\
&\qquad + 2\sigma^{max}(P)M_d, \\
&\overset{(4.18\text{f})}{=} q^j_{k+2|t}.
\end{aligned}
$$

Part 3: Constraint satisfaction We consider the candidate values $a^s_{k|t+1} = a^{s*}_{k+1|t}$ and $a^q_{k|t+1} = a^{q*}_{k+1|t}$ for $k = 0, \ldots, n_t - 1 - t$. Consequently, constraint (4.18i) is trivially satisfied at time $t + 1$. We have that

$$
\begin{aligned}
h_i(\bar{x}^j_{k|t+1}, \bar{u}^j_{k|t+1}) + c_i s^j_{k|t+1} &\overset{(4.13)}{\leq} h_i(\bar{x}^{j*}_{k+1|t}, \bar{u}^{j*}_{k+1|t}) + c_i V_\delta(\bar{x}^j_{k|t+1}, \bar{x}^{j*}_{k+1|t}) + c_i s^j_{k|t+1}, \\
&\overset{(4.21),(4.22)}{\leq} h_i(\bar{x}^{j*}_{k+1|t}, \bar{u}^{j*}_{k+1|t}) + c_i s^{j*}_{k+1|t}.
\end{aligned}
$$

Thus, (4.18g) gives

$$a^{s}_{k|t+1}\left(h_i(\bar{x}^{j}_{k|t+1}, \bar{u}^{j}_{k|t+1}) + c_i s^{j}_{k|t+1}\right) \leq a^{s*}_{k+1|t}\left(h_i(\bar{x}^{j*}_{k+1|t}, \bar{u}^{j*}_{k+1|t}) + c_i s^{j*}_{k+1|t}\right) \leq 0.$$

Inequality (4.22) ensures that (4.18e) holds.

Additionally, we have that

$$a^{q}_{k|t+1}\left(h_i(x^{j-1}_{t+k+1}, u^{j-1}_{t+k+1}) + c_i q^{j}_{k|t+1}\right) \overset{(4.24)}{\leq} a^{q*}_{k+1|t}\left(h_i(x^{j-1}_{t+k+1}, u^{j-1}_{t+k+1}) + c_i q^{j*}_{k+1|t}\right) \overset{(4.18h)}{\leq} 0.$$

The simultaneous satisfaction of constraints (4.18g), (4.18h), (4.18i) ensures that $(x^{j}_{t+1}, u^{j}_{t}) \in \mathcal{Z}$ for all $t = 0, \ldots n_t - 1$. □

Now that conditions for recursive feasibility in the time domain have been established, we then show that the updates to the estimated model parameters between task cycles do not result in a loss of feasibility of $R(\underset{,}{\Theta}^{j}, x_0, \dot{\boldsymbol{x}}^{j}, \dot{\boldsymbol{u}}^{j}, 0)$ over the iteration domain. This requires the following requirement to be placed on the adaptive scheme and initial problem feasibility.

Assumption 22. *The parameter estimates* $\underset{,}{\theta}^{j}$ *and* $\underset{,}{\theta}^{j+1}$ *and the set-valued functions* $\tilde{\underset{,}{\Theta}}^{j}$, $\tilde{\underset{,}{\Theta}}^{j+1}$, *and* $\Delta\tilde{\underset{,}{\Theta}}^{j+1}$ *that are generated by the adaptive scheme satisfy* $\underset{,}{\theta}^{j+1}(x) \oplus \tilde{\underset{,}{\Theta}}^{j+1}(x) \subseteq \underset{,}{\theta}^{j}(x) \oplus \tilde{\underset{,}{\Theta}}^{j}(x)$ *and* $\tilde{\underset{,}{\Theta}}^{j+1}(x) \oplus \Delta\tilde{\underset{,}{\Theta}}^{j+1}(x) \subseteq \tilde{\underset{,}{\Theta}}^{j}(x)$ *for all* $x \in \mathcal{X}$.

Assumption 23. *At iteration* 0 *and timestep* 0, $R(\underset{,}{\Theta}^{j}, x_0, \dot{\boldsymbol{x}}^{0}, \dot{\boldsymbol{u}}^{0}, 0)$ *is feasible.*

Lemma 25. *Let Assumptions 15, 16, 17, 19, 20, 21, 22, and 23 hold. Then* $R(\underset{,}{\Theta}^{j}, x^{j}_{0}, \dot{\boldsymbol{x}}^{j}, \dot{\boldsymbol{u}}^{j}, 0)$ *is feasible for all* $j \in \mathbb{I}_{\geq 0}$.

Proof. By nature of Algorithm 4, feasibility of $R(\underset{,}{\Theta}^{j+1}, x_0, \dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}, 0)$ is enforced at any iteration wherein an update to $(\dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1})$ has occurred. Hence, in the remainder of the proof, only the case where $(\dot{\boldsymbol{x}}, \dot{\boldsymbol{u}}) = (\dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}) = (\dot{\boldsymbol{x}}^{j}, \dot{\boldsymbol{u}}^{j})$ needs to be considered.

At time $t = 0$ of iteration $j + 1$, let

$$\begin{aligned} \bar{u}^{j+1}_{k|0} &= \kappa(\bar{x}^{j+1}_{k|t}, \bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}), \\ a^{j+1}_{s,k+1|0} &= a^{j*}_{s,k+1|0}, \\ a^{j+1}_{q,k+1|0} &= a^{j*}_{q,k+1|0}. \end{aligned} \tag{4.25}$$

denote a candidate solution to $R(\underset{,}{\Theta}^{j+1}, x_0, \dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}, 0)$, where, due to the parameter change $\Delta\underset{,}{\theta}^{j+1}$, the nominal model predicts

$$\bar{x}^{j+1}_{k+1|t} \overset{(4.18b)}{=} f_{\underset{,}{\theta}^{j+1}}(\bar{x}^{j+1}_{k|t}, \bar{u}^{j+1}_{k|t}) \overset{(4.3)}{=} f_{\underset{,}{\theta}^{j}}(\bar{x}^{j+1}_{k|t}, \bar{u}^{j+1}_{k|t}) + G(\bar{x}^{j+1}_{k|t}, \bar{u}^{j+1}_{k|t})\Delta\underset{,}{\theta}^{j+1}(\bar{x}^{j+1}_{k|t}).$$

We break up the proof into several parts. First, we show that the difference between $\bar{x}^{j*}_{\cdot|0}$ and $\bar{x}^{j+1}_{\cdot|0}$ can be bounded by some quantity $\tilde{s}^{j+1}_{k+1|0}$ such that

$$V_\delta(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}) \leq \tilde{s}^{j+1}_{k|0},\ k = 0, \ldots, n_t. \tag{4.26}$$

Here, we define

$$\tilde{s}^{j+1}_{0|0} = 0,\ \tilde{s}^{j+1}_{k+1|0} = \rho_{\c{\theta}^j} \tilde{s}^{j+1}_{k|0} + \tilde{w}_{\delta, \Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, \tilde{s}^{j+1}_{k|0}). \tag{4.27}$$

As $\bar{x}^{j+1}_{0|0} = \bar{x}^{j*}_{0|0} = x_0$ from (4.18b), (4.26) is satisfied at $k = 0$.

To show that (4.26) is satisfied at $k + 1$, assume that (4.26) holds at step k. Then,

$$\begin{aligned}
V_\delta(\bar{x}^{j+1}_{k+1|0}, \bar{x}^{j*}_{k+1|0}) &\overset{(4.18b)}{=} V_\delta(f_{\c{\theta}^{j+1}}(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}), f_{\c{\theta}^j}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0})), \\
&\leq V_\delta(f_{\c{\theta}^j}(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}), f_{\c{\theta}^j}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0})) \\
&\qquad + V_\delta(\bar{x}^{j*}_{k+1|0} + G(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}) \Delta\c{\theta}^{j+1}(\bar{x}^{j+1}_{k|0}), \bar{x}^{j*}_{k+1|0}), \\
&\overset{(4.7)(4.9a)}{\leq} \rho_{\c{\theta}^j} V_\delta(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}) + \tilde{w}_{\Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}), \\
&\overset{(4.9b)}{\leq} (\rho_{\c{\theta}^j} + L_{\Delta\tilde{\c{\Theta}}^{j+1}}) V_\delta(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}) + \tilde{w}_{\Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}), \\
&\overset{(4.26),(4.10)}{\leq} \rho_{\c{\theta}^j} \tilde{s}^{j+1}_{k|0} + \tilde{w}_{\delta, \Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, \tilde{s}^{j+1}_{k|0}), \\
&\overset{(4.27)}{\leq} \tilde{s}^{j+1}_{k+1|0}
\end{aligned}$$

where the second inequality arises from the triangle inequality.

We then show that the state $\bar{x}^{j+1}_{k|0}$ with a corresponding tube of radius $s^{j+1}_{k|0}$ lies inside the tube centered at $\bar{x}^{j*}_{k|0}$ with radius $s^{j*}_{k|0}$. We first observe the following

$$\begin{aligned}
w^{j+1}_{k|0} &\overset{(4.18c)}{=} \tilde{w}_{\delta, \tilde{\c{\Theta}}^{j+1}, \mathcal{D}}(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}, s^{j+1}_{k|0}) \overset{(4.11),(4.26)}{\leq} \tilde{w}_{\delta, \tilde{\c{\Theta}}^{j+1}, \mathcal{D}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}), \\
&\overset{(4.12)}{\leq} \tilde{w}_{\delta, \tilde{\c{\Theta}}^{j}, \mathcal{D}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}) - \tilde{w}_{\delta, \Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}), \\
&\overset{(4.10),(4.18c)}{=} w^{j*}_{k|0} + L_{\tilde{\c{\Theta}}^j}(s^{j+1}_{k|0} - s^{j*}_{k|0} + \tilde{s}^{j+1}_{k|0}) - \tilde{w}_{\delta, \Delta\tilde{\c{\Theta}}^{j+1}, \{0\}}(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}).
\end{aligned} \tag{4.28}$$

This will enable us to demonstrate that

$$s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0} \leq 0,\ k = 0, \ldots, n_t. \tag{4.29}$$

Note that from (4.18d) and (4.27) $s^{j+1}_{0|0} = s^{j*}_{0|0} = \tilde{s}^{j+1}_{0|0} = 0$ such that (4.29) is satisfied at $k = 0$.

Suppose at prediction step k that (4.29) is satisfied. Then,

$$
\begin{aligned}
& s^{j+1}_{k+1|0} + \tilde{s}^{j+1}_{k+1|0} - s^{j*}_{k+1|0} \\
& \quad \overset{(4.18\text{d}),(4.27)}{=} \rho_{\c{\theta}^{j+1}} s^{j+1}_{k|0} + w^{j+1}_{k|0} + \rho_{\c{\theta}^{j}} (\tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}) + \tilde{w}_{\delta,\Delta\tilde{\c{\Theta}}^{j+1},\{0\}} (\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, \tilde{s}^{j+1}_{k|0}) - w^{j*}_{k|0}, \\
& \quad \overset{(4.10)}{=} \rho_{\c{\theta}^{j+1}} s^{j+1}_{k|0} + w^{j+1}_{k|0} + \rho_{\c{\theta}^{j}} (\tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}) + \tilde{w}_{\delta,\Delta\tilde{\c{\Theta}}^{j+1},\{0\}} (\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j*}_{k|0}) - w^{j*}_{k|0} \\
& \qquad + L_{\Delta\tilde{\c{\Theta}}^{j+1}} (\tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}), \\
& \quad \overset{(4.8),(4.9\text{e})}{\leq} (\rho_{\c{\theta}^{j}} + L_{\Delta\tilde{\c{\Theta}}^{j+1}}) (s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}) + w^{j+1}_{k|0} + \tilde{w}_{\delta,\Delta\tilde{\c{\Theta}}^{j+1},\{0\}} (\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j*}_{k|0}) - w^{j*}_{k|0}, \\
& \quad \overset{(4.28)}{\leq} (\rho_{\c{\theta}^{j}} + L_{\Delta\tilde{\c{\Theta}}^{j+1}} + L_{\tilde{\c{\Theta}}^{j}}) (s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}) - \tilde{w}_{\delta,\Delta\tilde{\c{\Theta}}^{j+1},\{0\}} (\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}) \\
& \qquad + \tilde{w}_{\delta,\Delta\tilde{\c{\Theta}}^{j+1},\{0\}} (\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}, s^{j*}_{k|0}), \\
& \quad \overset{(4.10)}{=} (\rho_{\c{\theta}^{j}} + L_{\tilde{\c{\Theta}}^{j}}) (s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0} - s^{j*}_{k|0}) \overset{(4.29)}{\leq} 0.
\end{aligned}
$$

Hence, (4.29) holds by induction.

We then show that constraint (4.18g) continues to be satisfied. Here,

$$
\begin{aligned}
a^{j+1}_{s,k+1|t} (h_i(\bar{x}^{j+1}_{k|0}, \bar{u}^{j+1}_{k|0}) + c_i s^{j+1}_{k|0}) & \overset{(4.13),(4.26)}{\leq} a^{j+1}_{s,k+1|t} (h_i(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}) + c_i \tilde{s}^{j+1}_{k|0} + c_i s^{j+1}_{k|0}), \\
& \overset{(4.29)(4.25)}{\leq} a^{j*}_{s,k+1|t} (h_i(\bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}) + c_i s^{j*}_{k|0}) \overset{(4.18\text{g})}{\leq} 0.
\end{aligned}
$$

We now show that (4.18h) continues to be satisfied. For this, we first demonstrate that

$$
q^{j+1}_{k|0} \leq q^{j*}_{k|0} \tag{4.30}
$$

for all $k = 0, \ldots, n_t - 1$. As $x^{j+1}_0 = x^j_0 = \overset{\therefore}{x}^j_0 = x_0$, $\|\delta x^{j+1}_0\| = \|\delta x^j_0\| = 0$. Consequently, from (4.18f), $q^{j+1}_{0|0} = q^{j*}_{0|0} = 0$ such that (4.30) holds at $k = 0$. Suppose that (4.30) holds for $k \in \{0, \ldots, n_t - 2\}$. Then,

$$
q^{j+1}_{k+1|0} \overset{(4.18\text{f})}{=} \rho_{q,\c{\Theta}^{j+1}} q^{j+1}_{k|0} + \rho_{s,\c{\Theta}^{j+1}} s^{j+1}_{k|0} + \rho_{\bar{u},\overset{\therefore}{u},\c{\Theta}^{j+1}} \|\bar{u}^{j+1}_{k|0} - \overset{\therefore}{u}_k\| + 2\sigma^{max}(P) M_d.
$$

Since $\Theta^{j+1}(x) \subseteq \Theta^{j}(x)$ for all $x \in \mathcal{X}$, $\theta^{j+1}_{max} \leq \theta^{j}_{max}$. Therefore,

$$
\begin{aligned}
q^{j+1}_{k+1|0} &\overset{(4.14)}{\leq} \rho_{q,\Theta^j} q^{j+1}_{k|0} + \rho_{s,\Theta^j} s^{j+1}_{k|0} + \rho_{\bar{u},\dot{u},\Theta^j} \|\bar{u}^{j+1}_{k|0} - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\overset{(4.29),(4.25)}{\leq} \rho_{q,\Theta^j} q^{j+1}_{k|0} + \rho_{s,\Theta^j} (s^{j*}_{k|0} - \tilde{s}^{j+1}_{k|0}) + \rho_{\bar{u},\dot{u},\Theta^j} \|\kappa(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}) - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\leq \rho_{q,\Theta^j} q^{j+1}_{k|0} + \rho_{s,\Theta^j} (s^{j*}_{k|0} - \tilde{s}^{j+1}_{k|0}) + \rho_{\bar{u},\dot{u},\Theta^j} \|\kappa(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}, \bar{u}^{j*}_{k|0}) - \bar{u}^{j*}_{k|0}\| \\
&\qquad + \rho_{\bar{u},\dot{u},\Theta^j} \|\bar{u}^{j*}_{k|0} - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\overset{(4.6)}{\leq} \rho_{q,\Theta^j} q^{j+1}_{k|0} + \rho_{s,\Theta^j} (s^{j*}_{k|0} - \tilde{s}^{j+1}_{k|0}) + \kappa_{max} \rho_{\bar{u},\dot{u},\Theta^j} V_\delta(\bar{x}^{j+1}_{k|0}, \bar{x}^{j*}_{k|0}) \\
&\qquad + \rho_{\bar{u},\dot{u},\Theta^j} \|\bar{u}^{j*}_{k|0} - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\overset{(4.14),(4.26)}{\leq} \rho_{q,\Theta^j} q^{j+1}_{k|0} + \rho_{s,\Theta^j} s^{j*}_{k|0} + \rho_{\bar{u},\dot{u},\Theta^j} \|\bar{u}^{j*}_{k|0} - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\overset{(4.30)}{\leq} \rho_{q,\Theta^j} q^{j*}_{k|0} + \rho_{s,\Theta^j} s^{j*}_{k|0} + \rho_{\bar{u},\dot{u},\Theta^j} \|\bar{u}^{j*}_{k|0} - \dot{u}_k\| + 2\sigma^{max}(P) M_d, \\
&\overset{(4.18f)}{=} q^{j*}_{k+1|0}.
\end{aligned}
$$

Hence,

$$
a^{j+1}_{q,k|0}(h_i(\dot{x}_k, \dot{u}_k) + c_i q^{j+1}_{k|0}) \leq a^{j*}_{q,k|0}(h_i(\dot{x}_k, \dot{u}_k) + c_i q^{j*}_{k|t}) \leq 0 \tag{4.31}
$$

such that constraint (4.18h) continues to be satisfied.

Equation (4.29) gives that

$$
s^{j+1}_{k|0} \leq s^{j*}_{k|0} \overset{(4.18e)}{\leq} \delta_{loc}
$$

such that (4.18e) continues to be satisfied. Moreover, (4.25) ensures that $a^{j+1}_{s,k|0}$ and $a^{j+1}_{q,k|0}$ satisfy constraint (4.18i). Consequently, the candidate solution given by (4.25) is feasible for (4.18). □

Theorem 5. *Let Assumptions 15, 16, 17, 19, 20, 21, 22, and 23 hold. Then* $R(\Theta^j, x^j_t, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$ *is feasible for all timesteps* $t \in \mathbb{I}_{[0,n_t-2]}$ *and iterations* $j \in \mathbb{I}_{\geq 0}$.

Proof. The desired result is an immediate consequence of Lemmas 24 and 25. □

4.3.3.2 Cost and Benchmark Input Convergence

Now that conditions for recursive feasibility have been established, convergence of the integrated cost and benchmark input sequence is examined.

Theorem 6. *Suppose that Assumptions 15-23 hold. Then, the optimal integrated cost at timestep* 0 *monotonically decreases over the iterations such that*

$$J^{int*}_{\c{\Theta}^{j+1},\dot{\boldsymbol{x}}^{j+1},\dot{\boldsymbol{u}}^{j+1}} \leq J^{int*}_{\c{\Theta}^{j},\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j}}.$$

Proof. By nature of Algorithm 4, $\boldsymbol{J}^{int*}_{\c{\Theta}^{j+1},\boldsymbol{x}^{j},\boldsymbol{u}^{j}} < \boldsymbol{J}^{int*}_{\c{\Theta}^{j+1},\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j}}$ is enforced at any iteration wherein an update to $(\dot{\boldsymbol{x}}^{j+1},\dot{\boldsymbol{u}}^{j+1})$ has occurred. Hence, in the remainder of the proof, only the case where $(\dot{\boldsymbol{x}}^{j+1},\dot{\boldsymbol{u}}^{j+1}) = (\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j})$ needs to be considered.

Let a candidate solution to $P(\c{\Theta}^{j+1}, x_0, \dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}, 0)$ be given by (4.25), which, following from Lemma 25, is feasible. We then show that this candidate solution does not result in an increase in the integrated system cost. From (4.16) we have that

$$\begin{aligned}
\Omega_{\mathcal{X}}(\bar{x}^{j*}_{k|0}, s^{j*}_{k|0}) &= \{x \in \mathcal{X} : V_\delta(x, \bar{x}^{j*}_{k|0}) \leq s^{j*}_{k|0}\},\\
&\supseteq \{x \in \mathcal{X} : \|x - \bar{x}^{j+1}_{k|0}\|_P + \|\bar{x}^{j+1}_{k|0} - \bar{x}^{j*}_{k|0}\|_P \leq s^{j*}_{k|0}\},\\
&\overset{(4.29)}{\supseteq} \{x \in \mathcal{X} : \|x - \bar{x}^{j+1}_{k|0}\|_P + \|\bar{x}^{j+1}_{k|0} - \bar{x}^{j*}_{k|0}\|_P \leq s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0}\},\\
&\overset{(4.26)}{\supseteq} \{x \in \mathcal{X} : \|x - \bar{x}^{j+1}_{k|0}\|_P \leq s^{j+1}_{k|0}\},\\
&\overset{(4.16)}{=} \Omega_{\mathcal{X}}(\bar{x}^{j+1}_{k|0}, s^{j+1}_{k|0}).
\end{aligned}$$

Similarly, (4.17) gives

$$\begin{aligned}
\Omega_{\mathcal{U}}(\bar{u}^{j*}_{k|0}, s^{j*}_{k|0}) &= \{u \in \mathcal{U} : \|u - \bar{u}^{j*}_{k|0}\| \leq \kappa_{max} s^{j*}_{k|0}\},\\
&\supseteq \{u \in \mathcal{U} : \|u - \bar{u}^{j+1}_{k|0}\| + \|\bar{u}^{j+1}_{k|0} - \bar{u}^{j*}_{k|0}\| \leq \kappa_{max} s^{j*}_{k|0}\},\\
&\overset{(4.29)}{\supseteq} \{u \in \mathcal{U} : \|u - \bar{u}^{j+1}_{k|0}\| + \|\bar{u}^{j+1}_{k|0} - \bar{u}^{j*}_{k|0}\| \leq \kappa_{max}(s^{j+1}_{k|0} + \tilde{s}^{j+1}_{k|0})\},\\
&\overset{(4.6),(4.25),(4.26)}{\supseteq} \{u \in \mathcal{U} : \|u - \bar{u}^{j+1}_{k|0}\| \leq s^{j+1}_{k|0}\},\\
&\overset{(4.17)}{=} \Omega_{\mathcal{U}}(\bar{u}^{j+1}_{k|0}, s^{j+1}_{k|0}).
\end{aligned}$$

Given the definition of J^{int} in (4.5) and Assumption 18, we then have that

$$J^{int}(\bar{\boldsymbol{x}}^{j+1}_{\cdot|0}, \bar{\boldsymbol{u}}^{j+1}_{\cdot|0}, \boldsymbol{s}^{j+1}_{\cdot|0}, 0) \leq J^{int}(\boldsymbol{x}^{j*}_{\cdot|0}, \boldsymbol{u}^{j*}_{\cdot|0}, \boldsymbol{s}^{j*}_{\cdot|0}, 0).$$

Therefore, as $\bar{\boldsymbol{u}}^{j+1}_{\cdot|0}$ corresponds to a feasible solution to $P(\c{\Theta}^{j+1}, x_0, \dot{\boldsymbol{x}}^{j+1}, \dot{\boldsymbol{u}}^{j+1}, 0)$, but is not necessarily optimal, the desired result is achieved:

$$J^{int*}_{\c{\Theta}^{j+1},\dot{\boldsymbol{x}}^{j+1},\dot{\boldsymbol{u}}^{j+1}} \leq J^{int}(\bar{\boldsymbol{x}}^{j+1}_{\cdot|0}, \bar{\boldsymbol{u}}^{j+1}_{\cdot|0}, \boldsymbol{s}^{j+1}_{\cdot|0}, 0) \leq J^{int*}_{\c{\Theta}^{j},\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j}}.$$

□

Lemma 26. *Suppose Assumptions 15-23 hold. Then, Algorithm 4 ensures that the sequence* $\{\dot{\boldsymbol{u}}^j\}^j$ *converges to some* $\dot{\boldsymbol{u}}^\infty$ *over the iterations.*

Proof. As a result of Theorem 6, we observe that $J^{int*}_{\c{\Theta}^{j+1},\dot{\boldsymbol{x}}^{j+1},\dot{\boldsymbol{u}}^{j+1}} \leq J^{int*}_{\c{\Theta}^{j},\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j}}$. As J^{int} is continuous from Assumption 18, then by the Weierstrass theorem, it attains a minimum over the compact set $\mathcal{Z}$. Consequently, $J^{int*}_{\c{\Theta}^{j},\dot{\boldsymbol{x}}^{j},\dot{\boldsymbol{u}}^{j}}$ converges, which, as a consequence of the benchmark input update condition given by (4.19), implies that $\lim_{j\to\infty} \|\boldsymbol{u}^{j+1} - \dot{\boldsymbol{u}}^j\| = 0$. □

4.4 Uncertainty set adaptation

In this section, an adaptive scheme is proposed that addresses Assumptions 17, 20, and 22. Here, the objective is to reduce the amount of uncertainty in the model parameter estimates while continuing to ensure that $\theta(x) \in \c{\Theta}(x)^j$. Consequently, the radii of the $s-$ and $q-$tubes can be reduced when solving $P(\c{\Theta}^j, x^j_t, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$, which can enable the controller to design more aggressive input signals while continuing to guarantee robust constraint satisfaction.

4.4.1 Partitioned Parameter Adaptation

Based on the framework proposed in [77], an adaptive scheme is now proposed to identify improved estimates of the model parameters $\theta(x)$. However, whereas [77] is restricted to the case where the unknown model parameters are constant, here, the case where the model parameters vary as a function of the system states is addressed. Consequently, additional consideration must be taken to guarantee that $\theta(x) \in \c{\Theta}(x)^j$ for all $x \in \mathcal{X}$ without incurring unnecessary conservatism.

An immediate extension of the adaptive scheme proposed in [77] to the case where the unknown model parameters are state-varying could be achieved by approximating the model parameters as constants, and casting any impacts of state-dependent variations in the model parameters as additive noise to the system. In other words, two modifications can be made: 1) the uncertainty set would be defined as $\c{\Theta}^j(x) = B(\c{\theta}^j(x), z^{\c{\Theta}^j}(x)) = B(\c{\theta}^j, z^{\c{\Theta}^j})$ for a sufficiently large value of $z^{\c{\Theta}^j}$ such that $\theta(x) \in B(\c{\theta}^j, z^{\c{\Theta}^j})$ for all $x \in \mathcal{X}$, and 2): the value of M_d would need to be increased to include the disturbances caused by variations in the model parameters. If the Lipschitz constant L^x_θ is known such that these variations can be bounded over $\mathcal{X}$, this transformation is straightforward to perform. However, if $\mathcal{X}$ is big, or if L_{θ}^{x} is large, significant conservatism is required in the adaptive scheme to guarantee that the true model parameters remain within $B(\underset{\sim}{\theta}^{j}, z^{\underset{\sim}{\Theta}^{j}})$ across $\mathcal{X}$. In other words, safe reductions in $z^{\underset{\sim}{\Theta}^{j}}$ can become difficult to achieve.

To mitigate this issue, an alternative approach is proposed that relies on less extreme approximations of the model parameters. Namely, rather than treating the model parameters as constant over the entirety of $\mathcal{X}$, we approximate the model parameters as being *locally* constant over user-defined subsets of $\mathcal{X}$. Here, by partitioning $\mathcal{X}$ into a group of subsets, we allow the model parameter estimates $\underset{\sim}{\theta}(x)$ to vary from subset to subset, which enables $\underset{\sim}{\theta}(x)$ to serve as a more accurate estimate of $\theta(x)$.

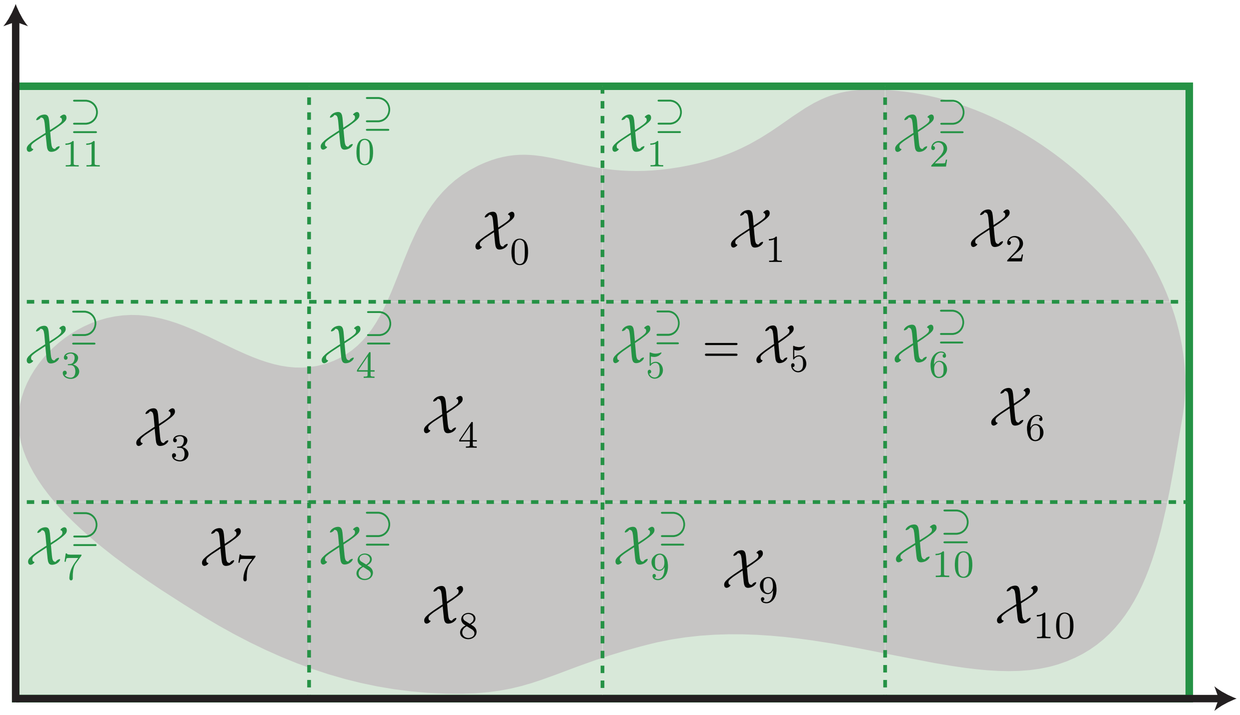
4.4.1.1 Partitioning of the State Space

We first outline the proposed method for partitioning the feasible state space. Let $\mathcal{X}^{\supseteq} \supseteq \mathcal{X}$ denote the smallest bounding hyperrectangle of $\mathcal{X}$. We consider the case where $\mathcal{X}^{\supseteq}$ and $\mathcal{X}$ are partitioned by a set of hyperplanes such that each partition of $\mathcal{X}^{\supseteq}$, denoted by $\mathcal{X}_i^{\supseteq}$, is a hyperrectangle. The resulting partitions of $\mathcal{X}$ formed by the hyperplanes, denoted individually as $\mathcal{X}_i$, can then each be classified as one of the following:

1. A 'boundary' partition, in that there exists an $x \in \mathcal{X}_i$ such that $x \in \partial\mathcal{X}$, where $\partial\mathcal{X}$ denotes the boundary of $\mathcal{X}$.

2. An 'interior' partition, in that there does not exist an $x \in \mathcal{X}_i$ such that $x \in \partial\mathcal{X}$.

As the sets $\mathcal{X}^{\supseteq}$ and $\mathcal{X}$ are partitioned using the same hyperplanes, and because $\mathcal{X}^{\supseteq} \supseteq \mathcal{X}$, the element $x \in \mathcal{X}_i$ lies in one and only one partition of $\mathcal{X}^{\supseteq}$. Consequently, for notational convenience, we index the partitions such that if $x \in \mathcal{X}_i$, then $\mathcal{X}_i \subseteq \mathcal{X}_i^{\supseteq}$. Specifically, $\mathcal{X}_i = \mathcal{X}_i^{\supseteq}$ when $\mathcal{X}_i$ is an interior partition, and $\mathcal{X}_i \subseteq \mathcal{X}_i^{\supseteq}$ when $\mathcal{X}_i$ is a boundary partition. The remaining partitions of $\mathcal{X}^{\supseteq}$ wherein $\mathcal{X}_i^{\supseteq} \cap \mathcal{X} = \emptyset$ do not share an index with any partition of $\mathcal{X}$. This idea is shown graphically for the case when $n_x = 2$ in Fig. 4.2. Let n_p denote the number of $\mathcal{X}_i$ partitions. For the remainder of Section 4.4.1 the partitions $\mathcal{X}_i$ will be leveraged to generate the proposed adaptive scheme. The partitions $\mathcal{X}_i^{\supseteq}$ will be revisited in Section 4.4.2.



$\mathcal{X}_{11}^{\supseteq}$
$\mathcal{X}_{0}^{\supseteq}$
$\mathcal{X}_{1}^{\supseteq}$
$\mathcal{X}_{2}^{\supseteq}$
$\mathcal{X}_0$
$\mathcal{X}_1$
$\mathcal{X}_2$
$\mathcal{X}_{3}^{\supseteq}$
$\mathcal{X}_{4}^{\supseteq}$
$\mathcal{X}_{5}^{\supseteq} = \mathcal{X}_5$
$\mathcal{X}_{6}^{\supseteq}$
$\mathcal{X}_3$
$\mathcal{X}_4$
$\mathcal{X}_6$
$\mathcal{X}_{7}^{\supseteq}$
$\mathcal{X}_7$
$\mathcal{X}_{8}^{\supseteq}$
$\mathcal{X}_{9}^{\supseteq}$
$\mathcal{X}_9$
$\mathcal{X}_{10}^{\supseteq}$
$\mathcal{X}_8$
$\mathcal{X}_{10}$

Figure 4.2: The feasible state set $\mathcal{X}$ (gray area) is bounded by the hyperrectangle $\mathcal{X}^{\supseteq}$ (green area). The partitions are generated by hyperplanes (dashed green lines). $\mathcal{X}_5$ is the sole interior partition, while all other $\mathcal{X}_i$ are boundary partitions that are subsets of their respective $\mathcal{X}_i^{\supseteq}$.

4.4.1.2 Parameter Estimate Update Law

The method for which the model parameter uncertainty sets are updated is now described. This adaptive scheme follows closely to the method described in [77], but contains additional considerations for the effects of state-dependence of the unknown parameters. As noted earlier, the rationale behind the adaptive scheme is to approximate $\theta(x)$ as a piecewise constant function over the feasible state space. Then, rather than attempting to identify the entirety of $\theta(x)$ simultaneously, the estimate of $\theta(x)$ is treated as a constant over a given partition $\mathcal{X}_i$, and the adaptive update law will be applied over each partition independently.

Let

$$\vartheta^{\mathcal{X}_i} = \{\vartheta : \vartheta = \theta(x), x \in \mathcal{X}_i\}$$

denote the set of all model parameter values that the system may take over state space partition $\mathcal{X}_i$.

Additionally,

$$\begin{aligned}(\theta^{\mathcal{X}_i}, z^{\vartheta^{\mathcal{X}_i}}) = \underset{\theta, z}{\text{argmin}} \quad & z \\ \text{subject to} \quad & \vartheta^{\mathcal{X}_i} \subseteq B(\theta, z)\end{aligned}$$

denotes the center and radius of the smallest ball containing $\vartheta^{\mathcal{X}_i}$. While $\theta^{\mathcal{X}_i}$ and $z^{\vartheta^{\mathcal{X}_i}}$ are unknown to the user, we assume that an estimate, $\hat{z}^{\vartheta^{\mathcal{X}_i}}$ is known that satisfies $\hat{z}^{\vartheta^{\mathcal{X}_i}} \geq z^{\vartheta^{\mathcal{X}_i}}$.

Remark 2. *If Assumption 15 holds such that $\theta(x)$ is locally Lipschitz continuous with Lipschitz constant L_θ^x, then $\hat{z}^{\vartheta^{\mathcal{X}_i}}$ can be computed as $\hat{z}^{\vartheta^{\mathcal{X}_i}} = L_\theta^x r(\mathcal{X}_i)$ where $r(\mathcal{X}_i)$ denotes the radius of $\mathcal{X}_i$.*

For partition $\mathcal{X}_i$, suppose that a known parameter uncertainty set $\hat{\Theta}^j_{\mathcal{X}_i} = B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$ exists satisfying $\vartheta^{\mathcal{X}_i} \subseteq \hat{\Theta}^j_{\mathcal{X}_i}$ and that state and input data $(\boldsymbol{x}, \boldsymbol{u}) = (\{x_0, \ldots, x_{n_t-1}\}, \{u_0, \ldots, u_{n_t-2}\})$ are available from a previous experiment wherein the states of system (4.1) were in partition $\mathcal{X}_i$ for at least a portion of the experiment. Let $t_{in}^{\mathcal{X}_i}$ denote the timestep that the system enters partition $\mathcal{X}_i$ and let $t_{out}^{\mathcal{X}_i}$ denote the last timestep before the system exits partition $\mathcal{X}_i$ or the last timestep before the experiment ends. We then wish to leverage data over the timestep sequence $\{t_{in}^{\mathcal{X}_i}, \ldots, t_{out}^{\mathcal{X}_i}\}$ to update the parameter uncertainty set $\hat{\Theta}^j_{\mathcal{X}_i}$. For simplicity of exposition we only describe the case where the system enters partition $\mathcal{X}_i$ a single time according to the state sequence $\boldsymbol{x}$. In the event that the system enters partition $\mathcal{X}_i$ multiple times over the course of an experiment, the adaptive update can be reinitialized at each $t_{in}^{\mathcal{X}_i}$. We first introduce a filtered form of the regressor matrix $G(x_t, u_t)$, denoted by ω_t, which is given as

$$\omega_{t+1} = \omega_t + G(x_t, u_t) - K_\omega \omega_t, \quad \omega_{t_{in}^{\mathcal{X}_i}} = 0 \tag{4.32}$$

for each $t \in \mathbb{I}_{[t_{in}^{\mathcal{X}_i}, t_{out}^{\mathcal{X}_i}]}$ with $K_\omega \in (0, 1)$ chosen by the user.

Let $\hat{\theta}_t$ denote a placeholder estimate of $\theta(x)$ in partition $\mathcal{X}_i$ with $\hat{\theta}_{t_{in}^{\mathcal{X}_i}} = \hat{\theta}^j_{\mathcal{X}_i}$. Additionally, let $z_t^{\hat{\Theta}}$ denote a user-known estimate of an upper bound on the error of $\hat{\theta}_t$ with $z^{\hat{\Theta}}_{t_{in}^{\mathcal{X}_i}} = z^{\hat{\Theta}^j_{\mathcal{X}_i}}$. We denote the corresponding uncertainty set as $\hat{\Theta}_t = B(\hat{\theta}_t, z_t^{\hat{\Theta}})$. Following (4.32), a filtered state variable, $\hat{x}_t$, is then defined as

$$\hat{x}_{t+1} = f(x_t, u_t) + G(x_t, u_t)\hat{\theta}_{t+1} + (K_\omega - 1)\tilde{x}_t - (1 - K_\omega)\omega_t(\hat{\theta}_t - \hat{\theta}_{t+1}), \;\; \hat{x}_{t_{in}^{\mathcal{X}_i}} = x_{t_{in}^{\mathcal{X}_i}} \tag{4.33}$$

where $\tilde{x}_t = x_t - \hat{x}_t$ denotes the error of the filtered states. Let θ^* denote an arbitrary point in $\vartheta^{\mathcal{X}_i}$. By substitution of (4.1), the dynamics of $\tilde{x}_t$ are then given as

$$\tilde{x}_{t+1} = G(x_t, u_t)\tilde{\hat{\theta}}_{t+1} + (1 - K_\omega)\tilde{x}_t + (1 - K_\omega)\omega_t(\hat{\theta}_t - \hat{\theta}_{t+1}) + v_t, \;\; \tilde{x}_{t_{in}^{\mathcal{X}_i}} = 0 \tag{4.34}$$

where $\tilde{\hat{\theta}}_t = \theta^* - \hat{\theta}_t$ and v_t is an element of the set

$$\mathcal{V}^{\mathcal{X}_i} = \{v \in \mathbb{R}^{n_x} : v = c + d, \|c\| \leq M_w^{\mathcal{X}_i}, \|d\| \leq M_d\}. \tag{4.35}$$

Here, $M_w^{\mathcal{X}_i}$ is some constant satisfying

$$M_w^{\mathcal{X}_i} \geq \max_{x \in \mathcal{X}_i, u \in \mathcal{U}} \|G(x,u)(\theta(x) - \theta^*)\|.$$

A valid value of $M_w^{\mathcal{X}_i}$ can be computed by the user as

$$M_w^{\mathcal{X}_i} = 2\hat{z}^{\vartheta \mathcal{X}_i} \max_{x \in \mathcal{X}_i, u \in \mathcal{U}} \|G(x,u)\|. \tag{4.36}$$

The signal v_t corresponds to the disturbance resulting from the combination of the noise d_t and the error in approximating $\theta(x)$ as a constant over $\mathcal{X}_i$.

We subsequently define an auxiliary variable $\eta_t = \tilde{x}_t - \omega_t \tilde{\hat{\theta}}_t$, which, from (4.32) and (4.34), evolves according to

$$\eta_{t+1} = (1 - K_\omega)\eta_t + v_t, \;\; \eta_{t_{in}^{\mathcal{X}_i}} = 0. \tag{4.37}$$

Since v_t is unknown, (4.37) cannot be solved directly by the user. Consequently, an estimate of η, $\hat{\eta}$, is generated by the dynamics

$$\hat{\eta}_{t+1} = (1 - K_\omega)\hat{\eta}_t.$$

From substitution of (4.37), this means that the auxiliary variable estimate error, $\tilde{\eta} = \eta - \hat{\eta}$, is given by

$$\tilde{\eta}_{t+1} = (1 - K_\omega)\tilde{\eta}_t + v_t. \tag{4.38}$$

We further define a user-known excitation parameter $\Sigma \in \mathbb{R}^{n_\theta \times n_\theta}$ according to

$$\Sigma_{t+1} = \Sigma_t + (\omega_t)^\top \omega_t, \;\; \Sigma_{t_{in}^{\mathcal{X}_i}} = \beta I \tag{4.39}$$

for some user-selected $\beta > 0$. As ω_t is real valued and because $\beta > 0$, from (4.39) we have that Σ_t is symmetric and positive definite. The dynamics of the inverse of Σ are subsequently given as

$$\Sigma_{t+1}^{-1} = \Sigma_t^{-1} - \Sigma_t^{-1}\omega_t^\top (I + \omega_t \Sigma_t^{-1} \omega_t^\top)^{-1} \omega_t \Sigma_t^{-1}, \;\; \Sigma_{t_{in}^{\mathcal{X}_i}}^{-1} = \frac{1}{\beta} I.$$

From these signals, a parameter estimate update law of the form

$$\hat{\theta}_{t+1} = \hat{\theta}_t + \Sigma_t^{-1}\omega_t^\top \left(I + \omega_t \Sigma_t^{-1}\omega_t^\top\right)^{-1} (\tilde{x}_t - \hat{\eta}_t), \tag{4.40}$$

$$= \hat{\theta}_t + \Sigma_t^{-1}\omega_t^\top \left(I + \omega_t \Sigma_t^{-1}\omega_t^\top\right)^{-1} (\omega_t \tilde{\hat{\theta}}_t + \tilde{\eta}_t) \tag{4.41}$$

is proposed. By leveraging the experimental data, (4.40) provides a user-implementable update law that can be translated to (4.41) as a consequence of the definitions of η_t and $\tilde{\eta}_t$. Upon examination of (4.41), we observe that the parameter update law relies on the auxiliary variable, which is influenced by the unknown parameter estimation error $\tilde{\hat{\theta}}_t$. Consequently, if ω_t is adequately excited, the presence of model parameter error can initiate updates to $\hat{\theta}_t$.

In the form given by (4.40), the parameter update law can cause the model parameter estimates to become unbounded. However, if it is known that $\vartheta^{\mathcal{X}_i} \subseteq B(\hat{\theta}_t, z_t^{\hat{\Theta}})$, this issue can be mitigated. Specifically, update law (4.40) is modified to

$$\bar{\theta}_{t+1} = \text{Proj}\left(\hat{\theta}_t + \Sigma_t^{-1}\omega_t^\top \left(I + \omega_t \Sigma_t^{-1}\omega_t^\top\right)^{-1} (\tilde{x}_t - \hat{\eta}_t), \hat{\Theta}_t)\right) \tag{4.42}$$

where $\text{Proj}(\cdot)$ denotes the projection operator such that $\bar{\theta}_{t+1} \in \hat{\Theta}_t$. Consequently, because $\hat{\Theta}_t$ is a convex set, if $\theta^* \in \hat{\Theta}_t$, then

$$(\tilde{\bar{\theta}}_{t+1})^\top \Sigma_{t+1} \tilde{\bar{\theta}}_{t+1} \leq (\tilde{\hat{\theta}}_{t+1})^\top \Sigma_{t+1} \tilde{\hat{\theta}}_{t+1} \tag{4.43}$$

where $\tilde{\bar{\theta}}_t = \theta^* - \bar{\theta}_t$. As a consequence of the use of the projected update law in (4.42), the parameter estimate error is, in the worst case, bounded by the set $B(0, z_{t_{in}^{\mathcal{X}_i}}^{\hat{\Theta}})$.

Lemma 27. *Let Assumptions 15 and 16 hold. Define* $V_{\tilde{\hat{\theta}}_t} = \tilde{\hat{\theta}}_t^\top \Sigma \tilde{\hat{\theta}}_t$. *Then parameter update law* (4.42) *ensures that*

$$V_{\tilde{\hat{\theta}}_{t+1}} = V_{\tilde{\hat{\theta}}_t} - (\tilde{x}_t - \hat{\eta}_t)^\top \left(I + \omega_t \Sigma_t^{-1}\omega_t^\top\right)^{-1} (\tilde{x}_t - \hat{\eta}_t) + \tilde{\eta}_t^\top \tilde{\eta}_t. \tag{4.44}$$

Proof. The proof is given in [77, Lemma 2]. However, whereas $\tilde{x}_t$ and $\tilde{\eta}_t$ are nonzero in [77] as a consequence of additive noise and parameter estimation error with respect to *constant* unknown model parameters, here these signals are nonzero as a consequence of additive noise and parameter estimation error with respect to *state-varying* unknown model parameters. □

The right hand side of (4.44) gives an upper bound on the error between the users parameter estimate $\hat{\theta}_t$ and any arbitrary point in $\vartheta^{\mathcal{X}_i}$. However, as $\tilde{\eta}$ is unknown to the user, this upper bound cannot be computed in practice. In the following subsection, it will be demonstrated that this issue can be mitigated.

4.4.1.3 Uncertainty Set Radius Update Law

After updating the model parameter estimates $\hat{\theta}_t$, the uncertainty set radius $z_t^{\hat{\Theta}}$ is recalculated to redefine the set $\hat{\Theta}_t$.

Although Lemma 27 provides a closed-form expression for the size of $\tilde{\hat{\theta}}_t$, because $\tilde{\hat{\theta}}_{t_{in}^{\mathcal{X}_i}}$ and $\tilde{\eta}_t$ are unknown to the user, (4.44) cannot be solved. However, if Assumptions 15 and 16 hold such that the user can identify the set $\mathcal{V}^{\mathcal{X}_i}$ according to (4.35), then, the dynamics of $\tilde{\eta}_t$ given in (4.38) enable the user to bound the size of $\tilde{\eta}_t$ according to

$$\tilde{\eta}_t^\top \tilde{\eta}_t \leq \left(\frac{M_d + M_w^{\mathcal{X}_i}}{K_\omega} \right)^2 . \tag{4.45}$$

Using this insight along with (4.44), the uncertainty set radius update is proposed as

$$\begin{aligned} z_t^{\hat{\Theta}} =& \sqrt{\frac{V_{z_t^{\hat{\Theta}}}}{\lambda_{min}(\Sigma_t)}}, \\ V_{z_{t+1}^{\hat{\Theta}}} =& V_{z_t^{\hat{\Theta}}} - (\tilde{x}_t - \hat{\eta}_t)^\top (I + \omega_t \Sigma_t^{-1} \omega_t^\top)^{-1} (\tilde{x}_t - \hat{\eta}_t) + \left(\frac{M_d + M_w^{\mathcal{X}_i}}{K_\omega} \right)^2 , \; V_{z^{\hat{\Theta}}_{t_{in}^{\mathcal{X}_i}}} = \beta (z^{\hat{\Theta}}_{t_{in}^{\mathcal{X}_i}})^2 . \end{aligned} \tag{4.46}$$

where $\lambda_{min}(\Sigma)$ gives the smallest eigenvalue of Σ.

Now that the adaptive update to $\hat{\Theta}_t$ has been established according to update laws (4.42) and (4.46), the update to $\hat{\Theta}_{\mathcal{X}_i}^j$ is performed at the conclusion of the j^{th} experiment as given by Algorithm 5.

Algorithm 5 Adaptive update to $\hat{\Theta}^j_{\mathcal{X}_i}$

```
Given: Experimental state and input data $(\boldsymbol{x}, \boldsymbol{u})$, parameters $K_\omega \in (0,1)$, $\beta > 0$,
    noise/disturbance bounds $M_d$ and $M_w^{\mathcal{X}_i}$, and parameter uncertainty set $\hat{\Theta}^j_{\mathcal{X}_i} = B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$.
Identify $t_{in}^{\mathcal{X}_i}$ and $t_{out}^{\mathcal{X}_i}$.
Initialize $\hat{\Theta}_{t_{in}^{\mathcal{X}_i}} = B(\hat{\theta}_{t_{in}^{\mathcal{X}_i}}, z^{\hat{\Theta}}_{t_{in}^{\mathcal{X}_i}}) = \hat{\Theta}^j_{\mathcal{X}_i}$ and $(\bar{\theta}, \bar{z}) = (\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$.
for $t \in \mathbb{I}_{[t_{in}^{\mathcal{X}_i}, t_{out}^{\mathcal{X}_i}]}$ do
    Calculate $\hat{\theta}_{t+1}$ from (4.42) and $z^{\hat{\Theta}}_{t+1}$ from (4.46).
    if
        $$z^{\hat{\Theta}}_{t+1} \leq z^{\hat{\Theta}}_t - \|\bar{\theta} - \hat{\theta}_t\| \qquad (4.47)$$
    then
        Set $(\bar{\theta}, \bar{z}) = (\hat{\theta}_{t+1}, z^{\hat{\Theta}}_{t+1})$.
    end if
end for
Update uncertainty set $\hat{\Theta}^{j+1}_{\mathcal{X}_i}$ with
    $$(\hat{\theta}^{j+1}_{\mathcal{X}_i}, z^{\hat{\Theta}^{j+1}_{\mathcal{X}_i}}) = (\bar{\theta}, \bar{z}). \qquad (4.48)$$
```

Assumption 24. *At iteration* 0*, an uncertainty set* $\hat{\Theta}^0_{\mathcal{X}_i}$ *is known that satisfies* $\vartheta^{\mathcal{X}_i} \subseteq \hat{\Theta}^0_{\mathcal{X}_i}$ *for each partition,* $\mathcal{X}_i$*.*

Lemma 28. *Suppose that Assumptions 15, 16, and 24 hold. Then implementation of Algorithm 5 over partition* $\mathcal{X}_i$ *ensures for all* $j \in \mathbb{I}_{\geq 0}$ *that*

$$\hat{\Theta}^{j+1}_{\mathcal{X}_i} \subseteq \hat{\Theta}^j_{\mathcal{X}_i}$$

and

$$\vartheta^{\mathcal{X}_i} \subseteq \hat{\Theta}^j_{\mathcal{X}_i}.$$

Proof. Assume $\hat{\Theta}^{j+1}_{\mathcal{X}_i} \nsubseteq \hat{\Theta}^j_{\mathcal{X}_i}$. Then, by the definition $\hat{\Theta}^j_{\mathcal{X}_i} = B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$

$$\sup_{\hat{\theta} \in \hat{\Theta}^{j+1}_{\mathcal{X}_i}} \|\hat{\theta} - \hat{\theta}^j_{\mathcal{X}_i}\| > z^{\hat{\Theta}^j_{\mathcal{X}_i}}. \tag{4.49}$$

However,

$$\begin{aligned}\sup_{\hat{\theta}\in\hat{\Theta}^{j+1}_{\mathcal{X}_i}} \|\hat{\theta}-\hat{\theta}^{j}_{\mathcal{X}_i}\| &\le \sup_{\hat{\theta}\in\hat{\Theta}^{j+1}_{\mathcal{X}_i}} \|\hat{\theta}-\hat{\theta}^{j+1}_{\mathcal{X}_i}\| + \|\hat{\theta}^{j+1}_{\mathcal{X}_i}-\hat{\theta}^{j}_{\mathcal{X}_i}\|,\\ &\le z^{\hat{\Theta}^{j+1}_{\mathcal{X}_i}} + \|\hat{\theta}^{j+1}_{\mathcal{X}_i}-\hat{\theta}^{j}_{\mathcal{X}_i}\|,\\ &\overset{(4.47)}{\le} z^{\hat{\Theta}^{j}_{\mathcal{X}_i}},\end{aligned}$$

where the second inequality arises from the definition of $\hat{\Theta}^{j}_{\mathcal{X}_i}$. However, this contradicts (4.49) which gives the first claim.

For an arbitrary $\theta^* \in \vartheta^{\mathcal{X}_i}$, suppose that

$$V_{\tilde{\hat{\theta}}_{t^{\mathcal{X}_i}_{in}}} \le V_{z^{\hat{\Theta}}_{t^{\mathcal{X}_i}_{in}}}.$$

Additionally, from (4.44), (4.45), and (4.46) we have that $V_{\tilde{\hat{\theta}}_{t+1}} - V_{\tilde{\hat{\theta}}_{t}} \le V_{z^{\hat{\Theta}}_{t+1}} - V_{z^{\hat{\Theta}}_{t}}$. Therefore, $V_{\tilde{\hat{\theta}}_{t}} \le V_{z^{\hat{\Theta}}_{t}}$ for all t. Hence,

$$\|\tilde{\hat{\theta}}_t\| \le \sqrt{\frac{V_{z^{\hat{\Theta}}_{t}}}{\lambda_{min}(\Sigma_t)}} = z^{\hat{\Theta}}_{t}.$$

Then, following from (4.48), if $\theta^* \in \hat{\Theta}^{j}_{\mathcal{X}_i}$, then $\theta^* \in \hat{\Theta}^{j+1}_{\mathcal{X}_i}$. Since θ^* is an arbitrary point in $\vartheta^{\mathcal{X}_i}$ and $\vartheta^{\mathcal{X}_i} \subseteq \hat{\Theta}^{0}_{\mathcal{X}_i}$ from Assumption 24, this therefore gives the second claim. □

Hence, by using the proposed adaptive scheme given by Algorithm 5, the unknown model parameter values remain within the parameter uncertainty set, which shrinks monotonically over time.

4.4.1.4 Enforcing Lipschitz Continuity of $\theta(x)$

In Sections 4.4.1.2 and 4.4.1.3 a methodology was proposed for updating the uncertainty sets $\hat{\Theta}^{j}_{\mathcal{X}_i}$ based on experimental data available from when the system states were in the set $\mathcal{X}_i$. As demonstrated in Lemma 28, this adaptive algorithm ensures that $\theta(x)$ remains in the uncertainty set for all $x \in \mathcal{X}_i$ after the uncertainty set update has occurred. However, if Assumption 15 holds such that $\theta(x)$ is Lipschitz continuous, then the update to $\hat{\Theta}^{j}_{\mathcal{X}_i}$ may also provide additional insight into the behavior of $\theta(x)$ in other partitions. For instance, if application of Algorithm 5 over partition $\mathcal{X}_a$ results in significant changes to $\hat{\Theta}^{j}_{\mathcal{X}_a}$ such that $\|\hat{\theta}^{j+1}_{\mathcal{X}_a} - \hat{\theta}^{j}_{\mathcal{X}_a}\|$ and $z^{\hat{\Theta}^{j}_{\mathcal{X}_a}} - z^{\hat{\Theta}^{j+1}_{\mathcal{X}_a}}$ are large, then the sets $\hat{\Theta}^{j+1}_{\mathcal{X}_a}$ and $\hat{\Theta}^{j}_{\mathcal{X}_b}$ may be quite different even though Lipschitz continuity should imply that $\theta(x)$ should not change drastically over $\mathcal{X}$. Consequently, application of Algorithm 5 on its own does not take full advantage of the user's knowledge of the behavior of $\theta(x)$, which can result in unnecessary conservatism in the model parameter estimation. Therefore we propose a method for updating the uncertainty sets based on state data available in other partitions by leveraging Lipschitz continuity of the unknown model parameters.

Let $m(\mathcal{X}_a, \mathcal{X}_b)$ denote the maximum possible distance between a point in the set $\mathcal{X}_a$ and a point in the set $\mathcal{X}_b$ according to

$$m(\mathcal{X}_a, \mathcal{X}_b) = \sup_{x^a \in \mathcal{X}_a, x^b \in \mathcal{X}_b} \|x^a - x^b\|_P.$$

Additionally, let $\mathcal{I}(\boldsymbol{x})$ denote the set of indices of the partitions $\mathcal{X}_i$ that were entered over the course of the state sequence $\boldsymbol{x}$. A methodology that updates the uncertainty sets to enforce Lipschitz continuity of the unknown model parameters is then given by Algorithm (6).

Algorithm 6 Enforcement of Lipschitz continuity of $\theta(x)$ on $\hat{\Theta}^j_{\mathcal{X}_i}$

```
Given: Experimental state and input data $(\boldsymbol{x}, \boldsymbol{u})$, parameters $K_\omega \in (0,1)$ and $\beta > 0$, Lipschitz constant $L^x_\theta$, matrix $P$, noise/disturbance bounds $M_d$ and $M^{\mathcal{X}_i}_w$, and parameter uncertainty sets $\hat{\Theta}^j_{\mathcal{X}_i} = B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$ for each $i \in \mathcal{I}(\boldsymbol{x})$.
for $i \in \mathcal{I}(\boldsymbol{x})$ do
    Update uncertainty set $\hat{\Theta}^j_{\mathcal{X}_i} \to \hat{\Theta}^{j+1}_{\mathcal{X}_i}$ by application of Algorithm 5.
    for $a \in \mathbb{I}_{[0,n_p-1]} \setminus i$ do
        Redefine uncertainty set $\hat{\Theta}^j_{\mathcal{X}_a}$ as
            $\hat{\Theta}^{j+1}_{\mathcal{X}_a} = \hat{\Theta}^j_{\mathcal{X}_a} \cap \hat{\Theta}^{j+1}_{\mathcal{X}_i} \oplus B\left(0, L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i)\right)$    (4.50)
    end for
end for
```

We now show that application of Algorithm 6 enables the set $\hat{\Theta}^j_{\mathcal{X}_a}$ to be less conservative while still maintaining robust identification of the unknown model parameters over $\mathcal{X}_a$.

Lemma 29. *Suppose that Assumptions 15, 16, and 24 hold. Then application of Algorithm 6 over partition $\mathcal{X}_i$ ensures that*

$$\hat{\Theta}^{j+1}_{\mathcal{X}_a} \subseteq \hat{\Theta}^j_{\mathcal{X}_a} \tag{4.51}$$

and

$$\vartheta^{\mathcal{X}_a} \subseteq \hat{\Theta}^{j+1}_{\mathcal{X}_a}. \tag{4.52}$$

for all $a \in \mathbb{I}_{[0,n_p-1]}$ *and* $j \in \mathbb{I}_{\geq 0}$.

Proof. For the case where $a = i$, Lemma 28 gives the above two claims.

We then examine the case where $a \neq i$. From the update law given by (4.50)

$$\hat{\Theta}^{j+1}_{\mathcal{X}_a} = \hat{\Theta}^{j}_{\mathcal{X}_a} \cap \hat{\Theta}^{j}_{\mathcal{X}_i} \oplus B\left(0, L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i)\right) \subseteq \hat{\Theta}^{j}_{\mathcal{X}_a}$$

which gives claim (4.51).

From the assumption on Lipschitz continuity of $\theta(x)$ in Assumption 15, for $x^a \in \mathcal{X}_a$ and $x^i \in \mathcal{X}_i$ we have that

$$\|\theta(x^a) - \theta(x^i)\| \leq L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i).$$

Consequently, as x^a and x^i are arbitrary points in $\mathcal{X}_a$ and $\mathcal{X}_i$, and since $\vartheta^{\mathcal{X}_i} \subseteq \hat{\Theta}^{j+1}_{\mathcal{X}_i}$, then

$$\vartheta^{\mathcal{X}_a} \subseteq \vartheta^{\mathcal{X}_i} \oplus B(0, L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i)) \subseteq \hat{\Theta}^{j+1}_{\mathcal{X}_i} \oplus B(0, L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i)).$$

Therefore, if $\vartheta^{\mathcal{X}_a} \subseteq \hat{\Theta}^{j}_{\mathcal{X}_a}$, then $\vartheta^{\mathcal{X}_a} \subseteq \hat{\Theta}^{j+1}_{\mathcal{X}_a} = \hat{\Theta}^{j}_{\mathcal{X}_a} \cap \hat{\Theta}^{j+1}_{\mathcal{X}_i} \oplus B\left(0, L^x_\theta m(\mathcal{X}_a, \mathcal{X}_i)\right)$. Since $\vartheta^{\mathcal{X}_a} \subseteq \hat{\Theta}^{0}_{\mathcal{X}_a}$ from Assumption 24, then (4.51) holds for all j. □

4.4.2 Generation of $\c{\Theta}^j(x)$

In Section 4.4.1 an adaptive scheme was proposed to generate uncertainty sets $\hat{\Theta}^{j}_{\mathcal{X}_i}$ that: 1) shrink monotonically, and 2) contain the true model parameter values in partition $\mathcal{X}_i$ at every iteration. Specifically, over the partition $\mathcal{X}_i$, Algorithm 6 generates a state-invariant parameter estimate $\hat{\theta}^{j}_{\mathcal{X}_i}$ and uncertainty set radius $z^{\hat{\Theta}^{j}_{\mathcal{X}_i}}$. However, for two partitions $\mathcal{X}_i$ and $\mathcal{X}_{i+1}$ that are adjacent in the state space, it does not hold in general that $\hat{\theta}^{j}_{\mathcal{X}_i} = \hat{\theta}^{j}_{\mathcal{X}_{i+1}}$ or $z^{\hat{\Theta}^{j}_{\mathcal{X}_i}} = z^{\hat{\Theta}^{j}_{\mathcal{X}_{i+1}}}$. Consequently, if the uncertainty set function $\c{\Theta}^j(x)$ that is used by the RAEILC controller in Algorithm 4 were to be defined as

$$\c{\Theta}^j(x) = \hat{\Theta}^{j}_{\mathcal{X}_i} \text{ if } x \in \mathcal{X}_i, i \in \mathbb{I}_{[0,n_p-1]},$$

then, while requirement (4.2) of Assumption 17 and Assumption 22 would hold, the assumption of Lipschitz continuity of $\c{\theta}^j(x)$ and $z^{\c{\Theta}^j}(x)$ would not be satisfied at the partition boundaries. This loss of continuity not only results in a loss of Assumption 17, but also makes it difficult to satisfy condition (4.9b) of Assumption 20 which effectively requires that the amount of uncertainty introduced to the system dynamics is Lipschitz with respect to the system state. In this section we identify a mitigation strategy to enable satisfaction of Assumption 17. Assumption 20 will then be addressed in Section 4.4.3.

Definition 7. *As described in [100], suppose there exists a hyperplane, H, such that $H \cap \mathcal{X}_i^{\supseteq} \neq \emptyset$ and that $\mathcal{X}^{\supseteq}$ lies entirely within a closed half-space generated by H. The set $H \cap \mathcal{X}_i^{\supseteq}$, whose dimension is n, is termed an 'n-face' of $\mathcal{X}_i^{\supseteq}$.*

As with all polytopes, the hyperrectangle partition $\mathcal{X}_i^{\supseteq}$ consists of n-faces whose dimensionality ranges from 0 to n_x. For instance, 0-faces correspond to vertices of $\mathcal{X}_i^{\supseteq}$, 1-faces represent edges of $\mathcal{X}_i^{\supseteq}$, 2-faces are faces of $\mathcal{X}_i^{\supseteq}$, etc. The n_x-face of $\mathcal{X}_i^{\supseteq}$ is given by $\mathcal{X}_i^{\supseteq}$ itself.

Using information about the n-faces of $\mathcal{X}_i^{\supseteq}$, we will now describe a method for further dividing $\mathcal{X}_i^{\supseteq}$ into a set of subpartitions. If v is a 0-face (vertex) of partition $\mathcal{X}_i^{\supseteq}$, let $\mathcal{F}(v, \mathcal{X}_i^{\supseteq})$ denote the set of n-faces of $\mathcal{X}_i^{\supseteq}$ for which v is an element. Then, given a set $\mathcal{F}$, let $\mathcal{C}(\mathcal{F})$ denote the set of centroids of the n-faces in $\mathcal{F}$. Consequently, $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))$ is the set of points corresponding to the centroids of the n-faces of $\mathcal{X}_i^{\supseteq}$ that contain the vertex, v. The convex hull of $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))$, denoted as conv$(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq})))$, is a polytope since $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))$ is a finite set. In fact, conv$(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq})))$ is itself a hyperrectangle that constitutes a subpartition of $\mathcal{X}_i^{\supseteq}$. Namely,

$$\bigcup_{a=1}^{2^{nx}} \text{conv}\left(\mathcal{C}(\mathcal{F}(v_a, \mathcal{X}_i^{\supseteq}))\right) = \mathcal{X}_i^{\supseteq} \tag{4.53}$$

where v_a denotes the a^{th} vertex of $\mathcal{X}_i^{\supseteq}$.

Let $\mathcal{P}(x)$ denote the set of indices, i, for which $x \in$ cl$(\mathcal{X}_i^{\supseteq})$ where cl$(\mathcal{A})$ denotes the closure of the set $\mathcal{A}$. In other words, $\mathcal{P}(x)$ identifies which partitions, $\mathcal{X}_i^{\supseteq}$, that x belongs to or lies on the boundary of.

For every element in $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))$, $B(\c{\theta}^j(x), z^{\c{\Theta}^j}(x))$ is constructed as the smallest ball containing the set $\bigcup_{i \in \mathcal{P}(x)} B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$ such that

$$B(\c{\theta}^j(x), z^{\c{\Theta}^j}(x)) \supseteq \bigcup_{i \in \mathcal{P}(x)} B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}}). \tag{4.54}$$

Note that $\hat{\theta}^j_{\mathcal{X}_i}$ and $z^{\hat{\Theta}^j_{\mathcal{X}_i}}$ may be undefined if $\mathcal{X}_i^{\supseteq} \cap \mathcal{X} = \emptyset$. In this case, we say that $B(\hat{\theta}^{\mathcal{X}_i}, z^{\Theta^{\mathcal{X}_i}}) = \emptyset$ in (4.54).

To define $\c{\Theta}^j(x)$ over the remainder of conv $\left(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))\right)$, $\c{\theta}^j(x)$ and $z^{\c{\Theta}^j}(x)$ are calculated at every $x \in$ conv $\left(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i^{\supseteq}))\right)$ through multilinear interpolation based on their values at the sample points $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_i))$.

Repeating this process for every vertex, v, of partition $\mathcal{X}_i^{\supseteq}$ allows $\c{\Theta}^j(x)$ to be defined over the entirety of $\mathcal{X}_i^{\supseteq}$. Conducting this process further at every partition satisfying $\mathcal{X}_i^{\supseteq} \cap \mathcal{X} \neq \emptyset$ then enables $\c{\Theta}^j(x)$ to be defined for every $x \in \mathcal{X}$. This process of constructing $\c{\Theta}^j(x)$ is shown graphically in Fig. 4.3.

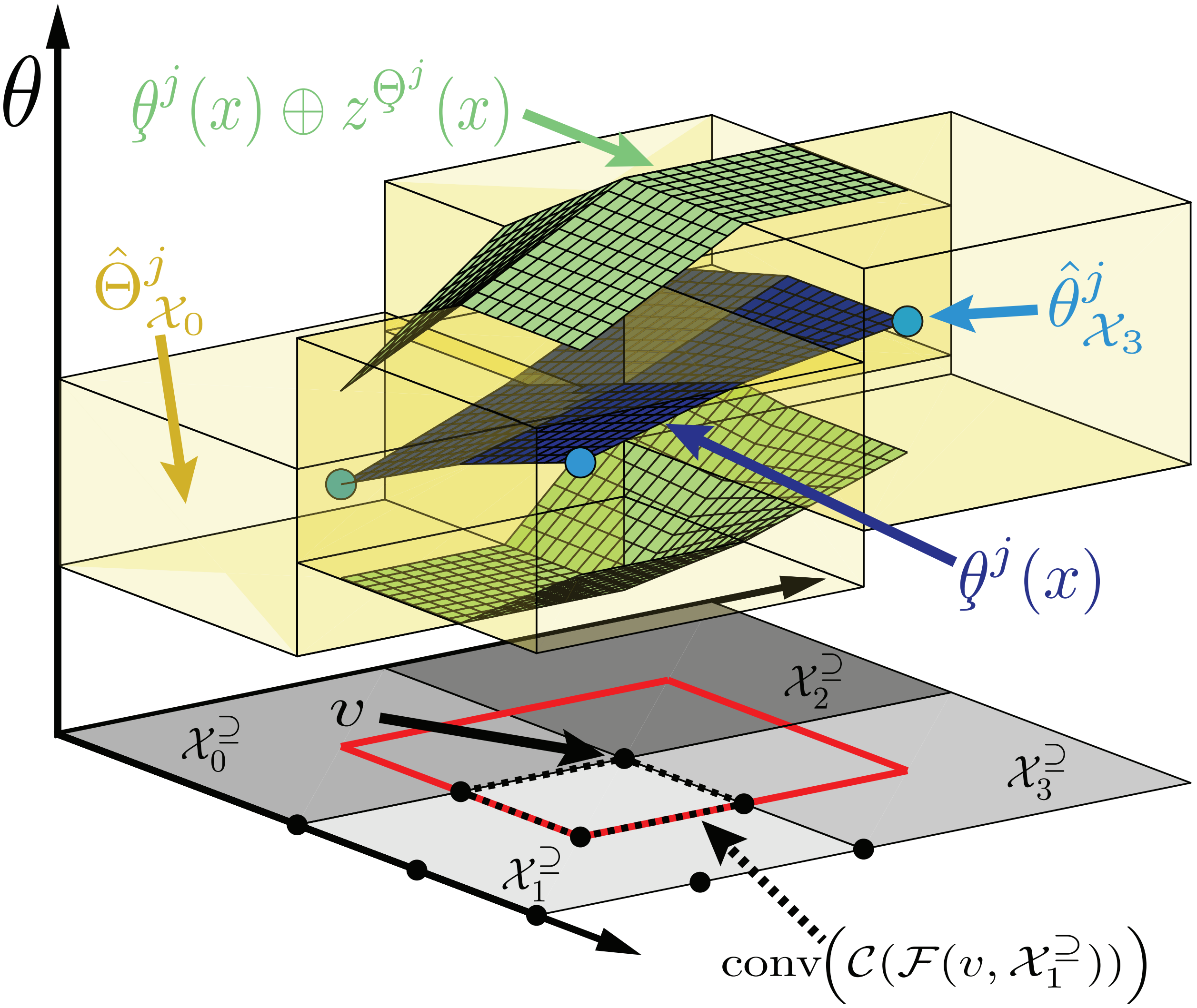

θ
$\c{\theta}^j(x) \oplus z^{\c{\Theta}^j}(x)$
$\hat{\Theta}^j_{\mathcal{X}_0}$
$\hat{\theta}^j_{\mathcal{X}_3}$
$\c{\theta}^j(x)$
v
$\mathcal{X}_0^{\supseteq}$
$\mathcal{X}_1^{\supseteq}$
$\mathcal{X}_2^{\supseteq}$
$\mathcal{X}_3^{\supseteq}$
$\mathrm{conv}\Big(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_1^{\supseteq}))\Big)$

Figure 4.3: An example of the construction of $\c{\Theta}(x)$ for $n_x = 2$ and $n_\theta = 1$ over four partitions $\mathcal{X}_i^{\supseteq}$ (gray areas). The uncertainty sets $\hat{\Theta}^j_{\mathcal{X}_i} = B(\hat{\theta}^j_{\mathcal{X}_i}, z^{\hat{\Theta}^j_{\mathcal{X}_i}})$ (yellow regions) are generated by Algorithm 6, but are not aligned at the boundaries of the partitions. For each partition, $\mathcal{X}_i^{\supseteq}$ the centroids of each of the $n-$faces of the system are identified (black dots are the centroids of the $n-$faces of partition $\mathcal{X}_1^{\supseteq}$.) For a vertex, v, $\c{\Theta}^j(x)$ is determined at each centroid that shares an $n-$face with v, i.e. at each element of $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_1^{\supseteq}))$. For all $x \in \mathrm{conv}(\mathcal{C}(\mathcal{F}(v, \mathcal{X}_1^{\supseteq})))$ (area enclosed by the dashed black lines), the values of $\c{\theta}^j(x)$ (blue surface) and $z^{\c{\Theta}^j}(x)$ are determined via multilinear interpolation of their values at the points in $\mathcal{C}(\mathcal{F}(v, \mathcal{X}_1^{\supseteq}))$. This process is repeated at every vertex of each partition, but, for the purposes of visualization, is only shown here when the procedure is applied at the vertex v over each of the partitions such that $\c{\Theta}^j(x)$ is depicted only for the x that lie within the red square. This strategy serves to generate a $\c{\Theta}^j(x)$ that is more conservative than the sets $\hat{\Theta}^j_{\mathcal{X}_i}$, but enables continuity at the partition boundaries.

We will now demonstrate that $\theta(x) \in \c{\Theta}^j(x)$ for all $x \in \mathcal{X}$. For this, we require the following lemma.

Lemma 30. *For some set $\mathcal{S}$, if $\mathcal{S} \subseteq B(x^a, z^a) \cap B(x^b, z^b)$, then*

$$\mathcal{S} \subseteq B(\lambda x^a + (1-\lambda)x^b, \lambda z^a + (1-\lambda)z^b) \tag{4.55}$$

for $\lambda \in [0,1]$.

Proof. Suppose for some $x \in \mathcal{S}$ that $x \notin B(\lambda x^a + (1-\lambda)x^b, \lambda z^a + (1-\lambda)z^b)$ such that

$$\|x - \lambda x^a - (1-\lambda)x^b\| = \|\lambda x - \lambda x^a + (1-\lambda)x - (1-\lambda)x^b\| > \lambda z^a + (1-\lambda)z^b.$$

By the triangle inequality and because $\lambda \in [0,1]$, we have

$$\lambda\|x - x^a\| + (1-\lambda)\|x - x^b\| > \lambda z^a + (1-\lambda)z^b.$$

But since $x \in B(x^a, z^a) \cap B(x^b, z^b)$, then

$$\|x - x^a\| < z^a, \quad \|x - x^b\| < z^b,$$

so

$$\lambda z^a + (1-\lambda)z^b > \lambda z^a + (1-\lambda)z^b$$

which is a contradiction. Therefore, since x is an arbitrary point in $\mathcal{S}$, we obtain (4.55). □

Theorem 7. *Suppose that Assumptions 15, 16, and 24 hold. Then, for all $x \in \mathcal{X}$,*

$$\theta(x) \in \c{\Theta}^j(x). \tag{4.56}$$

Proof. To demonstrate (4.56), we will show that $\theta(x) \in \c{\Theta}^j(x)$ for x in an arbitrary partition $\mathcal{X}_i$.

For vertex υ of partition $\mathcal{X}_i^{\supseteq}$ for which $\mathcal{X}_i^{\supseteq} \cap \mathcal{X} \neq \emptyset$, from (4.54) we have that

$$\c{\Theta}^j(x) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}, \ \forall x \in \mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq})). \tag{4.57}$$

Suppose that we are interested in calculating the value of $(\c{\theta}^j(x^{2^{n_x+1}-1}), z^{\c{\Theta}^j}(x^{2^{n_x+1}-1}))$ at some point $x^{2^{n_x+1}-1} \in \operatorname{conv}\big(\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))\big)$. Evaluating $(\c{\theta}^j(x^{2^{n_x+1}-1}), z^{\c{\Theta}^j}(x^{2^{n_x+1}-1}))$ via multilinear interpolation based on the values of $(\c{\theta}^j(x), z^{\c{\Theta}^j}(x))$ at the sample points in $\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))$ requires $n_x - 1$ linear interpolations to be performed, where $(\c{\theta}^j(x^a), z^{\c{\Theta}^j}(x^a))$ is evaluated based on the points $(\c{\theta}^j(x^{2(a-n_x)-1}), z^{\c{\Theta}^j}(x^{2(a-n_x)-1}))$ and $(\c{\theta}^j(x^{2(a-n_x)}), z^{\c{\Theta}^j}(x^{2(a-n_x)}))$ for $a \in \{2^{n_x} + 1, \ldots, 2^{n_x+1} - 1\}$ and the points $\{x^1, \ldots, x^{n_x}\}$ are the elements of $\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))$.

From Lemma 30, we have that, if $\c{\Theta}^j(x^{2(a-n_x)-1}) \cap \c{\Theta}^j(x^{2(a-n_x)}) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}$, then $\c{\Theta}^j(x^a) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}$. Since, from (4.57), we have that $\c{\Theta}^j(x^1), \ldots, \c{\Theta}^j(x^{n_x}) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}$, then, by induction,

$$\c{\Theta}^j(x^{2^{n_x+1}-1}) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}, \forall x^{2^{n_x+1}-1} \in \text{conv}\left(\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))\right). \tag{4.58}$$

Moreover, as (4.58) holds for an arbitrary vertex υ of $\mathcal{X}_i^{\supseteq}$, and because $\mathcal{X}_i^{\supseteq} \supseteq \mathcal{X}_i$, (4.53) then gives that

$$\c{\Theta}^j(x) \supseteq \hat{\Theta}^j_{\mathcal{X}_i}, \forall x \in \mathcal{X}_i. \tag{4.59}$$

Since, from Lemma 29, $\theta(x) \in \hat{\Theta}^j_{\mathcal{X}_i}$ is guaranteed by the adaptive law, and as (4.59) holds for any arbitrary partition $\mathcal{X}_i$ with $\mathcal{X} = \bigcup_i \mathcal{X}_i$, then $\theta(x) \in \c{\Theta}^j(x)$ for all $x \in \mathcal{X}$. □

As $\c{\Theta}^j(x)$ is identified via multilinear interpolation, $(\c{\theta}^j(x), z^{\Theta^j}(x))$ is therefore a polynomial of order n_x over the set conv $\left(\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))\right)$. Moreover, as conv $\left(\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))\right)$ is compact, then $(\c{\theta}^j(x), z^{\Theta^j}(x))$ is locally C^1 over $\mathcal{X}_i^{\supseteq}$. Given the definition of $\c{\Theta}^j(x)$ in (4.54), we additionally observe that on the boundary between the sets conv $\left(\mathcal{C}(\mathcal{F}(\upsilon_a, \mathcal{X}_b^{\supseteq}))\right)$ and conv $\left(\mathcal{C}(\mathcal{F}(\upsilon_c, \mathcal{X}_d^{\supseteq}))\right)$, that the points of $\mathcal{C}(\mathcal{F}(\upsilon_a, \mathcal{X}_b^{\supseteq}))$ and $\mathcal{C}(\mathcal{F}(\upsilon_c, \mathcal{X}_d^{\supseteq}))$ that lie on this boundary have identical values of $(\c{\theta}^j(x), z^{\Theta^j}(x))$. Consequently, interpolations performed along this boundary are based on identical sample values and therefore yield equivalent values of $(\c{\theta}^j(x), z^{\Theta^j}(x))$. Hence, $(\c{\theta}^j(x), z^{\Theta^j}(x))$ is C^0 and piecewise C^1 over $\mathcal{X}$. As there are a finite number of sets conv $\left(\mathcal{C}(\mathcal{F}(\upsilon, \mathcal{X}_i^{\supseteq}))\right)$, this implies that $(\c{\theta}^j(x), z^{\Theta^j}(x))$ is Lipschitz continuous over the entirety of $\mathcal{X}$ such that Assumption 17 is satisfied.

4.4.3 Enabling Continuity of $\tilde{w}_{\c{\Theta},\mathcal{D}}(\bar{x}, \bar{u})$

If $\c{\Theta}^j(x)$ is generated according to the method described in Section 4.4.2, we now propose a definition of $\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u})$ as

$$\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}) = \max_{\tilde{\c{\theta}} \in \tilde{\c{\Theta}}(\bar{x})} \|G(\bar{x}, \bar{u})\tilde{\c{\theta}}\|_P + \max_{d \in \mathcal{D}} \|d\|_P$$

such that conditions (4.9a)-(4.9e) of Assumption 20 can be satisfied.

Assumption 25. *For all $x \in \mathcal{X}$ and $j \in \mathbb{I}_{\geq 0}$, $z^{\Theta^j}(x)$ is lower bounded by some z^{Θ}_{min}.*

If $M_w^{\mathcal{X}_i}$ is calculated according to (4.36), Assumption 25 can be satisfied with

$$z^{\Theta}_{min} = \min_{i \in \mathbb{I}_{[0,n_p-1]}} \hat{z}^{\vartheta^{\mathcal{X}_i}}.$$

To demonstrate (4.9a), consider some $d_w \in \mathcal{W}_{\tilde{\Theta}^j,\mathcal{D}}(\bar{x}, \bar{u})$. Consequently,

$$V_\delta(\breve{x} + d_w, \breve{x}) = \|d_w\|_P \overset{(4.4)}{\leq} \max_{\tilde{\theta} \in \tilde{\Theta}^j(\bar{x}), d \in \mathcal{D}} \|G(\bar{x}, \bar{u})\tilde{\theta} + d\|_P,$$

$$\leq \max_{\tilde{\theta} \in \tilde{\Theta}^j(\bar{x})} \|G(\bar{x}, \bar{u})\tilde{\theta}\|_P + \max_{d \in \mathcal{D}} \|d\|_P = \tilde{w}_{\tilde{\Theta}^j,\mathcal{D}}(\bar{x}, \bar{u}).$$

To show (4.9b), we have

$$\begin{aligned}
&\tilde{w}_{\tilde{\Theta}^j,\mathcal{D}}(x, \kappa(x, \bar{x}, \bar{u})) - \tilde{w}_{\tilde{\Theta}^j,\mathcal{D}}(\bar{x}, \bar{u}) \\
&\quad = \max_{\tilde{\theta}_x \in \tilde{\Theta}^j(x)} \|G(x, \kappa(x, \bar{x}, \bar{u}))\tilde{\theta}_x\|_P + \max_{d \in \mathcal{D}} \|d\|_P - \max_{\tilde{\theta}_{\bar{x}} \in \tilde{\Theta}^j(\bar{x})} \|G(\bar{x}, \bar{u})\tilde{\theta}_{\bar{x}}\|_P - \max_{d \in \mathcal{D}} \|d\|_P, \\
&\quad \leq \max_{\tilde{\theta}_x \in \tilde{\Theta}^j(x)} \|G(x, \kappa(x, \bar{x}, \bar{u}))\tilde{\theta}_x - G(\bar{x}, \bar{u})\tilde{\theta}_x\|_P + \max_{\tilde{\theta}_x \in \tilde{\Theta}^j(x)} \|G(\bar{x}, \bar{u})\tilde{\theta}_x\|_P \\
&\qquad - \max_{\tilde{\theta}_{\bar{x}} \in \tilde{\Theta}^j(\bar{x})} \|G(\bar{x}, \bar{u})\tilde{\theta}_{\bar{x}}\|_P, \\
&\quad \leq z^{\Theta^j}(x) \max_{\tilde{\theta}_x \in B(0,1)} \|G(x, \kappa(x, \bar{x}, \bar{u}))\tilde{\theta}_x - G(\bar{x}, \bar{u})\tilde{\theta}_x\|_P + \max_{\tilde{\theta}_x \in \tilde{\Theta}^j(x)} \|G(\bar{x}, \bar{u})\tilde{\theta}_x\|_P \\
&\qquad - \max_{\tilde{\theta}_{\bar{x}} \in \tilde{\Theta}^j(\bar{x})} \|G(\bar{x}, \bar{u})\tilde{\theta}_{\bar{x}}\|_P.
\end{aligned}$$

As demonstrated in Section 4.4.2, $z^{\Theta^j}(x)$ is Lipschitz continuous for all j with some Lipschitz constant denoted by L_{z^Θ}. Consequently, by additionally leveraging the Lipschitz continuity of $G(\bar{x}, \bar{u})$,

$$\tilde{w}_{\tilde{\Theta}^j,\mathcal{D}}(x, \kappa(x, \bar{x}, \bar{u})) - \tilde{w}_{\tilde{\Theta}^j,\mathcal{D}}(\bar{x}, \bar{u}) \leq L_{G,\kappa} z^{\Theta^j}(x) V_\delta(x, \bar{x}) + G^P_{max} L_{z^\Theta} V_\delta(x, \bar{x}).$$

Let

$$z^\Theta_{max} = \max_{x \in \mathcal{X}} z^{\Theta^j}(x)$$

denote the largest uncertainty set radius over the feasible state space and define

$$z^{\Theta^j}_{ave} = \frac{\int_\mathcal{X} z^{\Theta^j}(x)dx}{V(\mathcal{X})}$$

as the average value of $z^{\Theta^j}(x)$ over $\mathcal{X}$ where $V(\mathcal{X})$ denotes the volume of $\mathcal{X}$. Then, for all $x \in \mathcal{X}$,

$$z^{\Theta^j}(x) \leq \frac{z^\Theta_{max}}{z^\Theta_{min}} z^{\tilde{\Theta}^j}_{ave}.$$

Let ϵ_z denote a value satisfying $\epsilon_z z^{\utilde{\Theta}}_{min} \geq G^P_{max} L_{z^{\utilde{\Theta}}}$. Consequently, any $L_{\tilde{\utilde{\Theta}}^j}$ satisfying $L_{\tilde{\utilde{\Theta}}^j} \geq L_{G,\kappa} \frac{z^{\utilde{\Theta}}_{max}}{z^{\utilde{\Theta}}_{min}} z^{\tilde{\utilde{\Theta}}^j}_{ave} + \epsilon_z z^{\tilde{\utilde{\Theta}}^j}_{ave}$ gives (4.9b).

For (4.9c) we have

$$\tilde{w}_{\tilde{\utilde{\Theta}}^j,\mathcal{D}}(\bar{x},\bar{u}) = \max_{\tilde{\utilde{\theta}} \in \tilde{\utilde{\Theta}}^j(x)} \|G(\bar{x},\bar{u})\tilde{\utilde{\theta}}\|_P + \max_{d \in \mathcal{D}} \|d\|_P = z^{\tilde{\utilde{\Theta}}^j}(\bar{x}) \max_{\tilde{\utilde{\theta}} \in B(0,1)} \|G(\bar{x},\bar{u})\tilde{\utilde{\theta}}\|_P + \max_{d \in \mathcal{D}} \|d\|_P.$$

Also,

$$\tilde{w}_{\tilde{\utilde{\Theta}}^{j+1},\mathcal{D}}(\bar{x},\bar{u}) = z^{\tilde{\utilde{\Theta}}^{j+1}}(\bar{x}) \max_{\tilde{\utilde{\theta}} \in B(0,1)} \|G(\bar{x},\bar{u})\tilde{\utilde{\theta}}\|_P + \max_{d \in \mathcal{D}} \|d\|_P,$$

and

$$\tilde{w}_{\Delta\tilde{\utilde{\Theta}},\{0\}}(\bar{x},\bar{u}) = (z^{\tilde{\utilde{\Theta}}^j}(\bar{x}) - z^{\tilde{\utilde{\Theta}}^{j+1}}(\bar{x})) \max_{\tilde{\utilde{\theta}} \in B(0,1)} \|G(\bar{x},\bar{u})\tilde{\utilde{\theta}}\|_P.$$

Thus,

$$\tilde{w}_{\tilde{\utilde{\Theta}},\mathcal{D}}(\bar{x},\bar{u}) = \tilde{w}_{\tilde{\utilde{\Theta}}^+,\mathcal{D}}(\bar{x},\bar{u}) + \tilde{w}_{\Delta\tilde{\utilde{\Theta}},\{0\}}(\bar{x},\bar{u})$$

such that (4.9c) holds.

For (4.9d) define

$$\begin{aligned} L_{\tilde{\utilde{\Theta}}^j} &= a_{L_{\tilde{\utilde{\Theta}}}} \left(L_{G,\kappa} \frac{z^{\utilde{\Theta}}_{max}}{z^{\utilde{\Theta}}_{min}} z^{\tilde{\utilde{\Theta}}^j}_{ave} + \epsilon_{z^{\utilde{\Theta}}} z^{\tilde{\utilde{\Theta}}^j}_{ave} \right), \\ L_{\tilde{\utilde{\Theta}}^{j+1}} &= a_{L_{\tilde{\utilde{\Theta}}}} \left(L_{G,\kappa} \frac{z^{\utilde{\Theta}}_{max}}{z^{\utilde{\Theta}}_{min}} z^{\tilde{\utilde{\Theta}}^{j+1}}_{ave} + \epsilon_{z^{\utilde{\Theta}}} z^{\tilde{\utilde{\Theta}}^{j+1}}_{ave} \right), \\ L_{\Delta\tilde{\utilde{\Theta}}} &= a_{L_{\tilde{\utilde{\Theta}}}} \left(L_{G,\kappa} \frac{z^{\utilde{\Theta}}_{max}}{z^{\utilde{\Theta}}_{min}} z^{\Delta\tilde{\utilde{\Theta}}}_{ave} + \epsilon_{z^{\utilde{\Theta}}} z^{\Delta\tilde{\utilde{\Theta}}}_{ave} \right) \end{aligned}$$

for some $a_{L_{\tilde{\utilde{\Theta}}}} > 1$. From the definition of $z^{\tilde{\utilde{\Theta}}}_{ave}$,

$$z^{\tilde{\utilde{\Theta}}^j}_{ave} = z^{\tilde{\utilde{\Theta}}^{j+1}}_{ave} + z^{\Delta\tilde{\utilde{\Theta}}}_{ave}.$$

Consequently (4.9d) holds.

Requirement (4.9e) is satisfied by choosing a value of $a_{L_{\tilde{\utilde{\Theta}}}}$ large enough.

Hence, by adopting the adaptive scheme outlined in Sections 4.4.1 and 4.4.2, Assumptions 17 and 20 can both be satisfied.

4.5 Simulation Example

A simulated case study of the RAEILC framework as given by Algorithm 4 is now presented. Here a discretized cart-pendulum system is studied using a sample period of $0.04s$ with its dynamics given by

$$x_{t+1} = x_t + 0.04 \begin{bmatrix} \dot{\phi}_t \\ \frac{39.2 \sin\phi_t - 0.2\dot{\phi}_t - \cos\phi_t \dot{\phi}_t^2 \sin\phi_t + \cos\phi_t u_t}{8 - \cos^2\phi_t} \\ \dot{p}_t \\ \frac{9.8 \sin\phi_t \cos\phi_t - 0.05 \cos\phi_t \dot{\phi}_t + 2u - 2\dot{\phi}_t^2 \sin\phi_t}{8 - \cos^2\phi_t} \end{bmatrix} + 0.04 \begin{bmatrix} 0 \\ -\frac{\cos\phi_t \dot{p}_t}{8 - \cos^2\phi_t} \\ 0 \\ -\frac{2\dot{p}_t}{8 - \cos^2\phi_t} \end{bmatrix} \theta(\dot{p}_t) + d_t$$

with $x_t = \begin{bmatrix} \phi_t & \dot{\phi}_t & p_t & \dot{p}_t \end{bmatrix}^\top$ where ϕ_t is the pendulum angle, $\dot{\phi}_t$ is the pendulum angular velocity, p_t is the cart position, and $\dot{p}_t$ is the cart velocity. u_t corresponds to an applied input force to the cart. The parameter $\theta(\dot{p}_t)$ corresponds to the unknown drag coefficient of the cart, which varies as a function of the cart speed. The feasible state and input sets are given by the inequalities

$$\mathcal{X} = \left\{ x : |x| \le \begin{bmatrix} 0.52 & 1.57 & 200 & 2.5 \end{bmatrix}^\top \right\}, \ \mathcal{U} = \{ u : |u| \le 5 \}.$$

The noise, d_t is bounded with $M_d = 0.001$. The true model parameter $\theta(\dot{p})$ evolves with the cart speed according to

$$\theta(\dot{p}) = 0.1 e^{-\frac{8}{\dot{p}^2 + 0.01}} \dot{p}^2 + 0.9.$$

As $\theta(\dot{p})$ is unknown to the user, parameter uncertainty sets are constructed. As θ is only expected to vary as a function of the cart velocity, $\mathcal{X}$ is not partitioned along the ϕ, $\dot{\phi}$, or p dimensions of the state space. Instead $\mathcal{X}$ is only divided along the $\dot{p}$ dimension into $n_p = 100$ equally sized partitions. Consequently, the uncertainty sets are given as $\hat{\Theta}^j_{\mathcal{X}_i} = B(1, 0.99)$ for all $i \in \mathbb{I}_{[0, n_p - 1]}$. To execute the adaptive parameter identification scheme outlined in Section 4.4, the update law parameters are chosen as $K_\omega = 0.95$ and $\beta = 0.01$.

The control objective is to maximize the position of the cart after $n_t = 201$ timesteps with an additional penalty placed on any deviations of the terminal state from the motionless, upright pendulum and cart equilibrium. Specifically, the cost function is given by $\ell(x, u) = -0.04\dot{p}$ and $V_f = \phi^2 + \dot{\phi}^2 + \dot{p}^2$.

The feedback law $\kappa(x, \bar{x}, \bar{u}) = \bar{u} + K(\bar{x}, \bar{u})(x - \bar{x})$ and matrix P are calculated by solving a group of linear matrix inequalities as per [78] with a contraction factor $\rho_{\underline{\theta}^0} = 0.9$. The algorithm is applied over 10 iterations and compared to a variant of the methodology described in [78] that has been modified from a receding horizon to a shrinking horizon formulation.

First, we compare the ability of the two adaptive methodologies to identify the true model parameters. As [78] is unable to directly address the dependence of the unknown model parameter on velocity, the impact of model parameter variations over $\mathcal{X}$ is instead treated as noise in this case. Specifically, using this adaptation methodology, $\mathcal{X}$ contains only a single partition, $\mathcal{X}_0$, which requires the value of $M_w^{\mathcal{X}_0}$ to be increased. In the context of (4.46), this makes a reduction in $z_t^{\hat{\Theta}}$ more difficult to achieve, which means that the adaptive algorithm in [78] requires additional conservatism in the parameter estimation. This effect is observed in Fig. 4.4. Here, although both algorithms are initiated with the same uncertainty set $\Theta^0(x)$ for all $x \in \mathcal{X}$, the RAEILC algorithm is able to more confidently estimate the model parameters by iteration 10 as shown by the reduction in $z^{\Theta^j}(x)$. Additionally, we see that the largest reductions in $z^{\Theta^j}(x)$ from its initial value $z^{\Theta^0}(x)$ occur at larger values of $\dot{p}$. This is because the stage cost encourages the cart to maximize its velocity. Consequently, a large majority of the historical state data that is available for implementing the adaptive scheme corresponds to high velocities, which allows more learning to occur in these partitions of the state space. Meanwhile, using the adaptation method from [78] requires so much additional conservatism from the controller in order to compensate for the state-dependent variations in the unknown model parameters that the uncertainty set $\Theta^j(x)$ does not shrink in size after 10 iterations. Consequently, any potential benefits to be gained by augmenting the controller with an adaptive algorithm are not realized.

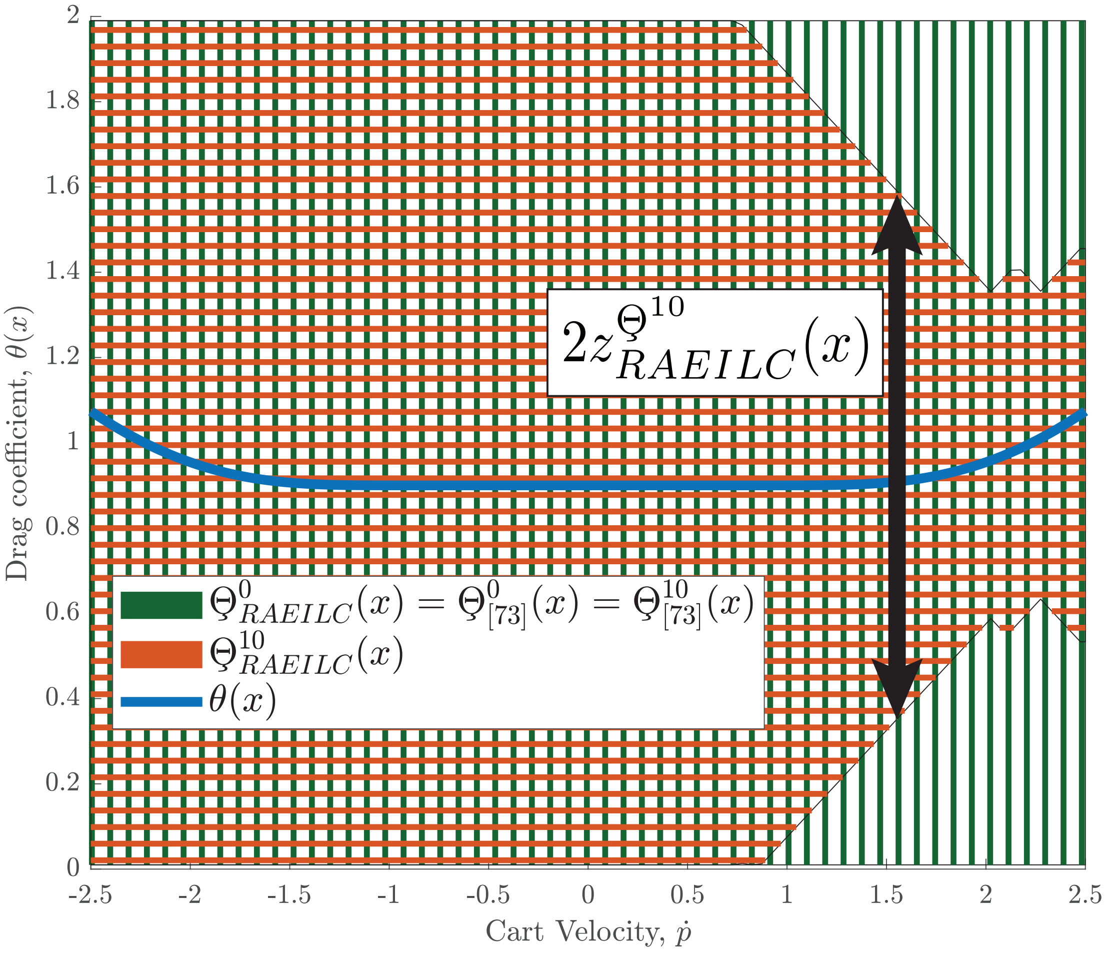


Figure 4.4: Comparing the evolution of $\Theta^{j}(x)$ when using the proposed adaptive scheme (area covered by orange hatched lines) versus an adaptive strategy that does not take state-dependent variations in the unknown model parameters into account (area covered by green hatched lines). After the 10^{th} iteration, both regions robustly contain the true model parameters (blue line), but the proposed adaptive scheme is less conservative. Specifically, the sets produced by $\Theta^{j}_{RAEILC}(x)$ are smaller than those produced by $\Theta^{j}_{[73]}(x)$, particularly at higher cart velocities.

A comparison of the state and input trajectories of the cart that were measured during application of the two control schemes is shown in Fig. 4.5. Here, by incorporating the partitioned uncertainty set adaptation and using the $q-$tube in conjunction with the $s-$tube, the RAEILC algorithm is able to apply more aggressive inputs in comparison to the control scheme outlined in [78]. Consequently, the cart is able to achieve greater displacement while the system states and input remain within the feasible set at all times. As a result, the cart is able to move 9.4% further at the final iteration in comparison to the first iteration. Moreover, whereas the iteration-averaged economic cost was -11.9 when using the controller from [78], the RAEILC incurred an average cost of -12.8 over the 10 iterations.

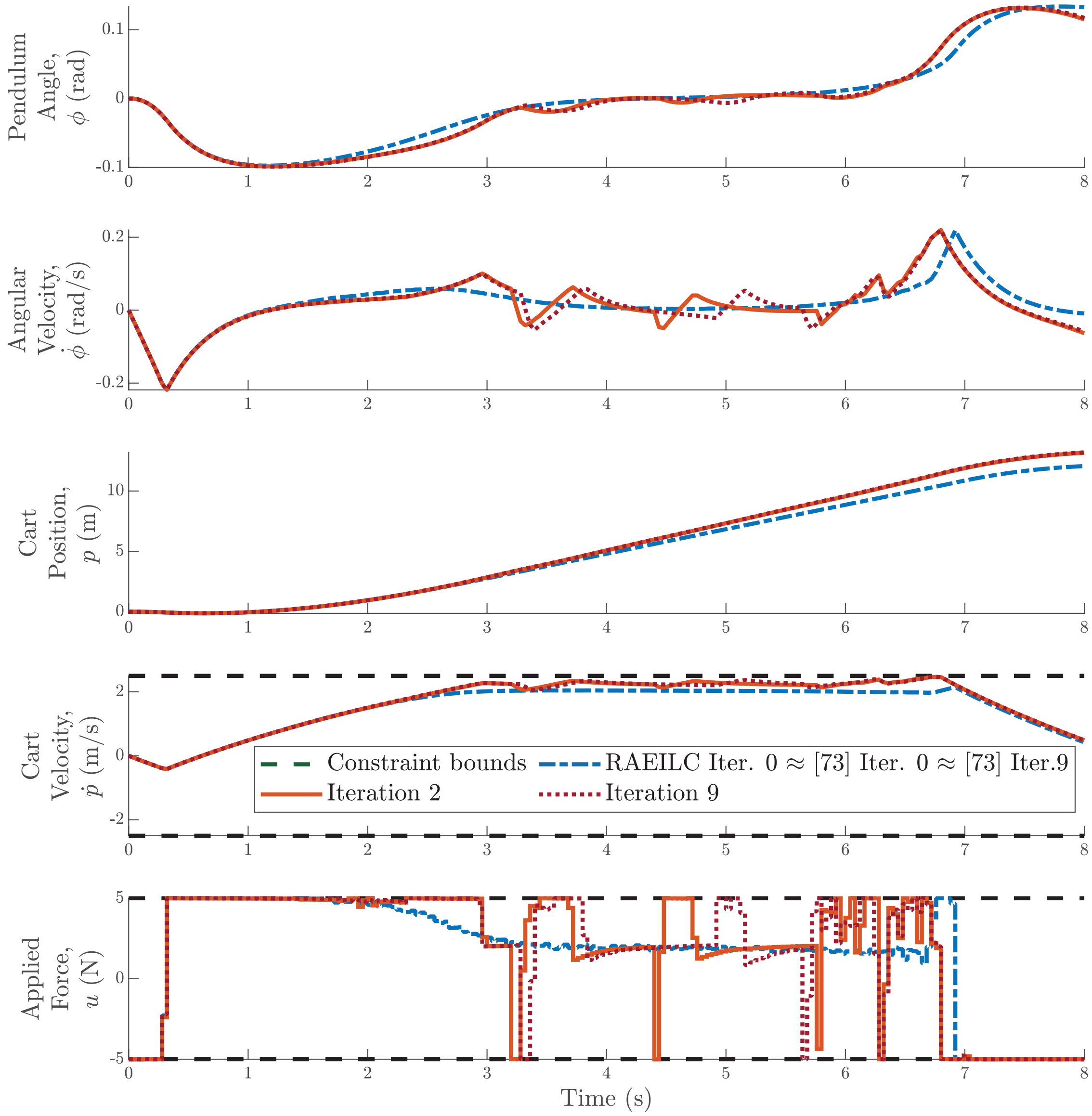


Figure 4.5: The state and input trajectories of the cart-pendulum system. The constraint boundaries (dashed black lines) are not shown for the states ϕ, $\dot{\phi}$, or p, as the cart position never neared these upper or lower limits. As the adaptive method used in [73] was unable to reduce the size of the adaptive uncertainty sets after 10 iterations and does not enable reduced conservatism through the use of the $q-$tube, no significant change in the control signal (ouside of the effects of noise) was observed between Iteration 0 and Iteration 9 in this case.

4.6 Conclusions

To enable improved economic control of uncertain nonlinear systems that perform repetitive, discontinuously operated processes, an RAEILC algorithm has been described. The ability of the framework to directly address the existence of unknown state-varying model parameters enable it to more accurately estimate the true system dynamics and consequently outperform similar tube-based MPC strategies. Moreover, by leveraging data from previous task executions to relax standard robust tube constraints, performance degradation caused by limits to the accuracy of the model parameter estimates can be further mitigated. Despite the existence of this uncertainty and noise, the proposed approach maintains recursive feasibility and convergence of the benchmark control signal and integrated cost.

The development of a supplementary adaptive algorithm provides a computationally tractable method for estimating the unknown model parameters with improved accuracy. Specifically, rather than assuming that the parameters are constant over the feasible state space, the parameter estimate update law treats the parameters as *piecewise* constant over the feasible state space. The efficacy of this framework with regards to model parameter identification and system performance was then demonstrated numerically to a simulated pendulum cart system.

The identification of methods to reduce the rate of growth of the $q-$tube remains a point of future investigation. Moreover, for systems with long trial durations wherein the value of n_t is large, the number of decision variables and constraints may become too large for $R(\underset{\varsigma}{\Theta}^j, x_t^j, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$ to be solved in real time. Consequently, methods for addressing this issue through the incorporation of terminal costs and constraints that enable reductions in the length of the prediction horizon are of additional interest for future research.

CHAPTER 5

Conclusions and Future Research Directions

The vast majority of engineered devices and systems are designed to complete a limited set of tasks many times in a repetitive fashion. For these systems, achieving high performance during each task execution becomes crucial as any suboptimality in performance would accumulate over the device's lifetime, resulting in unrealized utility. Moreover, with the advent of new and more advanced technologies, the design complexity and required capabilities of engineered systems will continue to grow. Hence, traditional control approaches that: 1) rely on system linearity, and 2) are designed solely to improve reference tracking behavior, restrict the achievable performance of these systems. In other words, nonlinearity in the plant dynamics, economic objectives wherein performance is driven by metrics that extend beyond reference tracking error minimization, and operational constraints placed on the system must all be considered to enable high performing control. As uncertainty exists in any practical controls application, whether in the form of modeling errors or noise and disturbances, addressing each of these systems properties simultaneously in a robust manner becomes an arduous task. Consequently, there exists a fundamental tradeoff between achieving good performance and maintaining robustness. This tradeoff is further accentuated when the amount of uncertainty in the system is large.

To reduce these impacts of uncertainty, learning-based approaches represent an enticing strategy for controller design. Here, the fundamental notion is that if new knowledge about a system can be developed based on available data, then a controller can be more assertive in its decision making. In other words, accompanying control algorithms with a learning component can reduce the required performance sacrifices that must be made for the sake of robustness. Fortunately, with the rapidly improving developments in the fields of sensing, data processing, and data storage, information is becoming more and more accessible. Consequently, research into formalized learning strategies has enjoyed a rapid increase in popularity in recent years [101]. For the particular case where the system of interest operates repetitively, repetitive control (RC) and iterative learning control (ILC) have served as two primary benchmarks for learning-based methods within the controls research commmunity.

In this dissertation, two popular classes of repetitive systems within the RC and ILC literature were investigated: continuously operated and discontinuously operated systems, which are distinguished based on the existence of an offline phase between task executions. By exploiting the repetitive nature of these systems and historical data available from previous iterations/cycles, new methods for learning-based control have been developed to improve economic performance while remaining cognizant of system constraints. Specifically, the following contributions have been made:

Contribution 1 - As described in Chapter 2, a novel methodology for learning-based control of discontinuously operated systems is developed based on a numerical method originating from the optimization research community. Specifically, by drawing parallels between the iterative nature of numerical optimization techniques and the repetitive behavior of the systems of interest, a mapping of a filter-based sequential quadratic programming (SQP) algorithm to an implementable control law is created. Here, through the careful synthesis of measured data available from previous task executions and (potentially inaccurate) model-based gradient information, the effects of uncertainty can be mitigated to achieve good performance while broadening the class of constraints and performance objectives that are addressed within the ILC literature.

Contribution 2 - As given in Chapter 3, a control algorithm is proposed that combines the design flexibility of model predictive control (MPC) and learning capabilities of adaptive control for application to economically driven, continuously operated repetitive systems. Specifically, a novel parametric uncertainty set update is generated for the case wherein system uncertainties appear periodically, and is integrated with a new economic MPC control law to ensure robust constraint enforcement. Sufficient conditions to ensure the recursive feasibility and robust convergence of the system's economic performance have been developed. Here, for a standard class of systems addressed within the RC community, uncertainty mitigation is enabled through improved estimates of unknown model parameters.

Contribution 3 - In Chapter 4, the advancements made within Contribution 2 are extended to discontinuously operated uncertain systems. Specifically, an adaptive methodology for identifying unknown state-varying model parameters is developed and combined with a shrinking horizon controller that is used to robustly enforce system constraints and promote good economic system performance. Moreover, by leveraging the resettability of the system state between task executions and historical state and input data, learning is further encouraged to reduce the conservatism incurred by purely model-based control methods. Conditions for which recursive feasibility and robust monotonic reductions in the system cost can be guaranteed have also been presented. Therefore, for systems commonly encountered within the ILC literature, methods for addressing system constraints and economic objectives under a new class of uncertainties has been developed.

5.1 Limitations and Future Research Directions

Through Contributions 1-3, this dissertation has outlined advancements in the field of learning-based control for repetitive systems. With the development of the sequential quadratic programming based iterative learning controller (SQP-ILC) outlined in Chapter 2, the robust adaptive economic model predictive controller (RAEMPC) described in Chapter 3, and robust adaptive economic iterative learning controller (RAEILC) presented in Chapter 4, the economic objectives of constrained nonlinear systems that operate both continuously and discontinuously are able to be addressed. While this work has progressed the applicability of RC and ILC to a broader range of systems, the presented methodologies still have limitations that offer avenues for future research.

First, in each of the SQP-ILC, RAEMPC, and RAEILC controllers, smoothness of the dynamics and performance costs is assumed in order to establish various theoretical properties of the control schemes. While this is a standard assumption in a large portion of the control theory literature, this requirement may be unreasonable or undesirable in practice. Consequently, the ability to incorporate discontinuities in the system dynamics through the use of hybrid models may be needed to enable the development of both simpler and more accurate models in a manner that purely smooth dynamic models cannot. As hybrid models are particularly prone to large prediction errors due to modeling uncertainty, the augmentation of optimization-based control schemes with learning-based techniques is an attractive strategy. Overcoming this smoothness assumption is nontrivial, but if achieved, can significantly extend the applicability of the developed controllers. A potential avenue for addressing this issue includes the use of mixed-integer programming, which has previously been applied to hybrid systems within the field of MPC [102].

Second, while the presented control schemes permit nonlinear system dynamics and objective functions, doing so can result in the construction of nonconvex control problems. Consequently, the proposed controllers are susceptible to convergence to local optima of system performance. Moreover, the numerical complexity of solving nonconvex optimization problems limits the ability to apply the RAEMPC and RAEILC algorithms in real time except for low-order systems or systems whose dynamics evolve over long timescales. Methods for bypassing the effects of nonconvexity remain a point of future research. One potential solution would be to leverage a linearized approximation of the system dynamics and convex approximation of the objective function, but a quantitative assessment on the effect of these approximations on closed-loop performance would need to be formally conducted.

Third, each of the SQP-ILC, RAEMPC, and RAEILC controllers is based on the assumption that the amount of uncertainty in the system dynamics is bounded and that this bound is known by the user *a priori*. However, establishing this upper bound can be a nontrivial task whose difficulty is exacerbated if the system is too complex for a physics-based uncertainty assessment to be made, or if the amount of uncertainty encountered by the system is state-dependent. In these cases, an upper bound may only be determined by conducting an arduous uncertainty analysis process or by performing a large number of potentially expensive experiments/simulations. While this limitation can be addressed by setting the upper bound to be very large, doing so can result in significant overconservatism and unnecessarily poor performance. Moreover, the proposed control schemes only consider the case where disturbances introduced by the environment to the plant dynamics are purely exogenous. However, in many practical applications, not only does the environment influence the plant behavior, but the plant behavior causes dynamic changes in the environment as well. Consequently, simply treating the environment as a purely external influence limits the achievable utility of the proposed control algorithms. Rather, in such cases it is reasonable to model the environment as another dynamic system. At present, problems wherein both the plant and the environment are considered to be dynamic agents are perhaps most appropriately addressed within the field of game theoretic control. An initial investigation into incorporating learning for economically-driven repetitive systems based on game theoretic methods is presented in [103] and provided in this dissertation within Appendix C. Here the problem of maximizing the power production of a wind farm is considered, where not only are the dynamics and energy generation capabilities of the turbines dependent upon the environmental wind field, but the propagation of low-speed wakes in the wind field is influenced by the controlled orientation of the turbines. The integration of learning-based control strategies within a game-theoretic problem structure for repetitive system applications is a relatively unexplored area within the RC and ILC literature, and presents a particularly intriguing pathway for further research.

Finally, validation of all of these strategies on physical systems should also be performed to provide a meaningful assessment of the assumptions made in this dissertation. While the simulation case studies presented in Chapters 2-4 have been embedded with various features to mimic the challenges that frequently arise when controlling physical systems, such as modeling errors, nonlinear dynamics, disturbances, and noise, these case studies are limited to low-dimensional systems. For more practical systems with potentially unmodeled high-order dynamics, the impact of unmodeled dynamics on system behavior can potentially deteriorate the economic performance guarantees outlined in this dissertation. Consequently, methods for bounding the impact of these unmodeled dynamics and translating these behaviors as noise terms in the system model remains a point of future work. Moreover, in Chapter 2, it was assumed that accurate measurements or calculations of the system constraint functions can be obtained from experimental data, while in Chapters 3 and 4 it was assumed that the system state can be accurately determined in real time. However, limitations on the availability of accurate sensor data can preclude these assumptions from being met. In particular, the presence of sensor noise and drift, resolution limitations, and challenges with measurement availability and signal delays can all challenge the validity of these assumptions and limit the practical effectiveness of the proposed control schemes. Additionally, while data processing and storage technologies have enjoyed significant advances in recent years, employing the control algorithms presented in this dissertation in real time may be difficult. For instance, the RAEMPC and RAEILC controllers both rely on the availability of the solution to a nonlinear program at each timestep of system operation. For complex or high-dimensional systems, solving such nonlinear programs during online operation may be too difficult. Consequently, the identification of strategies that intelligently leverage warm-starting of the numerical optimization solvers and approximate solutions to the various optimization problems posed in this dissertation are of particular interest to minimize the computational demands placed on the embedded controller.

5.2 Broader Impacts

The work presented in this dissertation has several ramifications within the field of learning-based controls research and has effects on the future of modern controls work. First, this work extends the range of systems that are addressed within the RC and ILC communities. Particularly, constrained control in these fields has been confined to a small class of constraints or system types, which has greatly restricted the practical applicability of these control strategies. By bridging the theoretical gap between RC/ILC, which have had limited application in industry to date, and strategies that have enjoyed greater industrial popularity, such as MPC, a wider adoption of learning-based control is possible. Here, RC and ILC are not meant to serve as complete replacements of standard control tools, but can be used, to some degree, in conjunction with more established control strategies to improve their performance.

Additionally, the SQP-ILC algorithm presented in Chapter 2 provides a benchmark framework for which other ILC strategies based on numerical optimization methods can be compared against. The numerical strategy in [39] is only one optimization method upon which other ILC frameworks may be based, and other methods may prove to be more efficient or enable improved performance for particular applications. However, the methodology with which information from data and model-based prediction is synthesized in Chapter 2 can act as a paradigm for future controller development.

Further, the adaptation schemes outlined in Chapters 3 and 4 enable extensions to systems with time-varying or spatiotemporally varying model parameters. The identification of time-varying model parameters has been a persistent challenge within the adaptive controls community [104], but by considering updates to parameteric uncertainty *sets* rather than solely updating estimates of the unknown model parameters themselves, issues resulting from parameter drift or time-dependence effects are perhaps more readily addressed while also facilitating integration with

MPC control schemes.

While the technical contributions of this dissertation have taken steps to advance the state of control theory, the results of this work also enable learning-based control to be used in a broader practical application space than previously achievable. Broadly speaking, the energy industry provides a wealth of applications for learning-based control of repetitive systems. A particularly exciting potential implementation of this work is the control of Airborne Wind Energy (AWE) systems operating in spatiotemporally varying wind environments that seek to maximize the amount of energy that they generate [105]. Modeling these systems with a high degree of accuracy is difficult on its own, but doing so while remaining amenable to real-time control is even more so if not impossible. When only low fidelity models of the AWE system and its operating environment are available, learning-based methods have already shown great promise [106, 107], and the methods outlined in this dissertation can be used to further augment the energy generation capabilities of these systems. For instance, the adaptive scheme outlined in Chapter 3 may be used to robustly identify a periodic wind disturbance that influences both the dynamics and energy generation capabilities of the AWE system. In other areas, tide-driven hydrokinetic energy and solar energy systems that operate under repetitive power production cycles, as well as energy management systems that must navigate periodic electricity demands all provide potential platforms for which the methods developed in this dissertation may be deployed.

Meanwhile, fully autonomous and semi-autonomous driving systems present an intriguing application space for this work, where safety constraints must be rigorously enforced and a variety of different economics objectives may exist (for instance, maximizing fuel efficiency [108] or rider comfort [109], or minimizing travel time [110]). The wealth of sensor data that is collected by these systems present exciting opportunities for extensions of the work presented in this dissertation, where data available from previous trips or retrieved via communication from other vehicles following similar paths along the road can enable performance improvements to be achieved. For instance, historical information about spatially-varying road grades or conditions could be leveraged by the adaptive scheme in Chapter 4 to reduce the effects of uncertainty on the decisions made by the RAEILC controller, thereby enabling more aggressive control actions to be taken while still maintaining safe operation of the vehicle.

Finally, the manufacturing industry is another area where the products of this dissertation may contribute. Here, given the highly regulated, structured, and self-contained nature of many manufacturing systems, this application space may prove to be one where the insights brought forth by this dissertation can have the most immediate impact. Specifically, the trial-to-trial similarity in the operating conditions of these systems could enable an SQP-ILC controller to reduce the time required to complete the production of a part such that throughput rate objectives can be maximized [111], or encourage more sustainable practices by reducing the emissions or waste of materials that arise during the manufacturing process [112].

The results of this dissertation not only appear as improvements within the field of learning-based control theory, but also lay a foundation for enabling effective economic control in a diverse set of application spaces. By addressing the challenge of improving economic performance in repetitive systems and highlighting the benefits of incorporating learning within controller development, this research opens up opportunities for further exploration and innovation. The findings and methodologies presented here can inspire future studies in controller design across various domains, both in terms of advancing control theory and improving the utility of the technologies of the future.

APPENDIX A

Proofs for SQP-ILC Lemmas

A.1 Proof of Lemma 4

Lemma 4 is proven as follows.

Proof. Since $j \notin \mathcal{R}$, this implies that TRQP(x_j, Δ_j) is compatible. Therefore $\boldsymbol{n}_j$ must satisfy (2.12) such that

$$\|\boldsymbol{n}_j\| \leq \kappa_\Delta \kappa_\mu \Delta_j^\mu.$$

Hence, combining with (2.22) gives

$$\theta_j \leq \frac{\kappa_\Delta \kappa_\mu}{\kappa_{lsc}} \Delta_j^\mu.$$

Thus (2.24) is demonstrated for any $\kappa_{ubt} \geq \frac{\kappa_\Delta}{\kappa_{lsc}}$.

The i^{th} constraint for the real system at $\boldsymbol{x}_j + \boldsymbol{s}_j$ can be written as

$$c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j) = c_i^R(\boldsymbol{x}_j) + \nabla c_i^R(\boldsymbol{x}_j)^\top \boldsymbol{s}_j + \frac{1}{2}\boldsymbol{s}_j^\top \nabla^2 c_i^R(\zeta_j)\boldsymbol{s}_j$$

for $i \in E \cup I$, which is obtained by applying Assumption 1 and the mean value theorem and where ζ_j lies on the line segment $[\boldsymbol{x}_j, \boldsymbol{x}_j + \boldsymbol{s}_j]$. From (2.9b) and (2.9c) we have

$$\begin{aligned} c_i^R(\boldsymbol{x}_j) + \nabla c_i^M(\boldsymbol{x}_j)^\top \boldsymbol{s}_j &= 0, i \in E, \\ c_i^R(\boldsymbol{x}_j) + \nabla c_i^M(\boldsymbol{x}_j)^\top \boldsymbol{s}_j &\leq 0, i \in I. \end{aligned}$$

Thus,

$$\begin{aligned} c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j) =& \frac{1}{2}\boldsymbol{s}_j^\top \nabla^2 c_i^R(\zeta_j)\boldsymbol{s}_j + (\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j))^\top \boldsymbol{s}_j, i \in E, \\ c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j) \leq& \frac{1}{2}\boldsymbol{s}_j^\top \nabla^2 c_i^R(\zeta_j)\boldsymbol{s}_j + (\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j))^\top \boldsymbol{s}_j, i \in I. \end{aligned}$$

The triangle inequality then gives

$$|c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j)| \leq \left| \frac{1}{2}\boldsymbol{s}_j^\top \nabla^2 c_i^R(\zeta_j)\boldsymbol{s}_j \right| + \left| (\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j))^\top \boldsymbol{s}_j \right|$$

for $i \in E \cup I$. From the Cauchy-Schwarz inequality, we have

$$\begin{aligned} |c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j)| \leq& \frac{1}{2}\|\nabla^2 c_i^R(\zeta_j)\|\|\boldsymbol{s}_j\|^2 + \|\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j)\|\|\boldsymbol{s}_j\|, \\ \leq& \frac{1}{2}\max_{\boldsymbol{x}\in X}\|\nabla^2 c_i^R(\boldsymbol{x})\|\|\boldsymbol{s}_j\|^2 + \|\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j)\|\|\boldsymbol{s}_j\|. \end{aligned}$$

From the triangle inequality, we have that

$$\|\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j)\| \leq \|\nabla c_i^R(\boldsymbol{x}_j)\| + \|\nabla c_i^M(\boldsymbol{x}_j)\|$$

for all $\boldsymbol{x} \in \mathcal{X}$. Invoking Assumption 2 then gives

$$\|\nabla c_i^R(\boldsymbol{x}_j) - \nabla c_i^M(\boldsymbol{x}_j)\| \leq M^{P_1} \leq M^{R_1} + M^{M_1}$$

for some constant M^{P_1} and

$$\max_{\boldsymbol{x}\in X}\|\nabla^2 c_i^R(\boldsymbol{x})\| \leq M^{R_2}.$$

Therefore

$$|c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j)| \leq \frac{1}{2}M^{R_2}\|\boldsymbol{s}_j\|^2 + M^{P_1}\|\boldsymbol{s}_j\|.$$

Constraint (2.9d) then gives

$$\theta(\boldsymbol{x}_j + \boldsymbol{s}_j) \leq |c_i^R(\boldsymbol{x}_j + \boldsymbol{s}_j)| \leq \frac{1}{2}M^{R_2}\Delta_j^2 + M^{P_1}\Delta_j$$

for all $i \in E \cup I$ which gives (2.25) for

$$\kappa_{ubt} = \max\left[\frac{1}{2}M^{R_2}, \frac{\kappa_\Delta \kappa_\mu}{\kappa_{lsc}}\right].$$

□

A.2 Proof of Lemma 9

Lemma 9 is proven as follows.

Proof. Because $\theta_j \leq \delta_n$, we know from Assumption 3 and Lemma 1 that

$$\kappa_{lsc}\theta_j \leq \|\boldsymbol{n}_j\| \leq \kappa_{usc}\theta_j. \tag{A.1}$$

Assume for the purpose of deriving a contradiction that $j \in \mathcal{R}$. From (2.12) and (2.31), this means that

$$\|\boldsymbol{n}_j\| > \kappa_\Delta \kappa_\mu \Delta_j^\mu. \tag{A.2}$$

As the algorithm is designed such that the restoration step is never run in consecutive iterations, this implies that $j-1 \notin \mathcal{R}$.

Suppose that iteration $j-1 \notin \mathcal{S}$. Lemma 8 gives

$$\boldsymbol{J}(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}) \leq \boldsymbol{J}(\boldsymbol{x}_{j-1}) - \gamma_\theta \theta_{j-1}. \tag{A.3}$$

Since by nature of the algorithm $\boldsymbol{x}_{j-1}$ is acceptable for the filter at the beginning of iteration $j-1$, $j-1 \notin \mathcal{S}$ implies that $\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}$ is not acceptable for the filter. Since (A.3) is satisfied, this implies from (2.18) that

$$\theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}) > (1 - \gamma_\theta)\theta_{j-1}.$$

Since $j-1$ is unsuccessful, $\boldsymbol{x}_j = \boldsymbol{x}_{j-1}$ and therefore $\theta_j = \theta_{j-1}$ which gives

$$\theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}) > (1 - \gamma_\theta)\theta_j.$$

Lemma 4 gives that

$$\theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}) \leq \kappa_{ubt}\Delta_{j-1}^2 + M^{P_1}\Delta_{j-1}.$$

Hence

$$(1 - \gamma_\theta)\theta_j < \kappa_{ubt}\Delta_{j-1}^2 + M^{P_1}\Delta_{j-1}.$$

Since, by nature of the algorithm, $\Delta_{j-1} \leq \frac{\Delta_j}{\gamma_0}$

$$(1-\gamma_\theta)\theta_j < \kappa_{ubt}\frac{\Delta_j^2}{\gamma_0^2} + M^{P_1}\frac{\Delta_j}{\gamma_0}.$$

Applying (A.1) and (A.2) gives

$$\kappa_\Delta \Delta_j < \kappa_{usc}\theta_j < \frac{\kappa_{usc}}{1-\gamma_\theta}\left(\kappa_{ubt}\frac{\Delta_j^2}{\gamma_0^2} + M^{P_1}\frac{\Delta_j}{\gamma_0}\right).$$

Dividing by Δ_j and rearranging terms yields

$$\frac{\kappa_{usc}}{1-\gamma_\theta}\left(\kappa_{ubt}\frac{\Delta_j}{\gamma_0^2} + \frac{M^{P_1}}{\gamma_0}\right) - \kappa_\Delta > 0.$$

However, this violates Assumption 6 and (2.31). Thus iteration $j-1$ must be successful, which implies that

$$\theta_j = \theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}).$$

From the assumption that $j \in \mathcal{R}$ and Assumption 3, this gives

$$\kappa_\Delta \Delta_j < \|\boldsymbol{n}_j\| \leq \kappa_{usc}\theta_j = \kappa_{usc}\theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}).$$

Again, we reuse Lemma 4 and the fact that for any iteration $\Delta_{j-1} \leq \frac{\Delta_j}{\gamma_0}$ to find

$$\begin{aligned}\kappa_\Delta \Delta_j &< \kappa_{usc}\theta(\boldsymbol{x}_{j-1} + \boldsymbol{s}_{j-1}),\\ &\leq \kappa_{usc}\left(\kappa_{ubt}\Delta_{j-1}^2 + M^{P_1}\Delta_{j-1}\right),\\ &\leq \kappa_{usc}\left(\kappa_{ubt}\frac{\Delta_j^2}{\gamma_0^2} + M^{P_1}\frac{\Delta_j}{\gamma_0}\right).\end{aligned}$$

Hence

$$\kappa_{usc}\left(\kappa_{ubt}\frac{\Delta_j^2}{\gamma_0^2} + M^{P_1}\frac{\Delta_j}{\gamma_0}\right) - \kappa_\Delta \Delta_j > 0.$$

But Assumption 6, (2.31) and the fact that $(1-\gamma_\theta) < 1$ show that this is not possible. Therefore, the assumption that $j \in \mathcal{R}$ is false, so $j \notin \mathcal{R}$. □

APPENDIX B

Proofs for RAEMPC Lemmas

B.1 Proof of Lemma 13

The proof of Lemma 13 is given as follows.

Proof. From (3.17), we have that

$$(\bar{\tilde{\theta}}^i_{j+1})^\top \Sigma^i_{j+1} \bar{\tilde{\theta}}^i_{j+1} - (\tilde{\theta}^i_j)^\top \Sigma^i_j \tilde{\theta}^i_j \leq (\tilde{\theta}^i_{j+1})^\top \Sigma^i_{j+1} \tilde{\theta}^i_{j+1} - (\tilde{\theta}^i_j)^\top \Sigma^i_j \tilde{\theta}^i_j = V_{\tilde{\theta}^i_{j+1}} - V_{\tilde{\theta}^i_j}.$$

From the definition of $\tilde{\theta}^i_j$ and (3.15), we have that

$$\tilde{\theta}^i_{j+1} = \tilde{\theta}^i_j - (\Sigma^i_j)^{-1}(\omega^i_j)^\top (I + \omega^i_j (\Sigma^i_j)^{-1} (\omega^i_j)^\top)^{-1} (\omega^i_j \tilde{\theta}^i_j + \tilde{\eta}^i_j),$$

upon which substitution of (3.13) yields

$$\tilde{\theta}^i_{j+1} = (\Sigma^i_{j+1})^{-1} \Sigma^i_j \tilde{\theta}^i_j - (\Sigma^i_j)^{-1} (\omega^i_j)^\top (I + \omega^i_j (\Sigma^i_j)^{-1} (\omega^i_j)^\top)^{-1} \tilde{\eta}^i_j. \tag{B.1}$$

Substitution of (3.12), and the definitions of η^i_j and $\tilde{\eta}^i_j$ into (B.1) yields

$$V_{\tilde{\theta}^i_{j+1}} - V_{\tilde{\theta}^i_j} = -(\tilde{x}^i_j - \hat{\eta}^i_j)^\top \left(I + \omega^i_j (\Sigma^i_j)^{-1} (\omega^i_j)^\top\right)^{-1} (\tilde{x}^i - \hat{\eta}^i_j) + (\tilde{\eta}^i_j)^\top \tilde{\eta}^i_j \tag{B.2}$$

which gives the first claim. Additionally, (B.2) gives

$$\begin{aligned} \lim_{j\to\infty} V_{\tilde{\theta}^i_j} =& V_{\tilde{\theta}^i_0} + \sum_{j=0}^{\infty} \Big((\tilde{\eta}^i_j)^\top \tilde{\eta}^i_j - (\tilde{x}^i_j - \hat{\eta}^i_j)^\top \left(I + \omega^i_j (\Sigma^i_j)^{-1} (\omega^i_j)^\top\right)^{-1} (\tilde{x}^i - \hat{\eta}^i_j) \Big), \\ \leq& V_{\tilde{\theta}^i_0} + \sum_{j=0}^{\infty} \left((\tilde{\eta}^i_j)^\top \tilde{\eta}^i_j - \gamma^i (\tilde{x}^i_j - \hat{\eta}^i_j)^\top (\tilde{x}^i - \hat{\eta}^i_j) \right). \end{aligned} \tag{B.3}$$

As a consequence of the first condition of (3.18), the right hand side of (B.3) is finite such that a limit of $V_{\tilde{\theta}^i_j}$ exists. The second condition of (3.18) then gives the second claim. □

B.2 Proof of Lemma 14

The proof of Lemma 14 is given as follows.

Proof. Claim (3.22) will be demonstrated by contradiction. First suppose that $\Theta^i_{j+1} \not\subseteq \Theta^i_j$. Then, given the definition of Θ^i_j, there exists a $\hat{\theta} \in \Theta^i_{j+1}$ such that

$$\|\hat{\theta} - \hat{\theta}^i_j\| > z^{\Theta^i_j}. \tag{B.4}$$

However, if $\Theta^i_{j+1} \neq \Theta^i_j$, Algorithm 2 requires that

$$\begin{aligned}\|\hat{\theta} - \hat{\theta}^i_j\| &\leq \|\hat{\theta} - \hat{\theta}^i_{j+1}\| + \|\hat{\theta}^i_{j+1} - \hat{\theta}^i_j\|,\\ &\leq z^{\Theta^i_{j+1}} + \|\hat{\theta}^i_{j+1} - \hat{\theta}^i_j\| \leq z^{\Theta^i_j}\end{aligned}$$

which contradicts (B.4). Therefore claim (3.22) holds.

We now note that as a consequence of the property $(\tilde{\eta}^i_j)^\top \tilde{\eta}^i_j \leq \left(\frac{v^{max}}{K_\omega}\right)^2$ with $\tilde{\theta}^i_0 \leq z^{\Theta^i_0}$, that (3.19) and (3.21) imply that $V_{\tilde{\theta}^i_j} \leq V_{z^{\Theta^i_j}}$. Hence,

$$\|\tilde{\theta}^i_j\|^2 \leq \frac{V_{\tilde{\theta}^i_j}}{\lambda_{min}(\Sigma^i_j)} \leq \frac{V_{z^{\Theta^i_j}}}{\lambda_{min}(\Sigma^i_j)} = (z^{\Theta^i_j})^2.$$

Therefore, $\tilde{\theta}^i_j \leq z^{\Theta^i_j}$ for all j, which gives (3.23). □

B.3 Proof of Lemma 21

The proof of Lemma 21 is given as follows.

Proof. We first expand $J^{\text{aux}}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom})$ as

$$\begin{aligned}&J^{\text{aux}}_{j+1}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom})\\ &\quad= \sum_{t=0}^{K_h-1} \Big(s_{j+1}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}^{nom}}_{\hat{\theta}_{j+1}}(tP, \bar{x}_{jP+P}), \mathring{\bar{u}}^{nom}_{tP}) - \mathring{\lambda}(f^P_{\hat{\theta}_{j+1}}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}^{nom}}_{\hat{\theta}_{j+1}}(tP, \bar{x}_{jP+P}), \mathring{\bar{u}}^{nom}_{tP}, 0))\\ &\qquad+ \mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}^{nom}}_{\hat{\theta}_{j+1}}(tP, \bar{x}_{jP+P}))\Big) + \mathring{\bar{V}}^f(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}^{nom}}_{\hat{\theta}_{j+1}}(n_h, \bar{x}_{jP+P})),\\ &\quad\leq \sum_{t=0}^{K_h-1} \Big(s_j(\mathring{\bar{x}}^{\bar{\boldsymbol{w}}^{jP}(P)}_{\hat{\theta}_j}(tP, \bar{x}_{jP+P}), \mathring{\bar{w}}^{jP}(tP+P)) + \mathring{\lambda}(\mathring{\bar{x}}^{\bar{\boldsymbol{w}}^{jP}(P)}_{\hat{\theta}_j}(tP, \bar{x}_{jP+P}))\\ &\qquad- \mathring{\lambda}(f^P_{\hat{\theta}_j}(\mathring{\bar{x}}^{\bar{\boldsymbol{w}}^{jP}(P)}_{\hat{\theta}_j}(tP, \bar{x}_{jP+P}), \mathring{\bar{w}}^{jP}(tP+P), 0))\Big) + \mathring{\bar{V}}^f(\mathring{\bar{x}}^{\bar{\boldsymbol{u}}^{nom}}_{\hat{\theta}_{j+1}}(n_h, \bar{x}_{jP+P})).\end{aligned}$$

where the inequality arises from Assumption 14 and the definition of $\bar{\boldsymbol{u}}^{nom}$. Similarly expanding $J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP}))$ yields

$$\begin{aligned}
&J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})) \\
&\quad = \sum_{t=0}^{K_h-1} \Big(s_j(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(tP, \bar{x}_{jP}), \mathring{\bar{u}}_{tP|j}^*(\bar{x}_{jP})) - \mathring{\lambda}(f_{\hat{\theta}_{j+1}}^P(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(tP, \bar{x}_{jP}), \mathring{\bar{u}}_{tP|j}^*(\bar{x}_{jP}), 0)) \\
&\qquad + \mathring{\lambda}(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(tP, \bar{x}_{jP}))\Big) + \mathring{\bar{V}}^f(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP})).
\end{aligned}$$

Since it holds by definition that $\mathring{\bar{w}}^{jP}(tP) = \mathring{\bar{u}}_{tP|j}^*(\bar{x}_{jP})$ for $t \in \mathbb{I}_{[0,K_h-1]}$ and $\mathring{\bar{w}}^{jP}(n_h) = \mathring{\kappa}_f(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}))$, then

$$\begin{aligned}
&J_{j+1}^{\text{aux}}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})) \\
&\quad \leq \mathring{\bar{L}}_j(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}), \mathring{\kappa}_f(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}))) + \mathring{\bar{V}}^f(\mathring{\bar{x}}_{\hat{\theta}_{j+1}}^{\bar{\boldsymbol{u}}^{nom}}(n_h, \bar{x}_{jP+P})) \\
&\qquad - \mathring{\bar{V}}^f(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP})) - \mathring{\bar{L}}_j(\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}).
\end{aligned}$$

Lemma 20 and (3.47) give that

$$\begin{aligned}
&J_{j+1}^{\text{aux}}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})) \\
&\quad \leq \alpha_1(|(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}), \mathring{\kappa}_f(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP})))|_{\pi_j}) \\
&\qquad + \alpha_2(|(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}), \mathring{\kappa}_f(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP})))|_{\pi_j}) - \alpha(|\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}|_{\pi_j}).
\end{aligned}$$

We now consider two possibilities: that $(\mathring{\bar{x}}_{jP})_{P-1} \notin \bar{\mathcal{X}}_j^f$, or that $(\mathring{\bar{x}}_{jP})_{P-1} \in \bar{\mathcal{X}}_j^f$. In the case $(\mathring{\bar{x}}_{jP})_{P-1} \notin \bar{\mathcal{X}}_j^f$, (3.50) gives

$$J_{j+1}^{\text{aux}}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})) < 0.$$

On the other hand, if $(\mathring{\bar{x}}_{jP})_{P-1} \in \bar{\mathcal{X}}_j^f$, (3.49) and (3.51) give

$$\begin{aligned}
&J_{j+1}^{\text{aux}}(\bar{x}_{jP+P}, \bar{\boldsymbol{u}}^{nom}) - J_j^{\text{aux}}(\bar{x}_{jP}, \bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})) \\
&\quad \leq \alpha(|(\mathring{\bar{x}}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP}), \mathring{\kappa}_f(\bar{x}_{\hat{\theta}_j}^{\bar{\boldsymbol{u}}_j^*(\bar{x}_{jP})}(n_h, \bar{x}_{jP})))|_{\pi_j}) - \alpha(|\mathring{\bar{x}}_{jP}, \mathring{\bar{u}}_{jP}|_{\pi_j}) \leq 0
\end{aligned}$$

where the second inequality is strict for any $\mathring{\bar{x}} \neq \mathring{\bar{x}}_j^p$. Therefore, a reduction in J_j^{aux} is achieved. □

APPENDIX C

Game Theoretic Wind Farm Control Based on Level-k Cognitive Modeling

C.1 Background and Motivation

In many controls engineering applications, environmental disturbances have a significant impact on system behavior and influence the desired manner in which the system is controlled. Numerous strategies, such as internal model principle methods [6], active disturbance rejection control [113], and robust control [114], seek to account for these environmental disturbances when designing the controller or control signal.

While the aforementioned methodologies have enabled the control of systems that operate in a wide variety of environments, these approaches typically treat environmental disturbances as purely exogenous entities. However, in many applications, not only does the environment influence the behavior of the controlled system, but the controlled system can have influence over the environmental behavior as well. This phenomenon, termed a ‘bidirectional coupling’ between the plant and environment, provides opportunities for a controller to better reject, or even exploit, environmental disturbances. Intuitively, if the plant can be controlled such that the environment is influenced to behave in an advantageous manner, improvements in performance may be achieved.

However, a primary challenge with controlling bidirectionally coupled systems is that high-fidelity environmental models may not exist or are not amenable to directly identify a control signal/policy. Such may be the case, for instance, if the environmental dynamics are governed by processes that are too complex for integration into a model-based controller.

One such example of bidirectional coupling is the task of maximizing the energy generated by a wind farm. Here, the performance of the plant, which consists of a collection of turbines, is dependent upon the behavior of the wind field environment. Moreover, the wind field is influenced by the turbine dynamics as well. For instance, as a turbine extracts energy from the wind, a wake is generated wherein the wind speed downstream of the turbine is reduced. This reduction in wind speed hinders the power production capabilities of downstream turbines, resulting in the well-documented 'wake effect' [115]. Wake steering control is an active field of research where the objective is to optimally deflect the wake so as to mitigate the power production losses of downstream turbines [116]. Unfortunately, as the underlying physics describing the interactions between the turbines and wind field are complex, directly incorporating wake effects in high-fidelity within a wake steering controller is impractical due to the required computational demand.

To mitigate this issue, several model-free strategies have been developed to make the problem of wind farm energy optimization more tractable [117–120]. Extremum seeking controllers, such as those described in [117] and [118], aim to identify optimal control parameters without utilizing a model of the wake dynamics. To bypass the need for a wake model, these approaches typically require that the wind behavior changes slowly in comparison to the rate at which the control signal is modified. However, due to the bidirectional coupling between the turbines and wind field, the plant and environmental dynamics will generally evolve over similar timescales. Alternative strategies given in [119, 120] pose the problem as a cooperative game where each turbine constitutes an agent, and control parameters are iteratively updated between simulations. While monotonic improvement in the baseline performance of the wind farm can be guaranteed in these cases, the control design space grows rapidly as the number of turbines increases or as the control design space is expanded. Consequently, implementing these algorithms in a high-fidelity simulation environment can become impractical for large-scale wind farms as it would require a large number of computationally intensive simulations to be conducted.

Control approaches that directly leverage low-fidelity wake models in the control law update have also been explored in the literature. For instance, dynamic programming approaches have been proposed wherein a controls-oriented wake model is used to predict the impact of upwind axial induction factors [121], and turbine yaw angles [122, 123], on downwind wake velocities. While computationally inexpensive, these strategies rely on the assumption that the low-fidelity models accurately describe the impact of the turbine control decisions on the wind field, which may not hold in practice. Consequently, the performance predicted by the controller may not reflect the true performance of the system.

Therefore, we seek to identify a methodology that merges the advantages of model-free and low-fidelity control strategies for bidirectionally coupled systems. Namely, the high performance that can be achieved through data-driven model-free methods, and the minimal computational demands of low-fidelity methods. The contributions of this work are:

1. the development of a level-k cognition interpretation of bidirectionally coupled systems,
2. a control law that seeks to optimize plant performance when an accurate model of the environmental dynamics is unavailable,
3. a simulation implementation of the proposed controller on a multi-turbine wind farm and

comparison to alternative control strategies.

To serve as a reference for the reader, a notation guide for the variables used in this chapter is provided in Appendix D.5. The contents of this chapter have been accepted to the 2023 7^{th} IEEE Conference on Control Technology and Applications (CCTA) as [103].

C.2 Problem Description

C.2.1 System Models

To describe the relationship between the plant and environment, we introduce the following difference equation model:

$$\begin{aligned} x_{i+1}^p &= f^p(x_i^p, x_i^e, u_i, i), \\ x_{i+1}^e &= f^e(x_i^p, x_i^e, u_i, i) \end{aligned} \tag{C.1}$$

where i denotes a step index, x^p and x^e are the plant and environment states, and u is the control input. Additionally, let the plant stage reward, z^p, be given by

$$z_i^p = r^p(x_i^p, x_i^e, u_i). \tag{C.2}$$

We assume that the plant states and reward, as well as the environmental states or substates that influence the plant dynamics and reward are measurable.

The control objective is given by the optimization problem

$$\begin{aligned} \underset{\{u_i\}\in\mathcal{U}}{\text{maximize}} \quad & \sum_{i=0}^{n_i-1} z_i^p, && \text{(C.3a)} \\ \text{subject to} \quad & x_{i+1}^p = f^p(x_i^p, x_i^e, u_i, i),\, x_0^p = x^p(0), && \text{(C.3b)} \\ & x_{i+1}^e = f^e(x_i^p, x_i^e, u_i, i),\, x_0^e = x^e(0), && \text{(C.3c)} \\ & z_i^p = r^p(x_i^p, x_i^e, u_i), && \text{(C.3d)} \end{aligned}$$

where n_i is the optimization horizon length and $\mathcal{U}$ denotes the feasible control region which is assumed to be convex.

As f^p, f^e, and r^p may be unknown, problem (C.3) is not directly solvable by the user. However, we assume that the user has access to estimates of the dynamic models and plant reward function which are denoted as $\hat{f}^p$, $\hat{f}^e$, and $\hat{r}^p$.

C.2.2 Level-k Cognitive Theory

One method to address problem (C.3) would be to substitute the unknown functions f^p, f^e, and r^p with the known models $\hat{f}^p$, $\hat{f}^e$, and $\hat{r}^p$ such that an estimate of the optimal input sequence can be derived. However, while this approach is intuitive, if a large amount of uncertainty exists in these user-available models, there may be significant mismatch between the estimated optimal input sequence and the solution to (C.3).

To mitigate this issue, we turn to a tool originating from Game Theory (GT) termed *level-k cognitive theory* [124]. In equilibrium theory of games, it is conventionally assumed that each player seeks to maximize their reward in a way that is *mutually consistent*, such that each player's expectation of other players' decisions is correct [125]. However, it is often difficult to have a completely accurate prediction of the behavior of others. Consequently, optimal decisions derived through equilibrium theory are not observed in reality.

Alternatively, level-k theory considers games where players have limited understanding about the methods that others use to make their decisions. Level-k theory posits that players may exhibit different 'levels' in their reasoning, starting at some level-0, and that each agent considers their own strategy to be the most sophisticated amongst all others. As suggested in [126], a level-0 player is defined as one who makes decisions without consideration of the behavior of other players. A level-1 player then makes a prediction of the reward-maximizing decision under the assumption that all others are utilizing a level-0 decision policy. Generalizing to an arbitrary cognitive level, k, a level-k player is one who assumes that all others are at level-$(k-1)$, generates predictions of the decisions made by the other players, and makes a corresponding decision to maximize its own reward. In the following section, we will describe how, by posing control problem (C.3) as a two player non-cooperative game where the controller and environment act as agents, level-k theory can be leveraged to combat model uncertainty.

C.3 Methodology

C.3.1 A Level-k Interpretation of Uncertain Bidirectional Coupling

In [127–129], level-k theory has previously been leveraged within the controls literature to address human-to-human, human-to-automation, or automation-to-automation interactions wherein all players are given as 'sentient' decision makers. Given that the environment is non-sentient in our case, it may seem counterintuitive to describe the environment as a strategic decision maker. Moreover, [127–129] rely on the assumption that the true dynamic and reward models are known by the user in order to identify the level-k decisions, whereas f^p, f^e, and r^p are unknown in this work.

Hence, to enable level-k theory to be applied when one of the players is a non-sentient environment with unknown dynamics, we propose a specific structure of the stage reward, z_i^e, that the environment seeks to maximize over n_i steps. First, we define the environmental 'decision' as the sequence of environmental states over the optimization horizon, $\{x_i^e\}_i$. Here, despite the fact that it is non-sentient, defining the environmental decision in this way will enable the environment to be treated as a strategic player. Subsequently, for some control sequence $\{u_i\}_i$, we define z_i^e as

$$z_i^e = \begin{cases} -\|x_0^e - x^e(0)\|, i = 0 \\ -\|x_i^e - f^e(x_{i-1}^p, x_{i-1}^e, u_{i-1}, i)\|, i > 0. \end{cases} \tag{C.4}$$

Following (C.4), the true dynamics of the environment given by (C.3c) ensure that z_i^e always equals 0. In other words, regardless of the selection of $\{u_i\}_i$, the environment 'chooses' its state sequence such that its reward is *always* maximized. In the context of level-k theory, this implies that if $\{u_i\}_i$ is a level-k control sequence, then the environment produces a level-$(k+1)$ state sequence in response.

This gives the primary benefit of interpreting the plant-environment interactions under the paradigm of level-k theory: although f^e is unknown in analytical form, we know that the environment always identifies an optimal decision in response to the controller's decision. In other words, although the user has uncertainty in the environmental model, it has certainty in the cognitive level of the environment.

C.3.2 Level-k Decision Identification

In Section C.3.1, control problem (C.3) was framed as a non-cooperative game where the controller and environment represent two players with different cognitive levels. We now describe how the decisions of a level-k controller or environment may be determined.

We first require a baseline level-0 control, $\{u_{i|0}\}_i$, upon which level-k environmental decisions, $\{x_{i|k}^e\}_i$, and level-k controller decisions, $\{u_{i|k}\}_i$, are based. To mimic level-k methodologies within the GT literature, this level-0 controller decision may be given by a naive control sequence in $\mathcal{U}$, and does not need to be derived using analytical or numerical methods. Let the level-k control and environmental decision sequences be decomposed as $\{u_{i|k}\}_i = \{u_{0|k}, \dots, u_{n_i-1|k}\}$ and $\{x_{i|k}^e\}_i = \{x_{0|k}^e, \dots, x_{n_i-1|k}^e\}$. Similarly, let $\{x_{i|k}^p\}_i = \{x_{0|k}^p, \dots, x_{n_i-1|k}^p\}$ denote the measured plant state sequence that results from applying control sequence $\{u_{i|k}\}_i$ to the true system.

The level-$(k-1)$ environmental response to a level-$(k-2)$ controller corresponds to the solution of

$$\begin{aligned}
&\max_{\{x_i^e\}_i} \sum_{i=0}^{n_i-1} z_i^e, \\
&\text{s.t. } x_{i+1}^p = f^p(x_i^p, x_i^e, u_{i|k-2}, i),\, x_0^p = x_{0|k-2}^p, \\
&\quad x_{i+1}^e = f^e(x_i^p, x_i^e, u_{i|k-2}, i),\, x_0^e = x_{0|k-1}^e, \\
&\quad z_i^e = \begin{cases} -\|x_0^e - x_{0|k-1}^e\|,\, i = 0 \\ -\|x_i^e - f^e(x_{i-1}^p, x_{i-1}^e, u_{i-1|k-2}, i)\|,\, i > 0. \end{cases}
\end{aligned} \tag{C.5}$$

As f^p and f^e are unknown, it is not possible to solve problem (C.5) directly using numerical optimization methods. However, based on the assumption that the environmental states (or substates) are measureable, we can instead run an experiment or simulation where the controller applies $\{u_{i|k-2}\}_i$ to the plant. By measuring the resulting environmental state sequence, $\{x_{i|k-1}^e\}_i$, the level-$(k-1)$ environmental decision, $\{x_{i|k-1}^e\}_i$ can be identified.

After $\{x_{i|k-1}^e\}_i$ has been determined, a prediction of the optimal controller response is made. First, we introduce the sequence $\{u_{i|k}^*\}_i = \{u_{0|k}^*, \ldots, u_{n_i-1|k}^*\}$ which is given by solving the following optimization problem:

$$\begin{aligned}
\{u_{i|k}^{i_0*}\}_i = \underset{\{u_i\}_i \in \mathcal{U}}{\text{argmax}} \quad & \sum_{i=i_0}^{n_i-1} z_i^p, \\
\text{subject to} \quad & x_{i+1}^p = \hat{f}^p(x_i^p, x_i^e, u_i, i),\, x_{i_0}^p = x_{i_0|k-2}^p, \\
& x_{i+1}^e = \hat{f}^e(x_i^p, x_i^e, u_i, i),\, x_{i_0}^e = x_{i_0|k-1}^e, \\
& z_i^p = \hat{r}^p(x_i^p, x_i^e, u_i)
\end{aligned} \tag{C.6}$$

with

$$u_{i_0|k}^* = \kappa(\{u_{i|k}^{i_0*}\}_i) \tag{C.7}$$

for each $i_0 \in \{0, \ldots, n_i - 1\}$. Here, $\kappa(\{a_i\}_i)$ produces the first element of the sequence $\{a_i\}_i$. The sequence $\{u_{i|k}^*\}_i$ provides an estimate of the optimal control decisions to be made in response to the state sequences $\{x_{i|k-2}^p\}_i$ and $\{x_{i|k-1}^e\}_i$ that were generated by applying the control $\{u_{i|k-2}\}_i$. In this manner, measured state data is incorporated directly in the identification of the control signal, rather than relying solely upon model-based state predictions. The update law given by (C.7) mimics a control structure that is typically utilized in MPC methods. However, whereas MPC establishes feedback by setting the initial conditions of the predicted state trajectories as $x_{i_0}^p = x_{i_0|k}^p$ and $x_{i_0}^e = x_{i_0|k+1}^e$, in (C.6), feedback is instead introduced by leveraging the state measurements obtained by applying a level-$(k-2)$ controller.

However, $\{u^*_{i|k}\}_i$ is likely to be a sub-optimal control decision for two reasons:

1. The functions $\hat{f}^p$, $\hat{f}^e$, and $\hat{r}^p$ may not accurately describe the true plant and environmental dynamics and plant reward. Hence, predictions based on these functions will, in general, be incorrect.

2. The initial conditions $x^p_{i_0|k-2}$ and $x^e_{i_0|k-1}$ used in problem (C.6) to identify the level-k control are obtained from data measured from interactions between a level-$(k-2)$ controller and level-$(k-1)$ environment. However, as noted in Section C.3.1, application of a level-k controller produces a level-$(k+1)$ environmental response. Hence, it cannot be guaranteed in general that $x^p_{i_0|k} = x^p_{i_0|k-2}$ and $x^e_{i_0|k+1} = x^e_{i_0|k-1}$. Consequently, $u^*_{i_0|k}$ will be identified based on an inaccurate initial condition, resulting in inexact estimates of the true plant performance.

Hence, to temper the effect of these sources of sub-optimality, the level-k control decision is not set as $\{u^*_{i|k}\}_i$, but is instead given by

$$u_{i|k} = u_{i|k-2} + \beta(u^*_{i|k} - u_{i|k-2}) \tag{C.8}$$

for some user-defined $\beta \in (0,1)$. Note that $\{u_{i|k}\} \in \mathcal{U}$ is guaranteed due to the convexity of $\mathcal{U}$. By defining the control law in this way, we can prevent the control signal from being dramatically changed in an improper manner due to calculations made from inaccurate models of the environment and plant dynamics or plant reward function. Additionally, for systems with smooth dynamics, restricting changes in the control decision ensures that the state trajectories between cognitive levels are similar, which can reduce undesired performance losses caused by disagreement between $x^p_{i_0|k}$ and $x^p_{i_0|k-2}$, and $x^e_{i_0|k+1}$ and $x^e_{i_0|k-1}$ in (C.6).

As a tuning guide, small values of β should be used when there is low confidence in the dynamic or performance models, or if the system dynamics are highly sensitive to changes in the control signal. In this way, the controller becomes more conservative when large sources of uncertainty exist at the expense of a reduced convergence rate.

C.3.3 J vs. k Model Development

As described in Section C.3.1, the environment is always exactly one cognitive level higher than the controller. Since the environment is a more sophisticated decision maker than the controller, and because the controller has imperfect knowledge of f^p, f^e, and r^p, there may be significant mismatch between the plant performance predicted by the controller and the true plant performance that is observed after running an experiment.

While one strategy to mitigate this mismatch would be to identify more accurate models, system identification techniques that seek to accurately characterize the vector-to-vector mappings $\hat{f}^p$ and $\hat{f}^e$, or vector-to-scalar mapping $\hat{r}^p$ may be computationally intensive. This computational complexity may become particularly burdensome if the number of plant or environmental states is large or if the dimension of u_i is high. Consequently, rather than identifying optimal control decisions purely based on an unknown plant reward structure and unknown plant and environmental dynamic models, we propose a strategy that instead seeks to identify the controller cognitive level that yields the greatest plant reward. Here, while a model relating plant performance to cognitive level may be unknown, generating an estimate of this relationship is computationally inexpensive, as the plant performance and cognitive level are both scalar values.

We define $J_k = \sum_{i=i_0}^{n_i-1} z^p_{i|k}$ as the experimentally measured cumulative plant reward where $z^p_{i|k}$ is the plant stage reward at step i observed by applying the level-k control decision. We subsequently define $\{J_k\}_k = \{J_0, J_2, \ldots, J_{\bar{k}}\}$ as the sequence of measured cumulative plant rewards corresponding to cognitive levels $\{k_k\}_k = \{0, 2, \ldots, \bar{k}\}$ where $\bar{k}$ is the highest cognitive level of the controller that has been experimentally applied to the plant.

Additionally, let $g(k, \theta) : 2\mathbb{N} \times \mathbb{R}^{n_\theta} \to \mathbb{R}$ denote a user-defined function where $2\mathbb{N}$ denotes the set of even natural numbers. Here, $g(k, \theta)$ maps the controller's cognitive level to an estimate of the cumulative plant reward based on a selection of the function parameters θ. θ is the set of parameters determined via regression analysis that best fits $g(k, \theta)$ from available cognitive level data to the measured cumulative plant reward data. Let θ_k signify the calculated model parameters based on the data $\{k_k\}_k$ and $\{J_k\}_k$.

After each experiment for which an input sequence given by a controller with a cognitive level greater than or equal to $2(n_\theta - 1)$ has been applied, we compute θ_k which generates the sequence $\{\theta_k\}_k = \{\theta_{2n_\theta-2}, \theta_{2n_\theta}, \ldots, \theta_{\bar{k}}\}$. For a user-defined n_g, $\{\Theta_k\}_k = \{\theta_{\bar{k}-2(n_g-1)}, \theta_{\bar{k}-2(n_g-2)}, \ldots, \theta_{\bar{k}}\}$ denotes a subsequence of $\{\theta_k\}_k$ with mean $\mu(\Theta_{\bar{k}})$. As the elements of θ_k may have varying magnitudes, we normalize $\{\Theta_k\}_k$ as $\{\overset{\circ}{\Theta}_k\}_k = \{\theta_{\bar{k}-2(n_g-1)} \oslash \mu(\Theta_{\bar{k}}), \theta_{\bar{k}-2(n_g-2)} \oslash \mu(\Theta_{\bar{k}}), \ldots, \theta_{\bar{k}} \oslash \mu(\Theta_{\bar{k}})\}$ where $\oslash$ is the Hadamard division operator. We say that the sequence $\{\theta_k\}_k$ has *converged* if

$$tr\left(C(\overset{\circ}{\Theta}_{\bar{k}})\right) < \epsilon \tag{C.9}$$

for a user-defined $\epsilon > 0$ where $tr(A)$ gives the trace of A and $C(\overset{\circ}{\Theta}_{\bar{k}})$ is the covariance matrix generated from $\{\overset{\circ}{\Theta}_k\}_k$.

Remark 3. *Condition* (C.9) *requires that the data from the last n_g experiments has resulted in sufficiently small modifications to the parameters θ_k. However, this criteria is simply a heuristic strategy for evaluating convergence, and does not guarantee for all $k > \bar{k}$ that $\theta_k \approx \theta_{k-2}$. A loss* *of convergence may occur, for example, if the user's selection of $g(k,\theta)$ poorly reflects the true J vs. k relationship. Consequently, we suggest the user to adopt the following strategies to improve the identification of the J vs. k model:*

1. ***Testing a variety of candidate $g(k,\theta)$.*** *As the structure of the true J vs. k relationship is unknown, at the conclusion of each simulation/experiment, the user should fit the data $\{k_k\}_k$ and $\{J_k\}_k$ to multiple $g(k,\theta)$ functions. Criteria* (C.9) *should then be evaluated for the $g(k,\theta)$ that most closely aligns with the model data.*

2. ***Setting n_g to be large and ϵ to be small.*** *This strategy makes condition* (C.9) *more strict such that greater confidence in parameter convergence can be achieved. However, this approach will, in general, require more experiments to be conducted.*

Identifying cases for which (C.9) guarantees that $\theta_k \approx \theta_{k-2}$ for all $k > \bar{k}$ is a point of further investigation.

Since the cognitive level, k, and cumulative plant reward, J, are scalars, identification of a θ that fits $g(k,\theta)$ to recorded data is inexpensive. This constitutes an additional benefit of leveraging level-k theory in control design, as it allows us to bypass the computationally demanding task of identifying $\hat{f}^p$, $\hat{f}^e$, and $\hat{r}^p$ in high-fidelity.

C.3.4 Game Theoretic Control Framework

The proposed GT strategy for control of uncertain bidirectionally coupled systems is given by Algorithm 7.

Note in Step 5 that the controller cognitive level is set to the solution to (C.11) rather than the maximizer of g, k^*. This is because g is only an *estimate*, and may not perfectly describe the relationship between J and k. Hence, in (C.11), we set the controller's cognitive level to the value of k that achieves the greatest known plant reward for the true system.

C.4 Wind Farm Wake Steering Control

Algorithm 7 is now demonstrated via simulation of a wind farm using the multiphysics software package FAST.Farm. Here, the wind farm, as depicted in Fig. C.1, consists of three 126m diameter turbines based on NREL's 5MW reference turbine [130] that are arranged colinearly along a line termed the 'x-axis' with a spacing of 700m. To mitigate wake effects, wake steering is performed via control of the turbine yaw angles. As an inlet condition to the simulation, the direction of the freestream wind is parallel to the x-axis with its speed v^∞ (m/s) at height h (m) given by the power

Algorithm 7 GT control scheme for bidirectional coupling

```
Step 0: Given $\hat{f}^p$, $\hat{f}^e$, $\hat{r}^p, \mathcal{U}$, $\beta$, $\{u_{i|0}\}_i$, $g$, $n_g$, and $\epsilon$, initialize $k = 0$.
Step 1: Apply $\{u_{i|k}\}_i$ to the plant and record $\{x^p_{i|k}\}_i$, $\{x^e_{i|k+1}\}_i$, and $J_k$.
Step 2: If $k \geq 2(n_\theta - 1)$, find the $\theta_k$ that fits $g(k, \theta)$ to measured data $\{k_k\}_k$ and $\{J_k\}_k$.
Step 3: If $k \geq 2(n_\theta + n_g - 2)$, evaluate convergence condition (C.9). If (C.9) is not met, skip to Step 6.
Step 4: Identify $k^*$ as the solution to

        $k^* = \underset{j \in 2\mathbb{N}}{\text{argmax}} \quad g(j, \theta_k)$        (C.10)

Step 5: If $k^* < k$, set $k$ as

        $k = \underset{j}{\text{argmax}} \quad \{J_j\}_j$        (C.11)

and stop. Otherwise, proceed to Step 6.
Step 6: Increment $k$ as $k = k + 2$ and identify $\{u_{i|k}\}_i$ from (C.8). Return to Step 1.
```

law

$$v^\infty(h) = v^\infty_{ref} \left(\frac{h}{87.6} \right)^{0.143}. \tag{C.12}$$

A description of how FAST.Farm calculates the resulting time-varying wake dynamics can be found in [131].

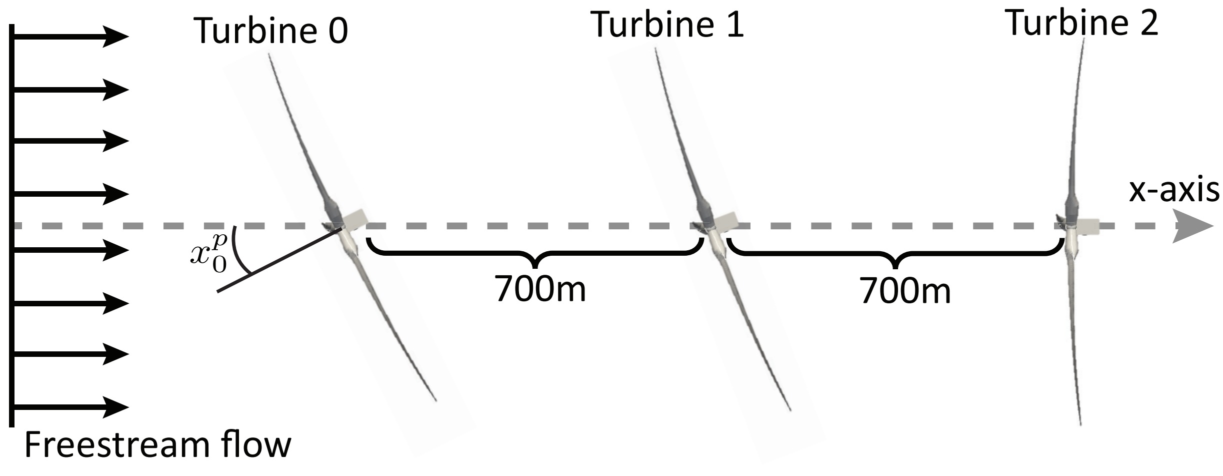


Figure C.1: The configuration of the three turbine wind farm. The plant states are given by the nacelle yaw angles of each of the turbines. x_0^p corresponds to the yaw angle of Turbine 0.

C.4.1 Control Algorithm Parameters

We now outline how the various parameters required by Algorithm 7 are defined to implement the wind farm yaw controller with the goal of maximizing the total energy generated by the farm over 600 seconds for a v_{ref}^{∞} of 15m/s.

First, the step index i is used to distinguish the various turbines in the wind farm. Specifically, $i = 0$ corresponds to the upwind turbine, while $i = 1$ denotes the middle turbine, and $i = 2$ refers to the most downwind turbine as depicted in Fig. C.1. The plant state, x_i^p, corresponds to the nacelle yaw angle of Turbine i, while the control input u_i corresponds to the *commanded* nacelle yaw angle of Turbine i which is restricted to lie in the set $\mathcal{U} = \{u | 0^\circ \leq u \leq 50^\circ\}$. For the purposes of generating a simple low-fidelity model, the controller assumes that the wake propagation is predominantly in the same direction as the freestream wind such that the environmental state x_i^e is given by the hub-height wind speed in the the x-axis direction at Turbine i.

The estimated plant reward function, $\hat{r}^p(x_i^p, x_i^e, u_i)$, makes a prediction of the generated power at Turbine i given the yaw angle, hub-height wind speed, and commanded yaw angle of the turbine. The low-fidelity model assumes that a low-level yaw controller exists that is capable of rapidly driving the nacelle yaw angle to the commanded yaw angle such that $x_i^p \approx u_i$. Consequently, $\hat{r}^p(x_i^p, x_i^e, u_i)$ is simplified as $\hat{r}^p(x_i^e, u_i)$. Similarly, since the controller assumes that the yaw angle of Turbine $i + 1$ is primarily given by u_{i+1}, this means that the yaw angle of Turbine i can be set without incurring yawing dynamics of Turbine $i + 1$ such that $\hat{f}^p = \emptyset$. Finally, the estimated environmental dynamic model, $\hat{f}^e(x_i^p, x_i^e, u_i, i)$, provides a prediction for how the nacelle yaw angle, hub-height wind speed, commanded yaw angle, and turbine index at Turbine i impacts the hub-height wind speed at Turbine $i+1$. The controller assumes that the environmental dynamics are invariant with respect to the turbine index, which, combined with the assumption that $x_i^p \approx u_i$, allows $\hat{f}^e(x_i^p, x_i^e, u_i, i)$ to be simplified as $\hat{f}^e(x_i^e, u_i)$. While these assumptions may only hold in approximation, we note that $\hat{r}^p$, $\hat{f}^e$, and $\hat{f}^e$ are only low-fidelity models and are not expected to perfectly describe the true system behavior that is captured by the FAST.Farm simulation environment but is not directly integrated into the control law given by (C.8).

To develop closed-form expressions for $\hat{r}^p$ and $\hat{f}^e$, a data-based approach is used. Here, a series of single-turbine simulations are conducted wherein the nacelle yaw angle is set to a fixed value between 0° and 50° and the inflow wind at the turbine is set to have a uniform velocity profile with v_{ref}^{∞} in (C.12) ranging from 5m/s to 15m/s. Output measurements of the generated power and the windspeed 700m downwind of the turbine are recorded over 600s of simulation which provides sufficient time for the output signals to reach steady-state. $\hat{r}^p$ and $\hat{f}^e$ are then generated from a polynomial fit relating yaw angle and inflow wind speed to the measured steady-state power and downwind speed as

$$\begin{aligned}\hat{r}^p(x_i^e, u_i) =& -75.7 + 72.6x_i^e + 0.9u_i - 0.1x_i^e u_i - 0.1u_i^2 - 10^{-3}x_i^e u_i^2 + 10^{-3}u_i^3, && \text{(C.13)}\\ \hat{f}^e(x_i^e, u_i) =& -0.1 + 0.7x_i^e - 8\cdot 10^{-3}u_i + 6\cdot 10^{-4}u_i^2 + 10^{-3}x_i^e u_i - 8\cdot 10^{-5}x_i^{e^2} + 10^{-5}x_i^{e^2}u_i \\ & + 9\cdot 10^{-5}x_i^e u_i^2 + 5\cdot 10^{-6}u_i^3. \\ & && \text{(C.14)}\end{aligned}$$

The definitions of $\hat{r}^p$ and $\hat{f}^e$ in (C.13) and (C.14) are specific to the windfarm configuration and freestream wind direction shown in Figure C.1. For different turbine configurations or inlet wind directions, new low-fidelity models can be produced without incurring additional computational complexity by modifying the location at which the downstream windspeed is measured. Additionally, alternative modeling choices may also be used to generate $\hat{r}^p$ and $\hat{f}^e$. For instance, in [120] and [123], the Park and Jensen wake model [132] is extended to include the effect of the upwind turbine yaw angle on the downwind wake velocity. In [133], another modeling choice for $\hat{r}^p$ is suggested where power generation is related to the nacelle yaw through a factor of $\cos^\gamma(x_i^p)$ for some potentially experimentally obtained value of γ.

To enable increased control authority, each turbine updates its yaw angle command every 3s based on the level-$(k-1)$ environmental state and level-$(k-2)$ input at that time. In other words, at time t, we determine $u_{i|k}$ based on the values of $x_{i|k-1}^e$ and $u_{i|k-2}$ measured at time t in the previously run level-$(k-2)$ controller simulation. Consequently a modification to J_k is made according to

$$J_k = \int_0^{600} \sum_{i=0}^{2} z_{i|k}^p(t)dt \tag{C.15}$$

where $z_{i|k}^p(t)$ is the stage reward of Turbine i at time t. The input update gain, β, used in (C.8) is set as 0.2.

We choose $g(k,\theta)$ to have the form of an underdamped second-order system step response:

$$g(k,\theta) = J_0 + M - \frac{Me^{-\zeta\omega_n k}}{\sqrt{1-\zeta^2}} \sin\left(\omega_n\sqrt{1-\zeta^2}k + \cos^{-1}(\zeta)\right) \tag{C.16}$$

where $\theta = (M, \zeta, \omega_n)$. In the context of the relationship between J and k, M (kilojoules) corresponds to the difference in energy generated by the wind farm when a level-0 control input is applied versus the estimated energy generated when a level-∞ control input is applied. ζ dictates how much estimated performance 'overshoot' there is relative to the performance of the level-∞ controller at some finite controller cognitive level. ω_n, which has units of radians per cognitive level, or (rad/k), influences which cognitive level the expected maximum performance overshoot occurs. As discussed in Section C.3.3, the specific form of $g(k, \theta)$ given in (C.16) was chosen from a group of candidate functions as it provided the most accurate fit between the measured $\{k_k\}_k$ and $\{J_k\}_k$ data with $n_g = 5$ and $\epsilon = 10^{-4}$.

The level-0 control input is given by $u_{i|0} = 0°$ for all i and all t, and corresponds to aligning the turbine rotors perpendicular to the freestream wind. This control decision is termed the 'greedy' control strategy, and is the conventional way in which yaw control is implemented in practice [120]. Using greedy control, each turbine seeks to maximize its individual power generation without considering the effect of its decisions on farm-level power production.

C.4.2 Simulation Results

Algorithm 7 was implemented on the wind farm system wherein the θ_k convergence condition, (C.9), was satisfied at $\bar{k} = 76$. The evolution of θ_k is shown in Fig. C.2, with $\theta_{76} = (2.8 \cdot 10^5, 0.58, 0.13)$. Here, although the values of M, ζ, and ω_n change drastically at small values of k, these parameters appear to approach equilibria as the controller's cognitive level continues to increase.

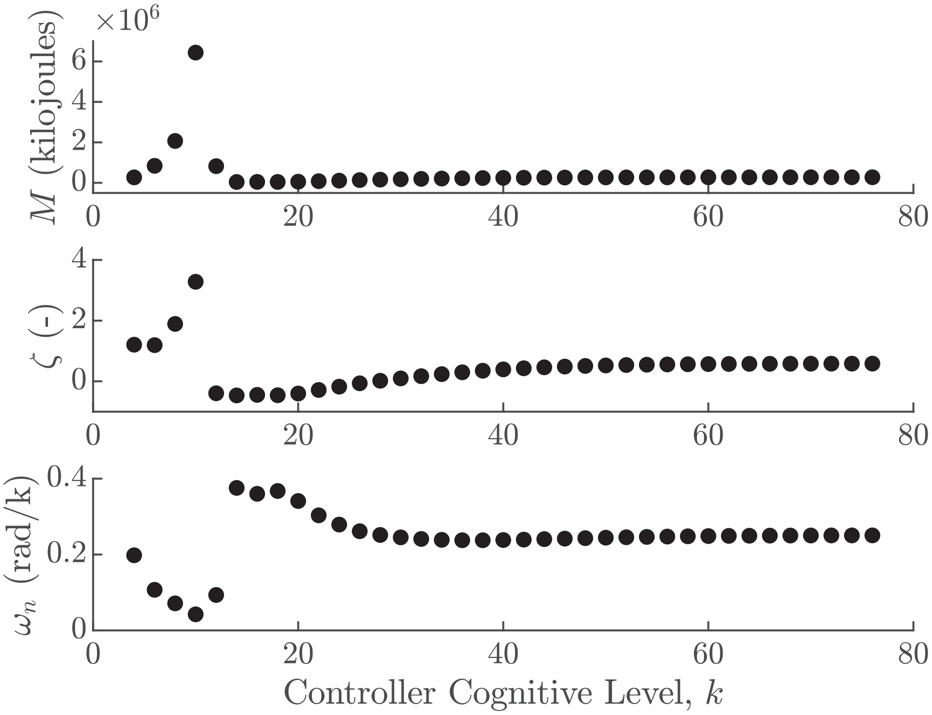


Figure C.2: Estimates of the J vs. k model parameters obtained at various controller cognitive level values.

For $\bar{k} = 76$, (C.10) gives $k^* = 30$. The measured cumulative performance at each controller cognitive level from $k = 0$ to $k = 76$ is depicted in Fig. C.3.

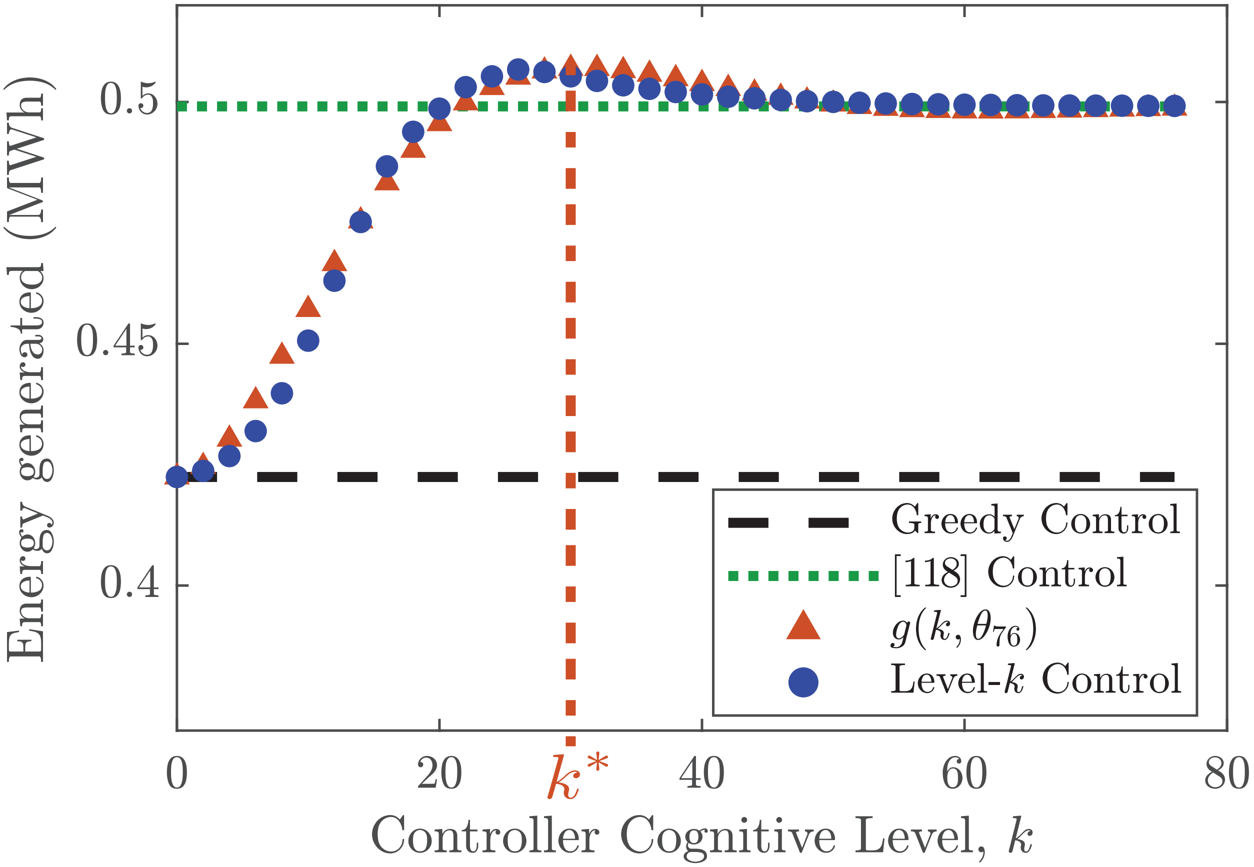


Figure C.3: Total energy generated by the wind farm over the $600s$ simulation for various control schemes. The $g(k, \theta_{76})$ model predicts an optimal cognitive level of $k^* = 30$ while the observed optimal occurs at $k = 26$.

As shown in Fig. C.3, the $g(k, \theta_{76})$ model closely resembles, but does not perfectly match, the measured J vs. k behavior. Consequently, k is not set to k^* in future simulations, but solving (C.11) instead identifies $k = 26$ as the optimal controller cognitive level. Comparing J_{26} to the performance of the greedy controller, the proposed algorithm enables a 20.0% increase in generated energy. We similarly compare the proposed control scheme to the purely model-based dynamic programming strategy given in [123]. Here, through the incorporation of historical plant and environmental data within (C.6) and (C.11), the level-26 controller is able to increase the produced energy by 1.5% in comparison to the controller given in [123]. Additionally, by not relying solely upon model-based state predictions, the use of data augmentation enables the proposed algorithm to be more robust to inaccuracies in the low-fidelity dynamic/reward models, and for the energy generation performance to scale better for larger wind farm systems in comparison to [123].

The wind field at the final timestep of the simulation of the level-27 environment is shown in Fig. C.4. Here, the two upwind turbines have large yaw errors with respect to the freestream flow, which deflects the wake away from downstream turbines. As the furthest downwind turbine, Turbine 2 is able to maximize its individual energy production by setting its nacelle yaw angle to nearly $0°$.

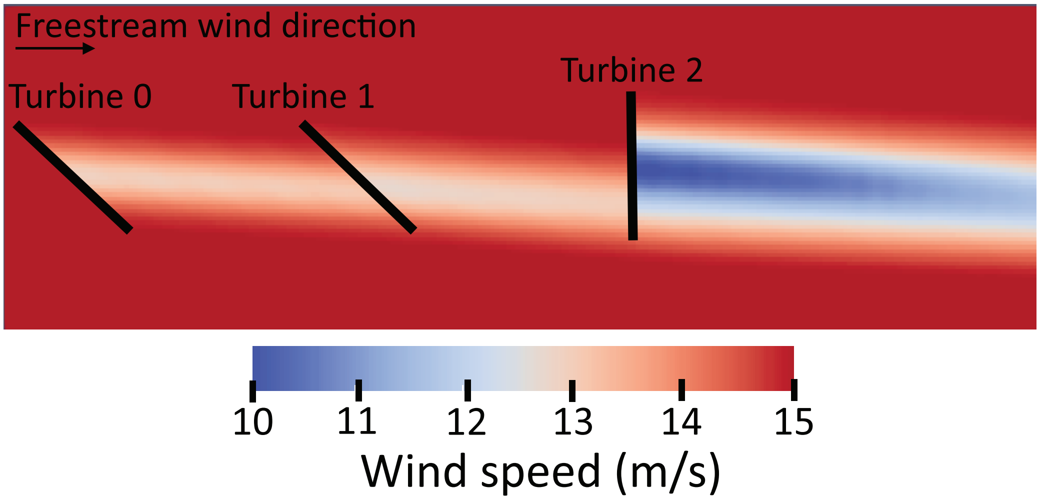


Figure C.4: Level-27 environment hub-height wind field after $600s$.

C.5 Conclusions

In this work we have proposed a framework that leverages tools from level-k cognitive theory for the control of systems that are bidirectionally coupled with their operating environments. The proposed strategy seeks to combat model uncertainty in the system dynamics and reward by posing the optimal control problem as a two player game wherein the environment acts as a more sophisticated decision maker than the controller. By relying on only low-fidelity dynamic and reward models, the controller is able to make computationally tractable predictions of the optimal control strategy. Further, to combat the inaccuracy of these low-fidelity models, historical data is leveraged to supplement predictions of the optimal control decisions in response to the environmental behavior. The control scheme is demonstrated by simulation of a wind farm wherein energy production is increased by 1.5% in comparison to a similar control strategy.

Future work includes the incorporation of a lower-level iterative learning controller to further improve performance, despite the existence of plant model uncertainty. Additionally, identifying methods to enable robust performance in the case of stochastic dynamics is a point of future investigation.

APPENDIX D

Variable Notation Guides

D.1 General Notation Guide

To help guide the reader, the table below provides a list of mathematical notation that is used throughout this dissertation.

General Notation Guide	
Variable	**Description**
$B(a, b)$	Closed Euclidean ball with center a and radius b
I	Identity matrix
$\mathbb{I}_{\geq a}$	Set of integers greater than or equal to a
$\mathbb{I}_{[a,b]}$	Set of integers in the range $[a, b]$
$\mathbb{N}$	Set of natural numbers
$r(\mathcal{A})$	Radius of the smallest closed Euclidean ball that contains set $\mathcal{A}$
$\mathbb{R}$	Set of real numbers
Proj$\{b, A\}$	Orthogonal projection of element b onto set A
$\sigma^{max}(A)$	Largest singular values of matrix A
$\sigma^{min}(A)$	Smallest singular values of matrix A
$\emptyset$	Empty set
$\|b\|_{\mathcal{A}}$	Distance between a point $b \in \mathbb{R}^{n_b}$ and set $\mathcal{A} \subseteq \mathbb{R}^{n_b}$ given as $\inf\limits_{a \in \mathcal{A}} \|\|b - a\|\|$
$\|\|A\|\|$	Euclidean norm of argument A, or the induced matrix norm if A is a matrix
$\lfloor a \rfloor$	Largest integer less than or equal to a
$(\mathcal{A})^n$	$\mathcal{A} \times \ldots \times \mathcal{A}$
$(\cdot)^\top$	Transpose operator
$(\cdot)^{-1}$	Inverse operator
$\mathcal{A} \oplus \mathcal{B}$	Minkowski sum of sets $\mathcal{A}$ and $\mathcal{B}$
$\mathcal{A} \ominus \mathcal{B}$	Pontryagin difference of sets $\mathcal{A}$ and $\mathcal{B}$

D.2 Variable Guide for Chapter 2

To help guide the reader, the tables below provide a list of variable nomenclature that appears recurrently throughout Chapter 2 - Economic Iterative Learning Control: Numerical Optimization-Inspired Controller Design.

Chapter 2 Variable Notation Guide	
Variable	**Description**
$\boldsymbol{A}_E^M$	Gradient of $\boldsymbol{c}_E^M$
$\boldsymbol{A}_I^M$	Gradient of $\boldsymbol{c}_I^M$
$\boldsymbol{A}_E^R$	Gradient of $\boldsymbol{c}_E^R$
$\boldsymbol{A}_I^R$	Gradient of $\boldsymbol{c}_I^R$
$\boldsymbol{c}_E^M$	Equality constraint functions based on the user's system model
$\boldsymbol{c}_I^M$	Inequality constraint functions based on the user's system model
$\boldsymbol{c}_E^R$	Equality constraint functions based on the real system model
$\boldsymbol{c}_I^R$	Inequality constraint functions based on the real system model
E	Set of equality constraint functions
f_x^c	Discrete time state dynamics function
f_x^c	Continuous time state dynamics function
f_y	Output dynamics function
$\mathcal{F}$	Filter
$\boldsymbol{g}$	Gradient of $\boldsymbol{J}$
$\boldsymbol{H}$	Hessian of $\boldsymbol{J}$
I	Set of inequality constraint functions
J	Stage cost (at a timestep)
$\boldsymbol{J}$	Cumulative (over an iteration) cost function
$\boldsymbol{J}^{min}$	Lower bound on cost function
k	Timestep index
m	Quadratic approximation of $\boldsymbol{J}$
M^{M_1}	Bound on the norm of the gradient of $\boldsymbol{c}_E^M$ and $\boldsymbol{c}_I^M$
M^{M_2}	Bound on the norm of the Hessian of $\boldsymbol{c}_E^M$ and $\boldsymbol{c}_I^M$
M^{R_1}	Bound on the norm of the gradient of $\boldsymbol{c}_E^R$ and $\boldsymbol{c}_I^R$
M^{R_2}	Bound on the norm of the Hessian of $\boldsymbol{c}_E^R$ and $\boldsymbol{c}_I^R$
M^{P_1}	Constant

Chapter 2 Variable Notation Guide	
Variable	**Description**
$\boldsymbol{n}$	Optimization normal step
n_τ	Number of timestep samples constituting an iteration
n_u	Dimension of the input space
n_y	Dimension of the output space
n_z	Dimension of the state space
$p^{(k)}$	Output transition function that gives the value of y_k
$\boldsymbol{p}$	Lifted output transition function
P	Projection operator
$\boldsymbol{r}$	Restoration step
$\mathcal{R}$	Set of indices denoting iterations where a restoration procedure was performed
$\boldsymbol{s}$	Candidate optimization iterate step
$\mathcal{S}$	Set of successful iterate indices
$\boldsymbol{t}$	Optimization tangent step
T	Feasible sample period set
$\mathcal{T}$	Lifted feasible sample period set, $\mathcal{T} = T \times \ldots \times T$
u	Inputs
$\boldsymbol{u}$	Lifted input sequence
U	Feasible input set
$\mathcal{U}$	Lifted feasible input set, $\mathcal{U} = U \times \ldots \times U$
$\boldsymbol{x}$	Lifted control trajectory. The decision variable used in the optimization problems
$\boldsymbol{x}^N$	Nominal decision variable plus the normal step
$\mathcal{X}$	Bounded domain of the lifted control trajectories
y	Outputs
$\boldsymbol{y}$	Lifted output sequence
Y	Feasible output set
$\mathcal{Y}$	Lifted feasible output set, $\mathcal{Y} = Y \times \ldots \times Y$
z	System states
z^0	Initial system state at each iteration
$\mathcal{Z}$	Set of iterations for which an iterate is added to the filter

Chapter 2 Variable Notation Guide

Variable	Description
β	$\|\boldsymbol{H}\| + 1$
γ_θ	User-defined parameter
γ_0	User-defined parameter
γ_1	User-defined parameter
γ_2	User-defined parameter
δ_f	Constant
δ_m	Constant
δ_n	Constant
$\delta_{\mathcal{R}}$	Constant
δ_θ	Constant
δ_ρ	Constant
Δ	Trust region radius
η_1	User-defined parameter
η_2	User-defined parameter
θ	Maximum measured real system constraint violation
θ^{max}	Upper bound on constraint violation
κ_{lsc}	Constant
κ_{tmd}	User-defined parameter
κ_{ubg}	Bound on the norm of the gradient of $\boldsymbol{J}$
κ_{ubh}	Constant
κ_{ubt}	Constant
κ_{umh}	Constant
κ_{usc}	Constant
κ_Δ	User-defined parameter
κ_θ	User-defined parameter
κ_μ	User-defined parameter
μ	User-defined parameter
ρ	Ratio of true cost decrease to quadratic approximation cost decrease
τ	Sample period length
$\boldsymbol{\tau}$	Lifted sample period sequence
τ_{max}	Maximum sample period length
χ	First order criticality metric
ψ	User-defined parameter

D.3 Variable Guide for Chapter 3

To help guide the reader, the tables below provide a list of variable nomenclature that appears recurrently throughout Chapter 3 - Robust Adaptive Economic Model Predictive Control.

Chapter 3 Variable Notation Guide

Variable	**Description**
d_j^i	State 'disturbance' caused by $\tilde{\theta}_j^i$ and v_j^i given as $d_j^i \triangleq G(x_j^i, u_j^i)\tilde{\theta}_j^i + v_j^i$
$d_{\tilde{\Theta}_j,\mathcal{A}}^{max}(\bar{x}, \bar{u}, c)$	Upper bound on the size of a disturbance in a neighborhood of size c around $\bar{x}$
$\bar{d}_{\tilde{\Theta}_j,\mathcal{A}}^{max}(\bar{x}, \bar{u})$	Upper bound on the size of a disturbance given state/input pair $(\bar{x}, \bar{u})$
$\mathring{d}$	Disturbance to the P-step system
$\mathcal{D}_j^i$	Known set bounding state disturbances caused by $\tilde{\theta}_j^i$ and v_j^i given as $\mathcal{D}_j^i \triangleq H(\tilde{\Theta}_j^i) \oplus \mathcal{V}$
$\mathcal{D}_j$	Known set bounding all state disturbances within cycle number j given as $\mathcal{D}_j \triangleq \mathcal{D}_j^0 \cup \mathcal{D}_j^1 \cup \ldots \cup \mathcal{D}_j^{n_c-1}$
e	State estimation error given as $e \triangleq x - \bar{x}$
$f_{\hat{\theta}_j^i}(x, u, d)$	Dynamic map based on model parameter values $\hat{\theta}_j^i$
$f_{\hat{\theta}_j}$	The sequence of mapping functions given by the various $\hat{\theta}_j^i \in \hat{\theta}_j$ given as $f_{\hat{\theta}_j} \triangleq \{f_{\hat{\theta}_j^0}, \ldots, f_{\hat{\theta}_j^{n_c-1}}\}$
$f_{\hat{\theta}_j}^P(\mathring{x}, \mathring{u}, \mathring{d})$	Dynamic map of the P-step system based on the sequence of model parameter values, $\hat{\theta}_j$
$F(x, u)$	Known component of system dynamic model
$G(x, u)$	Known regressor component of system dynamic model
$H(\tilde{\Theta}_j^i)$	Known set bounding state uncertainty caused by $\tilde{\theta}_j^i$ given as $H(\tilde{\Theta}_j^i) \triangleq \{G(x, u)\tilde{\theta} : x \in \mathcal{X}, u \in \mathcal{U}, \tilde{\theta} \in \tilde{\Theta}_j^i\}$
i	Intracycle step index given by $i = k \bmod n_c$
j	Intercycle step index given by $j = \lfloor \frac{k}{n_c} \rfloor$
$J_j^{\text{aux}}(\bar{x}, \bar{\boldsymbol{u}})$	Auxiliary objective function
$J_\Omega^{\text{MPC}}(\bar{x}, \bar{\mathbf{u}})$	MPC cost where $\bar{\mathbf{u}}$ denotes a nominal input sequence prediction
k	Timestep index
K_{G,Ω_j}	Upper bound on difference between $G(\bar{x}, \bar{u})$ and $G(x, \phi(\bar{u}, x, \bar{x}))$, normalized by $\|x - \bar{x}\|$
K_ω	Adaptation gain
$\ell(x, u)$	Economic cost function
$\ell^{\text{int}}(\bar{x}, \bar{u}, \Omega)$	Integrated stage cost

Chapter 3 Variable Notation Guide	
Variable	**Description**
$\mathring{\ell}^{\text{int}}(\mathring{\bar{x}}, \mathring{\bar{u}}, \Omega)$	P-step integrated stage cost
$\mathring{\bar{L}}_j(\mathring{\bar{x}}, \mathring{\bar{u}})$	Rotated stage cost
n_c	Cycle period length
n_h	MPC prediction horizon length
n_u	Dimension of u
n_x	Dimension of x
n_θ	Dimension of θ^i
P	Iteration length. Assumed to be an iteration multiple of n_c
$P_{\hat{\theta}}^{\text{RAEMPC}}(J, \bar{x})$	MPC optimization problem defined by cost function, J, initial condition, $\bar{x}$, and system model parameters, $\hat{\theta}$
$\mathring{\bar{\mathcal{P}}}^p_{\tilde{\Theta}_j}$	Set of all possible minimal periodic orbits of the nominal system given parameter error uncertainty set $\tilde{\Theta}_j$
$R^i(\Delta\hat{\theta}^i_{j+1})$	Set of all possible changes to $\bar{x}$ resulting from updating the model parameters at intracycle step i between cycle numbers j and $j+1$ according to $\Delta\hat{\theta}^i_{j+1}$
$R(\Delta\hat{\theta}_j)$	Given as $R(\Delta\hat{\theta}_j) \triangleq R^0(\Delta\hat{\theta}^0_j) \cup \ldots \cup R^{n_c-1}(\Delta\hat{\theta}^{n_c-1}_j)$
$s_j(\mathring{\bar{x}}, \mathring{\bar{u}})$	Supply rate
u	True system inputs
$\bar{u}$	Nominal system inputs
$\bar{\boldsymbol{u}}^{cand}$	Candidate input sequence for the first $n_h - P + 1$ timesteps in the $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ prediction horizon.
$\bar{\boldsymbol{u}}^{nom}$	Candidate input sequence for all n_h timesteps in the $P_{\hat{\theta}_j}^{\text{RAEMPC}}(J_{\Omega_j}^{\text{MPC}}, \bar{x})$ prediction horizon.
$\bar{u}^p_{j,t}$	t^{th} element of $\mathring{\bar{u}}^p_j$
$\mathring{u}$	True P-step system inputs
$\mathring{\bar{u}}^*_{tP\vert j}(\bar{x})$	P-step subsequence of $\bar{\boldsymbol{u}}^*_j(\bar{x})$ where $\mathring{\bar{u}}^*_{tP\vert j}(\bar{x}) = (\bar{u}^*_{tP\vert j}(\bar{x}), \ldots, \bar{u}^*_{tP+P-1\vert j}(\bar{x}))$
$\mathring{\bar{u}}^p_j$	Projection of π_j onto the set $\bar{\mathcal{U}}_j$
$\bar{\boldsymbol{u}}^*_j(\bar{x})$	Solution to $P_{\hat{\theta}}^{\text{RAEMPC}}(J_{\Omega}^{\text{MPC}}, \bar{x})$ which can be decomposed as $\bar{\boldsymbol{u}}^*_j(\bar{x}) = \Big(\bar{u}^*_{0\vert j}(\bar{x}), \ldots, \bar{u}^*_{n_h-1\vert j}(\bar{x})\Big)$
$\mathcal{U}$	Set of feasible u
$(\mathcal{U}_{\hat{\theta}_j, \mathcal{D}_j})^T(x)$	Set of length T input sequences that, starting from initial condition x, ensures $x_t \in \mathcal{X}$ for any disturbance sequence, $\boldsymbol{d} \in (\mathcal{D}_j)^T$. Given as $(\mathcal{U}_{\hat{\theta}_j, \mathcal{D}_j})^T(x) \triangleq \{\boldsymbol{u} \in (\mathcal{U})^T : x^{\boldsymbol{u}}_{\hat{\theta}_j}(t, x) \in \mathcal{X}, \forall \boldsymbol{d} \in (\mathcal{D}_j)^T, \forall t \in \mathbb{I}_{[0,T]}\}$
$\bar{\mathcal{U}}_j$	Projection of $\bar{\mathcal{Z}}_j$ onto $\mathcal{U}$
v	True system noise
v^{max}	Known upper bound on the size of v
$V_\delta(x, \bar{x})$	Distance function given as $V_\delta(x, \bar{x}) \triangleq \Vert x - \bar{x} \Vert$
$\bar{V}^f(\bar{x})$	MPC terminal cost function
$\mathring{\bar{V}}^f(\mathring{\bar{x}})$	Rotated P-step terminal cost

Chapter 3 Variable Notation Guide	
Variable	**Description**
$\mathcal{V}$	Domain of v given as $\mathcal{V} \triangleq \{v \in \mathbb{R}^{n_x} : \|v\| \leq v^{max}\}$
$\mathring{\bar{w}}^k$	P-step subsequence of $\bar{\boldsymbol{w}}^k$
$\bar{\boldsymbol{w}}^k$	Candidate input sequence to $P^{\text{RAEMPC}}_{\hat{\theta}}(J^{\text{MPC}}_{\Omega}, \bar{x})$ in the event that $\bar{\boldsymbol{x}}^{cand}$ and $\bar{\boldsymbol{u}}^{cand}$ are not compatible for $P^{\text{RAEMPC}}_{\hat{\theta}_j}(J^{\text{MPC}}_{\Omega_j}, \bar{x})$. Decomposed into P-step input sequences as $\bar{\boldsymbol{w}}^k(P) = \big(\mathring{\bar{w}}^k(P), \mathring{\bar{w}}^k(2P), \ldots, \mathring{\bar{w}}^k(n_h)\big)$
x	True system states
$x^{\mathbf{a}}_{\hat{\theta}_j}(t, x)$	State of the true system given by $f_{\hat{\theta}_j}$ at time t resulting from initial condition x, applied input sequence $\mathbf{a}$, and some disturbance sequence $\boldsymbol{d} \in (\mathcal{D})^t$
$\bar{x}$	Estimated/nominal system states
$\bar{x}^{\mathbf{a}}(t, \bar{x})$	State of the nominal system given by $f_{\hat{\theta}_j}$ at time t resulting from initial condition $\bar{x}$, applied input sequence $\mathbf{a}$, and the disturbance sequence $\boldsymbol{d} = (0)^t$
$\bar{x}^p_{j,t}$	t^{th} element of $\mathring{\bar{x}}^p_j$
$\bar{\boldsymbol{x}}^{cand}$	Candidate state sequence for the first $n_h - P + 1$ timesteps in the $P^{\text{RAEMPC}}_{\hat{\theta}_j}(J^{\text{MPC}}_{\Omega_j}, \bar{x})$ prediction horizon
$\bar{x}^{nom}_t$	Candidate state at timestep t within the $P^{\text{RAEMPC}}_{\hat{\theta}_j}(J^{\text{MPC}}_{\Omega_j}, \bar{x})$ prediction horizon with $t \in \mathbb{I}_{[0,n_h]}$
$\hat{x}$	Filtered system state
$\tilde{x}$	Filtered system state error given as $\tilde{x} \triangleq x - \hat{x}$
$\mathring{x}$	True states of the P-step system
$\mathring{x}^{\mathbf{a}}_{\hat{\theta}_j}(t, x)$	State of the true P-step system given by $f^P_{\hat{\theta}_j}$ at time t resulting from initial condition x, applied input sequence $\mathbf{a}$, and some disturbance sequence $\boldsymbol{d} \in (\mathcal{D})^t$
$\mathring{\bar{x}}^p_j$	Projection of π_j onto the set $\bar{\mathcal{X}}_j$
$\bar{x}^*_{t\|j}$	Predicted optimal nominal state at step t of the prediction horizon at cycle number j. Given as $\bar{x}^*_{t\|j} \triangleq \bar{x}^{\bar{\boldsymbol{u}}^*_j(\bar{x})}(t, \bar{x})$
$\mathcal{X}$	Set of feasible x
$\bar{\mathcal{X}}_j$	Projection of $\bar{\mathcal{Z}}_j$ onto $\mathcal{X}$
$\bar{\mathcal{X}}^f_j$	MPC terminal state constraint set at cycle number j
$\bar{\mathcal{X}}_{j,n_h}$	The set of initial conditions for which $P^{\text{RAEMPC}}_{\hat{\theta}}(J^{\text{MPC}}_{\Omega_j}, \bar{x})$ is feasible
$z_{\Delta\Theta^i_j}$	Radius of $\Delta\Theta^i_j$
$z_{\Theta^i_j}$	Radius of Θ^i_j at intracycle step i and cycle number j
$\bar{z}_{\Theta^i_j}$	Candidate radius of Θ^i_j at intracycle step i and cycle number j
$z^{\Theta_j}_{max}$	Maximum uncertainty set radius at cycle number j
z^{Ω_j}	Radius of the smallest ball containing Ω_j
$\mathcal{Z}$	The set of feasible true states and inputs given as $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{U}$
$\bar{\mathcal{Z}}_j$	Tightened constraint set for $\bar{x}$ and $\bar{u}$ at cycle number j
α	$\mathcal{K}_\infty$ function
α_1	$\mathcal{K}_\infty$ function
α_2	$\mathcal{K}_\infty$ function

Chapter 3 Variable Notation Guide

Variable	**Description**
β	Initial, user-selected eigenvalues of Σ
$\Delta\hat{\theta}^i_{j+1}$	Change in model parameter estimates between cycles j and $j+1$ at intracycle step i. Given as $\Delta\hat{\theta}^i_{j+1} \triangleq \hat{\theta}^i_{j+1} - \hat{\theta}^i_j$
$\Delta\hat{\theta}_j$	Set of all $\Delta\hat{\theta}^i_j$ at cycle number j given as $\Delta\hat{\theta}_j \triangleq \{\Delta\hat{\theta}^0_j, \ldots, \Delta\hat{\theta}^{n_c-1}_j\}$
$\Delta\Theta^i_{j+1}$	Change in the uncertainty set between cycles j and $j+1$ at intracycle step i. Given as $\Delta\Theta^i_{j+1} \triangleq \Theta^i_j \ominus \Theta^i_{j+1}$
$\Delta\tilde{\Theta}_{j+1}$	Difference between sets $\tilde{\Theta}_j$ and $\tilde{\Theta}_{j+1}$ given as $\Delta\tilde{\Theta}_{j+1} \triangleq \tilde{\Theta}_j \ominus \tilde{\Theta}_{j+1}$
η	Adaptation auxiliary variable
$\hat{\eta}$	Estimated adaptation auxiliary variable
$\tilde{\eta}$	Estimated adaptation auxiliary variable error given as $\tilde{\eta} \triangleq \eta - \hat{\eta}$
θ^i	Unknown model parameters at intracycle step i
$\hat{\theta}^i_j$	User estimate of θ^i at cycle number j
$\hat{\theta}_j$	Set of $\hat{\theta}^i_j$ at cycle number j given as $\hat{\theta}_j \triangleq \{\hat{\theta}^0_j, \ldots, \hat{\theta}^{n_c-1}_j\}$
$\bar{\hat{\theta}}^i_{j+1}$	Projection of parameter estimate $\hat{\theta}^i_{j+1}$ onto Θ^i_j at intracycle step i and cycle number j
$\tilde{\theta}^i_j$	Parameter estimate error at intracycle step i and cycle number j given as $\tilde{\theta}^i_j \triangleq \theta^i - \hat{\theta}^i_j$
$\tilde{\theta}_j$	Set of $\tilde{\theta}^i_j$ at cycle number j given as $\tilde{\theta}_j \triangleq \{\tilde{\theta}^0_j, \ldots, \tilde{\theta}^{n_c-1}_j\}$
$\bar{\tilde{\theta}}^i_j$	Projected parameter estimate error at intracycle step i and cycle number j given as $\bar{\tilde{\theta}}^i_j \triangleq \theta^i - \bar{\hat{\theta}}^i_j$
Θ^i_j	Parameter uncertainty set at intracycle step i and cycle number j designed such that $\theta^i \in \Theta^i_j \triangleq B(\hat{\theta}^i_j, z_{\Theta^i_j})$
$\tilde{\Theta}^i_j$	Parameter error uncertainty set at intracycle step i and cycle number j given as $\tilde{\Theta}^i_j \triangleq B(0, z^{\Theta^i_j})$
$\tilde{\Theta}_j$	Product of sets $\tilde{\Theta}^i_j$ at cycle number j given as $\tilde{\Theta}_j \triangleq \tilde{\Theta}^0_j \times \ldots \times \tilde{\Theta}^{n_c-1}_j$
$\mathring{\kappa}_f$	Terminal feedback law
$\mathring{\lambda}(\mathring{\bar{x}})$	Storage function
π_j	Nominal feasible P-periodic orbit of system $f^P_{\hat{\theta}_j}(\mathring{x}, \mathring{u}, 0)$. Consists of a sequence of state/input pairs $(\bar{x}^p_{j,t}, \bar{u}^p_{j,t})$. Dependency on $\tilde{\theta}_j$ gives the alternate notation $\pi(\tilde{\theta}_j)$
Π_j	Set of all π_j
$\rho_{\bar{\theta}_j}$	Feedback error contraction factor
Σ	Excitation matrix used for the adaptive law updates
$\phi(\bar{u}, x, \bar{x})$	Feedback law
Ψ_{Ω_j}	Set of $(x, \bar{x}, \bar{u})$ such that $(x, \phi(\bar{u}, x, \bar{x})) \in \mathcal{Z}$ and $(x - \bar{x}) \in \Omega_j$
ω^i_j	Filtered version of $G(x, u)$ at intracycle step i and cycle number j
Ω_j	Robust control invariant set under feedback $\phi(\bar{u}, x, \bar{x})$ at cycle number j

D.4 Variable Guide for Chapter 4

To help guide the reader, the tables below provide a list of variable nomenclature that appears recurrently throughout Chapter 4 - Robust Adaptive Economic Iterative Learning Control.

Chapter 4 Variable Notation Guide

Variable	Description
$a_{q,k\mid t}$	Coefficient for $q-$tube constraints
$a^{j*}_{q,k\mid t}$	Optimal $q-$tube constraint coefficient for $R(\underset{\varsigma}{\Theta}^j, x^j_t, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$
$a_{s,k\mid t}$	Coefficient for $s-$tube constraints
$a^{j*}_{s,k\mid t}$	Optimal $s-$tube constraint coefficient for $R(\underset{\varsigma}{\Theta}^j, x^j_t, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$
c_i	Lipschitz constant of constraint function h_i w.r.t. x under feedback κ
cl$(\mathcal{A})$	Closure of the set $\mathcal{A}$
conv$(\mathcal{A})$	Convex hull of $\mathcal{A}$
$\mathcal{C}(\mathcal{F})$	Set of centroids of the n-faces in $\mathcal{F}$
d	System noise
d_w	Disturbance signal resulting from combined effects of system noise and parameter estimation error
$\mathcal{D}$	Domain of the system noise
f	Known component of system dynamic model
$f_{\underset{\varsigma}{\theta}}$	Nominal dynamic model based on parameter estimate $\underset{\varsigma}{\theta}$
$\mathcal{F}(v, \mathcal{X}_i^{\supseteq})$	Set of n-faces of $\mathcal{X}_i^{\supseteq}$ for which vertex v is an element
G	Known regressor component of system dynamic model
G_{max}	Upper bound on the Euclidean-norm of G
G^P_{max}	Upper bound on the $P-$norm of G
h	Constraint functions
h_u	Constraint functions defining $\mathcal{U}$
h_x	Constraint functions defining $\mathcal{X}$
i	Partition index
$\mathcal{I}(\boldsymbol{x})$	Set of indices of the partitions $\mathcal{X}_i$ that were entered over the course of the state sequence $\boldsymbol{x}$
j	Iteration index
J	Economic cost function
J^{int}	Integrated economic cost over the $s-$tube
$\boldsymbol{J}^{int*}_{\underset{\varsigma}{\Theta}, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}}$	Optimal integrated cost of $R(\underset{\varsigma}{\Theta}, x_0, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}, 0)$

Chapter 4 Variable Notation Guide

Variable	Description
k	Prediction horizon index
K_ω	User-defined parameter
ℓ	Stage cost function
ℓ^{int}	Integrated stage cost over the $s-$tube
L_f^u	Lipschitz constant of f w.r.t. u
L_f^x	Lipschitz constant of f w.r.t. x
L_G^u	Lipschitz constant of G w.r.t. u
L_G^x	Lipschitz constant of G w.r.t. x
$L_{G,\kappa}$	Lipschitz constant of G w.r.t. x under feedback κ
$L_{z^{\c{\Theta}}}$	Lipschitz constant of $z^{\c{\Theta}}$ w.r.t. x
$L_{\Delta\c{\theta}}$	Lipschitz constant of $\Delta\c{\theta}(x)$ w.r.t. x
L_θ^x	Lipschitz constant of θ w.r.t. x
$L_{\tilde{\c{\Theta}}}$	Lipschitz constant of $\tilde{w}$ w.r.t. x under feedback κ
L_κ^x	Lipschitz constant of κ w.r.t. x
$L_\kappa^{\bar{x}}$	Lipschitz constant of κ w.r.t. $\bar{x}$
$L_\kappa^{\bar{u}}$	Lipschitz constant of κ w.r.t. $\bar{u}$
$L_{\rho,\c{\theta},\Delta\c{\Theta}}$	Constant
$m(\mathcal{X}_a, \mathcal{X}_b)$	Maximum distance between any point in $\mathcal{X}_a$ and any point in $\mathcal{X}_b$
M_d	Upper bound on the size of d
$M_w^{\mathcal{X}_i}$	Upper bound on the size of the disturbance caused by state-dependent variations in $\theta(x)$ in the partition $\mathcal{X}_i$
n_h	Number of constraint functions h_i
n_p	Number of partitions $\mathcal{X}_i$
n_t	Number of timestep samples constituting and iteration
n_θ	Dimension of θ
p	Cart position
$\dot{p}$	Cart velocity
P	Weighting matrix for distance metric V_δ
$\mathcal{P}(x)$	Set of partition indices of $\mathcal{X}^{\supseteq}$ for which $x \in \text{cl}(\mathcal{X}_i^{\supseteq})$
q	Upper bound on $V_\delta(x, \overset{\therefore}{x})$ under feedback κ
$q_{k\|t}^{j*}$	Optimal value of the $q-$tube radius for $R(\c{\Theta}^j, x_t^j, \overset{\therefore}{\boldsymbol{x}}^j, \overset{\therefore}{\boldsymbol{u}}^j, t)$
$R(\c{\Theta}, x, \overset{\therefore}{\boldsymbol{x}}, \overset{\therefore}{\boldsymbol{u}}, t)$	Shrinking horizon optimization problem
s	Upper bound on $V_\delta(x, \bar{x})$ under feedback κ

Chapter 4 Variable Notation Guide	
Variable	**Description**
$s_{k\|t}^{j*}$	Optimal value of the $s-$tube radius for $R(\c{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$
$\tilde{s}$	Bound on the change in the $s-$tube radius
t	Timestep index
$t_{in}^{\mathcal{X}_i}$	Timestep that the system enters $\mathcal{X}_i$
$t_{out}^{\mathcal{X}_i}$	Last timestep before the system exits $\mathcal{X}_i$
u	Inputs
$\bar{u}$	Nominal inputs
$\bar{u}_{k\|t}^{j*}$	Optimal input for $R(\c{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$
$\dot{\boldsymbol{u}}$	Benchmark input sequence
$\mathcal{U}$	Set of feasible inputs
v	Disturbance caused by noise and state-dependent variations in $\theta(x)$ over $\mathcal{X}_i$
V_f	Terminal cost function
V_f^{int}	Integrated terminal cost over the $s-$tube
$V_{z^{\hat{\Theta}}}$	Metric used to update $z^{\hat{\Theta}}$
V_δ	Distance metric function
$V_{\tilde{\hat{\theta}}}$	Measure of the size of $\tilde{\hat{\theta}}$
$\mathcal{V}^{\mathcal{X}_i}$	Domain of disturbances caused by the combined effects of noise and state-dependent variations in $\theta(x)$ over the partition $\mathcal{X}_i$
$\tilde{w}_{\tilde{\c{\Theta}},\mathcal{D}}(x, u)$	Upper bound on the size of d_w at (x, u)
$\tilde{w}_{\delta,\tilde{\c{\Theta}},\mathcal{D}}(\bar{x}, \bar{u}, s)$	Upper bound on the size of d_w in a neighborhood of size s around (x, u)
$w_{k\|t}^{j*}$	Optimal value of $\tilde{w}_{\delta,\tilde{\c{\Theta}}^j,\mathcal{D}}$ for $R(\c{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$
$\mathcal{W}_{\tilde{\c{\Theta}},\mathcal{D}}$	Domain of d_w
x	System states
x_0	Iteration-invariant initial condition
$\bar{x}$	Nominal system states
$\bar{x}_{k\|t}^{j*}$	Optimal states for $R(\c{\Theta}^j, x_t^j, \dot{\boldsymbol{x}}^j, \dot{\boldsymbol{u}}^j, t)$
$\dot{\boldsymbol{x}}$	Benchmark state sequence
$\hat{x}$	Filtered system states
$\tilde{x}$	Filtered system state error
$\mathcal{X}$	Set of feasible states
$\mathcal{X}_i$	Partition of $\mathcal{X}$
$\mathcal{X}^{\supseteq}$	Smallest bounding hyperrectangle of $\mathcal{X}$
$\mathcal{X}_i^{\supseteq}$	Partition of $\mathcal{X}^{\supseteq}$

Chapter 4 Variable Notation Guide

Variable	Description
$z^{\utilde{\Theta}}(x)$	Radius of $\utilde{\Theta}(x)$
$z^{\utilde{\Theta}^j}_{ave}$	Average value of $z^{\utilde{\Theta}^j}(x)$ over $\mathcal{X}$
$z^{\utilde{\Theta}}_{max}$	Upper bound on $z^{\utilde{\Theta}}(x)$
$z^{\utilde{\Theta}}_{min}$	Lower bound on $z^{\utilde{\Theta}}(x)$
$z^{\hat{\Theta}}$	Upper bound on the size of the error $\hat{\theta} - \theta(x)$
$z^{\hat{\Theta}_{\mathcal{X}_i}}$	Radius of $\hat{\Theta}_{\mathcal{X}_i}$
$z^{\vartheta^{\mathcal{X}_i}}$	Radius of the smallest ball containing $\vartheta^{\mathcal{X}_i}$
$\hat{z}^{\vartheta^{\mathcal{X}_i}}$	Known upper bound on $z^{\vartheta^{\mathcal{X}_i}}$
$\mathcal{Z}$	Set of feasible inputs and constraints, given by $\mathcal{X} \times \mathcal{U}$
β	User-defined parameter
$\gamma_{\boldsymbol{u}^j, \dot{\ddot{\boldsymbol{u}}}^j}$	User-defined parameter
δ_{loc}	Constant
δu	Difference between u and $\dot{\ddot{u}}$
δx	Difference between x and $\dot{\ddot{x}}$
$\Delta\utilde{\theta}(x)$	Iteration-to-iteration change in $\utilde{\theta}(x)$
$\Delta\utilde{\Theta}(x)$	Iteration-to-iteration change in $\utilde{\Theta}(x)$
ϵ_z	Constant
$\epsilon_{\Delta\utilde{\Theta}}$	Upper bound on the size of $\Delta\utilde{\theta}(x)$
η	Adaptation auxiliary variable
$\hat{\eta}$	Estimated adaptation auxiliary variable
$\tilde{\eta}$	Estimated adaptation auxiliary variable error given as $\tilde{\eta} = \eta - \hat{\eta}$
θ	Unknown state-varying model parameters
θ^*	Arbitrary point in $\vartheta^{\mathcal{X}_i}$
$\utilde{\theta}(x)$	User-estimate of $\theta(x)$
$\utilde{\theta}_{max}$	Upper bound on the size of $\utilde{\theta}(x)$
$\theta^{\mathcal{X}_i}$	Center of the smallest ball containing $\vartheta^{\mathcal{X}_i}$
$\hat{\theta}$	Placeholder estimate of $\theta(x)$
$\hat{\theta}_{\mathcal{X}_i}$	Center of $\hat{\Theta}_{\mathcal{X}_i}$
$\bar{\theta}$	Projection of the placeholder model parameter estimate onto $\hat{\Theta}$
$\tilde{\hat{\theta}}$	Given as $\theta^* - \hat{\theta}$
$\tilde{\bar{\theta}}$	Given as $\theta^* - \bar{\theta}$
$\vartheta^{\mathcal{X}_i}$	The set of values of $\theta(x)$ over $\mathcal{X}_i$
$\utilde{\Theta}(x)$	Model parameter uncertainty set function

Chapter 4 Variable Notation Guide	
Variable	**Description**
$\tilde{\c{\Theta}}(x)$	Model parameter error uncertainty set function
$\hat{\Theta}$	Placeholder model parameter uncertainty set over partition $\mathcal{X}_i$ given as $B(\hat{\theta}, z^{\hat{\Theta}})$
$\hat{\Theta}_{\mathcal{X}_i}$	Discontinuous model parameter uncertainty set over partition $\mathcal{X}_i$
κ	Feedback control law
$\kappa_{max} > 0$	Constant
$\rho_{q,\c{\Theta}}$	Growth factor for $q-$tube w.r.t. q
$\rho_{s,\c{\Theta}}$	Growth factor for $q-$tube w.r.t. s
$\rho_{\bar{u},\dot{\bar{u}},\c{\Theta}}$	Growth factor for $q-$tube w.r.t. δu
$\rho_{\c{\theta}}$	Feedback error contraction factor
Σ	Excitation matrix used for the adaptive law updates
υ	Hyperrectangle vertex
ϕ	Pendulum angle
$\dot{\phi}$	Pendulum angular velocity
Ψ	Set of $(x, \bar{x}, \bar{u})$ such that $(x, \kappa(x, \bar{x}, \bar{u}))$ is feasible and $V_\delta(x, \bar{x}) \leq \delta_{loc}$
ω	Filtered version of $G(x, u)$
$\Omega_{\mathcal{U}}(\bar{u}, s)$	Tube around $\bar{u}$ with radius $\kappa_{max}s$
$\Omega_{\mathcal{X}}(\bar{x}, s)$	$s-$tube around $\bar{x}$ with radius s

D.5 Variable Guide for Appendix C

To help guide the reader, the tables below provide a list of variable nomenclature that appears recurrently throughout Appendix C - Game Theoretic Wind Farm Control Based on Level-k Cognitive Modeling.

Appendix C Variable Notation Guide	
Variable	**Description**
f^e	Environment dynamics function
$\hat{f}^e$	Estimated environment dynamics function
f^p	Plant dynamics function
$\hat{f}^p$	Estimated plant dynamics function
g	Estimate of the mapping between the controller's cognitive level and cumulative plant performance

Appendix C Variable Notation Guide

Variable	Description
h	Windfield height
J	Measured cumulative plant reward over an experiment/simulation
k	Level of cognitive decision making
$\bar{k}$	Highest cognitive level that has previously been applied to the plant
k^*	Maximizer of $g(k, \theta_k)$
M	Gain parameter of the mapping g for the simulation case study
n_g	User-defined parameter
n_i	Optimization horizon length
n_θ	Dimension of θ
r^p	Plant Reward function
$\hat{r}^p$	Estimated plant Reward function
u	Control input
$u^*_{i_0\|k}$	Estimated optimal level-k control input at step i_0
$\{u_{i\|k}\}_i$	Level-k control sequence, i.e. the controller's decision
$\{u^*_{i\|k}\}_i$	Estimated optimal level-k control sequence
$\mathcal{U}$	Set of feasible u
$v^\infty(h)$	Freestream wind velocity at height h
v^∞_{ref}	Reference freestream wind velocity
x^e	Environment states
$\{x^e_{i\|k}\}_i$	Level-k environmental state sequence, i.e. the environment's decision
x^p	Plant states
$\{x^p_{i\|k}\}_i$	Level-k plant state sequence
z^e	Environmental stage reward
z^p	Plant stage reward
$z^p_{i\|k}$	Plant stage reward at step i observed by applying the level-k control decision
β	User-defined parameter
ϵ	User-defined parameter
ζ	Damping ratio parameter of the mapping g for the simulation case study
θ	Functional parameters for the mapping g
$\{\Theta_k\}_k$	Subsequence of θ_k
$\{\mathring{\Theta}_k\}_k$	Mean-normalization of $\{\Theta_k\}_k$
κ	Optimal control input identification policy
ω_n	Natural frequency parameter of the mapping g for the simulation case study

BIBLIOGRAPHY

[1] Tarunraj Singh. *Optimal Reference Shaping for Dynamical Systems: Theory and Applications*. CRC Press, October 2009. 1

[2] Donald E. Kirk. *Optimal Control Theory - An Introduction*. Dover Publications, 1998. 2, 41

[3] Richard W. Longman. On the Theory and Design of Linear Repetitive Control Systems. *European Journal of Control*, 16(5):447–496, January 2010. 4, 7

[4] D.A. Bristow, M. Tharayil, and A.G. Alleyne. A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3):96–114, June 2006. Conference Name: IEEE Control Systems Magazine. 4, 11, 77

[5] Youqing Wang, Furong Gao, and Francis J. Doyle. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, 19(10):1589–1600, December 2009. 4

[6] B. A. Francis and W. M. Wonham. The internal model principle of control theory. *Automatica*, 12(5):457–465, September 1976. 4, 133

[7] T. Inoue, M. Nakano, T. Kubo, S. Matsumoto, and H. Baba. High Accuracy Control of a Proton Synchrotron Magnet Power Supply. *IFAC Proceedings Volumes*, 14(2):3137–3142, August 1981. 4

[8] Kok Kia Chew and Masayoshi Tomizuka. Steady-State and Stochastic Performance of a Modified Discrete-Time Prototype Repetitive Controller. *Journal of Dynamic Systems, Measurement, and Control*, 112(1):35–41, March 1990. 4

[9] Ying Liu, Shanmei Cheng, Bowen Ning, and Yesong Li. Robust Model Predictive Control With Simplified Repetitive Control for Electrical Machine Drives. *IEEE Transactions on Power Electronics*, 34(5):4524–4535, May 2019. Conference Name: IEEE Transactions on Power Electronics. 4

[10] Noud Mooren, Gert Witvoet, and Tom Oomen. Gaussian Process Repetitive Control With Application to an Industrial Substrate Carrier System With Spatial Disturbances. *IEEE Transactions on Control Systems Technology*, 31(1):344–358, January 2023. Conference Name: IEEE Transactions on Control Systems Technology. 4

[11] Li Cuiyan, Zhang Dongchun, and Zhuang Xianyi. A survey of repetitive control. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 2, pages 1160–1166 vol.2, September 2004. 5

[12] Tohru Omata, Shinji Hara, and Michio Nakano. Nonlinear repetitive control with application to trajectory control of manipulators. *Journal of Robotic Systems*, 4(5):631–652, 1987. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.4620040505. 5

[13] Hiromitsu Hikita, Mitsuhisa Yamashita, and Yuzuru Kubota. Repetitive Control for a Class of Nonlinear Systems. *JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing*, 36(4):430–434, 1993.

[14] J. Ghosh and B. Paden. Nonlinear repetitive control. *IEEE Transactions on Automatic Control*, 45(5):949–954, May 2000. Conference Name: IEEE Transactions on Automatic Control. 5

[15] M. Norrlof. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251, April 2002. Conference Name: IEEE Transactions on Robotics and Automation. 5

[16] M. Mezghani, G. Roux, M. Cabassud, M.V. Le Lann, B. Dahhou, and G. Casamatta. Application of iterative learning control to an exothermic semibatch chemical reactor. *IEEE Transactions on Control Systems Technology*, 10(6):822–834, November 2002. Conference Name: IEEE Transactions on Control Systems Technology. 5

[17] Mickaël Guth, Thomas Seel, and Jörg Raisch. Iterative Learning Control with variable pass length applied to trajectory tracking on a crane with output constraints. In *52nd IEEE Conference on Decision and Control*, pages 6676–6681, December 2013. ISSN: 0191-2216. 5

[18] Baolin Wu, Danwei Wang, and Eng Kee Poh. High Precision Satellite Attitude Tracking Control via Iterative Learning Control. *Journal of Guidance, Control, and Dynamics*, 38(3):528–534, 2015. Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.G000497. 5

[19] Chris Freeman and Ying Tan. Point-to-point iterative learning control with mixed constraints. In *Proceedings of the 2011 American Control Conference*, pages 3657–3662, June 2011. ISSN: 2378-5861. 6

[20] Bing Chu, Chris T. Freeman, and David H. Owens. A Novel Design Framework for Point-to-Point ILC Using Successive Projection. *IEEE Transactions on Control Systems Technology*, 23(3):1156–1163, May 2015. Conference Name: IEEE Transactions on Control Systems Technology. 6

[21] Yiyang Chen, Bing Chu, and Christopher T. Freeman. Point-to-Point Iterative Learning Control With Optimal Tracking Time Allocation. *IEEE Transactions on Control Systems Technology*, 26(5):1685–1698, September 2018. Conference Name: IEEE Transactions on Control Systems Technology. 6, 11

[22] Yiyang Chen, Bing Chu, and Christopher T. Freeman. A coordinate descent approach to optimal tracking time allocation in point-to-point ILC. *Mechatronics*, 59:25–34, May 2019. 6, 45, 77

[23] Ingyu Lim and Kira L. Barton. Pareto iterative learning control: Optimized control for multiple performance objectives. *Control Engineering Practice*, 26:125–135, May 2014. 6, 11

[24] Kevin L. Moore. *Iterative Learning Control for Deterministic Systems*. Advances in Industrial Control. Springer-Verlag, London, 1993. 7, 78

[25] G. Hillerstrom. Adaptive suppression of vibrations - a repetitive control approach. *IEEE Transactions on Control Systems Technology*, 4(1):72–78, January 1996. Conference Name: IEEE Transactions on Control Systems Technology. 8

[26] Minghe Tian, Bo Wang, Yong Yu, Qinghua Dong, and Dianguo Xu. Discrete-Time Repetitive Control-Based ADRC for Current Loop Disturbances Suppression of PMSM Drives. *IEEE Transactions on Industrial Informatics*, 18(5):3138–3149, May 2022. Conference Name: IEEE Transactions on Industrial Informatics. 8

[27] James Reed, Joshua Daniels, Ayaz Siddiqui, Mitchell Cobb, Michael Muglia, and Chris Vermillion. Optimal Cyclic Control of an Ocean Kite System in a Spatiotemporally Varying Flow Environment. In *2021 American Control Conference (ACC)*, pages 596–601, May 2021. ISSN: 2378-5861. 8

[28] Michael Quann, Lauro Ojeda, William Smith, Denise Rizzo, Matthew Castanier, and Kira Barton. Off-road ground robot path energy cost prediction through probabilistic spatial mapping. *Journal of Field Robotics*, 37(3):421–439, 2020. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21927. 8

[29] Maxwell J. Wu, Mitchell Cobb, Chris Vermillion, and Kira Barton. A Flexible-Time Iterative Learning Control Framework for Linear, Time-Based Performance Objectives. In *2020 American Control Conference (ACC)*, pages 4792–4797, July 2020. ISSN: 2378-5861. 11, 12, 39, 45

[30] Yiyang Chen, Bing Chu, and Christopher T. Freeman. Iterative Learning Control for Path-Following Tasks With Performance Optimization. *IEEE Transactions on Control Systems Technology*, pages 1–13, 2021. Conference Name: IEEE Transactions on Control Systems Technology. 11, 45, 77

[31] Ugo Rosolia and Francesco Borrelli. Minimum time learning model predictive control. *International Journal of Robust and Nonlinear Control*, 31(18):8830–8854, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5284. 11, 12, 45

[32] Siddharth H. Nair, Ugo Rosolia, and Francesco Borrelli. Output-Lifted Learning Model Predictive Control. *IFAC-PapersOnLine*, 54(6):365–370, January 2021.

[33] Ugo Rosolia, Xiaojing Zhang, and Francesco Borrelli. Robust Learning Model Predictive Control for Linear Systems Performing Iterative Tasks. *IEEE Transactions on Automatic Control*, pages 1–1, 2021. Conference Name: IEEE Transactions on Automatic Control. 11

[34] Yushen Long, Lihua Xie, and Shuai Liu. Nontracking type iterative learning control based on economic model predictive control. *International Journal of Robust and Nonlinear Control*, 30(18):8564–8582, 2020. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5261. 12, 45, 77

[35] Yiyang Chen, Bing Chu, and Christopher T. Freeman. Generalized Iterative Learning Control Using Successive Projection: Algorithm, Convergence, and Experimental Verification. *IEEE Transactions on Control Systems Technology*, 28(6):2079–2091, November 2020. Conference Name: IEEE Transactions on Control Systems Technology. 12, 77

[36] Dominic Liao-McPherson, Efe C. Balta, Alisa Rupenyan, and John Lygeros. On Robustness in Optimization-Based Constrained Iterative Learning Control. *IEEE Control Systems Letters*, 6:2846–2851, 2022. Conference Name: IEEE Control Systems Letters. 12, 77

[37] Sandipan Mishra, Ufuk Topcu, and Masayoshi Tomizuka. Optimization-Based Constrained Iterative Learning Control. *IEEE Transactions on Control Systems Technology*, 19(6):1613–1621, November 2011. Conference Name: IEEE Transactions on Control Systems Technology. 12, 20, 77

[38] Katrin Baumgärtner and Moritz Diehl. Zero-Order Optimization-Based Iterative Learning Control. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 3751–3757, December 2020. ISSN: 2576-2370. 12

[39] Roger Fletcher, Nicholas I. M. Gould, Sven Leyffer, Philippe L. Toint, and Andreas Wächter. Global Convergence of a Trust-Region SQP-Filter Algorithm for General Nonlinear Programming. *SIAM Journal on Optimization*, 13(3):635–659, January 2002. Publisher: Society for Industrial and Applied Mathematics. 12, 16, 17, 25, 26, 27, 28, 30, 32, 123

[40] Maxwell Wu, Chris Vermillion, and Kira Barton. Economic Iterative Learning Control Based on Sequential Quadratic Programming. *Submitted to IEEE Transactions on Automatic Control*. 13

[41] J. E. Dennis and Luís N. Vicente. On the Convergence Theory of Trust-Region-Based Algorithms for Equality-Constrained Optimization. *SIAM Journal on Optimization*, 7(4):927–950, November 1997. Publisher: Society for Industrial and Applied Mathematics. 19

[42] J. E. Dennis, Mahmoud El-Alem, and Maria C. Maciel. A Global Convergence Theory for General Trust-Region-Based Algorithms for Equality Constrained Optimization. *SIAM Journal on Optimization*, 7(1):177–207, February 1997. Publisher: Society for Industrial and Applied Mathematics. 19

[43] PH. L. Toint. Global Convergence of a of Trust-Region Methods for Nonconvex Minimization in Hilbert Space. *IMA Journal of Numerical Analysis*, 8(2):231–252, April 1988. 20, 30

[44] Michael C. Grant and Stephen P. Boyd. CVX: Matlab Software for Disciplined Convex Programming, version 2.1, March 2014. 37

[45] Michael C. Grant and Stephen P. Boyd. Graph Implementations for Nonsmooth Convex Programs. In Vincent D. Blondel, Stephen P. Boyd, and Hidenori Kimura, editors, *Recent Advances in Learning and Control*, volume 371, pages 95–110. Springer London, London, 2008. ISSN: 0170-8643 Series Title: Lecture Notes in Control and Information Sciences. 37

[46] Matthijs van de Vosse, Tyler W. Toner, Maxwell J. Wu, Dawn M. Tilbury, and Kira L. Barton. Using Economic Iterative Learning Control for Time-Optimal Control of a Redundant Manipulator. In *Accepted to the 2023 19th IEEE International Conference on Automation Science and Engineering (CASE)*. 38

[47] Basil Kouvaritakis and Mark Cannon. *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. Springer International Publishing, Cham, 2016. 41

[48] Alberto Bemporad and Manfred Morari. Robust model predictive control: A survey. In A. Garulli and A. Tesi, editors, *Robustness in identification and control*, volume 245, pages 207–226. Springer London, London, 1999. Series Title: Lecture Notes in Control and Information Sciences. 42

[49] Davide Martino Raimondo, Daniel Limon, Mircea Lazar, Lalo Magni, and Eduardo Ferná ndez Camacho. Min-max Model Predictive Control of Nonlinear Systems: A Unifying Overview on Stability. *European Journal of Control*, 15(1):5–21, January 2009. 42

[50] Ali Mesbah. Stochastic Model Predictive Control: An Overview and Perspectives for Future Research. *IEEE Control Systems Magazine*, 36(6):30–44, December 2016. Conference Name: IEEE Control Systems Magazine. 42

[51] D. Q. Mayne, M. M. Seron, and S. V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, February 2005. 42

[52] D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho. Robust tube-based MPC for tracking of constrained linear systems with additive disturbances. *Journal of Process Control*, 20(3):248–260, March 2010. 42

[53] Shuyou Yu, Christoph Maier, Hong Chen, and Frank Allgöwer. Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems. *Systems & Control Letters*, 62(2):194–200, February 2013. 42, 56

[54] Ilya Kolmanovsky and Elmer G. Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical Problems in Engineering*, 4(4):317–367, 1998. Publisher: Hindawi. 42

[55] Brett T. Lopez, Jean-Jacques E. Slotine, and Jonathan P. How. Dynamic Tube MPC for Nonlinear Systems. In *2019 American Control Conference (ACC)*, pages 1655–1662, July 2019. ISSN: 2378-5861. 43

[56] András Sasfi, Melanie N. Zeilinger, and Johannes Köhler. Robust adaptive MPC using control contraction metrics. *Automatica*, 155:111169, September 2023. 43, 76

[57] David Angeli, Rishi Amrit, and James B. Rawlings. On Average Performance and Stability of Economic Model Predictive Control. *IEEE Transactions on Automatic Control*, 57(7):1615–1626, July 2012. Conference Name: IEEE Transactions on Automatic Control. 44, 49

[58] Matthias A. Müller, Lars Grüne, and Frank Allgöwer. On the role of dissipativity in economic model predictive control. *IFAC-PapersOnLine*, 48(23):110–116, January 2015. 44

[59] Mario Zanon, Lars Grüne, and Moritz Diehl. Periodic Optimal Control, Dissipativity and MPC. *IEEE Transactions on Automatic Control*, 62(6):2943–2949, June 2017. Conference Name: IEEE Transactions on Automatic Control. 44

[60] Johannes Köhler, Matthias A. Müller, and Frank Allgöwer. On Periodic Dissipativity Notions in Economic Model Predictive Control. *IEEE Control Systems Letters*, 2(3):501–506, July 2018. Conference Name: IEEE Control Systems Letters. 44, 60

[61] Sergio Lucia, Joel A. E. Andersson, Heiko Brandt, Moritz Diehl, and Sebastian Engell. Handling uncertainty in economic nonlinear model predictive control: A comparative case study. *Journal of Process Control*, 24(8):1247–1259, August 2014. 44

[62] Florian A. Bayer, Matthias A. Müller, and Frank Allgöwer. Tube-based robust economic model predictive control. *Journal of Process Control*, 24(8):1237–1246, August 2014. 56, 59, 86

[63] Lukas Schwenkel, Johannes Köhler, Matthias A. Müller, and Frank Allgöwer. Robust Economic Model Predictive Control without Terminal Conditions. *IFAC-PapersOnLine*, 53(2):7097–7104, January 2020. 44

[64] Timothy J. Broomhead, Chris Manzie, Rohan C. Shekhar, and Peter Hield. Robust periodic economic MPC for linear systems. *Automatica*, 60:30–37, October 2015. 44

[65] Kim P. Wabersich, Florian A. Bayer, Matthias A. Müller, and Frank Allgüwer. Economic Model Predictive Control for Robust Periodic Operation with Guaranteed Closed-Loop Performance. In *2018 European Control Conference (ECC)*, pages 507–513, June 2018. 44, 47, 60, 64, 73

[66] Monimoy Bujarbaruah, Xiaojing Zhang, Ugo Rosolia, and Francesco Borrelli. Adaptive MPC for Iterative Tasks. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 6322–6327, December 2018. ISSN: 2576-2370. 45, 76

[67] Mitchell Cobb, James Reed, Joshua Daniels, Ayaz Siddiqui, Max Wu, Hosam Fathy, Kira Barton, and Chris Vermillion. Iterative Learning-Based Path Optimization With Application to Marine Hydrokinetic Energy Systems. *IEEE Transactions on Control Systems Technology*, pages 1–15, 2021. Conference Name: IEEE Transactions on Control Systems Technology. 45

[68] Kirti D. Mishra, James Reed, Maxwell Wu, Kira Barton, and Chris Vermillion. Hierarchical Structures for Economic Repetitive Control. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5838–5844, December 2021. ISSN: 2576-2370. 45

[69] A. P. Page and C. T. Freeman. Point-to-point repetitive control of functional electrical stimulation for drop-foot. *Control Engineering Practice*, 96:104280, March 2020. 45

[70] Maxwell Wu, Chris Vermillion, and Kira Barton. Point-to-Point Repetitive Control with Optimal Tracking Time. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5845–5850, December 2021. ISSN: 2576-2370. 45

[71] Matthias Lorenzen, Mark Cannon, and Frank Allgöwer. Robust MPC with recursive model update. *Automatica*, 103:461–471, May 2019. 45, 76

[72] Monimoy Bujarbaruah, Siddharth H. Nair, and Francesco Borrelli. A Semi-Definite Programming Approach to Robust Adaptive MPC under State Dependent Uncertainty. In *2020 European Control Conference (ECC)*, pages 960–965, May 2020.

[73] Monimoy Bujarbaruah, Xiaojing Zhang, Marko Tanaskovic, and Francesco Borrelli. Adaptive Stochastic MPC Under Time-Varying Uncertainty. *IEEE Transactions on Automatic Control*, 66(6):2840–2845, June 2021. Conference Name: IEEE Transactions on Automatic Control. 76

[74] Xiaonan Lu, Mark Cannon, and Denis Koksal-Rivet. Robust adaptive model predictive control: Performance and parameter estimation. *International Journal of Robust and Nonlinear Control*, 31(18):8703–8724, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5175.

[75] Stefano Di Cairano and Claus Danielson. Indirect adaptive model predictive control and its application to uncertain linear systems. *International Journal of Robust and Nonlinear Control*, 31(18):8678–8702, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5166. 45

[76] Veronica Adetola and Martin Guay. Robust adaptive MPC for constrained uncertain nonlinear systems. *International Journal of Adaptive Control and Signal Processing*, 25(2):155–167, 2011. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/acs.1193. 45, 76, 81

[77] Guilherme A. A. Gonçalves and Martin Guay. Robust discrete-time set-based adaptive predictive control for nonlinear systems. *Journal of Process Control*, 39:111–122, March 2016. 49, 50, 52, 53, 54, 81, 97, 99, 102

[78] Johannes Köhler, Peter Kötting, Raffaele Soloperto, Frank Allgöwer, and Matthias A. Müller. A robust adaptive model predictive control framework for nonlinear uncertain systems. *International Journal of Robust and Nonlinear Control*, 31(18):8725–8749, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5147. 45, 66, 68, 76, 81, 82, 83, 114, 115, 116

[79] Mohamed L. Shaltout, Zheren Ma, and Dongmei Chen. An Adaptive Economic Model Predictive Control Approach for Wind Turbines. *Journal of Dynamic Systems, Measurement, and Control*, 140(5):051007, May 2018. 45

[80] Bijan Sakhdari and Nasser L. Azad. Adaptive Tube-Based Nonlinear MPC for Economic Autonomous Cruise Control of Plug-In Hybrid Electric Vehicles. *IEEE Transactions on Vehicular Technology*, 67(12):11390–11401, December 2018. Conference Name: IEEE Transactions on Vehicular Technology. 45

[81] M. Guay and V. Adetola. Adaptive economic optimising model predictive control of uncertain nonlinear systems. *International Journal of Control*, 86(8):1425–1437, August 2013. 45, 46, 76, 81

[82] Maxwell Wu, Chris Vermillion, and Kira Barton. Robust Adaptive Economic Model Predictive Control for Nonlinear Repetitive Systems. *Submitted to IEEE Transactions on Automatic Control*. 47

[83] Matthias A. Müller and Lars Grüne. Economic model predictive control without terminal constraints for optimal periodic behavior. *Automatica*, 70:128–139, August 2016. 61

[84] Lars Grüne and Vryan Gil Palma. Robustness of performance and stability for multistep and updated multistep MPC schemes. *Discrete and Continuous Dynamical Systems*, 35(9):4385–4414, 2015. 62

[85] Ronald R. Mohler. I Introduction. In *Bilinear Control Processes: With Applications to Engineering, Ecology, and Medicine*, volume 106 of *Mathematics in Science and Engineering*, pages 1–20. Elsevier, 1973. ISSN: 0076-5392. 72

[86] Florian Bayer, Mathias Bürger, and Frank Allgöwer. Discrete-time Incremental ISS: A framework for Robust NMPC. In *2013 European Control Conference (ECC)*, pages 2068–2073, July 2013. 72

[87] Lukas Hewing, Kim P. Wabersich, Marcel Menner, and Melanie N. Zeilinger. Learning-Based Model Predictive Control: Toward Safe Learning in Control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(1):269–296, 2020. _eprint: https://doi.org/10.1146/annurev-control-090419-075625. 76

[88] Brett Thomas Lopez. *Adaptive robust model predictive control for nonlinear systems*. Thesis, Massachusetts Institute of Technology, 2019. Accepted: 2019-10-04T21:32:05Z. 76

[89] Suguru Arimoto, Sadao Kawamura, and Fumio Miyazaki. Bettering operation of Robots by learning. *Journal of Robotic Systems*, 1(2):123–140, 1984. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.4620010203. 77

[90] Abdelhamid Tayebi and Chiang-Ju Chien. A Unified Adaptive Iterative Learning Control Framework for Uncertain Nonlinear Systems. *IEEE Transactions on Automatic Control*, 52(10):1907–1913, October 2007. 77

[91] Jian-Xin Xu and Xu Jin. State-Constrained Iterative Learning Control for a Class Of MIMO Systems. *IEEE Transactions on Automatic Control*, 58(5):1322–1327, May 2013. Conference Name: IEEE Transactions on Automatic Control.

[92] Berk Altın and Kira Barton. Robust iterative learning for high precision motion control through L1 adaptive feedback. *Mechatronics*, 24(6):549–561, September 2014.

[93] Ruikun Zhang, Zhongsheng Hou, Ronghu Chi, and Honghai Ji. Adaptive iterative learning control for nonlinearly parameterised systems with unknown time-varying delays and input saturations. *International Journal of Control*, 88(6):1133–1141, June 2015.

[94] Hocine Benslimane, Abdesselem Boulkroune, and Hachemi Chekireb. Adaptive Iterative Learning Control of Nonlinearly Parameterized Pure Feedback Nonlinear Systems. *Journal of Control, Automation and Electrical Systems*, 28(4):457–469, August 2017.

[95] Miao Yu and Chaoyong Li. Robust Adaptive Iterative Learning Control for Discrete-Time Nonlinear Systems With Time-Iteration-Varying Parameters. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7):1737–1745, July 2017. Conference Name: IEEE Transactions on Systems, Man, and Cybernetics: Systems.

[96] Zhongjie He and Jianning Li. Barrier Robust Iterative Learning Control for Nonlinear Systems With Both Nonparametric Uncertainties and Time-Iteration-Varying Parametric Uncertainties Under Alignment Condition. *IEEE Access*, 10:85918–85928, 2022. Conference Name: IEEE Access.

[97] Zhi Yang, Wei Wang, Yuntao Zhang, Qiuzhen Yan, and Jianping Cai. High-Order Internal Model Based Barrier Iterative Learning Control for Time-Iteration-Varying Parametric Uncertain Systems With Arbitrary Initial Errors. *IEEE Access*, 10:17619–17628, 2022. Conference Name: IEEE Access. 77

[98] Maxwell Wu and Kira Barton. Robust Adaptive Economic Iterative Learning Control for State-Varying Parametric Uncertainties. *In preparation*. 78

[99] Johannes Köhler, Raffaele Soloperto, Matthias A. Müller, and Frank Allgöwer. A Computationally Efficient Robust Model Predictive Control Framework for Uncertain Nonlinear Systems. *IEEE Transactions on Automatic Control*, 66(2):794–801, February 2021. Conference Name: IEEE Transactions on Automatic Control. 81

[100] Branko Grünbaum. *Convex Polytopes*, volume 221 of *Graduate Texts in Mathematics*. Springer, New York, NY, 2003. 108

[101] Raffaele Pugliese, Stefano Regondi, and Riccardo Marini. Machine learning-based approach: global trends, research directions, and regulatory standpoints. *Data Science and Management*, 4:19–29, December 2021. 119

[102] E. F. Camacho, D. R. Ramirez, D. Limon, D. Muñoz de la Peña, and T. Alamo. Model predictive control techniques for hybrid systems. *Annual Reviews in Control*, 34(1):21–31, April 2010. 121

[103] Maxwell Wu, Chris Vermillion, and Kira Barton. Game Theoretic Wind Farm Control based on Level-k Cognitive Modeling. In *Accepted to the 2023 7th IEEE Conference on Control Technology and Applications (CCTA)*, 2023. 122, 135

[104] Petros A. Ioannou and Jing Sun. *Robust Adaptive Control*. Courier Corporation, December 2012. Google-Books-ID: pXWFY_vbg1MC. 123

[105] Chris Vermillion, Mitchell Cobb, Lorenzo Fagiano, Rachel Leuthold, Moritz Diehl, Roy S. Smith, Tony A. Wood, Sebastian Rapp, Roland Schmehl, David Olinger, and Michael Demetriou. Electricity in the air: Insights from two decades of advanced control research and experimental flight testing of airborne wind energy systems. *Annual Reviews in Control*, 52:330–357, January 2021. 124

[106] Mitchell K. Cobb, Kira Barton, Hosam Fathy, and Chris Vermillion. Iterative Learning-Based Path Optimization for Repetitive Path Planning, With Application to 3-D Crosswind Flight of Airborne Wind Energy Systems. *IEEE Transactions on Control Systems Technology*, 28(4):1447–1459, July 2020. Conference Name: IEEE Transactions on Control Systems Technology. 124

[107] Mitchell Cobb, Maxwell Wu, Kira Barton, and Chris Vermillion. Flexible-Time Economic Iterative Learning Control: A Case Study in Airborne Wind Energy. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5580–5586, December 2019. ISSN: 2576-2370. 124

[108] Seyed Amin Sajadi-Alamdari, Holger Voos, and Mohamed Darouach. Nonlinear Model Predictive Control for Ecological Driver Assistance Systems in Electric Vehicles. *Robotics and Autonomous Systems*, 112:291–303, February 2019. 124

[109] Pedro F. Lima, Marco Trincavelli, Mattias Nilsson, Jonas Mårtensson, and Bo Wahlberg. Experimental evaluation of economic model predictive control for an autonomous truck. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 710–715, June 2016. 124

[110] Robin Verschueren, Mario Zanon, Rien Quirynen, and Moritz Diehl. Time-optimal race car driving using an online exact hessian based nonlinear MPC algorithm. In *2016 European Control Conference (ECC)*, pages 141–147, June 2016. 124

[111] Jiuping Xu and Ziqiang Zeng. A discrete time optimal control model with uncertainty for dynamic machine allocation problem and its application to manufacturing and construction industries. *Applied Mathematical Modelling*, 36(8):3513–3544, August 2012. 124

[112] Lin Li and Zeyi Sun. Dynamic Energy Control for Energy Efficiency Improvement of Sustainable Manufacturing Systems Using Markov Decision Process. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(5):1195–1205, September 2013. Conference Name: IEEE Transactions on Systems, Man, and Cybernetics: Systems. 125

[113] Jingqing Han. From PID to Active Disturbance Rejection Control. *IEEE Transactions on Industrial Electronics*, 56(3):900–906, March 2009. 133

[114] Kemin Zhou and John Comstock Doyle. *Essentials of Robust Control*, volume 104. Prentice Hall, Upper Saddle River, NJ, October 1998. 133

[115] Rabia Shakoor, Mohammad Yusri Hassan, Abdur Raheem, and Yuan-Kang Wu. Wake effect modeling: A review of wind farm layout optimization using Jensen's model. *Renewable and Sustainable Energy Reviews*, 58:1048–1059, May 2016. 134

[116] Michael F. Howland, Sanjiva K. Lele, and John O. Dabiri. Wind farm power optimization through wake steering. *Proceedings of the National Academy of Sciences*, 116(29):14495–14500, July 2019. Publisher: Proceedings of the National Academy of Sciences. 134

[117] Kathryn E. Johnson and Geraldine Fritsch. Assessment of Extremum Seeking Control for Wind Farm Energy Production. *Wind Engineering*, 36(6):701–715, December 2012. Publisher: SAGE Publications. 134

[118] Umberto Ciri, Mario A. Rotea, and Stefano Leonardi. Model-free control of wind farms: A comparative study between individual and coordinated extremum seeking. *Renewable Energy*, 113:1033–1045, December 2017. 134

[119] Jason R. Marden, Shalom D. Ruben, and Lucy Y. Pao. A Model-Free Approach to Wind Farm Control Using Game Theoretic Methods. *IEEE Transactions on Control Systems Technology*, 21(4):1207–1214, July 2013. 134

[120] P. M. O. Gebraad, F. W. Teeuwisse, J. W. van Wingerden, P. A. Fleming, S. D. Ruben, J. R. Marden, and L. Y. Pao. Wind plant power optimization through yaw control using a parametric model for wake effects—a CFD simulation study. *Wind Energy*, 19(1):95–114, 2016. 134, 144, 145

[121] Mario A. Rotea. Dynamic Programming Framework for Wind Power Maximization. *IFAC Proceedings Volumes*, 47(3):3639–3644, January 2014. 134

[122] Jinkyoo Park, Soonduck Kwon, and Kincho H. Law. Wind farm power maximization based on a cooperative static game approach. In *Active and Passive Smart Structures and Integrated Systems 2013*, volume 8688, pages 204–218. SPIE, April 2013. 134

[123] Zamiyad Dar, Koushik Kar, Onkar Sahni, and Joe H. Chow. Windfarm Power Optimization Using Yaw Angle Control. *IEEE Transactions on Sustainable Energy*, 8(1):104–116, January 2017. 134, 144, 146

[124] Colin F. Camerer, Teck-Hua Ho, and Juin-Kuan Chong. A Cognitive Hierarchy Model of Games. *The Quarterly Journal of Economics*, 119(3):861–898, August 2004. 136

[125] George J. Mailath. Do People Play Nash Equilibrium? Lessons from Evolutionary Game Theory. *Journal of Economic Literature*, 36(3):1347–1374, 1998. Publisher: American Economic Association. 136

[126] Tomasz Strzalecki. Depth of reasoning and higher order beliefs. *Journal of Economic Behavior & Organization*, 108:108–122, December 2014. 136

[127] Scott Backhaus, Russell Bent, James Bono, Ritchie Lee, Brendan Tracey, David Wolpert, Dongping Xie, and Yildiray Yildiz. Cyber-Physical Security: A Game Theory Model of Humans Interacting Over Control Systems. *IEEE Transactions on Smart Grid*, 4(4):2320–2327, December 2013. 136

[128] Nan Li, Ilya Kolmanovsky, Anouck Girard, and Yildiray Yildiz. Game Theoretic Modeling of Vehicle Interactions at Unsignalized Intersections and Application to Autonomous Vehicle Control. In *2018 Annual American Control Conference (ACC)*, pages 3215–3220, June 2018. ISSN: 2378-5861.

[129] Sisi Li, Nan Li, Anouck Girard, and Ilya Kolmanovsky. Decision making in dynamic and interactive environments based on cognitive hierarchy theory, Bayesian inference, and predictive control. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 2181–2187, December 2019. ISSN: 2576-2370. 136

[130] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-MW Reference Wind Turbine for Offshore System Development. Technical Report NREL/TP-500-38060, National Renewable Energy Lab. (NREL), Golden, CO (United States), February 2009. 141

[131] Jason Jonkman and Kelsey Shaler. FAST.Farm User's Guide and Theory Manual. Technical Report NREL/TP-5000-78785, National Renewable Energy Lab. (NREL), Golden, CO (United States), 2021. 142

[132] N.O. Jensen. A note on wind generator interaction. Report 87-550-0971-9, Risø National Laboratory, Roskilde, 1983. 144

[133] K. Boorsma. Power and loads for wind turbines in yawed conditions. Analysis of field measurements and aerodynamic predictions. Technical Report ECN-E-12-047, Energy research Centre of the Netherlands ECN, Petten (Netherlands), November 2012. 144